(1) Requirement: The monitoring system bandwidth measured at its 20 dB down points must be equal to or greater than the emission bandwidth of the intended transmission.

Response: Tx emission bw is best represented by mode 6 in Table 23, Rx BW is represented in Table 41 of Mozart_Production_Analog_DVT_Report_138088_v33_20Apr2011.doc

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	269.56 KHz	271.41 KHz	275.52 KHz	267.91 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	81.11 KHz	81.46 KHz	81.74 KHz	81.32 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	205.33 KHz	203.83 KHz	204.01 KHz	204.10 KHz	PASS	Regulatory subtest #1

Table 1 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

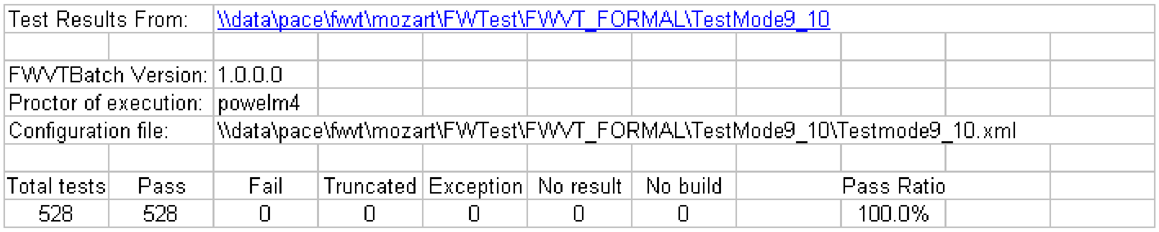
Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Frequency (-3dB) TelC Narrowband	**26 kHz min**	34KHz	34KHz	34KHz	33KHz	PASS	Measured using baseband, not at carrier frequency. (Spec is 52 kHz at carrier frequency)
Frequency (-3dB) TelC Wideband	**110 kHz min**	134KHz	135KHz	129KHz	134KHz	PASS	Measured using baseband, not at carrier frequency. (Spec is 220 kHz at carrier frequency)

Table 2 Rx Filter -3dB Bandwidth (TelC)

(2) Requirement: Within 5 seconds prior to initiating a communications session, circuitry associated with a MedRadio programmer/control transmitter must monitor the channel or channels the system devices intend to occupy for a minimum of 10 milliseconds per channel.

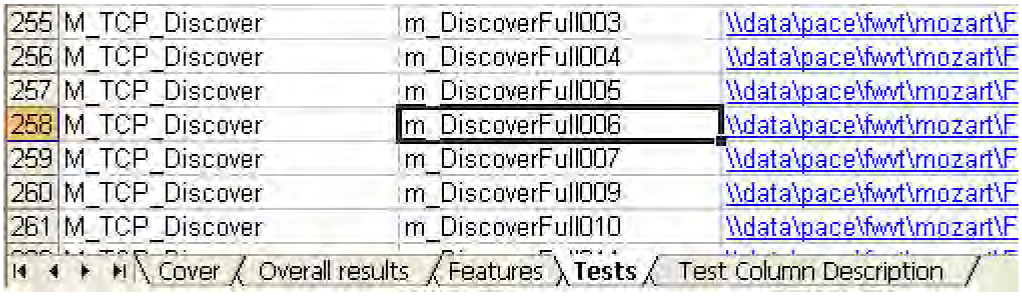
Response: The timing specific tests were verified by Firmware VT. The attached document DSN001255, titled Tel M FWVT Test Results & Status Phase 1 TCP-M, lists each test and the run status of Pass. The Traceability Report (DSN001257) maps the requirement to the test name(s). The requirement is listed in telM_protocol_C_A36244.pdf.

Test Results (from DSN001255.xls)::



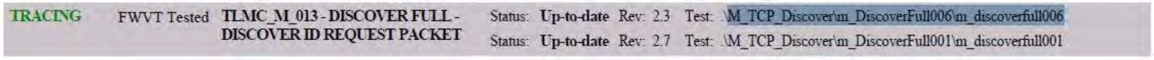
Test Results From:		\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10								
FWVTBatch Version:		1.0.0.0								
Proctor of execution:		powelm4								
Configuration file:		\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Testmode9_10.xml								
Total tests	Pass	Fail	Truncated	Exception	No result	No build		Pass Ratio		
528	528	0	0	0	0	0		100.0%		

Test Name (from DSN001255.xls):



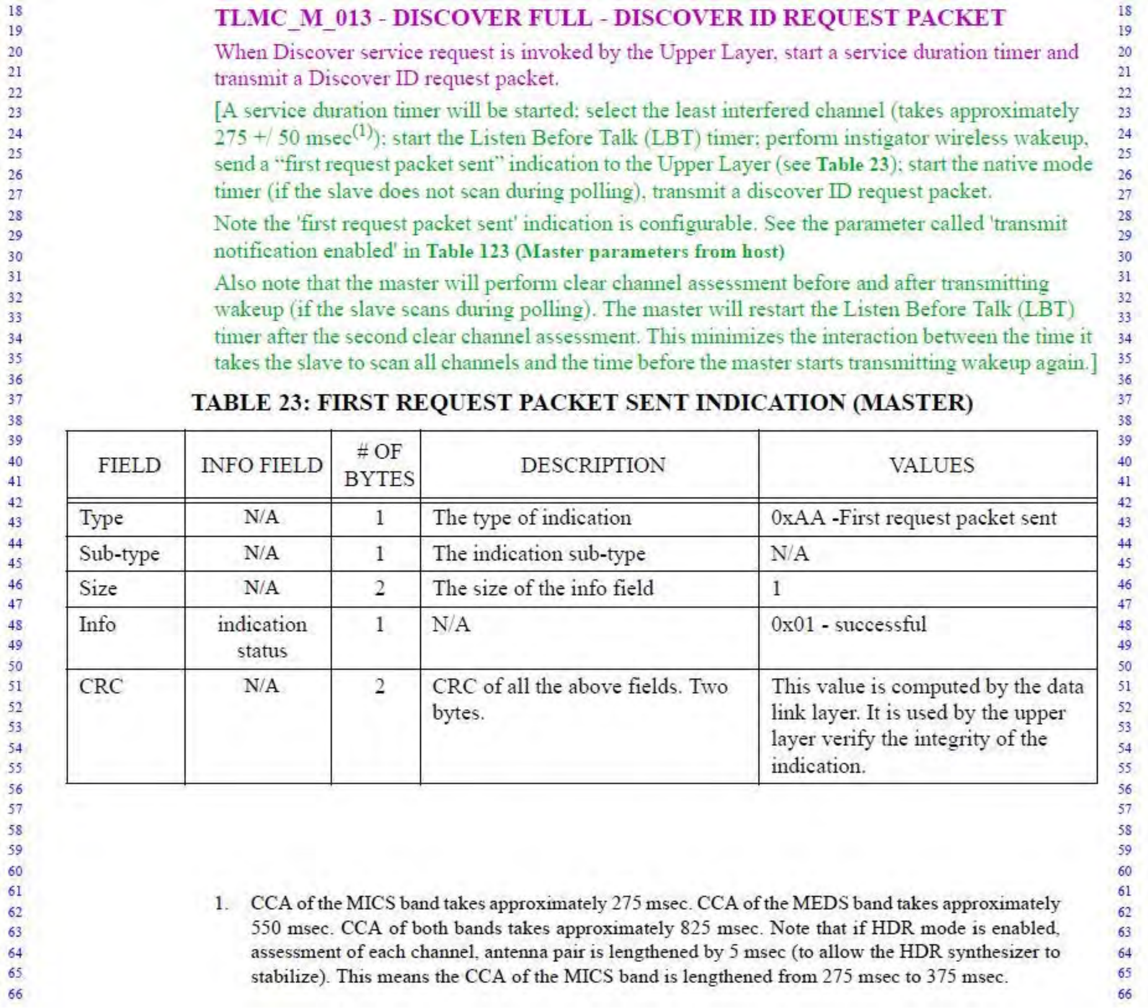
255	M_TCP_Discover	m_DiscoverFull003	\\data\pace\fwvt\mozart\F
256	M_TCP_Discover	m_DiscoverFull004	\\data\pace\fwvt\mozart\F
257	M_TCP_Discover	m_DiscoverFull005	\\data\pace\fwvt\mozart\F
258	M_TCP_Discover	m_DiscoverFull006	\\data\pace\fwvt\mozart\F
259	M_TCP_Discover	m_DiscoverFull007	\\data\pace\fwvt\mozart\F
260	M_TCP_Discover	m_DiscoverFull009	\\data\pace\fwvt\mozart\F
261	M_TCP_Discover	m_DiscoverFull010	\\data\pace\fwvt\mozart\F

Mapping test name to Requirement (from DNS001257.pdf):



TRACING	FWVT Tested	TLMC_M_013 - DISCOVER FULL - DISCOVER ID REQUEST PACKET	Status: **Up-to-date** Rev: 2.3 Test: \M_TCP_Discover\m_DiscoverFull006\m_discoverfull006 Status: **Up-to-date** Rev: 2.7 Test: \M_TCP_Discover\m_DiscoverFull001\m_discoverfull001

Requirement (from telM_protocol_C_A36244.pdf



TLMC_M_013 - DISCOVER FULL - DISCOVER ID REQUEST PACKET

When Discover service request is invoked by the Upper Layer, start a service duration timer and transmit a Discover ID request packet.

[A service duration timer will be started; select the least interfered channel (takes approximately 275 +/ 50 msec(1)); start the Listen Before Talk (LBT) timer; perform instigator wireless wakeup, send a "first request packet sent" indication to the Upper Layer (see **Table 23**); start the native mode timer (if the slave does not scan during polling), transmit a discover ID request packet.

Note the 'first request packet sent' indication is configurable. See the parameter called 'transmit notification enabled' in **Table 123 (Master parameters from host)**

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TABLE 23: FIRST REQUEST PACKET SENT INDICATION (MASTER)

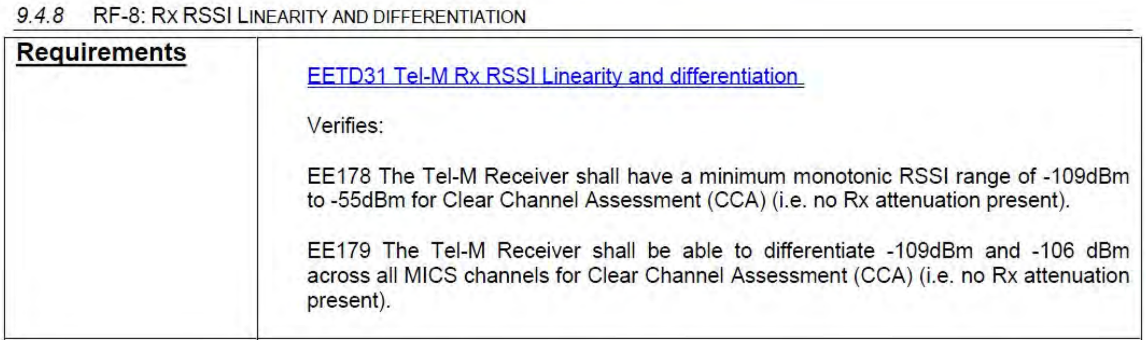
FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xAA -First request packet sent
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	The size of the info field	1
Info	indication status	1	N/A	0x01 - successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

---

1. CCA of the MICS band takes approximately 275 msec. CCA of the MEDS band takes approximately 550 msec. CCA of both bands takes approximately 825 msec. Note that if HDR mode is enabled, assessment of each channel, antenna pair is lengthened by 5 msec (to allow the HDR synthesizer to stabilize). This means the CCA of the MICS band is lengthened from 275 msec to 375 msec.

(3) Requirement: Based on use of an isotropic monitoring system antenna, the monitoring threshold power level must not be more than 10logB(Hz) −150 (dBm/Hz) + G(dBi), where B is the emission bandwidth of the MedRadio communications session transmitter having the widest emission and G is the MedRadio programmer/control transmitter monitoring system antenna gain relative to an isotropic antenna. For purposes of showing compliance with the above provision, the above calculated threshold power level must be increased or decreased by an amount equal to the monitoring system antenna gain above or below the gain of an isotropic antenna, respectively.

Response: Attached document NDHF1405-124574 CTM2 RF DVT Report.pdf , section 9.4.8 verifies the capabilities of RSSI (received signal strength indicator). The report summarizes that the CTM is capable of discerning different signals that vary by 2 dB down to a level of -112 worst case. This means the radio can tell the difference between a signal at -112 dBm and one at -110 dBm at worst. This was testing was done over all 10 channels we use in the MICS band as well as over temperature at each of those channels.

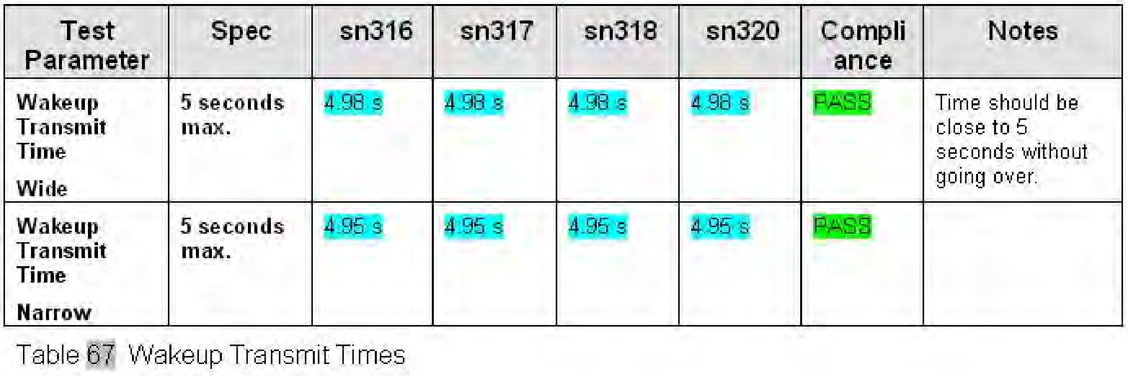


9.4.8 RF-8: Rx RSSI Linearity and Differentiation

Requirements	EETD31 Tel-M Rx RSSI Linearity and differentiation Verifies: EE178 The Tel-M Receiver shall have a minimum monotonic RSSI range of -109dBm to -55dBm for Clear Channel Assessment (CCA) (i.e. no Rx attenuation present). EE179 The Tel-M Receiver shall be able to differentiate -109dBm and -106 dBm across all MICS channels for Clear Channel Assessment (CCA) (i.e. no Rx attenuation present).

(4) Requirement: If no signal in a MedRadio channel above the monitoring threshold power level is detected, the MedRadio programmer/control transmitter may initiate a MedRadio-communications session involving transmissions to and from a medical implant or medical body-worn device on that channel. The MedRadio communications session may continue as long as any silent period between consecutive data transmission bursts does not exceed 5 seconds. If a channel meeting the criteria in paragraph (a)(3) of this section is unavailable, MedRadio transmitters that are capable of operating on multiple channels may transmit on the alternate channel accessible by the device with the lowest monitored ambient power level. Except as provided in paragraph (b) of this section, MedRadio transmitters that operate on a single channel and thus do not have the capability of operating on alternate channels may not transmit unless no signal on the single channel of operation exceeds the monitoring threshold power level.

Response:
Section 11.2 of Mozart_Production_Analog_DVT_Report_138088_v33_20Apr2011.doc



Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Wakeup Transmit Time Wide	5 seconds max.	4.98 s	4.98 s	4.98 s	4.98 s	PASS	Time should be close to 5 seconds without going over.
Wakeup Transmit Time Narrow	5 seconds max.	4.95 s	4.95 s	4.95 s	4.95 s	PASS	

Table 67 Wakeup Transmit Times

(5) Requirement: When a channel is selected prior to a MedRadio communications session, it is permissible to select an alternate channel for use if communications are interrupted, provided that the alternate channel selected is the next best choice using the above criteria. The alternate channel may be accessed in the event a communications session is interrupted by interference. The following criteria must be met:

(i) Before transmitting on the alternate channel, the channel must be monitored for a period of at least 10 milliseconds.

(ii) The detected power level during this 10 millisecond or greater monitoring period must be no higher than 6dB above the power level detected when the channel was chosen as the alternate channel.

(iii) In the event that this alternate channel provision is not used by the MedRadio system or if the criteria in paragraphs (a)(5)(i) and (ii) are not met, a channel must be selected using the access criteria specified in paragraphs (a)(1) through (a)(4) of this section."

Response: N/A, Tel M does not use alternate channel

Mozart Production Analog DVT Report

Change History

Issue	Date	Description of Change
V1	20 Aug 2010	First draft from Alan Mathieu – not reviewed
V2	8 Sep 2010	RSSI section 5.11 updated per ECO 76
V3	17 Sep 2010	Test results added for the week of Sep 17.
V4	23 Sep 2010	Test results added for part of week of Sep 24. Added section for DVT plan deviations. Added Deviation #1. Added text under Regulatory Tests to allow DVT to progress with regulatory testing.
V5	27 Sep 2010	Test results added for week ending Sep. 24.
V6	27 Sep 2010	Test results added for part of week ending Oct. 1. Added Deviation #2.
V7	01 Oct 2010	Test results added for week ending Oct. 1.
V8	11 Oct 2010	Test results added for part of week ending Oct. 15. Minor update to Deviation #2 and added Deviation #3.
V9	15 Oct 2010	Test results added for week ending Oct. 15.
V10	19 Oct 2010	Removed complex impedance data in preparation for re-run. Added test results up to Oct. 19.
V11	21 Oct 2010	Completed actions from Oct.19 review. Added test results up to Oct. 21
V12	29 Oct 2010	Updated data in External Spurious Response Table 24. Added data integrity and completeness section. Added test results up to Oct. 29
V13	05 Nov 2010	Added results up to week ending Nov. 5
V14	12 Nov 2010	Added results up to week ending Nov. 12
V15	17 Nov 2010	Test results added for part of week ending Nov. 19. Added Deviation #6 and Deviation #13.
V16	17 Nov 2010	Updated procedure steps under section 5.4 Deviation #6 “Summary” to have numbered steps instead of lettered.
V17	21 Nov 2010	Added results up to week ending Nov. 19
V18	07 Dec 2010	Test results added for part of week ending Dec. 10. Added Deviation #7 and Deviation #14.
V19	10 Dec 2010	Corrected results for previously verified tables. Added results for week ending Dec. 10.
V20	19 Dec 2010	Added results for week ending Dec. 17.
V21	4 Jan 2011	Added results for Jan 4, 2011
V22	12 Jan 2011	Deviation #10 added.
V23	19 Jan 2011	Removed Deviation #10 (is now represented in Bench DVT Report), Revised WU Dynamic Range results per Action Tracker #142396. Added Deviation #4, updated revised deviation #7 and added deviation #12.
V24	23 Jan 2011	Added results for Jan 23, 2011
V25	01 Feb 2011	Updated the text in section “2.0 Scope” to clarify that this report only covers the Analog DVT part of Mozart DVT results. Added text to section “6.0 DVT Plan Deviations” to clarify that deviations detailed in this report only covers Analog DVT and added here a reference to the Mozart DVT Plan Deviations spreadsheet for a full summary of all deviations to the Mozart DVT plan. Added reference [4] Mozart DVT Plan Deviations to References table. Test results added for of week of Feb 01. Added Deviation #18.

V26	03Feb 2011	Test Results added for week of Feb 03. Updated Deviation #4 and Deviation #12 due to test updates or finding during data reviews.
V27	04Feb2011	Added results for Feb. 4, 2011
V28	07Feb2011	Added results for Feb. 7, 2011 Added Deviation #19.
V29	22Feb2011	Added results for Feb. 9, 2011 Updated Deviation #12.
V30	28Feb2011	Added results for Feb. 24, 2011 Minor clarification to deviation #12, table 1e. Added Deviation #20.
V31	01Mar2011	Updated per review held on 01-Mar-2011. Added note, DVT plan variance, to section 11.8 Wakeup External Spurious Response and Blocking and section 9.6 External Spurious Response Rejection (TelM and TelC) regarding “overtesting” compared to DVT plan. Updated version numbers for [1] and [4] in the reference table
V32	30Mar2011	Updated according to feedback from Medtronic received in an e-mail from John Pohl on March 28th, 2011 (subject: “Feedback on DVT Reports”).
V33	15Apr2011	Updated acronym table based on feedback from Medtronic received in an email from John Pohl on March 31st, 2011. Added section 3.4 that defines the location of raw test results. Updated Approvals list based on input from MDT/John Pohl received in email on April 19th 2011 (subject: “Names on reports/L1.L2”). Updated section “Mozart Inductor Change” based on updated test plan, acceptance criteria and results.

Table of Contents

Definitions, Abbreviations and Acronyms

Abbreviation	Meaning	Description
ACR	Adjacent Channel Rejection	Metric for how well a receiver can receive a transmitted signal in the presence of a high-level interfering signal in the adjacent channel.
ACPR	Adjacent Channel Power Ratio	The ratio between total power of the adjacent channel to the main channel's power.
AltCR	Alternate Channel Rejection	Metric for how well a receiver can receive a transmitted signal in the presence of a high-level interfering signal in the alternate channel.
AltCPR	Alternate Channel Power Ratio	The ratio between total power of the alternate channel to the main channel's power.
AM	Amplitude Modulation	In the AM Rejection test, the interferer is an AM signal.
ATE	Automated Test Equipment	Refers to the Catalyst tester used to probe wafers and modules in manufacturing.
CCA	Clear Channel Assesment	Function to determine the current state of use of a wireless medium.
CW	Continuous Wave	An unmodulated signal.
DUT	Device Under Test	Mozart module or IC being tested.
DVT	Design Verification Test	Testing program that is performed to deliver objective, comprehensive testing verifying all product specifications.
ECO	Engineering Change Order	The specification or the implementation is likely to change during development. These changes are referred to as engineering change orders (ECOs)
EVM	Error Vector Magnitude	Metric used to quantify the performance of a digital radio transmitter.
IMR	Intermodulation Rejection	The difference between the fundamental and intermodulation signal levels.
MDR	Missed Detect Rate	Metric for Wake Up sensitivity. Number of wakeup packets not successfully detected divided by the total number of wakeup packets transmitted.
MEDS	MEdical Data Serice	The name of a specification using frequency bands 401-402MHz and 405-406MHz.
MICS	Medical Implant Communication Service	The name of a specification using the frequency band between 402MHz and 405MHz.
Mozart	Program name	Also used to refer to a module.
NB	Narrow Band	Mode of operation in the Mozart radio using 100KHz channel spacing.
PER	Packet Error Rate	Metric for sensitivity. Number of “bad” packets received divided by the total number of transmitted packets. This is after the Reed-Solomon error correction.
TelM	Telemetry M	Medtronic's name for the Mozart radio protocol and system.
WB	Wide Band	Mode of operation in the Mozart radio using 300KHz channel spacing.
WU	Wake-Up	Mode of operation in the Mozart radio.

1 Purpose

This document contains the summary of test and analysis results from the formal Analog DVT (Design Verification Test) on production Mozart Modules per design requirements documented in “Tel M RF Module Requirement Specification” [1], revision I and Engineering Change Orders as documented in “Mozart DVT Plan Deviations” [4]. The summary of data provided in this report provides the required evidence that the design satisfy the design requirements.

2 Scope

This document only covers/reports results referred to as Analog DVT efforts in the “Mozart DVT Plan” [2]. Other DVT efforts, referred to as Bench DVT and Digital DVT, are conducted and reported separately as detailed in the “Mozart DVT Plan” [2].

3 Assumptions

3.1 Devices Used in Production DVT

The module serial numbers and the tests they are assigned to are available in document “Mozart Production DVT Devices” [5].

ATE data on DVT devices: ATE DVT data from 14 Aug 10 has been entered into cdca as “Mozart Proto 3 DVT data from ATE” [7] and “Mozart Proto 3 DVT data from ATE (stdf file)” [8].

3.2 Mozart Inductor Change

The wire diameter for the 43nH inductors (L1 and L2) was changed from 28um to 32um to address potential wire break issue. In order to investigate any potential module performance change related to changing the inductors, the ATE distributions for specific inductor dependant tests was compared between samples of rev006 modules and rev008 modules. For test plan and acceptance criteria see “Mozart 70470 006 vs 008 ATE analysis for inductor change DVT confirmation” [3].

The comparison and shift analysis showed passing results as per the acceptance criteria and no DVT regression is required. The null hypothesis for all tests were rejected and engineer rationale describes all shifts is acceptable. The variances for all tests are essentially equal for rev006 and rev008 and are acceptable. For more inofmration see “Mozart 70470 006 vs 008 ATE analysis for inductor change DVT confirmation RESULT” [11] and “Mozart Module L1 & L2 Inductor Change Statistical Analysis” [12].

3.3 DVT Software Version

DVT started with software Version 5.1 and ended with Version 6.9, software changes were managed and documented in “Mozart DVT Test System” [10]. Each raw test result has the used software version recorded in its logfiles.

3.4 Raw Test Results Location

All raw test results have been copied from the testers to a common directory on the mozart_dvt drive. The results are divided into three directories; RX, TX and SPUR. They correspond to the different type of testers that was used during Analog DVT. The directories are found here:

mozart_dvt\Test Executive Results\Formal DVT

4 Approvals

Name	Title	Company
Marcus Strom	Mozart DVT Lead	Proprietary Company Name Removed.
Alan Mathieu	Mozart DVT Engineer	Proprietary Company Name Removed.
Randy Koploy	Mozart DVT Engineer	Proprietary Company Name Removed.
Guy Delight	Mozart Project Engineer	Proprietary Company Name Removed.
Dan Wiley	Mozart Project Manager	Proprietary Company Name Removed.
Jon Klaren	Mozart DVT Engineer	Proprietary Company Name Removed.
Peter Nanawa	Mozart DVT Engineer	Proprietary Company Name Removed.
Bill Verhoef	Design Verification Test Engineering	Medtronic Inc.
Chuck Farlow	Senior Program Manager	Medtronic Inc.
Quentin Denzene	Analog Hardware Engineering Leader	Medtronic Inc.
George Rosar	Digital Hardware Engineering Leader	Medtronic Inc.
James Henry	Quality Leader	Medtronic Inc.

5 References

Ref	Document #	Title
[1]	112631 v13	Mozart Requirement Specification (MDT Document #A17245); rev I + ECOs
[2]	125759 v20	Mozart DVT Plan
[3]	140098 v2	Mozart 70470 006 vs 008 ATE analysis for inductor change DVT confirmation
[4]	139706 v11	Mozart DVT Plan Deviations
[5]	139732 v1	Mozart Production DVT Devices
[6]	126599 v4	Mozart DVT Test Matrix
[7]	138069 v1	Mozart Proto 3 DVT data from ATE
[8]	138070 v1	Mozart Proto 3 DVT data from ATE (stdf file)
[9]	139681 v1	Measurement comparison of a CISPR Quasi peak detector method and an approximated CISPR peak detector method
[10]	139839 v9	Mozart DVT Test System
[11]	140498 v2	Mozart 70470 006 vs 008 ATE analysis for inductor change DVT confirmation RESULT
[12]	140579 v6	Mozart Module L1 & L2 Inductor Change Statistical Analysis

6 **DVT Plan Deviations**

If deviations to the DVT plan are needed during the execution of formal DVT, the following procedures shall be followed to ensure quality and correctness in the execution:

1) ZLK DVT lead will provide MDT DVT lead a formal request via email for a deviation including details of the issue, the cause, and the reason for why the deviation is needed and acceptable.

2) Detailed communication and discussion of the deviation will occur between MDT and ZLK. Persons participating in these discussions will be at the discretion of the DVT Leads and based on the nature of the issue.

3) The deviation and associated details will be reviewed and preliminarily approved by MDT DVT lead and communicated to ZLK DVT lead via email. All preliminary approved deviations will be communicated to the Program Managers. The Program Managers may at their discretion request a formal ECO for the deviation.

4) The details of the preliminary approved deviation (details defined by ZLK in a request email identified in step 1, or if needed, additional details defined by MDT in the preliminary approval email) will be documented in the DVT report and sent to MDT. MDT will then formally approve the deviation and this will be noted in the DVT report with an approval date and the document will then be released in eBinder. This process will be a formality but the formal approval from MDT will be described and stated explicitly as an approval, and acknowledged by as such, in an email communication or another written form of communication method deemed acceptable by both DVT Leads.

5) ZLK can (after they have received the formal approval) implement the agreed details of the deviation and execute the needed testing.

6) The new results (if any) will be documented in the DVT report, which is revision controlled in eBinder and is reviewed on a regular basis between ZLK and MDT.

Each deviation to the DVT plan has been assigned a dedicated number. This report only covers deviations related to Analog DVT. For a summary of all deviations to the “Mozart DVT plan [2], please see the spreadsheet Mozart DVT Plan Deviations [4].

Some deviations describe new or current text in the Mozart DVT plan. Note that references to references in such text may <u>not</u> match the reference table in this document since they are intended to match the reference table in the Mozart DVT plan document.

6.1 Deviation #1

Reason: The DVT Plan was not followed but results are OK to be used for a specific purpose.

Formal Approval Date: 9/23/2010

Summary:

The first run of formal DVT Tx Stability data (Section 6.3.5) incorrectly used 1MHz RBW setting for frequencies above 1GHz. This data was originally taken for Tx Stability review and cannot be used for this task but can be used to determine if it is okay to proceed with Regulatory testing or not. This additional task of reviewing Tx Stability data by Medtronic before proceeding with Regulatory testing is a requirement called out in section 6.3.5 of the Test Plan.

Justification:

TX Stability Testing requires a lower resolution bandwidth (RBW) than the 1 MHz resolution bandwidth used above 1 GHz to take the initial DVT TX Stability data. The lower bandwidth is required to effectively lower the noise floor of the measurement so that there is adequate dynamic range relative to the TX RF Output to allow the test to detect TX instabilities if present. Even though the incorrect resolution bandwidths were used for this initial TX Stability test data, it is still usable for performing the TX Emissions Regulatory Pre-screening Review (Pre-screen Review) also called out in the plan. This is allowed due to the lower dynamic range requirements for the Pre-screen Review relative to the TX Stability Testing (the Pre-screen Review requires only a -57 dBm or less measurement above 1 GHz). This initial TX Stability Data will therefore be used to perform the TX Emissions Regulatory Pre-screening Review as called out in the DVT Test Plan.

6.2 Deviation #2

Reason: The DVT Plan can be relieved since results will still be OK to verify specification.

Formal Approval Date: 9/28/2010

Summary:

The 10 KHz RBW setting in section 6.3.5 of the Test Plan for the frequencies both below and above 1GHz can be relieved to 30 KHz RBW. This will still give Medtronic the data dynamic range they need to verify the Tx Stability specification while improving the test time for the test duration. Additionally, this section for the TX Stability Test did not originally call out a maximum frequency step size for the spectrum data. It will be 300 KHz or less for the frequencies both below and above 1GHz. The use of a Peak detector with Max Hold for 10 sweep averages is now also explicitly identified. This deviation requires a full run of the Tx Stability tests over all the vectors using these new settings, in order to review the data for Tx Stability requirements.

Justification:

TX Stability Testing measurements rely on a minimum dynamic range to be able to discern TX instabilities, if present, from the Tel M RF Module. The resolution bandwidth (RBW) of the spectrum analyzer making this measurement directly affects the measurement noise floor. Choosing a narrower resolution bandwidth reduces the noise floor. Note that the narrower the resolution bandwidth, the slower the measurement, and the longer the test time. For very large frequency sweeps, the test time can be prohibitively long if too narrow a RBW is used. An equivalent spectrum analyzer noise floor of -63 dBm, referenced to the properly terminated DUT RF Port, was determined to be acceptable for making effective TX Stability assessments. This level is based on: estimations of potential spurious oscillation level at the TX power amplifier output, the bandwidth of the potential oscillation, and the out-of-band SAW filter amplitude response. A RBW of 30 KHz meets the required noise-floor for this the TX Stability Test, while also allowing for practical test execution times. Thus 30 KHz RBW will be used for the 50 KHz to 1 GHz, and the 1 - 20 GHz measurement ranges for the TX Stability Test. A maximum frequency step size of 300 KHz is now being added to ensure that potential spurious oscillations, which are not generally frequency-locked, nor of such narrow bandwidth as to avoid detection in the 300 KHz frequency step size, will be detected. Additionally, the use of a Peak detector with Max Hold for 10 sweep averages is be added to capture any time varying spectrum sufficiently above the noise floor.

6.3 **Deviation #3**

Reason: An approved ECO modifies specification limits and is referenced in the Test Plan. The current DVT Plan does not reflect these changes and must be adjusted accordingly.

Formal Approval Date: 10/12/2010

Summary:

Section 6.3.2 Transmit Modulation Response in the DVT plan contains limits from the Mozart Requirement Specification (MDT Document #A17245); rev I but the applicable parts of the approved ECO #067 change these limits. ECO #067 is referenced in the Test Plan (see Reference section of the Test Plan) and so must properly be reflected in Section 6.3.2. See the Table below for a summary of the updated limits. The DVT Test has been modified to test to the limits in ECO #067 and the Formal DVT Report will report the results accordingly. Please see ECO #067 for more information. Here below is a new an updated table to replace table 12 in the DVT plan.

Measurement	min	max	Units	Note
Data Rate for Mode 1	48.71795 (-100ppm)	48.71795 (+100ppm)	kb/s	Should be within +/-12ppm over temperature/voltage
Data Rate for Mode 2	97.43590 (-100ppm)	97.43590 (+100ppm)	kb/s	
Data Rate for Mode 3	190 (-100ppm)	190 (+100ppm)	kb/s	
Data Rate for Mode 4	380 (-100ppm)	380 (+100ppm)	kb/s	
EVM modes 1-4	0	8.4	%	-4 to -30 dBm MICS band
EVM modes 1-4	0	12.6	%	< -30 dBm MICS band
EVM modes 1-4	0	8.4	%	-7 to -30 dBm MEDS band
EVM modes 1-4	0	12.6	%	< -30 dBm MEDS band

Justification:

An approved ECO is a specification change and the DVT plan needed to be adjusted for updates to the specification.

6.4 Deviation #4

Reason: The DVT Plan Section 6.4.13.3 RSSI Response Time contains power levels indicated for RF Signal Generator waveforms #1 and #2. ECO 090 modifies those power limits. The DVT Plan must update the test procedure accordingly. Additionally, updates were made to the test signal modulation parameters to align with radio regulatory requirements. The test method has also been updated to make the initiation of the RF generator waveforms to be asynchronous with the start of the CCA measurement in order to mimic more closely the function of the product application and the test method at the regulatory test agencies.

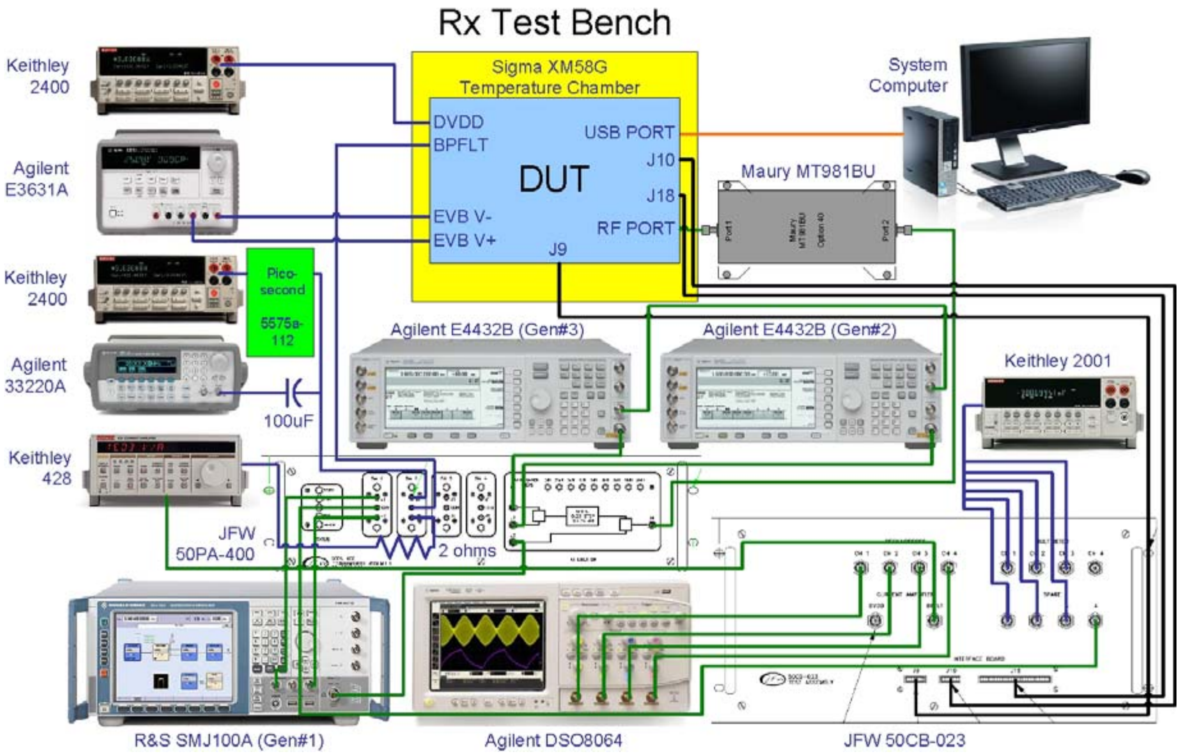
Additionally, an update will be made to the Mozart DVT Test Matrix [6] to indicate the testing all 30 channels for Mode 2 (Narrowband includes 10 MICS and 20 MEDS) and the 10 MICS channels for Mode 4 (Wideband) rather than the currently indicated single channel. The “tested channel” in this instance is the channel that has the Generator waveform #1 signal applied.

As a result of these test plan changes and waveform updates, a change to the Rx Test System must be made to route a signal from Generator #3 to Generator #2. This will be a permanent update to this tester and will not affect other DVT tests run on this tester. There is an update to the Rx test system block diagram in Section 3.2.1, Figure 1, to reflect this change.

Formal Approval Date: 02/03/2011

Summary:

Update to 3.2.1 Figure 1 Receiver Test System Hardware Diagram:



Previous wording of Section 6.4.13.3 in the DVT Test Plan:

Test Module: RSSI Response Test

Generator #1 shall generate a pulse of modulated RF 100us long. The pulse can be essentially a square pulse, meaning it does not have to have a sine squared or some other shape. The modulation type shall be FSK with 10kHz deviation and data rate of 4.7kbps. The data shall be alternating 1 and 0 (10101010). The level of the pulse shall be –97dBm.
Generator #2 shall output an FSK signal with 10kHz deviation and 9.5kbps data rate. The data pattern shall be alternating 1 and 0 (10101010). The level of the signal shall be –100dBm.

Procedure:

1. Set Generator #1 to the channel to be measured for an RSSI pulse.
2. Initiate CCA.
3. Wait for the dwell time plus 1ms.
4. Trigger Generator #1 to output one RF pulse on one channel as described in the set-up within 9ms of the start of the CCA. The pulse must occur within the 10ms CCA window.
5. Read reg_wu_rdf_value and record the RSSI level.
6. Enable Generator #2 on all other channels one at a time.
7. Initiate CCA.
8. Wait for the dwell time plus 1ms.
9. Read reg_wu_rdf_value and record the RSSI level.
10. Turn off Generator #2.
11. Repeat steps 1 – 11 for all channels and modes and any other condition in the vector Matrix [2], filling up a table of data.
12. Analyze the data and check that all the pulse results are higher than any results from Generator #2.

Updated wording of Section 6.4.13.3 in the DVT Test Plan (Replacing the text above):

Test Module: RSSI Response Test

RF Generator waveform #1 shall output a 2FSK RF signal with 20 kHz deviation and 2.0 kbps data rate. The data pattern shall be a 1010 data pattern as defined and made available by the Agilent E443x Series RF signal generators. The power level of this signal shall be -83.75 dBm for the narrowband mode, and -78.75 dBm when in the wideband mode. Power levels will be defined with all modulations off (CW).

Generator waveform #2 shall be defined as a repetitive 100us pulse of FM modulated RF with a 10 mS period. The Frequency Modulation of this waveform shall be identical to Generator waveform #1 indicated above. The RF pulse envelope can be essentially a square pulse, meaning it does not have to have a sine squared or some other shape. The power levels of the pulsed RF signals shall be -81 dBm for the narrowband mode, and -76 dBm when in the wideband mode. Power levels will be defined with all modulations off (CW).

Procedure:

1. Set Generator waveform #1 to the center frequency of the channel to be measured.
2. Enable Generator waveform #1 to output the defined 2FSK waveform.
3. Wait 10 mS.

4. Initiate a single channel CCA on the channel to be measured. The initiation of this CCA measurement is to be asynchronous with the enabled RF waveform.
5. Wait for the dwell time plus 1ms.
6. Read reg_wu_rdf_value and record the RSSI level.
7. Turn off Generator waveform #1.
8. Enable Generator waveform #2 to output the defined 2FSK waveform gated with Pulse Modulation. Set this waveform on the center frequency of all other appropriate channels (based on the Generator waveform #1 channel, tested against all other appropriate channels specific to a MICS or MEDS band only CCA) testing one channel at a time.
9. Wait 10 mS.
10. Initiate a single channel CCA on the channel to be measured. The initiation of this CCA measurement is to be asynchronous with the enabled RF waveform.
11. Wait for the dwell time plus 1ms.
12. Read reg_wu_rdf_value and record the RSSI level.
13. Repeat steps 8 – 12 until all other MICS or MEDS band appropriate channels have the reg_wu_rdf_value measured.
14. Turn off Generator #2.
15. Analyze the data and check that all waveform #2 results are higher than the result from waveform #1. This comparison is based on channels specific to the MICS or MEDS only CCA.
16. Repeat steps 1 – 15, repeating the measurement so that Generator waveform #1 is applied to all channels using all combination of conditions identified in the vector Matrix [2], filling up a table of data.

Justification:

1) The DVT plan needs to be adjusted for updates to the specification per ECO 090.
2) The new RF modulation scheme is more representative of CCA use in the product application, and is better aligned with radio regulatory test methods.
3) The test should include the “real world” possibility that the CCA sample window starts somewhere within or outside the Generator #1 pulse and ends somewhere within or outside the pulse. This relative timing between the pulse and the CCA window can potentially produce different results and should be part of the test.
4) Vector updates to add channels were needed as this was more representative of the radio regulatory test methods while still meeting the intent of the test to show compliance to the requirement.
5) The update to the tester is to add a signal path that was not there before. This path is new and not a modification of a current Rx tester signal path. This feature and path is accessed for only this test and does not affect the function of the other tests run on this tester.

6.5 **Deviation #6**

Reason: An approved ECO modifies the specification from that referenced in the DVT Plan. The current DVT Plan does not reflect these changes and must be adjusted accordingly.

Formal Approval Date: 11/17/2010

Summary:

ECO 079 changes the measurement bandwidth and detector type for the measurement of emissions above 1 GHz in the Mozart Requirement Specification (MDT Document #A17245);

rev I. The previously specified CISPR 16 / Quasi-Peak detector is changed to a 1 MHz bandwidth peak detector for emissions above 1 GHz in this specification.

It should be noted that Mozart Requirement Specification (MDT Document #A17245); rev I required the use of a CISPR 16 / Quasi-Peak Detector above 1 GHz, while the DVT Test Plan called out a 1 MHz bandwidth detector above 1 GHz (note: peak detection with the 1 MHz detector was not specified in the existing DVT Test Plan). The CISPR 16 detector called out by the Mozart Requirement Specification (MDT Document #A17245); rev I vs. the 1 MHz bandwidth detector called out by the DVT Test Plan was a discrepancy which would have required correction by itself. However, ECO 079 further added the requirements of not only using a 1 MHz bandwidth detector above 1 GHz, but that it also is a “peak detector”.

The update to the DVT Test Plan involves Section 6.4.14 Receiver Conducted Spurious Emissions. This will change the 1 MHz bandwidth detector used for measuring RX conducted spurious emissions above 1 GHz be a “peak detector”.

As part of this deviation, there are also corrections to the number of “points” measured, and the addition of “Peak Detection with 10 Trace Averaging” to Table 39.

Additionally, there are explanations/justifications referenced for alternate measurement techniques which are used in both the existing and proposed test methodology where they appear to disagree with the Mozart Requirement Specification (MDT Document #A17245); rev I.

The updated procedure will be as follows:

1. Connect RF input to Spectrum Analyzer.
2. Set the Spectrum Analyzer for Peak Detection, 0 dB attenuation and RBW and VBW ratio to 1:1 (auto), and turn on trace averaging over 10 sweeps
3. Set Center Frequency, Frequency Span, and number of measurement points in order to achieve the settings called out in Table 39.
4. Set up DUT for receive always with the setup file.
5. Start TelM Rx.
6. If other modes are called for in the Test Vector Matrix [2], load setup files and start the reception the same way it is done in the Rx Sensitivity test for the required mode.
7. Measure powers across the spectrum specified in the Limits table and store all data. Record frequencies and power exceeding spec.
8. Repeat for all conditions in the test vector Matrix [2]

This table shall replace table 39 Limits - Rx Conducted Spurious Emissions in the DVT plan:

Measurement	**min**	**max**	**Units**	**Note**
Power from 9kHz to 2 MHz		-57	dBm	1kHz RBW (min of 2201 points)
Power from 2 MHz to 1 GHz		-57	dBm	100kHz RBW (min of 15201 points)
Power 1 GHz to 4.06 GHz		-57	dBm	1MHz RBW (min of 6,001 points)

Justification:

1) Please note that a 100 KHz bandwidth peak detector was specified instead of the CISPR 16/quasi-peak detector for measurements at 1 GHz and below as called out in the Mozart Requirement Specification (MDT Document #A17245); rev I. This substitution was done to speed up test time by a factor of 2x to 3x, knowing that a 100 KHz bandwidth peak detector always provides measurements at an indicated level greater than, or equal to, a CISPR 16/quasi-peak detector. The 120 KHz bandwidth of the CISPR quasi-peak detector was also shown to be essentially equivalent to the 100 KHz peak detector. Section D1.0 of Agilent Application Note AN 1302: Making Radiated and Conducted Compliance Measurements with EMI Receivers specifically calls out using a peak detector as a valid approach to speeding up EMI measurements which call out a quasi-peak detector. This is not a change to test methodology as it's the same for both current/old and proposed/new test methodologies, but is merely mentioned here for completeness.

2) As seen from Table 1 below, data acquired below 2 MHz does not use a CISPR 16 detector as called out in the Mozart Requirement Specification (MDT Document #A17245); rev I. This is because at very low frequencies, the spectrum analyzer has an issue with detecting its own DC component, which leads to erroneous data for some wider resolution bandwidths. A narrower, 1000 Hz resolution bandwidth was chosen at lower frequencies to eliminate this issue. This is not a change to test methodology as it's the same for both current/old and proposed/new test methodologies, but is merely mentioned here for completeness.

3) The original test method was not specific enough to provide overlapping measurement bandwidths (e.g. the frequency span divided by the number of points was less than the RBW of the measurement system). This allowed frequency "gaps" in the data. This will be corrected per this deviation (e.g. the "span between steps" is less than the RBW of the measurement). See the update to Table 39 and the modified Test Procedure.

4) The number of measurement "points" called out in Table 39 was changed in the updated Table 39 to reflect the number of points actually needing to be measured for the specific RBW used.

    a) For frequencies from 9 kHz to 1 GHz, the original Table 39 called out 10000 points. The ECO updates to the requirements and also the need to have overlapping measurement bandwidth (See # 3 above) for complete test coverage. The Test Plan will be updated to call out a minimum of 2201 points from 9 KHz to 2 MHz, and minimum of 15201 points from 2 MHz to 1 GHz.

    b) For frequencies from 1 GHz to 4.06 GHz, the original Table 39 called out 3060 points. The ECO updates to the requirements and also the need to have overlapping measurement bandwidth (See # 3 above) for complete test coverage. The Test Plan will be updated to call out a minimum of 6001 points from 1 GHz to 4.06 GHz.

5) The "Max Hold" function was previously used to process acquired spectrum analyzer data for this measurement. "Max Hold" is not required and will therefore be removed.

Peak Detection with 10-Trace Averaging will be applied instead per this deviation. See the updated Test Procedure above.

6.6 Deviation #7

Reason: The DVT Plan, section 6.3.4 calls out the Mozart Requirement Specification (MDT Document #A17245); rev I section 2.3.4.6.4 to be the guidance for detailed test methodology. When reviewing this section and comparing it to the test conditions outlined in the DVT test Matrix, it does not specify enough output power levels, using these methods, to fully cover the conditions required to test the module requirements in the Mozart Requirement Specification (MDT Document #A17245); rev I. This deviation details the missing vectors and justifies the need for them.

Formal Approval Date: 01/19/2011

Summary:

To fully and accurately test the Mozart Requirement Specification (MDT Document #A17245); rev I regarding MEDS into MICS requirements, the DVT test matrix need 5 additional output power level vectors applied to the appropriate tests outlined below.

1. Under section 2.3.4.6.4.2.3.2 MEDS into MICS (FCC) in Mozart Requirement Specification (MDT Document #A17245); rev I (last row of table 18) it states TX output power set to at least -20dBm in channel 11-19 or 22-30 the module shall meet -57.2dBm maximum emission power in the neighboring MICS channel using Marker-Delta-Marker method. Higher powers are allowed (such as -15dBm in this example) to be used as an acceptable worst case substitute for lower powers as long as the requirement is met. This was done because it met the intent of the test and reduced the number of tests needed. In this case a power closer to -20 dBm is needed. By adding an additional trimmed output power level test vector at -19.4 dBm for a 50 ohm antenna match (with adjustments for different antenna match relaxations identified in table 13 of the Mozart Requirement Specification (MDT Document #A17245); rev I) we will be able to satisfy the conditions needed to test this requirement. This will create new DVT Test Matrix test conditions that will be similar to rows 800-806, 864-870, 914-920, 964-970 and 1014-1020 currently in the DVT Test Matrix.

   Within that same section 2.3.4.6.4.2.3.2 it also states TX output power set to at least -51.8 dBm in channel 20 or 21 the module shall meet -52.2dBm maximum emission measured power in the neighboring MICS channel using Marker-Delta-Marker method. By adding a test vector at -52 dBm for a 50 ohm antenna match (with adjustments for different antenna match relaxations identified in table 13 of the Mozart Requirement Specification (MDT Document #A17245); rev I) we will be able to satisfy the conditions needed to test this requirement. This new test vector will be identical to the rows 821-827, 885-891, 935-941, 985-991 and 1035-1041 in the DVT test Matrix with the difference of used Tx power level.

   For the changes in section 1, the difference is the use of the new -19.4 and -52 dBm Tx power levels. The rows indicated contain vectors for the *Regulatory_TX_Harmonics_Spurious* test in the DVT software. The affected TE scripts that will need to have added updates for the additional power levels are Tx_Sc15a, Tx_Sc15b, Tx_Sc16, Tx_Sc17, Tx_Sc18, Tx_Sc19 and Tx_Sc20.

2. Under section 2.3.4.6.4.2.3.3 MEDS into MICS (ETSI) in Mozart Requirement Specification (MDT Document #A17245); rev I (last row of table 20) it states TX output power set to at least -20dBm in channel 11-19 or 22-30 the module shall meet -59.85dBm maximum emission measured power in the neighboring MICS channel using a test methodology consistent with CEPT/ERC/RECOMMENDATION 74-01E. Higher powers are allowed (such as -15dBm in this example) to be used as an acceptable worst case substitute for lower powers as long as the requirement is met. This was done because it met the intent of the test and reduced the number of tests needed. In this case a power closer to -20 dBm is needed. By adding an additional trimmed output power level test vector at -19.4 dBm for a 50 ohm antenna match (with adjustments for different antenna match relaxations identified in table 13 of the Mozart Requirement Specification (MDT Document #A17245); rev I) we will be able to satisfy the conditions needed to test this requirement. This will create new DVT Test Matrix test conditions that will be similar to rows 800-806, 864-870, 914-920, 964-970 and 1014-1020 currently in the DVT Test Matrix.

   Within that same section 2.3.4.6.4.2.3.3 it also states TX output power set to at least -15 dBm in channel 20 or 21 the module shall meet -34.85 dBm maximum emission measured power in the neighboring MICS channel using a test methodology consistent with CEPT/ERC/RECOMMENDATION 74-01E. By adding a test vector at -15 dBm for a 50 ohm antenna match (with adjustments for different antenna match relaxations identified in table 13 of the Mozart Requirement Specification (MDT Document #A17245); rev I) we will be able to satisfy the conditions needed to test this requirement. This new test vector will be identical to the rows 821-827, 885-891, 935-941, 985-991 and 1035-1041 in the DVT test Matrix with the difference of used Tx power level.

   Also under this section 2.3.4.6.4.2.3.3 MEDS into MICS (ETSI) in Mozart Requirement Specification (MDT Document #A17245); rev I (including text just above table 21 and table 21 itself) we can extract the following requirements. With TX output power set to at least -25dBm in channel 20 or 21 we shall meet -54.85 dBm maximum emission measured power in the neighboring MICS channel using a test methodology consistent with CEPT/ERC/RECOMMENDATION 74-01E. By adding a test vector at -25 dBm (with adjustments for different antenna relaxations) we will be able to satisfy this requirement. This new test vector will be identical to the rows 821-827, 885-891, 935-941, 985-991 and 1035-1041 in the DVT test Matrix with the difference of used Tx power level.

   For these changes, the difference is the use of the new -15, -19.4 and -25 dBm Tx power levels. The rows indicated contain vectors for the *Regulatory_TX_Harmonics_Spurious* test in the DVT software. The affected TE scripts that will need to have added updates for the additional power levels are Tx_Sc15a, Tx_Sc15b, Tx_Sc16, Tx_Sc17, Tx_Sc18, Tx_Sc19 and Tx_Sc20.

Justification:

Without these additional vectors to set the output power as it is required for each of these tests, we cannot fully demonstrate compliance to the requirements in the Mozart Requirement Specification (MDT Document #A17245); rev I section 2.3.4.6.4.

6.7 Deviation #12

Reason: The DVT Plan has several errors in the Wakeup Receptor Section 6.6. The DVT Tests related to this section have been written and run correctly. All Table References to the Mozart Requirement Specification (MDT Document #A17245); rev I in the Test Plan section 6.6 need to be updated to match the Mozart Requirement Specification (MDT Document #A17245); rev I.

Formal Approval Date: 02/23/2011

Summary:
The following changes to the DVT Plan are necessary.

1. The sensitivity levels in DVT Plan sections 6.6.1 - 6.6.4 are incorrect in the Limits tables.

    a. Replace Table 46 and the Note below it in section 6.6.1 with the table and Note below.

MEASUREMENT	min	max	Units	Note
MDR		5	%	Sensitivity Requirement: Narrowband (center channel) @ -103dBm. Dynamic Range: Narrowband (center channel) @ measured sensitivity to measured sensitivity + 3dB
MDR		2	%	Dynamic Range: Narrowband (center channel) @ measured sensitivity + 3dB to -44dBm
MDR		5	%	Sensitivity Requirement: Narrowband (side channels) @ -101.5dBm Dynamic Range: Narrowband (side channels) @ measured sensitivity to measured sensitivity + 3dB
MDR		2	%	Dynamic Range: Narrowband (side channels) @ – measured sensitivity + 3dB to -44dBm
MDR		5	%	Sensitivity Requirement: Wideband (center channel) @ -97.5dBm Dynamic Range: wideband (center channel) @ measured sensitivity to measured sensitivity + 3dB
MDR		2	%	Dynamic Range: wideband (center channel) @ – measured sensitivity + 3dB to -44dBm
MDR		5	%	Sensitivity Requirement: Wideband (side channels) @ -96dBm Dynamic Range: wideband (side channels) @ measured sensitivity to measured sensitivity + 3dB
MDR		2	%	Dynamic Range: wideband (side channels) @ measured sensitivity + 3dB to -44dBm

Note: The signal levels that apply to the limits should be adjusted by the amount of attenuation in the Rx front-end. For example, the 5% MDR applies to Narrowband center channel from -101 dBm to -42 dBm when 2 dB of attenuation is used at the receiver front end.

    b. Replace Table 48 in section 6.6.2 with the table below.

MEASUREMENT	min	max	Units	Note
IMR	44		dB	5% MDR Narrowband (center channel) wanted signal @ measured sensitivity + 3dB
IMR	44		dB	5% MDR Narrowband (side channels) wanted signal @ measured sensitivity + 3dB
IMR	44		dB	5% MDR wideband (center channel) wanted signal @ measured sensitivity + 3dB
IMR	44		dB	5% MDR wideband (side channels) wanted signal @ measured sensitivity + 3dB

c. Replace Table 50 in section 6.6.3 with the table below.

MEASUREMENT	min	max	Units	Note
ACR	25		dB	5% MDR Narrowband (center channel) wanted signal @ measured sensitivity + 3dB
ACR	25		dB	5% MDR Narrowband (side channels) wanted signal @ measured sensitivity + 3dB
ACR	25		dB	5% MDR wideband (center channel) wanted signal @ measured sensitivity + 3dB
ACR	25		dB	5% MDR wideband (side channels) wanted signal @ measured sensitivity + 3dB

d. Replace Table 52 in section 6.6.4 with the table below.

Measurement	min	max	Units	Note
Image Rejection	25		dB	5% MDR Narrowband (side channels) wanted signal @ measured sensitivity + 6dB
Image Rejection	25		dB	5% MDR wideband (side channels) wanted signal @ measured sensitivity + 6dB

Note: This note is added to clarify the difference between Image Rejection and Alternate Channel Rejection (AltCR). For each desired side-channel there are two "alternate" channels, one that is two channels higher in frequency and one that is two channels lower in frequency. In the case of Wakeup Weaver Lower Channel (one below the center channel), the Image Rejection is the AltCR on the upper side. The Alternate Channel Rejection is the AltCR on the lower side and is included in the spurious specification. In the case of Wakeup Weaver Upper Channel (one above the center channel), the Image Rejection is the AltCR on the lower side. The Alternate Channel Rejection therefore is the AltCR on the upper side and is included in the spurious specification.

e. Replace Table 55 in section 6.6.5 with the table below.

Measurement	min	max	Units	Note
Interferer Level NB - Center Channel Center of Alternate channel through Alternate channel + 2	-60		dBm	5% MDR wanted signal @ specified sensitivity + 6dB
Interferer Level NB - Center Channel Alternate channel + 3 and beyond	-55		dBm	5% MDR wanted signal @ specified sensitivity + 6dB
Interferer Level WB - Center Channel Center of Alternate channel and beyond	-55		dBm	5% MDR wanted signal @ specified sensitivity + 6dB

Interferer Level relative to specified sensitivity NB & WB - Side (Weaver) Channels **Channels including 3rd Harmonic of Weaver LO**	20		dB	5% MDR wanted signal @ specified sensitivity + 6dB Includes Alternate channel for weaver side and alternate channel + 2 for image side.
Interferer Level relative to specified sensitivity NB & WB - Side (Weaver) Channels **Alternate channel + 1 and beyond except channels including 3rd harmonic of Weaver LO**	25		dB	5% MDR wanted signal @ specified sensitivity + 6dB

Change section 6.6.5 "Test Executive Script:
Spur_Sc8" to "Test Executive Scripts: Spur_Sc8, Spur_Sc9, Spur_Sc10".

Change section 6.5.5 External spurious response rejection procedure by adding a 4th step as follows:
MEDS (Narrowband mode): Run a "Coarse Spurious" test for the frequency range of 400.1 MHz to 406.9 MHz, excluding the "wanted" channel and 200kHz above and below the center of the wanted channel. The "Coarse" test shall consist of a MDR measurement on the wanted channel with the wanted signal set 6dB above the specified sensitivity. The interferer signal shall be FM modulated with 10.7kHz sine wave and a deviation of 50kHz and set at -48dBm. The test shall take 100kHz steps across the frequency range and record as a spur any frequency that produces a MDR greater than 5%. A subsequent "Fine Spurious" test shall be run using the spur list generated by the "Coarse Spurious" test. The "Fine" test shall be the same as Step 3 in the section 6.5.5 procedures except using the Wakeup spur list generated by this Step 4. The spur list should be examined for spurs that are closer than the "inner" edge of the Alternate+1 channel. Any spur closer should be moved to be 250kHz from the center of the wanted channel so that the "Fine" test does not test closer than 200kHz from the center of the wanted channel. The results of the "Fine" spurious test must show that all 10 frequencies tested within the 100kHz span surrounding each spur at the specified level do not produce a spurious response greater than 5%.

Change section 6.5.5 External spurious response rejection procedure by adding a 5th step as follows:
MICS (wideband mode and narrowband mode): Run a "Coarse Spurious" test for the frequency range of 401.4 MHz to 405.6 MHz, excluding the "wanted" channel and 600kHz above and below the center of the wanted channel. The "Coarse" test shall consist of a MDR measurement on the wanted channel with the wanted signal set 6dB above the specified sensitivity. The interferer signal shall be FM modulated with 10.7kHz sine wave and a deviation of 150kHz and set at -48dBm. The test shall take 300kHz steps across the frequency range and record as a spur any frequency that produces a MDR greater than 5%. A subsequent "Fine Spurious" test shall be run using the spur list generated by the "Coarse Spurious" test. The "Fine" test shall use the Wakeup spur list generated by this Step 5. The spur list should be examined for spurs that are closer than the "inner" edge of the Alternate+1

channel. Any spur closer should be moved to be 750kHz from the center of the wanted channel so that the “Fine” test does not test closer than 600kHz from the center of the wanted channel. The results of the “Fine” spurious test must show that all 10 frequencies tested with 30kHz spacing within the 300kHz span surrounding each spur at the specified level do not produce a spurious response greater than 5%.

Background Note:
The WU External Spurious DVT testing was completed using the spur list developed from the Coarse Spurious Test on the Main Radio. This “coarse” testing skipped the desired channel and 950kHz on either side for narrowband and 750kHz on either side for wideband. The WU spurious testing therefore was not performed on the center WU channel from the Alternate channel out to the frequencies just mentioned. The side “Weaver” WU channels were partially tested in this frequency gap by virtue of the Weaver LO harmonics that were added to the spur list. The alternate channels of the Weaver side channels, however, were tested at the same time as the Image Rejections and therefore had the measured sensitivity as the reference level rather than the specified sensitivity level. Additionally, the alternate + 1 channel was not completely tested for the Weaver side channels. Additional testing for the Weaver side channels, as well as the center channel, is therefore required. Testing intended to cover this gap uncovered specification non-compliance that is now addressed by ECO 094. The tables above contain the specification changes.

2. DVT Plan section 6.6.2 Procedure step 3 incorrectly says to use the specified sensitivity level. Replace with the following: Start sending Wake-up packets from the generator at the measured sensitivity level + 3dB.
3. DVT Plan section 6.6.3 Procedure step 3 incorrectly says to use the specified sensitivity level. Replace with the following: Start sending Wake-up packets from the generator at the measured sensitivity level + 3dB.
4. DVT Plan section 6.6.4 Procedure step 3 incorrectly says to use the specified sensitivity level and does not state how much to raise the wanted signal. Replace with the following: Start sending Wake-up packets from the generator at the measured sensitivity level + 6dB.
5. DVT Plan section 6.6.4, in the paragraph for the Set-up, incorrectly says that “Generator #1 shall be the wanted signal on channel at 3 dB above sensitivity.” The 3dB should be 6dB.
6. DVT Plan section 6.6.5: The 10 kHz intervals should be 30 kHz and the range should be 150 kHz below to 150 kHz above the spur frequency in the spur list. In the paragraph entitled “External spurious response rejection procedure:”, replace step 3 with the following: “For every spur frequency found by the Main Radio external spurious response test, re-test Wakeup MDR at 30kHz intervals from 150kHz below the problem frequency to 150kHz above. The interferer should now be a CW without the 7dB overtest. When the actual spur is found to within 30kHz, find the interferer level at which the MDR passes the spec.” See the Justification section below for an explanation.

   Similarly, in the paragraph entitled “Side (Weaver) Channels Test”, replace step 2 with the following: “For each spur frequency in the spur file, re-test at 30kHz intervals from 150kHz below the spur frequency to 150kHz above. The interferer should now be a CW without the additional 7dB. When an actual spur frequency is found to within 30kHz, find the interferer level at which the MDR passes the spec.
7. Change to section 6.6.1: Replace “Requirement paragraph: 2.3.4.8.4 Table 12 & 14” with “Requirement paragraph: 2.3.4.8.4 Table 26 & 28”.

8. Change to section 6.6.1: Remove procedure step 17. Attenuation vector is not required by the Test Matrix. Dynamic range with attenuation is tested by the main radio modes.
9. Change to section 6.6.2: Replace “Requirement paragraph: 2.3.4.8.4 Table 14” with “Requirement paragraph: 2.3.4.8.4 Table 28”.
10. Change to section 6.6.4: Replace “Requirement paragraph: 2.3.4.8.4 Table 14” with “Requirement paragraph: 2.3.4.8.4 Table 28”.
11. Change to section 6.6.4 Set-up paragraph: Replace “Generator #2 shall have a signal modulated according to the definition in section 6.4.5.1 with data uncorrelated and non-synchronous with the data on generator #1. It shall be 1 channel (200kHz for narrowband, 600kHz for wideband) above (and then below) the channel used by generator #1.” with “Generator #2 shall have a signal modulated according to the definition in section 6.4.5.1 with data uncorrelated and non-synchronous with the data on generator #1. It shall be 2 channels (200kHz for MEDS, 600kHz for MICS) above (and then below) the channel used by generator #1.”
12. Change to section 6.6.1 Procedure step 9: replace step 9 with “Change signal level and repeat MDR test until 5% MDR is reached with ½ dB resolution. Store level as measured sensitivity.”
13. Change to section 6.6.1 Procedure step 16: replace step 16 with “Dynamic Range Test: For wideband and narrowband modes on the worst case channel, measure MDR with a wanted signal level at the measured sensitivity level up to –44dBm in 2dB steps.”

Justifications: The following reasoning is behind the changes in the Summary section above:

1. **a)** The sensitivity levels are specified by the Mozart Requirement Specification (MDT Document #A17245); rev I Table 28.

   b - c) The 3dB increase, above measured sensitivity, to the wanted signal level at which the MDR spec is to be met is not stated in the Mozart Requirement Specification (MDT Document #A17245); rev I but comes from test methodology from TIA 603b section 2.1.6.2 Method of Measurement paragraph c). This standard is referenced in the Mozart Requirement Specification (MDT Document #A17245); rev I in section 2.3.4.5 Receiver Specifications.

   d) The 6dB increase, above measured sensitivity, to the wanted signal level is found in ECO 089.

   e) The DVT Plan is aligned with the Wakeup External Spurious Response requirement, which is defined in Mozart Requirement Specification (MDT Document #A17245); rev I and ECO 094. The procedure changes are needed to completely test the frequency band defined by the Mozart Requirement Specification (MDT Document #A17245); rev I.
2. The Mozart Requirement Specification (MDT Document #A17245); rev I does not state to use “specified sensitivity” as the reference so the default test methodology from TIA 603b applies. See TIA 603b section 2.1.6 paragraph b).
3. The Mozart Requirement Specification (MDT Document #A17245); rev I does not state to use “specified sensitivity” as the reference so the default test methodology from TIA 603b applies. See TIA 603b section 2.1.6 paragraph b).
4. The Mozart Requirement Specification (MDT Document #A17245); rev I does not state to use “specified sensitivity” as the reference so the default test methodology from TIA 603b applies. See TIA 603b section 2.1.6 paragraph b). This Alternate Channel Rejection test is

part of the External Spurious Response Rejection test covering the alternate channel. The wanted signal is therefore raised 6dB above the measured sensitivity. See note 10 below Table 28 in the Mozart Requirement Specification (MDT Document #A17245); rev I.

5. The information in the DVT Plan for setting up the generator should be consistent with the procedure addressed in item #4.
6. The approved DVT Plan section 6.4.8 External Spurious Response Rejection for TelM/TelC Modes was modified before approval by MDT to use 300 kHz coarse steps and 30kHz fine steps to reduce the test time. The deviation of the interferer was correspondingly increased. The DVT Plan section 6.4.8 “External Spurious Response Rejection for TelM/TelC Modes” procedure step 3 for coarse testing states “The interferer should be FM modulated with 10.3kHz and a deviation of 150kHz.” The fine test, described in the same section procedure step 6, states “The interferer should now be a CW without the additional 6dB.” [The approved DVT Plan section 6.6.5 Wakeup External Spurious and Blocking procedure step 3 states that the Wakeup Spurious Fine test will use the spurs found by the “main radio” spurious test. The Fine External Spurious Response Rejection for Wakeup must therefore check across the 300 kHz, not 100 kHz, to find the spur location more closely. The test was written for 10 steps so the step size must be 30 kHz rather than the 10 kHz stated in the DVT Plan. This was originally intended from the vector reduction effort.
7. The Mozart Requirement Specification (MDT Document #A17245); rev I was revised after the DVT Plan was written and the table numbers changed. The tables being referred to have not changed.
8. Attenuation vector is not required by the Test Matrix, which is considered the authority for vectors rather than the DVT Plan document. Dynamic range with attenuation is tested by the main radio modes. This was part of the vector reduction effort made by the combined Medtronic/ team.
9. The Mozart Requirement Specification (MDT Document #A17245); rev I was revised after the DVT Plan was written and the table numbers changed. The tables being referred to have not changed.
10. The Mozart Requirement Specification (MDT Document #A17245); rev I was revised after the DVT Plan was written and the table numbers changed. The tables being referred to have not changed.
11. This change is required because the Test Matrix requires a narrowband test in the MICS band where the offset needs to be 600kHz, not 200kHz. The 200kHz offset only applies in the MEDS band and 600kHz exclusively applies to the MICS band even if a narrowband mode is being tested.
12. This is clarification of what is stored and does not represent a change to the test. The purpose of the test is to measure the sensitivity level.
13. This is a clarification that measured sensitivity, not specified sensitivity, is used for the lowest level of the dynamic range test. The Mozart Requirement Specification (MDT Document #A17245); rev I Table 26 requirement description for the Missed Detect Rate states “The missed detect rate in the presence of AWGN and impulse noise at sensitivity shall be 5%. At levels 3dB above sensitivity the missed detect rate shall be 2%.” The “sensitivity” in this description is taken to mean measured sensitivity although it is not

explicitly stated. The Table 28 entry in the Mozart Requirement Specification (MDT Document #A17245); rev I section “Wakeup Signal Dynamic Range” gives the “specified sensitivity” as the minimum of the range. The actual test software was written to use measured sensitivity as the reference. This part of the deviation provides alignment between the test plan and the test software. This alignment still meets the purpose and intent of the test. The required “specified sensitivity” can be substituted by “measured sensitivity” in the Test Plan, for in all cases, using measured sensitivity is more stringent test method to show compliance to the requirement.

6.8 Deviation #13

Reason: An approved ECO modifies specification limits and those are referenced in the DVT Plan. The current DVT Plan does not reflect these changes and must be adjusted accordingly

Formal Approval Date: 11/17/2010

Summary:

In Section 6.4.13.2 RSSI Linearity Narrow and Wide Band, calls out a "differentiable level" of 2 dB. ECO 82 changes the "differentiable level" for MICS narrowband and wideband and for MEDS narrowband to 2.5dB. The RSSI step size for the test is changed from 1dB to 0.5dB and the limits table (Table 36) is changed from 2 to 2.5dB for the appropriate entries. Change Procedure Step 5 under DVT Plan section 6.4.13.2 to the following: “Increase the RF signal level by 0.5 dB”. Change the Limits of Table 36 to the following:

Measurement	Min	Max	Units	Note
Absolute level (narrowband – MICS – 0dB atten.)	-108	-104	dBm	Absolute level at expected CCA output
Absolute level (narrowband – MEDS – 0dB atten.)	-108	-104	dBm	Absolute level at expected CCA output
Absolute level (narrowband – MICS – **with** atten.)	-112	-104	dBm	Absolute level at expected CCA output
Absolute level (narrowband – MEDS – **with** atten.)	-112	-104	dBm	Absolute level at expected CCA output
Absolute level (wideband – MICS – 0dB atten.)	-103	-99	dBm	Absolute level at expected CCA output
Absolute level (wideband – MICS – **with** atten.)	-107	-99	dBm	Absolute level at expected CCA output
Differentiable Level (MICS -106 to -68 dBm) (no atten.) NB		2.5	dB	
Differentiable Level (MICS -68 to -55 dBm) (no atten.) NB		4	dB	
Differentiable Level (MEDS -106 to -68 dBm) (no atten.) NB		2.5	dB	
Differentiable Level (MEDS -68 to -55 dBm) (no atten.) NB		4	dB	
Differentiable Level (MICS -106 to -68 dBm) (**with** atten.) NB		4	dB	
Differentiable Level (MICS -68 to -55 dBm) (**with** atten.) NB		6	dB	
Differentiable Level (MEDS -106 to -68 dBm) (**with** atten.) NB		6	dB	
Differentiable Level (MEDS -68 to -55 dBm) (**with** atten.) NB		8	dB	
Differentiable Level (MICS -101 to -68 dBm) (no atten.) **WB**		2.5	dB	
Differentiable Level (MICS -68 to -55 dBm) (no atten.) **WB**		4	dB	
Differentiable Level (MICS -101 to -68 dBm) (**with** atten.) **WB**		4	dB	
Differentiable Level (MICS -68 to -55 dBm) (**with** atten.) **WB**		6	dB	

Justification:
An approved ECO is a specification change and the DVT plan needed to be adjusted for updates to the specification.

6.9 Deviation #14

Reason: To remove redundant, and therefore unnecessary, DVT testing related to the Wakeup External Spurious Response Rejection test, and also clarify alignment between the note made in the Test matrix for this test and the one formally stated in the DVT test plan, Table 54.

Formal Approval Date: 12/07/2010

Summary:

The DVT plan under section 6.6.5 Wakeup External Spurious and Blocking, paragraph “Side (Weaver) Channels Test”, indicates that 14 Weaver LO spurs shall be added to each side channel that is tested. In the DVT test Matrix (referenced from the DVT plan), the tests for Wakeup External Spurious Response Rejection (row 1390, 1391, 1393 and 1394 covered by script Spur_Sc8) also indicate in column X, BW and BX that 14 Weaver LO spurs shall be added to each channel that is tested. This deviation shows that testing on all channels is redundant and so it reduces the DVT test requirement to test 14 weaver LO spurs on only one channel in the MEDS band for narrowband wakeup (tested with lower and upper Wakeup hardware channels) and one channel in the MICS band for wideband wakeup (tested with lower and upper Wakeup hardware channels). These channels are channel 5 for MICS Wide band and channel 19 for MEDS Narrow band. These channels were arbitrarily chosen with the one restriction that they could not be a channel at the edge of the band where either the upper or lower Wakeup Weaver channel could not be tested.

Table # 54 in section 6.6.5 of the Test Plan states the specific LO harmonics tested. These are to be centered on the center channel and have LO frequency steps of 100 KHz (MEDS) or 300 KHz (MICS) within a range starting at Alternate +1 channel (3rd harmonic of Weaver LO) and ending at +/-900 KHz for MEDS and +/-2700 KHz for MICS band. This creates a total of 14 extra spurs to add to the spur list for testing side channel performance. The Test Vector Matrix makes similar statements but limits the frequency range to +/-700 KHz for MEDS and +/-2100 KHz on MICS on either side of the side channel frequency. The values should be +/-900 KHz for MEDS and +/-2700 KHz for both documents. With this alignment it meets the intent of the test as already stated correctly in the DVT Test Plan.

Justification:

1. <u>Filtered Interference in the Rx Path</u> - The Weaver mixer, where the Weaver LO is present, is separated from the RF front end by the Weaver Filter. The harmonics of the Weaver LO will only mix with what has already been downconverted and has made it through the filter. Any signal frequencies outside the filter cutoff will be attenuated to a level that removes any effect on spurious performance. This undesired mixing will be the same regardless of which RF channel frequency is used for the testing; therefore it is sufficient to test only one channel for spec compliance in the presence of weaver LO spurs.

2. <u>Unintended Coupling</u> - Other spurious responses that could be created, unrelated to the type of coupling identified in justification #1, requires that the Weaver LO signal couple into the RF mixer or the synthesizer via alternate paths specific to the physical layout of the IC. The revised wake-up side-channel receiver spurious response rejection requirement of 25 dB eliminates unintended coupling of noise on the IC as a cause for unanticipated receiver spurious responses at this level. Coupling of this nature is also

mitigated as the Weaver Receiver LO signal is a low frequency (100 KHz or 300 kHz and harmonics) signal located well away from the synthesizer and RF mixer and is electrically separated from the RF mixer by the Weaver filter. There are other higher frequency and higher power level digital signals associated with the channel select filter in the same physical area of the IC. The main radio spurious response DVT test did not show these digital signals levels to cause spurs.

3. Unintended Weaver LO harmonic coupling to the RF LO - Weaver LO or LO harmonics that do couple to the RF Mixer will cause mixing products at frequencies m*RFLO +/- n*WeaverLO. The largest products will be when m = 1. This case can be tested on any inband channel. The response to the 3rd harmonic frequency of the Weaver LO is tested as part of the Weaver Image Rejection test on 4 MEDS channels and 5 MICS channels. If m = 2, then the range of n when a narrowband Weaver LO (100 kHz) is used is 3961 to 4109 for the product to fall back in band. The range of n when a wideband Weaver LO (300 kHz) is used is 1328 to 1362.

6.10 Deviation #18

Reason: In the DVT plan, section 6.7.4 DVDD Supply Current, the wrong test module and test executive script is called out. There is also an incorrect comment in the notes column in the DVT test Matrix for the same test. Table 67 “Limits - DVDD Supply Current” needs updates to fully cover requirements in Table 3 from section 2.3.1.2 DVDD of the Mozart Requirement Specification (MDT Document #A17245); rev I.

Formal Approval Date: 02/01/2011

Summary:

1. The DVT plan section 6.7.4 DVDD Supply Current shows that the test module “Wake Up Current Latency” will be used to take the measurements using test executive script “Rx_Sc20”. “Wake Up Current Latency” is to be replaced by the test module “Standby Current” for disabled static current measurements and the test module “RSSI linearity” for functional static current measurements. The test modules will be executed from the test executive script “Rx_Sc22”. This will change the Script Name column (column T) in the DVT Test Matrix for row 1432 -1436 to “Rx_Sc22” instead of “Rx_Sc20”.
2. The notes column (column X) in the Mozart DVT Test Matrix for row 1403 – 1414 indicates that the Wake Up Current Latency test will be updated so that it reports DVDD and BPFLT separately (as well as the sum of currents - as it already does). This was never done and is not needed since bullet 1 in this deviation replaces the need for this software change.
3. Table 67 “Limits - DVDD Supply Current” in the Mozart DVT Plan under section 6.7.4 needs updates to fully cover the requirements on DVDD currents when DVDD = BPFLT+200mV (when BPFLT = 1.85V) in the Mozart Requirement Specification (MDT Document #A17245); rev I. Replace table 67 with the following table

	Min	Mean	Max	Note
Functional Static Current with BPFLT = 3.5V and DVDD = 3.5V			**10 uA**	**55C**
Functional Static Current with BPFLT = 2.0V and DVDD = 2.2V			**10.1 uA**	**55C**

Disabled Static Current @25C with BPFLT = 3.5V and DVDD = 3.5V			**50 nA**	
Disabled Static Current @37C with BPFLT = 3.5V and DVDD = 3.5V			**100 nA**	
Disabled Static Current over 0C to 55C with BPFLT = 3.5V and DVDD = 3.5V			**500 nA**	
Disabled Static Current @25C with BPFLT = 2.0V and DVDD = 2.2V			**150 nA**	
Disabled Static Current @37C with BPFLT = 2.0V and DVDD = 2.2V			**200 nA**	
Disabled Static Current over 0C to 55C with BPFLT = 2.0V and DVDD = 2.2V			**600 nA**	

Note: All tests, with the exception of Wake Up Current Latency and Standby Current, are run with a BPFLT voltage representative of the switching supply modelled in the Mozart Requirement Specification (MDT Document #A17245); rev I. The supply voltage waveform is generated by an arbitrary waveform generator and AC coupled into the actual supply circuit. Due to this configuration the BPFLT voltage is constrained to a minimum of 2.0V in the software to ensure that the arbitrary waveform generator never brings the BPFLT voltage below 1.85V. Because of this constraint it was chosen to use BPFLT = 2.0V and DVDD = 2.2V to cover this requirement.

Justification:

1. It was found that the test module “Standby Current” together with the test module “RSSI linearity” is available to take the measurements for DVDD Supply Currents which is equivalent to the planned Wake Up Current Latency test module. Using test executive script Rx_Sc22 instead of the planned Rx_Sc20 is equivalent.
2. Usage of the test module “Standby Current” together with the test module “RSSI linearity” replaces the indicated need for an update to the test module Wake Up Current Latency.
3. The measurements to cover the requirements on DVDD currents when DVDD = BPFLT+200mV (when BPFLT = 1.85V) is missing in Table 67 “Limits - DVDD Supply Current” and had to be added. Because of the constraint in software that BPFLT never can be lower than 2.0V, it was decided to use BPFLT = 2.0V and DVDD = 2.2V to cover this requirement. This is equivalent if not more conservative since higher voltage results in more leakage.

6.11 **Deviation #19**

Reason: The DVT Plan, section 6.3.4 calls out the Mozart Requirement Specification (MDT Document #A17245); rev I section 2.3.4.6.4 to be the guidance for detailed test methodology. When reviewing this section and comparing it to the test conditions outlined in the DVT test Matrix, it does not specify enough vectors, using these methods, to fully cover the conditions required to test the module requirements in the Mozart Requirement Specification (MDT Document #A17245); rev I. This deviation details the missing vectors and justifies the need for them.

Formal Approval Date: 02/07/2011

Summary:

To fully and accurately test the Mozart Requirement Specification (MDT Document #A17245); rev I regarding Unwanted Emissions requirements (ETSI regulations), the DVT test matrix need channel 30 added for 2 of the current tested output power levels applied to the appropriate test outlined below. In #3 and #4 immediately below and subsection Additional Background #1 and #2 below, higher power levels than the requirement may be tested with the same rationale as shown in the notes of section 8.5.3 in the DVT report.

1. Under section 2.3.4.6.4.2.4.2 Implant Mode (ETSI) in Mozart Requirement Specification (MDT Document #A17245); rev I it states TX output power between -7dBm and -15dBm in all MEDS channels shall meet -10.85dBm maximum power (or equal to the peak fundamental power output, whichever is less) in each emission across 25MHz to 401MHz and 406MHz to 4.06GHz.

2. Under section 2.3.4.6.4.2.4.5 External Mode (ETSI) in Mozart Requirement Specification (MDT Document #A17245); rev I (first row of table 23) it states TX output power between -15dBm and -20dBm in all MEDS channels shall meet -30.85dBm maximum unwanted emissions within the 400MHz to 401Mhz, 406MHz to 407MHz, and 800MHz to 814MHz region.

3. Under section 2.3.4.6.4.2.2 In-band Spectral Mask in Mozart Requirement Specification (MDT Document #A17245); rev I it states that for all MEDS channels with output power set to minimum -7dBm the emission bandwidth shall be < 100kHz.

4. Under section 2.3.4.6.4.2.3.1 General FCC (under section 2.3.4.6.4.2.3 Unwanted Radiation) in the Mozart Requirement Specification (MDT Document #A17245); rev I it that states that for all MEDS channels with output power set to minimum -7dBm, the emissions in the immediate out-of-band shall be -21dB maximum with respect to the peak power in the emission bandwidth.

5. Under section 2.3.4.6.4.2.3.2 MEDS into MICS (FCC) in Mozart Requirement Specification (MDT Document #A17245); rev I it states that for all MEDS channels other than the two MEDS channels that neighbor the MICS band, the power in the MICS band shall be -32.2dBm for transmitter power levels between -7dBm and -15dBm. For transmitter output power levels between -15dBm and -20dBm the maximum measured unwanted emission within the MICS band shall be -52.2dBm.

6. Under section 2.3.4.6.4.2.3.3 MEDS into MICS (ETSI) in Mozart Requirement Specification (MDT Document #A17245); rev I it states that for all MEDS channels other than the two MEDS channels that neighbor the MICS band, the power in the MICS band shall be -34.85dBm for transmitter power levels between -7dBm and -15dBm. For transmitter output power levels between -15dBm and -20dBm the maximum measured unwanted emission within the MICS band shall be -54.85dBm.

7. Under section 2.3.4.6.4.2.4.3 External Mode in the Mozart Requirement Specification (MDT Document #A17245); rev I it states that for all MEDS channels and output power levels less than or equal to -15dBm the following list provides the maximum measured level of each emission as a function of frequency band:

    - 25MHz to 30MHz (ETSI): -35.9 dBm
    - 30MHz to 88MHz (FCC): -63.2 dBm
    - 88MHz to 216MHz (FCC): -59.7 dBm

- 216MHz t to 400MHz(FCC): -57.2 dBm
- 407MHz to 800MHz:(FCC): -57.2 dBm
- 814MHz to 960MHz (FCC): -57.2 dBm
- 960MHz to 4.06GHz: -49.2 dBm

The DVT Test Matrix details these tests to be run at channel 11, 19, 22 and 29. These requirements state all MEDS channels so for full coverage of these requirements channel 30 must be considered and therefore channel 30 shall be added to the test vectors for these power levels. Note that this is different than for FCC regulations where channel 30 has a different requirement than other MEDS channels. This will create new DVT Test Matrix test conditions by filling column BJ (corresponds to channel 30) for rows 793-806, 857-870, 907-920, 957-970 and 1007-1020 currently in the DVT Test Matrix.

These changes add the testing of channel 30 at power levels -7dBm and -15dBm. The rows indicated above contain vectors for the *Regulatory_TX_Harmonics_Spurious* test in the DVT software. The affected TE scripts that will need to have added updates for the additional channel are Tx_Sc15a, Tx_Sc15b, Tx_Sc16, Tx_Sc17, Tx_Sc18, Tx_Sc19 and Tx_Sc20.

Additional Background

1. ECO 093 has changed section 2.3.4.6.4.2.4.1 Implant Mode (FCC) in Mozart Requirement Specification (MDT Document #A17245); rev I. It now states TX output power between -7.7dBm and -15.7dBm in channel 30 shall meet -32.2dBm maximum power in each emission across 25MHz to 401MHz and 406MHz to 4.06GHz.
2. ECO 093 has changed section 2.3.4.6.4.2.4.4 External Mode (FCC) in Mozart Requirement Specification (MDT Document #A17245); rev I. It now states TX output power between -27.1dBm and -29.1dBm in channel 30 shall meet -52.2dBm maximum power in each emission across 25MHz to 401MHz and 406MHz to 4.06GHz. Tx output power levels below -29.1dBm in channel 30 shall meet -53.7dBm maximum power in each emission across 25MHz to 401MHz and 406MHz to 4.06GHz.

Justification:
Without these additional vectors to test channel 30, as it is required for each of these tests, we cannot fully demonstrate compliance to the requirements in the Mozart Requirement Specification (MDT Document #A17245); rev I Section 2.3.4.6.4. and ECO 093.

6.12 **Deviation #20**

Reason: ECO 096 modifies specification limits which are referenced in the DVT Plan. The current DVT Plan does not reflect these changes and must be adjusted accordingly.

Formal Approval Date: 02/28/2011

Summary:

DVT Plan section 6.4.5 Intermodulation Rejection for TelM/TelC Modes contains requirement limits from Mozart Requirement Specification (MDT Document #A17245); rev I section 2.3.4.5.4. ECO 096 changes one of these limits; specifically, Tel-M Mode 4 Intermodulation Rejection is reduced 2 dB to 42.3 dB. There is no impact to DVT test methodology. Below is an updated table to replace Table 27 “Limits - TelM/TelC IMR” in the DVT plan.

Measurement	min	max	Units	Note
Mode 1, 3 and 6	47		dB	Rejection
Mode 2	45.3		dB	
Mode 4	42.3		dB	Relaxed with 2dB by ECO 096
Mode 5 nominal deviation	41.6		dB	
Mode 5 low deviation (47 kHz)	40.6		dB	

Justification:
An approved ECO is a specification change and the DVT plan needed to be adjusted for updates to the specification.

7 Data integrity and completeness

Data collected from DVT is formatted into pivot tables and plots to evaluate worst case results which are presented in the tables of this report. Along with the worst case results, the best case results are also plotted in the same pivot charts to confirm data integrity and make sure there are no rogue data points or abnormal looking data. Additionally, the number of tests for each script shown in the “Mozart DVT Test Matrix” [6] is compared against the actual number of tests completed in the script which is reported in the raw .XML file header. This methodology validates data integrity and confirms data completeness for all data presented in the tables of this report.

8 Transmit Test Results

8.1 Output Power

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.2

DVT Plan: 6.3.1.2

Test Parameter	Spec (min)	sn302	sn311	sn313	sn314	Compliance	Script
MICS, Implant, 50 Ohms	**-4 dBm min**	-3.70 dBm	-3.72 dBm	-3.71 dBm	-3.84 dBm	PASS	Tx_Sc2
MICS, Implant, 50-j60 Ohms	**-4.6 dBm min**	-4.18 dBm	-4.50 dBm	-4.33 dBm	-4.47 dBm	PASS	Tx_Sc3
MICS, Implant, 5+j0 Ohms	**-5.9 dBm min**	-5.41 dBm	-5.45 dBm	-5.12 dBm	-5.51 dBm	PASS	Tx_Sc4
MICS, Implant, 5-j20 Ohms	**-6.8 dBm min**	-6.39 dBm	-6.58 dBm	-6.41 dBm	-6.57 dBm	PASS	Tx_Sc6
MICS, Implant, 9-j20 Ohms	**-5.7 dBm min**	-5.16 dBm	-5.13 dBm	-5.08 dBm	-5.17 dBm	PASS	Tx_Sc7
MICS, Implant, 5-j60 Ohms	**-8 dBm min**	-6.72 dBm	-6.84 dBm	-6.76 dBm	-6.83 dBm	PASS	Tx_Sc5
MICS, Implant, 2:1VSWR, 50 Ohms	**-6.1 dBm min**	-4.91 dBm	-4.55 dBm	-4.52 dBm	-4.86 dBm	PASS	Tx_Sc1
MICS, Instrument, 50 Ohms	**-12 dBm min**	-11.76 dBm	-11.75 dBm	-11.67 dBm	-11.52 dBm	PASS	Tx_Sc2
MICS, Instrument, 50-j60 Ohms	**-12.6 dBm min**	-12.13 dBm	-12.29 dBm	-12.22 dBm	-12.48 dBm	PASS	Tx_Sc3
MICS, Instrument, 5+j0 Ohms	**-13.9 dBm min**	-13.38 dBm	-13.41 dBm	-13.18 dBm	-13.44 dBm	PASS	Tx_Sc4
MICS, Instrument, 5-j20 Ohms	**-14.8 dBm min**	-14.30 dBm	-14.42 dBm	-14.34 dBm	-14.51 dBm	PASS	Tx_Sc6

MICS, Instrument, 9-j20 Ohms	**-13.7 dBm min**	-13.14 dBm	-13.06 dBm	-13.00 dBm	-13.10 dBm	PASS	Tx_Sc7
MICS, Instrument, 5-j60 Ohms	**-16 dBm min**	-14.94 dBm	-14.81 dBm	-14.84 dBm	-14.69 dBm	PASS	Tx_Sc5
MICS, Instrument, 2:1VSWR, 50 Ohms	**-14.1 dBm min**	-12.97 dBm	-12.86 dBm	-12.70 dBm	-12.44 dBm	PASS	Tx_Sc1
MICS, Instr/Impl, 2:1VSWR, 50 Ohms	**-32.1 dBm min**	-30.10 dBm	-30.27 dBm	-28.82 dBm	-28.64 dBm	PASS	Tx_Sc1
MICS, Instr/Impl, 50 Ohms	**-52 dBm min**	-49.84 dBm	-50.57 dBm	-49.88 dBm	-49.96 dBm	PASS	Tx_Sc2
MICS, Instr/Impl, 50-j60 Ohms	**-52.6 dBm min**	-50.43 dBm	-50.44 dBm	-50.52 dBm	-50.64 dBm	PASS	Tx_Sc3
MICS, Instr/Impl, 5+j0 Ohms	**-53.9 dBm min**	-51.86 dBm	-52.51 dBm	-50.56 dBm	-50.73 dBm	PASS	Tx_Sc4
MICS, Instr/Impl, 5-j20 Ohms	**-54.8 dBm min**	-52.57 dBm	-52.69 dBm	-52.25 dBm	-52.14 dBm	PASS	Tx_Sc6
MICS, Instr/Impl, 9-j20 Ohms	**-53.7 dBm min**	-52.52 dBm	-53.26 dBm	-52.34 dBm	-50.92 dBm	PASS	Tx_Sc7
MICS, Instr/Impl, 5-j60 Ohms	**-56 dBm min**	-53.85 dBm	-53.34 dBm	-54.24 dBm	-54.44 dBm	PASS	Tx_Sc5
MICS, Instr/Impl, 2:1VSWR, 50 Ohms	**-54.1 dBm min**	-51.27 dBm	-52.08 dBm	-49.49 dBm	-50.37 dBm	PASS	Tx_Sc1
MEDS, Implant, 50 Ohms	**-7 dBm min**	-6.60 dBm	-6.85 dBm	-6.80 dBm	-6.83 dBm	PASS	Tx_Sc2
MEDS, Implant, 50-j60 Ohms	**-7.6 dBm min**	-7.24 dBm	-7.07 dBm	-7.06 dBm	-7.25 dBm	PASS	Tx_Sc3
MEDS, Implant, 5+j0 Ohms	**-8.9 dBm min**	-8.01 dBm	-8.09 dBm	-7.95 dBm	-8.09 dBm	PASS	Tx_Sc4
MEDS, Implant, 5-j20 Ohms	**-9.8 dBm min**	-9.45 dBm	-9.47 dBm	-9.24 dBm	-9.45 dBm	PASS	Tx_Sc6
MEDS, Implant, 9-j20 Ohms	**-9.6 dBm min**	-8.90 dBm	-8.88 dBm	-8.81 dBm	-8.88 dBm	PASS	Tx_Sc7
MEDS, Implant, 5-j60 Ohms	**-11 dBm min**	-9.84 dBm	-9.59 dBm	-9.75 dBm	-9.57 dBm	PASS	Tx_Sc5

MEDS, Implant, 2:1VSWR, 50 Ohms	**-9.1 dBm min**	-7.51 dBm	-7.64 dBm	-7.40 dBm	-7.77 dBm	PASS	Tx_Sc1
MEDS, Instrument, 50 Ohms	**-15 dBm min**	-14.90 dBm	-14.61 dBm	-14.45 dBm	-14.45 dBm	PASS	Tx_Sc2
MEDS, Instrument, 50-j60 Ohms	**-15.6 dBm min**	-15.01 dBm	-15.15 dBm	-14.98 dBm	-15.07 dBm	PASS	Tx_Sc3
MEDS, Instrument, 5+j0 Ohms	**-16.9 dBm min**	-15.79 dBm	-16.13 dBm	-15.64 dBm	-15.83 dBm	PASS	Tx_Sc4
MEDS, Instrument, 5-j20 Ohms	**-17.8 dBm min**	-17.22 dBm	-17.46 dBm	-17.08 dBm	-17.46 dBm	PASS	Tx_Sc6
MEDS, Instrument, 9-j20 Ohms	**-17.6 dBm min**	-16.66 dBm	-16.70 dBm	-16.50 dBm	-16.67 dBm	PASS	Tx_Sc7
MEDS, Instrument, 5-j60 Ohms	**-19 dBm min**	-17.80 dBm	-17.34 dBm	-17.76 dBm	-17.35 dBm	PASS	Tx_Sc5
MEDS, Instrument, 2:1VSWR, 50 Ohms	**-17.1 dBm min**	-15.34 dBm	-15.50 dBm	-15.45 dBm	-15.24 dBm	PASS	Tx_Sc1
MEDS, Instr/Impl, 50 Ohms	**-30 dBm min**	-29.38 dBm	-29.29 dBm	-29.30 dBm	-28.71 dBm	PASS	Tx_Sc2
MEDS, Instr/Impl, 50-j60 Ohms	**-30.6 dBm min**	-29.91 dBm	-30.25 dBm	-29.68 dBm	-29.47 dBm	PASS	Tx_Sc3
MEDS, Instr/Impl, 5+j0 Ohms	**-31.9 dBm min**	-30.77 dBm	-31.17 dBm	-30.45 dBm	-31.11 dBm	PASS	Tx_Sc4
MEDS, Instr/Impl, 5-j20 Ohms	**-32.8 dBm min**	-32.08 dBm	-32.77 dBm	-31.97 dBm	-32.38 dBm	PASS	Tx_Sc6
MEDS, Instr/Impl, 9-j20 Ohms	**-32.6 dBm min**	-31.32 dBm	-31.97 dBm	-31.30 dBm	-27.64 dBm	PASS	Tx_Sc7
MEDS, Instr/Impl, 5-j60 Ohms	**-34 dBm min**	-32.90 dBm	-33.11 dBm	-32.87 dBm	-30.19 dBm	PASS	Tx_Sc5
MEDS, Instr/Impl, 2:1VSWR, 50 Ohms	**-32.1 dBm min**	-29.87 dBm	-30.55 dBm	-29.42 dBm	-29.37 dBm	PASS	Tx_Sc1
MEDS, Instr/Impl, 50 Ohms	**-52 dBm min**	-50.05 dBm	-50.84 dBm	-49.84 dBm	-50.30 dBm	PASS	Tx_Sc2

MEDS, Instr/Impl, 50-j60 Ohms	**-52.6 dBm min**	-50.69 dBm	-51.43 dBm	-50.88 dBm	-51.20 dBm	PASS	Tx_Sc3
MEDS, Instr/Impl, 5+j0 Ohms	**-53.9 dBm min**	-52.74 dBm	-53.31 dBm	-50.93 dBm	-51.68 dBm	PASS	Tx_Sc4
MEDS, Instr/Impl, 5-j20 Ohms	**-54.8 dBm min**	-53.16 dBm	-53.99 dBm	-52.95 dBm	-53.29 dBm	PASS	Tx_Sc6
MEDS, Instr/Impl, 9-j20 Ohms	**-54.6 dBm min**	-52.70 dBm	-52.76 dBm	-52.86 dBm	-52.13 dBm	PASS	Tx_Sc7
MEDS, Instr/Impl, 5-j60 Ohms	**-56 dBm min**	-53.83 dBm	-53.01 dBm	-54.42 dBm	-54.28 dBm	PASS	Tx_Sc5
MEDS, Instr/Impl, 2:1VSWR, 50 Ohms	**-54.1 dBm min**	-50.87 dBm	-52.14 dBm	-50.03 dBm	-51.15 dBm	PASS	Tx_Sc1

Table 1 Output Power

Conclusion: All results from Output Power tests show compliance with the specification.

8.2 Transmit Modulation Response (EVM) and Data Rates

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.5 and 2.3.4.2 (data rate)

DVT Plan: 6.3.2

This test has a deviation (Deviation #3) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

Test Executive Scripts: Tx_Sc8 through Tx_Sc14

Impedance: 50 Ohm

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Data Rate, Mode 1	**48.72282 kb/s max** **48.71308 kb/s min**	48.720 kb/s	48.720 kb/s	48.720 kb/s	48.720 kb/s	PASS	
Data Rate, Mode 2	**97.44564 kb/s max** **97.42616 kb/s min**	97.440 kb/s	97.440 kb/s	97.440 kb/s	97.440 kb/s	PASS	
Data Rate, Mode 3	**190.019 kb/s max** **189.981 kb/s min**	190.000 kb/s	190.000 kb/s	190.000 kb/s	190.000 kb/s	PASS	
Data Rate, Mode 4	**380.038 kb/s max** **379.962 kb/s min**	380.000 kb/s	380.000 kb/s	380.000 kb/s	380.000 kb/s	PASS	
EVM, Modes 1-4, Power Level -4 to -30dBm, MICS	**8.4% (max)**	3.89%	4.08%	4.46%	3.98%	PASS	
EVM, Modes 1-4, Power Level below -30dBm, MICS/MEDS	**12.6% (max)**	3.09%	3.48%	4.35%	4.30%	PASS	
EVM, Modes 1-2, Power Level -7 to -30dBm, MEDS	**8.4% (max)**	2.96%	3.62%	3.33%	2.75%	PASS	

Table 2 Transmit Modulation Response (EVM) and Data Rate

VSWR 2:1

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
EVM, Modes 1-4, Power Level -6.1 to -32.1 dBm, MICS	**8.4% (max)**	6.20%	6.13%	6.28%	5.08%	PASS	
EVM, Modes 1-4, Power Level below -32.1 dBm, MICS/MEDS	**12.6% (max)**	3.31%	3.52%	4.54%	4.38%	PASS	
EVM, Modes 1-2, Power Level -9.1 to -32.1 dBm, MEDS	**8.4% (max)**	4.63%	5.21%	4.94%	4.17%	PASS	

Table 3 Transmit Modulation Response (EVM) and Data Rates

Impedance 50 – j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
EVM, Modes 1-4, Power Level -4.6 to -30.6 dBm, MICS	**8.4% (max)**	3.85%	5.43%	5.16%	4.92%	PASS	
EVM, Modes 1-4, Power Level below -30.6 dBm, MICS/MEDS	**12.6% (max)**	3.13%	3.42%	4.43%	4.62%	PASS	
EVM, Modes 1-2, Power Level -7.6 to -30.6 dBm, MICS	**8.4% (max)**	2.96%	3.70%	3.13%	2.70%	PASS	

Table 4 Transmit Modulation Response (EVM) and Data Rates

Impedance 5 + j0

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
EVM, Modes 1-4, Power Level -5.9 to -31.9 dBm, MICS	**8.4% (max)**	4.32%	4.31%	4.72%	3.92%	PASS	
EVM, Modes 1-4, Power Level below -31.9 dBm, MICS/MEDS	**12.6% (max)**	3.36%	3.61%	4.42%	4.52%	PASS	

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
EVM, Modes 1-2, Power Level -8.9 to -31.9 dBm, MEDS	**8.4% (max)**	2.95%	3.65%	3.11%	2.70%	PASS	

Table 5 Transmit Modulation Response (EVM) and Data Rates

Impedance 5 – j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
EVM, Modes 1-4, Power Level -8 to -34dBm, MICS	**8.4% (max)**	5.25%	5.20%	5.88%	5.13%	PASS	
EVM, Modes 1-4, Power Level below -34dBm, MICS/MEDS	**12.6% (max)**	3.42%	3.46%	4.96%	5.05%	PASS	
EVM, Modes 1-2, Power Level -11 to -34dBm, MEDS	**8.4% (max)**	2.99%	4.56%	3.48%	3.36%	PASS	

Table 6 Transmit Modulation Response (EVM) and Data Rates

Impedance 5 – j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
EVM, Modes 1-4, Power Level -6.8 to -32.8 dBm, MICS	**8.4% (max)**	4.83%	5.06%	5.10%	4.65%	PASS	
EVM, Modes 1-4, Power Level below -32.8 dBm, MICS/MEDS	**12.6% (max)**	3.73%	3.66%	4.42%	4.81%	PASS	
EVM, Modes 1-2, Power Level -9.8 to -32.8 dBm, MICS	**8.4% (max)**	5.22%	3.76%	3.06%	2.75%	PASS	

Table 7 Transmit Modulation Response (EVM) and Data Rates

Impedance 9 – j20

Test Parameter	**Spec**	**sn302**	**sn311**	**sn313**	**sn314**	**Compliance**	**Notes**
EVM, Modes 1-4 Power Level -5.7 to -31.7 dBm, MICS	**8.4% (max)**	5.47%	5.24%	5.31%	4.50%	PASS	
EVM, Modes 1-4, Power Level below -31.7 dBm, MICS	**12.6% (max)**	3.36%	3.63%	4.71%	4.74%	PASS	
EVM, Modes 1-2, Power Level -9.6 to -32.6 dBm, MEDS	**8.4% (max)**	2.95%	3.65%	3.14%	2.76%	PASS	
EVM, Modes 1-2, Power Level below -32.6 dBm, MEDS	**12.6% (max)**	2.31%	2.93%	3.40%	2.37%	PASS	

Table 8 Transmit Modulation Response (EVM) and Data Rates

Conclusion: All results from Transmit Modulation Response (EVM) tests show compliance with the specification.

8.3 TelC Deviation and Data Rate

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.5 and 2.3.4.2

DVT Plan: 6.3.3

Test Executive Script: Tx_Sc8

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Data Rate, Mode 5	**14.84523 kb/s max** **14.84227 kb/s min**	14.8437 kb/s	14.8437 kb/s	14.8438 kb/s	14.8438 kb/s	PASS	
Data Rate, Mode 6	**100.01 kb/s max** **99.99 kb/s min**	100.00 kb/s	100.00 kb/s	100.00 kb/s	100.00 kb/s	PASS	
Deviation Narrowband	**14.67 kHz min,** **17.93 kHz max**	15.3629 KHz	15.3629 KHz	15.3684 KHz	15.3729 KHz	PASS	
Deviation Wideband	**45 kHz min** **55 kHz max**	48.0649 KHz	48.1549 KHz	48.1633 KHz	48.2550 KHz	PASS	

Table 9 TelC Deviation and Data Rate

Note: The test signal used in the data rate test differs from the one identified in the DVT plan.
DVT Plan: acm_mod_clk_en
Tx_TelC_continuous_whitened: data_clk_en

Conclusion: All results from TelC Deviation and Data Rate tests show compliance with the specification.

8.4 Stability - Tx Spurious into 12:1 VSWR Load

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.4

DVT Plan: 6.3.5

Test Executive Script: Tx_Sc22

Specification: The requirement is that there be no instability. As stated in the DVT Plan, during the performance of these tests, integer multiple harmonics of the operating frequency of the "DUT" (such as the second harmonic of the modulated carrier) will be present in the frequency sweeps. Synthesizer spurs at other frequencies may also be present but are not related to transmitter stability. The known transmitter spurious and integer multiple harmonics that are products of the design will be present at all reflection coefficient magnitudes and angles. Instabilities, if present, would only occur at specific or narrow ranges of reflection coefficient magnitudes and angles.

Medtronic Engineering is required to examine the data results from this test and confirm that there are no instabilities.

This test has a deviation (Deviation #1) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

This test has a deviation (Deviation #2) from the DVT Plan. See Section 6 DVT plan Deviations for more information

8.4.1 Stability Test with 1MHz RBW

Stability Conclusion: Data not appropriate for Stability review. See.

Gate for Regulatory Testing: Medtronic has processed and reviewed this data and has approved proceeding to Regulatory testing. See.

Pre and Post Test Results:

Test Parameter	Spec	sn327	sn331	sn334	sn336	Compliance	Notes
RSSI **Mode 4 0 dB Attenuation**	Absolute -103 dBm min -99 dBm max Diff level -106 to -68: 2.5dB Diff level -68 to -55: 4 dB	-101.16 dBm min -100.52 dBm max Min code diff = 5 Min code diff = 3	-100.68 dBm min -100.68 dBm max Min code diff = 5 Min code diff = 4	-101 dBm min -100.84 dBm max Min code diff = 5 Min code diff = 4	-100.68 dBm min -100.68 dBm max Min code dff = 5 Min code diff = 4	PASS	Deltas and absolute numbers are worst case.

Sensitivity Mode 4 0dB Attenuation	-95dBm	Delta = 0.25dB, -101.5 dBm worst case	Delta = 0 dB, -101.5 dBm worst case	Delta = 0.25 dB, -101.5 dBm worst case	Delta = 0.25 dB, -101 dBm worst case	PASS	
Sensitivity Mode 4 2dB Attenuation	-93dBm	Delta = 0.25dB, -100.5 dBm worst case	Delta = 0dB, -100.75 dBm worst case	Delta = 0.25 dB, -100.5 dBm worst case	Delta = 0dB, -100.25 dBm worst case	PASS	
Sensitivity Mode 4 22dB Attenuation	-73dBm	Delta = 0dB, -82 dBm worst case	Delta = 0.25dB, -81.75 dBm worst case	Delta = 0 dB, -82.25 dBm worst case	Delta = 0dB, -82 dBm worst case	PASS	
AM Rejection Mode 4	-58dBm	Delta = 0.25 dB, -52 dBm	Delta = 0.25 dB, -49 dBm	Delta = 0.75 dB, -48.25 dBm	Delta = 0.5 dB, -50.25 dBm	PASS	
Standby Current (Disabled Static Current @25C)	50nA	Delta = 0.147nA, 2.83nA	Delta = 0.092nA, 3.01nA	Delta = 0.030nA, 2.76nA	Delta = 0.082nA, 2.54nA	PASS	
Tx Power	-4dBm min for MICS band	Delta = 0.063dB, -3.52 dBm	Delta = 0.056dB, -3.55 dBm	Delta = 0.050dB, -3.55 dBm	Delta = 0.047dB, -3.58 dBm	PASS	

Table 10 Pre and Post Tests for Stability (1MHz RBW)

8.4.2 **Stability Test with 30kHz RBW**

As stated in the Test Plan section 6.3.5, to fulfil the testing of this requirement, Medtronic has reviewed 264 screen capture plots representing the 264 vectors combinations across DUT’s with serial numbers sn327, sn331, sn334 and sn336. It has been reviewed using the agreed upon criteria also called out in the same section of the Test Plan. See Figure 1 for an example screen capture.

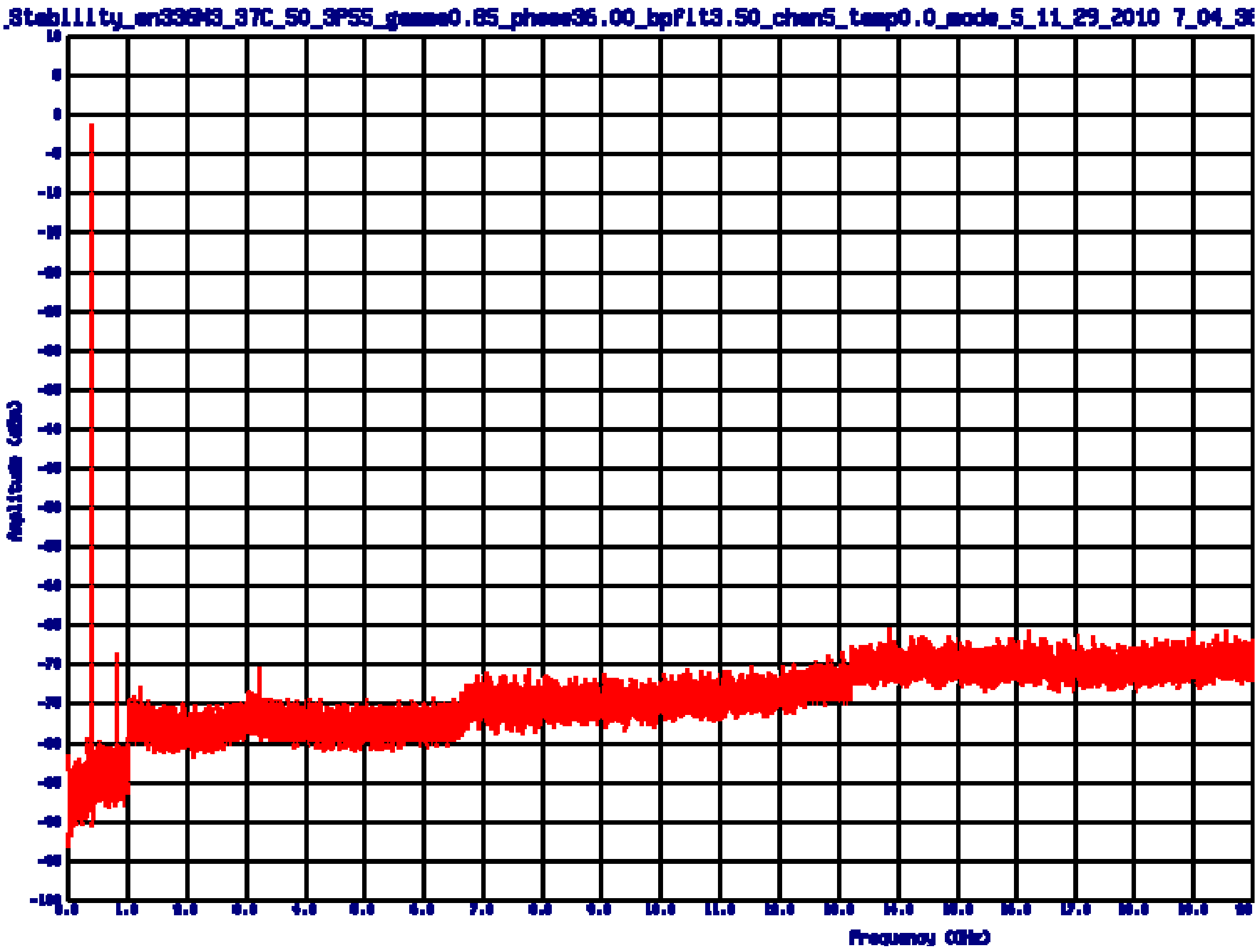


Figure 1 : Tx_Sc22 Stability Test with 30KHz RBW

Post stability test results.

Test Parameter	Spec	sn327	sn331	sn334	sn336	Compliance	Notes
RSSI **Mode 4 0 dB Attenuation**	Absolute -103 dBm min -99 dBm max	-100.84 dBm	-101.32 dBm	-101.00 dBm	-101.32 dBm	PASS	Deltas and absolute numbers are worst case.
	Diff level -106 to -68 dBm: 2.5dB	Min code diff = 5	Min code diff = 5	Min code diff = 5	Min code diff = 5		
	Diff level -68 to -55 dBm: 4 dB	Min code diff = 7	Min code diff = 7	Min code diff = 6	Min code diff = 6		
Sensitivity Mode 4 0dB Attenuation	-95dBm max	-101.25 dBm	-101.5 dBm	-101.25 dBm	-101.50 dBm	PASS	

Sensitivity Mode 4 2dB Attenuation	-93dBm max	-101.00 dBm	-100.5 dBm	-100.75 dBm	-100.75 dBm	PASS	
Sensitivity Mode 4 22dB Attenuation	-73dBm max	-82.25 dBm	-81.75 dBm	-82.25 dBm	-82.00 dBm	PASS	
AM Rejection Mode 4	-58dBm min	-51.75 dBm	-48.75 dBm	-47.50 dBm	-50.50 dBm	PASS	
Standby Current (Disabled Static Current @25C)	50nA max	1.89 nA	2.63 nA	2.18 nA	1.81 nA	PASS	
Tx Power	-4dBm min for MICS band	-3.51 dBm	-3.59 dBm	-3.24 dBm	-3.30 dBm	PASS	

Table 11 Post Test for Stability (30KHz RBW)

Conclusion for Post Stability Test: It was found that all results from tests run after Stability (30KHz RBW) show compliance with the specification.

Conclusion for Stability Test: It was found that the Stability Test (30KHz RBW) shows compliance with the specification since no evidence of instabilities on the spectrum analyzer plots for each sweep has been found.

8.5 Regulatory Test Results

Gate for Regulatory Testing:

The task of reviewing Tx Stability data by Medtronic before proceeding with Regulatory testing is a requirement called out in section 6.3.5 of the DVT Plan. Medtronic has for this purpose reviewed the first Tx Stability data taken with 1MHz RBW setting for frequencies above 1GHz with the justification of Deviation #1 to the DVT plan. Medtronic reviewed all four data sets representing the four Modules and Medtronic approve that DVT can move forward with the Regulatory testing.

This includes Tx Harmonic and Spurious, ACPR, AltCPR and Spectral Mask

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.4

DVT Plan: 6.3.4

Note: Regulatory tests in sections 4.5.2 through 4.5.4 are implemented in the test module ZRLNK_Regulatory_Tx_Harmonics_Spurious. This test module has 23 subtests. The subtest that covers a parameter in the results tables is shown in the Notes column.

Note on use of Peak Detector vs CISPR Detector: The Regulatory Tx Harmonics Spurious and Rx Conducted Spurious Emissions tests have requirements to test for unwanted emissions using a CISPR detector in the spectrum analyzer. The DVT tests use a peak detector instead in order to save time. Close to the MEDS/MICS bands the CISPR detector is used but it is slow. This substitution of a peak detector for the CISPR detector is justified by lab data reported in “Measurement comparison of a CISPR Quasi peak detector method and an approximated CISPR peak detector method” [9]. The conclusion of that document states: “The CISPR method measures a peak power that is 0.5dB lower than the approximation method. This means that the approximation method is conservative when measuring emissions and safe to use.”

8.5.1 ACPR and AltCPR

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.4.1.1 (MICS) and 2.3.4.6.4.2.1 (MEDS)

Test Executive Scripts: Tx_Sc1 through Tx_Sc8

Note: The specification for ACPR and AltCPR in the DVT Plan are stated to be “maximum” with positive values. They should have been “minimum”.

Note: Negative numbers in the raw data were adapted to positive numbers to match the Mozart Requirement Specification (MDT Document #A17245); rev I.

Impedance: 2:1 VSWR

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
ACPR, MICS, -6.1 to -14.1 dBm	**22.7 dB min**	27.62 dB	28.6 dB	26.67 dB	29.19dB	PASS	Tested power levels reflect relaxation from -4 to -12 dBm

ACPR, MICS, -14.1 dBm and below	**34 dB min**	40.89 dB	39.97 dB	40.44 dB	39.98 dB	PASS	
AltCPR, MICS, -6.1 to -14.1 dBm	**28 dB min**	51.79 dB	52.68 dB	50.51 dB	53.21 dB	PASS	
AltCPR, MICS, -14.1 dBm and below	**40 dB min**	54.35 dB	53.79 dB	53.72 dB	54.09 dB	PASS	
ACPR, MEDS, -9.1 to -17.1 dBm	**26 dB min**	31.43 dB	31.11 dB	30.9 dB	31.89 dB	PASS	Tested power levels reflect relaxation from -7 to -15 dBm
ACPR, MEDS, -17.1 dBm and below	**35 dB min**	40.60 dB	39.69 dB	40.52 dB	39.93 dB	PASS	
AltCPR, MEDS, -9.1 to -17.1 dBm	**31 dB min**	53.50 dB	53.33 dB	54.08 dB	54.14 dB	PASS	
AltCPR, MEDS, -17.1 dBm and below	**40 dB min**	53.70 dB	53.41 dB	53.69 dB	53.49 dB	PASS	

Table 12 Regulatory Tests – 2:1 VSWR :ACPR, AltCPR

Impedance: 50 +j0

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
ACPR, MICS, -4 to -12 dBm	22.7 dB min	30.96 dB	32.89 dB	29.94 dB	33.83 dB	PASS	Tested power levels reflect relaxation from -4 to -12 dBm
ACPR, MICS, -12 dBm and below	34 dB min	40.62 dB	39.94 dB	40.52 dB	40.27 dB	PASS	
AltCPR, MICS, -4 to -12 dBm	28 dB min	53.47 dB	53.70 dB	53.11 dB	53.16 dB	PASS	
AltCPR, MICS, -12 dBm and below	40 dB min	52.90 dB	52.37 dB	52.85 dB	52.22 dB	PASS	
ACPR, MEDS, -7 to -15 dBm	26 dB min	34.94 dB	35.20 dB	34.14 dB	35.68 dB	PASS	Tested power levels reflect relaxation from -7 to -15 dBm

ACPR, MEDS, -15 dBm and below	35 dB min	40.48 dB	39.63 dB	40.38 dB	39.94 dB	PASS	
AltCPR, MEDS, -7 to -15 dBm	31 dB min	54.71 dB	54.30 dB	54.74 dB	54.60 dB	PASS	
AltCPR, MEDS, -15 dBm and below	40 dB min	53.57 dB	53.18 dB	53.64 dB	53.66 dB	PASS	

Table 13 Regulatory Tests – 50+j0 :ACPR, AltCPR

Impedance: 50 –j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
ACPR, MICS, -4.6 to -12.6 dBm	22.7 dB min	31.76 dB	29.42 dB	29.41 dB	29.66 dB	PASS	Tested power levels reflect relaxation from -4 to -12 dBm
ACPR, MICS, -12.6 dBm and below	34 dB min	40.88 dB	40.21 dB	40.32 dB	40.30 dB	PASS	
AltCPR, MICS, -4.6 to -12.6 dBm	28 dB min	53.51 dB	53.57 dB	53.14 dB	53.43 dB	PASS	
AltCPR, MICS, -12.6 dBm and below	40 dB min	53.17 dB	53.02 dB	52.54 dB	52.72 dB	PASS	
ACPR, MEDS, -7.6 to -15.6 dBm	26 dB min	36.08 dB	36.13 dB	34.9 dB	36.35 dB	PASS	Tested power levels reflect relaxation from -7 to -15 dBm
ACPR, MEDS, -15.6 dBm and below	35 dB min	40.60 dB	39.99 dB	40.26 dB	40.09 dB	PASS	
AltCPR, MEDS, -7.6 to -15.6 dBm	31 dB min	54.42 dB	54.44 dB	55.05 dB	54.83 dB	PASS	
AltCPR, MEDS, -15.6 dBm and below	40 dB min	53.47 dB	53.14 dB	51.71 dB	52.18 dB	PASS	

Table 14 Regulatory Tests – 50-j60 :ACPR, AltCPR

Impedance: 5+ j0

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
ACPR, MICS, **-5.9 to -13.9 dBm**	22.7 dB min	29.35 dB	31.53 dB	29.84 dB	31.54 dB	PASS	Tested power levels reflect relaxation from -4 to -12 dBm
ACPR, MICS, **-13.9 dBm and below**	34 dB min	40.48 dB	39.79 dB	40.31 dB	39.88 dB	PASS	
AltCPR, MICS, **-5.9 to -13.9 dBm**	28 dB min	54.10 dB	53.65 dB	53.58 dB	53.67 dB	PASS	
AltCPR, MICS, **-13.9 dBm and below**	40 dB min	52.76 dB	52.88 dB	52.61 dB	52.88 dB	PASS	
ACPR, MEDS, **-8.9 to -16.9 dBm**	26 dB min	37.73 dB	37.29 dB	36.74 dB	37.59 dB	PASS	Tested power levels reflect relaxation from -7 to -15 dBm
ACPR, MEDS, **-16.9 dBm and below**	35 dB min	40.09 dB	39.37 dB	40.59 dB	39.77 dB	PASS	
AltCPR, MEDS, **-8.9 to -16.9 dBm**	31 dB min	54.70 dB	54.56 dB	55.04 dB	54.45 dB	PASS	
AltCPR, MEDS, **-16.9 dBm and below**	40 dB min	53.63 dB	53.16 dB	50.92 dB	51.16 dB	PASS	

Table 15 Regulatory Tests – 5+j0 :ACPR, AltCPR

Impedance: 5- j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
ACPR, MICS, **-8 to -16 dBm**	22.7 dB min	27.32 dB	27.81 dB	26.01 dB	28.37 dB	PASS	Tested power levels reflect relaxation from -4 to -12 dBm
ACPR, MICS, **-16 dBm and below**	34 dB min	40.23 dB	40.11 dB	39.88 dB	39.58 dB	PASS	
AltCPR, MICS, **-8 to -16 dBm**	28 dB min	51.64 dB	50.40 dB	49.05 dB	52.18 dB	PASS	

AltCPR, MICS, -16 dBm and below	40 dB min	53.05 dB	52.61 dB	52.88 dB	52.43 dB	PASS	
ACPR, MEDS, -11 to -19 dBm	26 dB min	35.17 dB	32.04 dB	33.58 dB	33.78 dB	PASS	Tested power levels reflect relaxation from -7 to -15 dBm
ACPR, MEDS, -19 dBm and below	35 dB min	40.39 dB	39.8 dB	39.52 dB	39.67 dB	PASS	
AltCPR, MEDS, -11 to -19 dBm	31 dB min	55.03 dB	53.79 dB	55.03 dB	54.39 dB	PASS	
AltCPR, MEDS, -19 dBm and below	40 dB min	53.60 dB	48.94 dB	48.71 dB	50.81 dB	PASS	

Table 16 Regulatory Tests – 5-j60 :ACPR, AltCPR

Impedance: 5- j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
ACPR, MICS, -6.8 to -14.8 dBm	22.7 dB min	29.93 dB	30.39 dB	28.69 dB	29.38 dB	PASS	Tested power levels reflect relaxation from -4 to -12 dBm
ACPR, MICS, -14.8 dBm and below	34 dB min	40.52 dB	39.57 dB	39.89 dB	39.61 dB	PASS	
AltCPR, MICS, -6.8 to -14.8 dBm	28 dB min	53.32 dB	53.11 dB	52.78 dB	53.12 dB	PASS	
AltCPR, MICS, -14.8 dBm and below	40 dB min	53.29 dB	52.70 dB	52.51 dB	52.91 dB	PASS	
ACPR, MEDS, -9.8 to -17.8 dBm	26 dB min	37.00 dB	37.18 dB	37.22 dB	37.84 dB	PASS	Tested power levels reflect relaxation from -7 to -15 dBm
ACPR, MEDS, -17.8 dBm and below	35 dB min	40.53 dB	39.14 dB	40.10 dB	39.53 dB	PASS	
AltCPR, MEDS, -9.8 to -17.8 dBm	31 dB min	55.06 dB	54.46 dB	55.08 dB	54.69 dB	PASS	

AltCPR, MEDS, **-17.8 dBm and below**	40 dB min	53.24 dB	53.24 dB	50.32 dB	50.15 dB	PASS	

Table 17 Regulatory Tests – 5-j20 :ACPR, AltCPR

Impedance: 9- j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
ACPR, MICS, **-5.7 to -13.7 dBm**	22.7 dB min	28.59 dB	30.36 dB	29.49 dB	30.63 dB	PASS	Tested power levels reflect relaxation from -4 to -12 dBm
ACPR, MICS, **-13.7 dBm and below**	34 dB min	40.49 dB	39.44 dB	40.38 dB	39.69 dB	PASS	
AltCPR, MICS, **-5.7 to -13.7 dBm**	28 dB min	53.51 dB	53.39 dB	53.19 dB	53.29 dB	PASS	
AltCPR, MICS, **-13.7 dBm and below**	40 dB min	53.47 dB	52.83 dB	52.95 dB	52.80 dB	PASS	
ACPR, MEDS, **-9.6 to -17.6 dBm**	26 dB min	38.08 dB	37.66 dB	37.59 dB	37.55 dB	PASS	Tested power levels reflect relaxation from -7 to -15 dBm
ACPR, MEDS, **-17.6 dBm and below**	35 dB min	40.00 dB	38.95 dB	39.95 dB	39.61 dB	PASS	
AltCPR, MEDS, **-9.6 to -17.6 dBm**	31 dB min	55.20 dB	54.17 dB	54.87 dB	54.50 dB	PASS	
AltCPR, MEDS, **-17.6 dBm and below**	40 dB min	50.94 dB	50.63 dB	50.54 dB	51.35 dB	PASS	

Table 18 Regulatory Tests – 9-j20 :ACPR, AltCPR

Conclusion: All results from TX Regulatory ACPR and AltCPR tests show compliance with the specification.

8.5.2 **In-band Spectral Mask**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.4.1.2 (MICS) and 2.3.4.6.4.2.2 (MEDS)

Test Executive Scripts: Tx_Sc15 through Tx_Sc20

This test has a deviation (Deviation #19) from the DVT Plan. See Section 6 DVT Plan Deviations for more information. Note that deviation #19 only specifies the need of data from missing vectors in the DVT plan. Data that aligns with all requirements in deviation #19, as well as requirements specified in ECO 093, was gathered before the deviation was approved. Data processing, review and incorporation of results into the DVT report occurred after ECO 093 and deviation #19 was approved.

Impedance: 50 Ohm

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	267.95 KHz	267.02 KHz	268.13 KHz	266.97 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	80.58 KHz	81.49 KHz	81.20 KHz	81.17 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	204.97 KHz	204.11 KHz	205.47 KHz	203.69 KHz	PASS	Regulatory subtest #1

Table 19 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

Impedance: 50 Ohm (2:1 VSWR)

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	268.15 KHz	271.65 KHz	274.88 KHz	268.35 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	80.98 KHz	81.8 KHz	82.1 KHz	81.64 KHz	PASS	Regulatory subtest #1

Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	204.84 KHz	205.05 KHz	205.54 KHz	204.08 KHz	PASS	Regulatory subtest #1

Table 20 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

Impedance: 50-j60 Ohm

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	267.51 KHz	269.92 KHz	269.15 KHz	265.84 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	81.18 KHz	81.35 KHz	81.39 KHz	81.23 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	205.13 KHz	204.95 KHz	203.95 KHz	203.82 KHz	PASS	Regulatory subtest #1

Table 21 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

Impedance: 5-j0

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	268.22 KHz	266.74 KHz	266.55 KHz	266.45 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	81.56 KHz	81.27 KHz	81.26 KHz	81.45 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	204.89 KHz	204.14 KHz	204.36 KHz	202.87 KHz	PASS	Regulatory subtest #1

Table 22 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

Impedance: 5-j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	269.56 KHz	271.41 KHz	275.52 KHz	267.91 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	81.11 KHz	81.46 KHz	81.74 KHz	81.32 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	205.33 KHz	203.83 KHz	204.01 KHz	204.10 KHz	PASS	Regulatory subtest #1

Table 23 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

Impedance: 5-j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	267.3 KHz	266.24 KHz	266.84 KHz	268.48 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	81.16 KHz	81.27 KHz	81.42 KHz	81.55 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	204.79 KHz	204.38 KHz	204.69 KHz	203.25 KHz	PASS	Regulatory subtest #1

Table 24 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

Impedance: 9-j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Emission Bandwidth, -21 dB **Wideband Mode 4(MICS)**	**300 kHz max**	269.47 KHz	268.35 KHz	269.71 KHz	269.17 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Narrowband Mode 2(MEDS)**	**100 kHz max**	81.16 KHz	81.10 KHz	81.56 KHz	81.16 KHz	PASS	Regulatory subtest #1
Emission Bandwidth, -21 dB **Mode 6**	**300KHz max**	205.43 KHz	204.05 KHz	204.16 KHz	203.18 KHz	PASS	Regulatory subtest #1

Table 25 Regulatory Tests– In-band Spectral Mask – Emission Bandwidth

Conclusion: All results from TX Regulatory Emission Bandwidth tests show compliance with the specification.

8.5.3 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph:

2.3.4.6.4.1.3 (MICS immediate out-of-band),

2.3.4.6.4.2.3.1 (MEDS immediate out-of-band),

2.3.4.6.4.2.3.2 (MEDS into MICS FCC),

2.3.4.6.4.2.3.3 (MEDS into MICS ETSI)

This test has a deviation (Deviation #7) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

This test has a deviation (Deviation #19) from the DVT Plan. See Section 6 DVT Plan Deviations for more information. Note that deviation #19 only specifies the need of data from missing vectors in the DVT plan. Data that aligns with all requirements in deviation #19, as well as requirements specified in ECO 093, was gathered before the deviation was approved. Data processing, review and incorporation of results into the DVT report occurred after ECO 093 and deviation #19 was approved.

Test Executive Scripts: Tx_Sc15 through Tx_Sc20

Impedance: 50 ohm

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Spurious Emissions, immediately out-of-band, MICS/MEDS **Tx power:** **-4 dBm MICS** **-7 dBm MEDS**	**-21 dB max**	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	PASS	Regulatory subtest #11,12
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -7 to -15 dBm**	**-32.2 dBm max**	-51.8 dBm	-50.7 dBm	-53.3 dBm	-55.0 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -15 dBm**	**-52.2 dBm max**	-60.8 dBm	-60.5 dBm	-61.1 dBm	-59.9 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -20 dBm**	**-57.2 dBm max**	-63.7 dBm	-65.3 dBm	-62.2 dBm	-64.0 dBm	PASS	Regulatory subtest #15,13 See Note (3)

MEDS into MICS (FCC), **Ch20,21** **Tx power -30.3 to -51.8 dBm**	**-32.2 dBm max**	-38.1dBm	-39.8 dBm	-39.3 dBm	-39.4 dBm	PASS	Regulatory subtest #15,13 See Note (1),(2)
MEDS into MICS (FCC), **Ch20,21** **Tx power -51.8 dBm**	**-52.2 dBm max**	-57.3 dBm	-60.9 dBm	-58.6 dBm	-59.8 dBm	PASS	Regulatory subtest #15,13 See Note (2)
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -7 to -15 dBm**	**-34.85 dBm max**	-54.1 dBm	-52.8 dBm	-53.8 dBm	-53.5 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -15 dBm**	**-54.85 dBm max**	-60.2 dBm	-59.5 dBm	-60.1 dBm	-60.4 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -20 dBm**	**-59.85 dBm max**	-64.7 dBm	-64.2 dBm	-64.3 dBm	-64.5 dBm	PASS	Regulatory subtest #15,14 See Note (3)
MEDS into MICS (ETSI), **Ch20,21** **Tx power -15 to -25 dBm**	**-34.85 dBm max**	-47.5 dBm	-47.1 dBm	-47.3 dBm	-46.6 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -25 dBm**	**-54.85 dBm max**	-58.1 dBm	-56.5 dBm	-57.1 dBm	-56.5 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -30 dBm**	**-59.85 dBm max**	-62.6dBm	-61.5 dBm	-62.0 dBm	-60.8 dBm	PASS	Regulatory subtest #15,14

Table 26 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Impedance: 50 Ohm (2:1 VSWR)

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Spurious Emissions, immediately out-of-band, MICS/MEDS **Tx power:** **-6.1dBm MICS** **-9.1dBm MEDS**	**-21 dB max**	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	PASS	Regulatory subtest #11,12
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -9.1 to -17.1 dBm**	**-32.2 dBm max**	-51.1 dBm	-52.3 dBm	-54.7 dBm	-50.1 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -17.1 dBm**	**-52.2 dBm max**	-58.4 dBm	-58.3 dBm	-61.6 dBm	-57.8 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -22.1 dBm**	**-57.2 dBm max**	-64.4 dBm	-60.6 dBm	-66.1 dBm	-62.8 dBm	PASS	Regulatory subtest #15,13 See Note (3)
MEDS into MICS (FCC), **Ch20,21** **Tx power -32.4 to -53.9 dBm**	**-32.2 dBm max**	-38.8 dBm	-37.8 dBm	-39.5 dBm	-37.1 dBm	PASS	Regulatory subtest #15,13 See Note (1),(2)
MEDS into MICS (FCC), **Ch20,21** **Tx power -53.9 dBm**	**-52.2 dBm max**	-57.2 dBm	-59.0 dBm	-58.9 dBm	-58.8 dBm	PASS	Regulatory subtest #15,13 See Note (2)
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -9.1 to -17.1 dBm**	**-34.85 dBm max**	-53.6 dBm	-53.1 dBm	-53.3 dBm	-53.4 dBm	PASS	Regulatory subtest #15,14

MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -17.1 dBm**	**-54.85 dBm max**	-60.4 dBm	-59.8 dBm	-60.3 dBm	-60.4 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -22.1 dBm**	**-59.85 dBm max**	-64.9 dBm	-64.2 dBm	-64.7 dBm	-64.3 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -17.1 to -25 dBm**	**-34.85 dBm max**	-48 dBm	-46.8 dBm	-47.4 dBm	-46.4 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -27.1 dBm**	**-54.85 dBm max**	-57.9 dBm	-57.3 dBm	-57.1 dBm	-56.2 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -32.1 dBm**	**-59.85 dBm max**	-62.6 dBm	-61.6 dBm	-61.9 dBm	-60.9 dBm	PASS	Regulatory subtest #15,14

Table 27 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Impedance: 50-j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Spurious Emissions, immediately out-of-band, MICS/MEDS **Tx power:** **-4.6dBm MICS** **-7.6dBm MEDS**	**-21 dB max**	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	PASS	Regulatory subtest #11,12
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -7.6 to -15.6 dBm**	**-32.2 dBm max**	-53.0 dBm	-52.1 dBm	-54.1 dBm	-49.7 dBm	PASS	Regulatory subtest #15,13

MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -15.6 dBm**	**-52.2 dBm max**	-62.2 dBm	-59.6 dBm	-60.7 dBm	-61.8 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -20.6 dBm**	**-57.2 dBm max**	-67.1 dBm	-63.9 dBm	-66.4 dBm	-62.7 dBm	PASS	Regulatory subtest #15,13 See Note (3)
MEDS into MICS (FCC), **Ch20,21** **Tx power -30.9 to -52.4 dBm**	**-32.2 dBm max**	-40.0 dBm	-40.0 dBm	-38.6 dBm	-37.9 dBm	PASS	Regulatory subtest #15,13 See Note (1),(2)
MEDS into MICS (FCC), **Ch20,21** **Tx power -52.4 dBm**	**-52.2 dBm max**	-61.3 dBm	-59.2 dBm	-59.3 dBm	-58.9 dBm	PASS	Regulatory subtest #15,13 See Note (2)
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -7.6 to -15.6 dBm**	**-34.85 dBm max**	-54.7 dBm	-53.9 dBm	-54.0 dBm	-54.1 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -15.6 dBm**	**-54.85 dBm max**	-61.2 dBm	-60.3 dBm	-60.3 dBm	-60.5 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -20.6 dBm**	**-59.85 dBm max**	-65.5 dBm	-64.4 dBm	-64.7 dBm	-64.8 dBm	PASS	Regulatory subtest #15,14 See Note (3)
MEDS into MICS (ETSI), **Ch20,21** **Tx power -15.6 to -25.6 dBm**	**-34.85 dBm max**	-47.8 dBm	-47.4 dBm	-47.5 dBm	-46.9 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -25.6 dBm**	**-54.85 dBm max**	-58.5 dBm	-56.9 dBm	-57.0 dBm	-56.5 dBm	PASS	Regulatory subtest #15,14

MEDS into MICS (ETSI), **Ch20,21** **Tx power -30.6 dBm**	**-59.85 dBm max**	-63.0 dBm	-61.9 dBm	-61.9 dBm	-61.3 dBm	PASS	Regulatory subtest #15,14

Table 28 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Impedance: 5+j0

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Spurious Emissions, immediately out-of-band, MICS/MEDS **Tx power: -5.9dBm MICS -8.9dBm MEDS**	**-21 dB max**	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	PASS	Regulatory subtest #11,12
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -8.9 to -16.9 dBm**	**-32.2 dBm max**	-53.4 dBm	-53.3 dBm	-53.8 dBm	-55.6 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -16.9 dBm**	**-52.2 dBm max**	-62.0 dBm	-62.8 dBm	-62.6 dBm	-63.8 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -21.9 dBm**	**-57.2 dBm max**	-64.9 dBm	-63.1 dBm	-64.6 dBm	-64.9 dBm	PASS	Regulatory subtest #15,13 See Note (3)
MEDS into MICS (FCC), **Ch20,21** **Tx power -32.2 to -53.7 dBm**	**-32.2 dBm max**	-39.2 dBm	-40.7 dBm	-40.0 dBm	-40.2 dBm	PASS	Regulatory subtest #15,13 See Note (1),(2)
MEDS into MICS (FCC), **Ch20,21** **Tx power -53.7 dBm**	**-52.2 dBm max**	-60.8 dBm	-62.3 dBm	-60.4 dBm	-59.6 dBm	PASS	Regulatory subtest #15,13 See Note (2)

MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -8.9 to -16.9 dBm**	**-34.85 dBm max**	-55.6 dBm	-54.9 dBm	-54.7 dBm	-55.1 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -16.9 dBm**	**-54.85 dBm max**	-61.8 dBm	-61.4 dBm	-61.4 dBm	-61.6 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -21.9 dBm**	**-59.85 dBm max**	-65.9 dBm	-65.8 dBm	-65.7 dBm	-65.5 dBm	PASS	Regulatory subtest #15,14 See Note (3)
MEDS into MICS (ETSI), **Ch20,21** **Tx power -16.9 to -26.9 dBm**	**-34.85 dBm max**	-48.2 dBm	-48.1 dBm	-48.2 dBm	-47.9 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -26.9 dBm**	**-54.85 dBm max**	-58.5 dBm	-57.5 dBm	-57.9 dBm	-56.7 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -31.9 dBm**	**-59.85 dBm max**	-63.4 dBm	-62.4 dBm	-62.5 dBm	-62.6 dBm	PASS	Regulatory subtest #15,14

Table 29 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Impedance: 5-j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Spurious Emissions, immediately out-of-band, MICS/MEDS **Tx power:** **-8 dBm MICS** **-11 dBm MEDS**	**-21 dB max**	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	PASS	Regulatory subtest #11,12

MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -11 to -19 dBm**	**-32.2 dBm max**	-57.1 dBm	-55.8 dBm	-57.9 dBm	-54.8 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -19 dBm**	**-52.2 dBm max**	-62.7 dBm	-61.0 dBm	-60.7 dBm	-63.8 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -24 dBm**	**-57.2 dBm max**	-68.1 dBm	-65.6 dBm	-68.1 dBm	-65.8 dBm	PASS	Regulatory subtest #15,13 See Note (3)
MEDS into MICS (FCC), **Ch20,21** **Tx power -34.3 to -55.8 dBm**	**-32.2 dBm max**	-41.8 dBm	-41.4 dBm	-42.5 dBm	-39.4 dBm	PASS	Regulatory subtest #15,13 See Note (1),(2)
MEDS into MICS (FCC), **Ch20,21** **Tx power -55.8 dBm**	**-52.2 dBm max**	-62.6 dBm	-60.1 dBm	-62.5 dBm	-61.8 dBm	PASS	Regulatory subtest #15,13 See Note (2)
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -11 to -19 dBm**	**-34.85 dBm max**	-57.2 dBm	-57.0 dBm	-56.6 dBm	-56.0 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -19 dBm**	**-54.85 dBm max**	-63.1 dBm	-62.4 dBm	-62.6 dBm	-63.0 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -24 dBm**	**-59.85 dBm max**	-67.6 dBm	-66.9 dBm	-67.1 dBm	-66.3 dBm	PASS	Regulatory subtest #15,14 See Note (3)
MEDS into MICS (ETSI), **Ch20,21** **Tx power -19 to -29 dBm**	**-34.85 dBm max**	-50.2 dBm	-48.7 dBm	-49.8 dBm	-48.5 dBm	PASS	Regulatory subtest #15,14

MEDS into MICS (ETSI), **Ch20,21** **Tx power -29 dBm**	**-54.85 dBm max**	-60.7 dBm	-59.0 dBm	-60.2 dBm	-58.9 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -34 dBm**	**-59.85 dBm max**	-64.9 dBm	-63.9 dBm	-64.4 dBm	-61.7 dBm	PASS	Regulatory subtest #15,14

Table 30 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Impedance: 5-j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Spurious Emissions, immediately out-of-band, MICS/MEDS **Tx power:** **-6.8 dBm MICS** **-9.8dBm MEDS**	**-21 dB max**	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	PASS	Regulatory subtest #11,12
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -9.8 to -17.8 dBm**	**-32.2 dBm max**	-54.4 dBm	-54.3 dBm	-55.0 dBm	-51.4 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -17.8 dBm**	**-52.2 dBm max**	-62.0 dBm	-60.3 dBm	-61.1 dBm	-62.2 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -22.8 dBm**	**-57.2 dBm max**	-66.7 dBm	-64.2 dBm	-66.6 dBm	-64.2 dBm	PASS	Regulatory subtest #15,13 See Note (3)
MEDS into MICS (FCC), **Ch20,21** **Tx power -33.1 to -54.6 dBm**	**-32.2 dBm max**	-41.2 dBm	-41.0 dBm	-40.4 dBm	-39.8 dBm	PASS	Regulatory subtest #15,13 See Note (1),(2)

MEDS into MICS (FCC), **Ch20,21** **Tx power -54.6 dBm**	**-52.2 dBm max**	-60.8 dBm	-61.0 dBm	-62.0 dBm	-60.7 dBm	PASS	Regulatory subtest #15,13 See Note (2)
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -9.8 to -17.8 dBm**	**-34.85 dBm max**	-56.5 dBm	-55.9 dBm	-56.0 dBm	-56.1 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -17.8 dBm**	**-54.85 dBm max**	-62.9 dBm	-62.7 dBm	-62.6 dBm	-63.0 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -22.8 dBm**	**-59.85 dBm max**	-67.0 dBm	-66.5 dBm	-66.9 dBm	-66.8 dBm	PASS	Regulatory subtest #15,14 See Note (3)
MEDS into MICS (ETSI), **Ch20,21** **Tx power -17.8 to -27.8 dBm**	**-34.85 dBm max**	-49.9 dBm	-49.1 dBm	-49.3 dBm	-49.3 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -27.8 dBm**	**-54.85 dBm max**	-60.1 dBm	-59.0 dBm	-59.0 dBm	-59.0 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -32.8 dBm**	**-59.85 dBm max**	-64.4 dBm	-63.8 dBm	-64.1 dBm	-63.4 dBm	PASS	Regulatory subtest #15,14

Table 31 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Impedance: 9-j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Spurious Emissions, immediately out-of-band, MICS/MEDS **Tx power:** **-5.7 dBm MICS** **-9.6dBm MEDS**	**-21 dB max**	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	No spurs found above -26dB	PASS	Regulatory subtest #11,12
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -9.6 to -17.6 dBm**	**-32.2 dBm max**	-52.7 dBm	-50.4 dBm	-57.0 dBm	-56.4 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -17.6dBm**	**-52.2 dBm max**	-60.1 dBm	-63.1 dBm	-61.9 dBm	-63.0 dBm	PASS	Regulatory subtest #15,13
MEDS into MICS (FCC), **Ch11-19, 22-30** **Tx power -21.6 dBm**	**-57.2 dBm max**	-67.1 dBm	-67.8 dBm	-65.0 dBm	-67.2 dBm	PASS	Regulatory subtest #15,13 See Note (3)
MEDS into MICS (FCC), **Ch20,21** **Tx power -32 to -53.7dBm**	**-32.2 dBm max**	-40.0 dBm	-39.8 dBm	-40.8 dBm	-38.8 dBm	PASS	Regulatory subtest #15,13 See Note (1),(2)
MEDS into MICS (FCC), **Ch20,21** **Tx power -53.7 dBm**	**-52.2 dBm max**	-62.6 dBm	-61.0 dBm	-62.9 dBm	-60.0 dBm	PASS	Regulatory subtest #15,13 See Note (2)
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -9.6 to -17.6 dBm**	**-34.85 dBm max**	-56.3 dBm	-55.2 dBm	-55.7 dBm	-55.8 dBm	PASS	Regulatory subtest #15,14

MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -17.6dBm**	**-54.85 dBm max**	-62.3 dBm	-62.0 dBm	-62.1 dBm	-62.1 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch11-19, 22-30** **Tx power -21.7dBm**	**-59.85 dBm max**	-66.7 dBm	-66.5 dBm	-66.4 dBm	-66.0 dBm	PASS	Regulatory subtest #15,14 See Note (3)
MEDS into MICS (ETSI), **Ch20,21** **Tx power -17.6 to -27.6dBm**	**-34.85 dBm max**	-49.4 dBm	-48.0 dBm	-49.0 dBm	-47.9 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -27.6dBm**	**-54.85 dBm max**	-59.2 dBm	-58.2 dBm	-58.7 dBm	-57.4 dBm	PASS	Regulatory subtest #15,14
MEDS into MICS (ETSI), **Ch20,21** **Tx power -31.7dBm**	**-59.85 dBm max**	-64.2 dBm	-63.2 dBm	-63.4 dBm	-62.3 dBm	PASS	Regulatory subtest #15,14

Table 32 Unwanted Emissions – Immediate Out-of-band and MEDS into MICS

Notes: (1):Tests requiring a power level of -30.3 dBm plus relaxations depending on impedance are tested with a power level of -30.0 dBm plus relaxations depending on impedance.
(2):Tests requiring a power level of -51.8 dBm plus relaxations depending on impedance are tested with a power level of -52.0 dBm plus relaxations depending on impedance.
(3):Tests requiring a power level of -22.0 dBm plus relaxations depending on impedance are tested with a power level of -19.4dBm plus relaxations depending on impedance.

Conclusion: All results from Unwanted Emissions – Immediate Out-of-band and MEDS into MICS tests show compliance with the specification.

8.5.4 Unwanted Emissions Including Harmonics

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph:

2.3.4.6.4.1.4 (MICS) and sub-paragraphs,

2.3.4.6.4.2.4 (MEDS) and sub-paragraphs

Test Executive Scripts: Tx_Sc15 through Tx_Sc20

This test has a deviation (Deviation #19) from the DVT Plan. See Section 6 DVT Plan Deviations for more information. Note that deviation #19 only specifies the need of data from missing vectors in the DVT plan. Data that aligns with all requirements in deviation #19, as well as requirements specified in ECO 093, was gathered before the deviation was approved. Data processing, review and incorporation of results into the DVT report occurred after ECO 093 and deviation #19 was approved.

Impedance: 50 ohms

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Implant Mode (FCC) MICS, **Power Level -4 to -12 dBm**	**-32.2 dBm max**	-54.6 dBm	-51.5 dBm	-54.9 dBm	-55.3 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.1 Regulatory subtest 2-8,16, 17-23
Implant Mode (FCC) MEDS ch11-20,21-29. **Power Level -7 to -15 dBm**	**-32.2 dBm max**	-53.8 dBm	-50.9 dBm	-55.2 dBm	-55.1 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23
Implant Mode (FCC) MEDS ch 30 **Power Level -7.7 to -15.7 dBm**	**-32.2 dBm max**	-37.3 dBm	-37.7 dBm	-38.0 dBm	-37.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23 See Note (7)
Implant Mode (ETSI) MICS, **Power Level -4 to -12 dBm**	**-10.85 dBm max or equal to average fundamental output power (which-ever is less)**	-30.9 dBm	-33.2 dBm	-29.7 dBm	-33.7 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23

Implant Mode (ETSI) MEDS, **Power Level -7 to -15 dBm**	**-10.85 dBm max or equal to peak fundamental output power (which-ever is less)**	-38.9 dBm	-38.8 dBm	-38.3 dBm	-38.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23
External Mode (MICS) **Power Level -12 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-401MHz** **E: 406-802MHz** **F: 812-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-76.3 B.-73.2 C.-73.2 D.-64.9 E.-66.0 F.-73.2 G.-55.0 dBm	A.-77.1 B.-73.5 C.-73.5 D.-64.4 E.-65.2 F.-73.5 G.-55.6 dBm	A.-76.0 B.-73.6 C.-73.6 D.-64.0 E.-65.3 F.-72.4 G.-55.2 dBm	A.-77.0 B.-73.9 C.-73.9 D.-65.2 E.-66.4 F.-73.5 G.-55.5 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (MEDS) **Power Level -15 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-400MHz** **E: 407-800MHz** **F: 814-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-76.3 B.-72.7 C.-72.7 D.-65.5 E.-65.5 F.-72.9 G.-54.7 dBm	A.-76.0 B.-73.4 C.-73.4 D.-65.4 E.-64.8 F.-73.2 G.-55.4 dBm	A.-76.2 B.-73.4 C.-73.4 D.-64.8 E.-64.7 F.-72.3 G.-55.3 dBm	A.-76.5 B.-74.1 C.-74.1 D.-65.6 E.-66.8 F.-72.7 G.-55.4 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (FCC) outside MICS band edges **Power Levels: A: -12 dBm** **B: -17 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-63.0 dBm B.met at -12dBm	A.-61.6 dBm B.met at -12 dBm	A.-61.8 dBm B.met at -12 dBm	A.-62.7 dBm B.met at -12 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 5-8, 17 For B see Note (1)

External Mode (FCC) MICS Harmonics **Power Levels: A: -12 dBm** **B: -17 dBm**	**A: -44.2** **B: -49.2** **dBm max**	A.-76.4 dBm B.met at -12dBm	A.-77.5 dBm B.met at -12dBm	A.-76.6 dBm B.met at -12dBm	A.-77.5 dBm B.met at -12dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 20 For B see Note (2)
External Mode (FCC) outside MEDS band edges and Harmonics **All MEDS channels except ch 30.** **Power Levels: A: -15 dBm** **B: -20 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-60.6 dBm B.-64.8 dBm	A.-61.3 dBm B.-64.6 dBm	A.-62.0 dBm B.-65.3 dBm	A.-58.6 dBm B.-64.1 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (5)
External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only **Power Level: A:-27.1 dBm** **B:-29.1 dBm**	**A.-52.2** **B.-53.7** **dBm max**	A.-62.6 dBm B.met at -27.1dBm	A.-59.6 dBm B.met at -27.1dBm	A.-60.9 dBm B.met at -27.1dBm	A.-57.8 dBm B.met at -27.1dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (6)
External Mode (ETSI) outside MICS band edges **Power Levels: A: -12 dBm** **B: -17 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-48.2 dBm B.met at -12dBm	A.-48.2 dBm B.met at -12dBm	A.-48.0 dBm B.met at -12dBm	A.-48.0 dBm B.met at -12dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 5, 9, 10, 17 For B see Note (3)
External Mode (ETSI) MICS Harmonics **Power Levels: A: -12 dBm** **B: -17 dBm**	**A: -22.85** **B: -27.85** **dBm max**	A.-76.4 dBm B. met at -12dBm	A.-77.5 dBm B.met at -12dBm	A.-76.6 dBm B.met at -12dBm	A.-77.5 dBm B.met at -12dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 20 For B see Note (4)

External Mode (ETSI) outside MEDS band edges and Harmonics **All MEDS channels** **Power Levels: A: -15 dBm** **B: -20 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-49.6 dBm B.-65.1 dBm	A.-48.8 dBm B.-64.6 dBm	A.-48.6 dBm B.-64.8 dBm	A.-48.6 dBm B.-64.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.5 Regulatory subtest 5, 9,10, 17, 20 For B see Note (5)

Table 33 Unwanted Emissions Including Harmonics

Impedance: 50 Ohms (2:1 VSWR)

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Implant Mode (FCC) MICS, **Power Level -6.1 to -14.1 dBm**	**-32.2 dBm max**	-47.7 dBm	-51.3 dBm	-47.6 dBm	-50.9 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.1 Regulatory subtest 2-8,16, 17-23
Implant Mode (FCC) MEDS ch11-20,21-29. **Power Level -9.1 to -17.1 dBm**	**-32.2 dBm max**	-52.4 dBm	-49.6 dBm	-50.3 dBm	-51.9 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23
Implant Mode (FCC) MEDS ch 30 **Power Level -9.8 to -17.8 dBm**	**-32.2 dBm max**	-38.6 dBm	-36.6 dBm	-36.4 dBm	-37.0 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23 See Note (7)
Implant Mode (ETSI) MICS, **Power Level -6.1 to -14.1 dBm**	**-10.85 dBm max or equal to average fundamental output power (which-ever is less)**	-27.6 dBm	-28.0 dBm	-26.4 dBm	-28.0 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23

Implant Mode (ETSI) MEDS, **Power Level -9.1 to -17.1 dBm**	**-10.85 dBm max or equal to peak fundamental output power (which-ever is less)**	-36.3 dBm	-37.9 dBm	-35.8 dBm	-37.2 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23
External Mode (MICS) **Power Level -14.1 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-401MHz** **E: 406-802MHz** **F: 812-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-76.7 B.-73.1 C.-73.1 D.-65.4 E.-66.8 F.-72.4 G.-54.4 dBm	A.-75.7 B.-73.8 C.-73.8 D.-64.4 E.-65.9 F.-73.1 G.-55.6 dBm	A.-76.2 B.-73.1 C.-73.1 D.-64.7 E.-64.9 F.-72.4 G.-55.0 dBm	A.-76.9 B.-74.3 C.-74.3 D.-65.4 E.-66.5 F.-73.4 G.-55.6 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (MEDS) **Power Level -17.1 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-400MHz** **E: 407-800MHz** **F: 814-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-76.3 B.-73.2 C.-73.2 D.-64.0 E.-64.9 F.-72.9 G.-54.7 dBm	A.-75.7 B.-73.9 C.-73.9 D.-64.6 E.-65.8 F.-72.7 G.-54.4 dBm	A.-76.1 B.-73.1 C.-73.1 D.-66.3 E.-66.5 F.-72.4 G.-55.0 dBm	A.-76.2 B.-73.4 C.-73.4 D.-64.5 E.-67.3 F.-72.5 G.-54.5 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (FCC) outside MICS band edges **Power Levels: A: -14.1 dBm** **B: -19.1 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-62.4 dBm B. met at -14.1dBm	A.-62.0 dBm B. met at -14.1dBm	A.-63.0 dBm B. met at -14.1dBm	A.-62.4 dBm B. met at -14.1dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 5-8, 17 For B see Note (1)

External Mode (FCC) MICS Harmonics **Power Levels: A: -14.1 dBm** **B: -19.1 dBm**	**A: -44.2** **B: -49.2** **dBm max**	A.-75.0 dBm B. met at -14.1dBm	A.-75.8 dBm B. met at -14.1dBm	A.-74.9 dBm B. met at -14.1dBm	A.-75.9 dBm B. met at -14.1dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 20 For B see Note (2)
External Mode (FCC) outside MEDS band edges and Harmonics **All MEDS channels except ch 30.** **Power Levels: A: -17.1 dBm** **B: -22.1 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-60.6 B.-65.7 dBm	A.-60.5 B.-64.4 dBm	A.-61 B.-66.1 dBm	A.-62.5 B.-64.0 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (5)
External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only **Power Levels: A: -29.2 dBm** **B: -31.2 dBm**	**A: -52.2** **B: -53.7** **dBm max**	A.-59.2 dBm B.met at -29.2dBm	-60.9 dBm B.met at -29.2dBm	-60.4 dBm B.met at -29.2dBm	-58.1 dBm B.met at -29.2dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (6)
External Mode (ETSI) outside MICS band edges **Power Levels: A: -14.1 dBm** **B: -19.1 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-47.2 dBm B. met at -14.1dBm	A.-47.2 dBm B. met at -14.1dBm	A.-47.1 dBm B. met at -14.1dBm	A.-47.1 dBm B. met at -14.1dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 5, 9, 10, 17 For B see Note (3)
External Mode (ETSI) MICS Harmonics **Power Levels: A: -14.1 dBm** **B: -19.1 dBm**	**A: -22.85** **B: -27.85** **dBm max**	A.-75.0 dBm B. met at -14.1dBm	A.-75.8 dBm B. met at -14.1dBm	A.-74.9 dBm B. met at -14.1dBm	A.-75.9 dBm B. met at -14.1dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 20 For B see Note (4)

External Mode (ETSI) outside MEDS band edges and Harmonics **All MEDS channels** **Power Levels: A: -17.1 dBm** **B: -22.1 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-49.2 B.-54.2 dBm	A.-48.7 B.-52.6 dBm	A.-48.2 B.-52.9 dBm	A.-47.9 B.-52.5 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.5 Regulatory subtest 5, 9,10, 17, 20 For B see Note (5)

Table 34 Unwanted Emissions Including Harmonics

Impedance: 50-j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Implant Mode (FCC) MICS, **Power Level -4.6 to -12.6 dBm**	**-32.2 dBm max**	-55.1 dBm	-53.1 dBm	-51.8 dBm	-50.9 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.1 Regulatory subtest 2-8,16, 17-23
Implant Mode (FCC) MEDS ch11-20,21-29. **Power Level -7.6 to -15.6 dBm**	**-32.2 dBm max**	-54.5 dBm	-53.6 dBm	-54.8 dBm	-54.9 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23
Implant Mode (FCC) MEDS ch 30 **Power Level -8.3 to -16.3 dBm**	**-32.2 dBm max**	-37.0 dBm	-39.2 dBm	-37.5 dBm	-38.7 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23 See Note (7)
Implant Mode (ETSI) MICS, **Power Level -4.6 to -12.6 dBm**	**-10.85 dBm max or equal to average fundamental output power (which-ever is less)**	-30.7 dBm	-29.7 dBm	-29.2 dBm	-29.5 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23

Implant Mode (ETSI) MEDS, **Power Level -7.6 to -15.6 dBm**	**-10.85 dBm max or equal to peak fundamental output power (which-ever is less)**	-38.8 dBm	-37.7 dBm	-38.2 dBm	-38.0 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23
External Mode (MICS) **Power Level -12.6 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-401MHz** **E: 406-802MHz** **F: 812-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-76.2 B.-72.2 C.-72.2 D.-64.8 E.-66.7 F.-73.0 G.-55.4 dBm	A.-76.8 B.-74.1 C.-74.1 D.-65.2 E.-66.5 F.-73.4 G.-55.7 dBm	A.-75.8 B.-73.3 C.-73.3 D.-64.8 E.-65.9 F.-72.9 G.-55.5 dBm	A.-76.7 B.-73.6 C.-73.6 D.-65.4 E.-66.4 F.-73.1 G.-55.7 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (MEDS) **Power Level -15.6 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-400MHz** **E: 407-800MHz** **F: 814-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-75.9 B.-72.7 C.-72.7 D.-66.3 E.-66.8 F.-72.7 G.-54.8 dBm	A.-76.4 B.-73.4 C.-73.4 D.-64.7 E.-66.5 F.-73.0 G.-55.2 dBm	A.-75.9 B.-72.3 C.-72.3 D.-65.9 E.-66.1 F.-72.6 G.-54.7 dBm	A.-75.8 B.-73.2 C.-73.2 D.-65.0 E.-67.1 F.-72.4 G.-55.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (FCC) outside MICS band edges **Power Levels: A: -12.6 dBm** **B: -17.6 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-62.5 dBm B.met at -12.6dBm	A.-62.6 dBm B.met at -12.6dBm	A.-62.9 dBm B.met at -12.6dBm	A.-63.1 dBm B.met at -12.6dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 5-8, 17 For B see Note (1)

External Mode (FCC) MICS Harmonics **Power Levels: A: -12.6 dBm** **B: -17.6 dBm**	**A: -44.2** **B: -49.2** **dBm max**	A.-75.9 dBm B.met at -12.6dBm	A.-73.0 dBm B.met at -12.6dBm	A.-73.6 dBm B.met at -12.6dBm	A.-72.7 dBm B.met at -12.6dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 20 For B see Note (2)
External Mode (FCC) outside MEDS band edges and Harmonics **All MEDS channels except ch 30.** **Power Levels: A: -15.6 dBm** **B: -20.6 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-61.9 dBm B.-66.5 dBm	A.-63.1 dBm B.-65.6 dBm	A.-61.5 dBm B.-66.2 dBm	A.-61.4 dBm B.-65.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (5)
External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only **Power Levels: A: -27.7 dBm** **B: -29.7 dBm**	**A.-52.2** **B.-53.7** **dBm max**	A.-62.7 dBm B.met at -27.7dBm	A.-60.6 dBm B.met at -27.7dBm	A.-61.9 dBm B.met at -27.7dBm	A.-61.2 dBm B.met at -27.7dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (6)
External Mode (ETSI) outside MICS band edges **Power Levels: A: -12.6 dBm** **B: -17.6 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-47.3 dBm B.met at -12.6dBm	A.-46.7 dBm B.met at -12.6dBm	A.-47.2 dBm B.met at -12.6dBm	A.-46.6 dBm B.met at -12.6dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 5, 9, 10, 17 For B see Note (3)
External Mode (ETSI) MICS Harmonics **Power Levels: A: -12.6 dBm** **B: -17.6 dBm**	**A: -22.85** **B: -27.85** **dBm max**	A.-75.9 dBm B. met at -12.6dBm	A.-73.0 dBm B.met at -12.6dBm	A.-73.6 dBm B.met at -12.6dBm	A.-72.7 dBm B.met at -12.6dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 20 For B see Note (4)

External Mode (ETSI) outside MEDS band edges and Harmonics **All MEDS channels** **Power Levels: A: -15.6 dBm** **B: -20.6 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-49.5 dBm B.-54.0 dBm	A.-48.2 dBm B.-52.9 dBm	A.-49.0 dBm B.-53.3 dBm	A.-48.3 dBm B.-52.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.5 Regulatory subtest 5, 9,10, 17, 20 For B see Note (5)

Table 35 Unwanted Emissions Including Harmonics

Impedance: 5+j0

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Implant Mode (FCC) MICS, **Power Level -5.9 to -13.9 dBm**	**-32.2 dBm max**	-51.8 dBm	-55.3 dBm	-51.5 dBm	-53.2 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.1 Regulatory subtest 2-8,16, 17-23
Implant Mode (FCC) MEDS ch11-20,21-29. **Power Level -8.9 to -16.9 dBm**	**-32.2 dBm max**	-56.3 dBm	-55.1 dBm	-55.2 dBm	-55.1 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23
Implant Mode (FCC) MEDS ch 30 **Power Level -9.6 to -17.6 dBm**	**-32.2 dBm max**	-40.2 dBm	-39.7 dBm	-42.3 dBm	-39.4 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23 See Note (7)
Implant Mode (ETSI) MICS, **Power Level -5.9 to -13.9 dBm**	**-10.85 dBm max or equal to average fundamental output power (which-ever is less)**	-26.9 dBm	-29.5 dBm	-26.4 dBm	-29.2 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23

Implant Mode (ETSI) MEDS, **Power Level -8.9 to -16.9 dBm**	**-10.85 dBm max or equal to peak fundamental output power (which-ever is less)**	-35.4 dBm	-35.7 dBm	-35.5 dBm	-35.0 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23
External Mode (MICS) **Power Level -13.9 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-401MHz** **E: 406-802MHz** **F: 812-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-76.7 B.-72.5 C.-72.5 D.-67.0 E.-66.3 F.-72.5 G.-55.5 dBm	A.-76.0 B.-73.2 C.-73.2 D.-66.4 E.-67.2 F.-71.9 G.-55.5 dBm	A.-76.1 B.-72.4 C.-72.4 D.-65.9 E.-66.0 F.-72.2 G.-55.4 dBm	A.-76.0 B.-73.5 C.-73.5 D.-65.9 E.-68.0 F.-72.8 G.-55.3 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (MEDS) **Power Level -16.9 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-400MHz** **E: 407-800MHz** **F: 814-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-76.4 B.-72.6 C.-72.6 D.-64.6 E.-67.0 F.-71.9 G.-54.3 dBm	A.-77.4 B.-73.3 C.-73.3 D.-66.1 E.-68.4 F.-72.3 G.-55.7 dBm	A.-76.2 B.-72.4 C.-72.4 D.-66.6 E.-66.8 F.-72.5 G.-54.9 dBm	A.-76.5 B.-72.6 C.-72.6 D.-66.6 E.-67.9 F.-72.2 G.-55.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (FCC) outside MICS band edges **Power Levels: A: -13.9 dBm** **B: -18.9 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-65.2 dBm B.met at -13.9dBm	A.-64.3 dBm B.met at -13.9 dBm	A.-63.6 dBm B.met at -13.9 dBm	A.-65.2 dBm B.met at -13.9 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 5-8, 17 For B see Note (1)

External Mode (FCC) MICS Harmonics **Power Levels: A: -13.9 dBm** **B: -18.9 dBm**	**A: -44.2** **B: -49.2** **dBm max**	A.-77.0 dBm B.met at -13.9dBm	A.-76.8 dBm B.met at -13.9dBm	A.-77.0 dBm B.met at -13.9dBm	A.-76.7 dBm B.met at -13.9dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 20 For B see Note (2)
External Mode (FCC) outside MEDS band edges and Harmonics **All MEDS channels except ch 30.** **Power Levels: A: -16.9 dBm** **B: -21.9 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-63.0 dBm B.-66.5 dBm	A.-60.4 dBm B.-65.6 dBm	A.-63.4 dBm B.-66.2 dBm	A.-62.4 dBm B.-65.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (5)
External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only **Power Levels: A.-29 dBm** **B.-32 dBm**	**A:-52.2** **B:-53.7** **dBm max**	A.-62.4 dBm B.met at -29dBm	A.-61.3 dBm B.met at -29dBm	A.-61.0 dBm B.met at -29dBm	A.-60.7 dBm B.met at -29dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (6)
External Mode (ETSI) outside MICS band edges **Power Levels: A: -13.9 dBm** **B: -18.9 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-44.4 dBm B.met at -13.9dBm	A.-45.0 dBm B.met at -13.9dBm	A.-44.3 dBm B.met at -13.9dBm	A.-44.4 dBm B.met at -13.9dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 5, 9, 10, 17 For B see Note (3)
External Mode (ETSI) MICS Harmonics **Power Levels: A: -13.9 dBm** **B: -18.9 dBm**	**A: -22.85** **B: -27.85** **dBm max**	A.-77.0 dBm B. met at -13.9dBm	A.-76.8 dBm B.met at -13.9dBm	A.-77.0 dBm B.met at -13.9dBm	A.-76.7 dBm B.met at -13.9dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 20 For B see Note (4)

External Mode (ETSI) outside MEDS band edges and Harmonics **All MEDS channels** **Power Levels: A: -16.9 dBm** **B: -21.9 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-46.1 dBm B.-50.7 dBm	A.-45.2 dBm B.-50.1 dBm	A.-45.4 dBm B.-50.2 dBm	A.-45.3 dBm B.-49.7 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.5 Regulatory subtest 5, 9,10, 17, 20 For B see Note (5)

Table 36 Unwanted Emissions Including Harmonics

Impedance: 5-j60

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Implant Mode (FCC) MICS, **Power Level -8 to -16 dBm**	**-32.2 dBm max**	-53.0 dBm	-54.9 dBm	-46.5 dBm	-52.5 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.1 Regulatory subtest 2-8,16, 17-23
Implant Mode (FCC) MEDS ch11-20,21-29. **Power Level -11 to -19 dBm**	**-32.2 dBm max**	-55.0 dBm	-55.8 dBm	-55.5 dBm	-56.0 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23
Implant Mode (FCC) MEDS ch 30 **Power Level -11.7 to -19.7 dBm**	**-32.2 dBm max**	-42.2 dBm	-39.2 dBm	-40.7 dBm	-40.7 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23 See Note (7)
Implant Mode (ETSI) MICS, **Power Level -8 to -16 dBm**	**-10.85 dBm max or equal to average fundamental output power (which-ever is less)**	-22.4 dBm	-22.9 dBm	-20.6 dBm	-22.1 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23

Implant Mode (ETSI) MEDS, **Power Level -11 to -19 dBm**	**-10.85 dBm max or equal to peak fundamental output power (which-ever is less)**	-34.3 dBm	-31.8 dBm	-33.5 dBm	-32.4 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23
External Mode (MICS) **Power Level -16 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-401MHz** **E: 406-802MHz** **F: 812-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-75.5 B.-71.9 C.-71.9 D.-66.6 E.-67.7 F.-71.6 G.-55.3 dBm	A.-77.0 B.-71.9 C.-71.9 D.-67.0 E.-67.7 F.-70.8 G.-55.8 dBm	A.-76.5 B.-70.6 C.-70.6 D.-66.4 E.-68.5 F.-71.5 G.-54.0 dBm	A.-76.5 B.-71.0 C.-71.0 D.-67.2 E.-69.0 F.-71.5 G.-55.4 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (MEDS) **Power Level -19 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-400MHz** **E: 407-800MHz** **F: 814-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-75.9 B.-70.8 C.-70.8 D.-69.5 E.-68.1 F.-71.5 G.-54.7 dBm	A.-76.2 B.-72.5 C.-72.5 D.-68.5 E.-69.5 F.-71.1 G.-55.7 dBm	A.-76.0 B.-71.2 C.-71.2 D.-68.5 E.-68.1 F.-71.0 G-55.2 dBm	A.-75.9 B.-71.4 C.-71.4 D.-68.9 E.-69.0 F.-71.0 G.-54.2 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (FCC) outside MICS band edges **Power Levels: A: -16 dBm** **B: -21 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-63.9 dBm B. met at -16dBm	A.-64.1 dBm B. met at -16dBm	A.-65.8 dBm B. met at -16dBm	A.-65.5 dBm B. met at -16dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 5-8, 17 For B see Note (1)

External Mode (FCC) MICS Harmonics **Power Levels: A: -16 dBm** **B: -21 dBm**	**A: -44.2** **B: -49.2** **dBm max**	A.-76.7 dBm B. met at -16dBm	A.-76.9 dBm B. met at -16dBm	A.-76.6 dBm B. met at -16dBm	A.-76.9 dBm B. met at -16dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 20 For B see Note (2)
External Mode (FCC) outside MEDS band edges and Harmonics **All MEDS channels except ch 30.** **Power Levels: A: -19 dBm** **B: -24 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-64.5 dBm B.-69.6 dBm	A.-64.9 dBm B.-66.8 dBm	A.-63.6 dBm B.-67.1 dBm	A.-63.4 dBm B.-67.4 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (5)
External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only **Power Levels: A. -31.1 dBm** **B. -33.5 dBm**	**A.-52.2** **B.-53.7** **dBm max**	A.-64.3 dBm B.met at -31.1dBm	A.-62.1 dBm B.met at -31.1dBm	A.-62.2 dBm B.met at -31.1dBm	A.-59.0 dBm B.met at -31.1dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (6)
External Mode (ETSI) outside MICS band edges **Power Levels: A: -16 dBm** **B: -21 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-42.7 dBm B. met at -16dBm	A.-42.2 dBm B. met at -16dBm	A.-41.9 dBm B. met at -16dBm	A.-42.1 dBm B. met at -16dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 5, 9, 10, 17 For B see Note (3)
External Mode (ETSI) MICS Harmonics **Power Levels: A: -16 dBm** **B: -21 dBm**	**A: -22.85** **B: -27.85** **dBm max**	A.-76.7 dBm B. met at -16dBm	A.-76.9 dBm B. met at -16dBm	A.-76.6 dBm B. met at -16dBm	A.-76.9 dBm B. met at -16dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 20 For B see Note (4)

External Mode (ETSI) outside MEDS band edges and Harmonics **All MEDS channels** **Power Levels: A: -19 dBm** **B: -24 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-45.0 B.-49.6 dBm	A.-42.6 B.-47.3 dBm	A.-43.7 B.-48.3 dBm	A.-42.7 B.-46.7 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.5 Regulatory subtest 5, 9,10, 17, 20 For B see Note (5)

Table 37 Unwanted Emissions Including Harmonics

Impedance: 5-j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Implant Mode (FCC) MICS, **Power Level -6.8 to -14.8 dBm**	**-32.2 dBm max**	-54.6 dBm	-55.7 dBm	-54.1 dBm	-56.1 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.1 Regulatory subtest 2-8,16, 17-23
Implant Mode (FCC) MEDS ch11-20,21-29. **Power Level -9.8 to -17.8 dBm**	**-32.2 dBm max**	-54.2 dBm	-55.8 dBm	-55.4 dBm	-55.4 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23
Implant Mode (FCC) MEDS ch 30 **Power Level -10.5 to -18.5 dBm**	**-32.2 dBm max**	-42.3 dBm	-40.0 dBm	-40.2 dBm	-39.7 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23 See Note (7)
Implant Mode (ETSI) MICS, **Power Level -6.8 to -14.8 dBm**	**-10.85 dBm max or equal to average fundamental output power (which-ever is less)**	-27.6 dBm	-28.7 dBm	-27.5 dBm	-27.9 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23

Implant Mode (ETSI) MEDS, **Power Level -9.8 to -17.8 dBm**	**-10.85 dBm max or equal to peak fundamental output power (which-ever is less)**	-36.1 dBm	-36.8 dBm	-36.6 dBm	-35.9 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23
External Mode (MICS) **Power Level -14.8 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-401MHz** **E: 406-802MHz** **F: 812-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-75.5 B.-71.9 C.-71.9 D.-67.0 E.-68.0 F.-71.8 G.-54.7 dBm	A.-76.6 B.-73.2 C.-73.2 D.-66.4 E.-67.6 F.-72.2 G.-54.8 dBm	A.-76.5 B.-72.5 C.-72.5 D.-67.0 E.-66.9 F.-71.9 G.-55.9 dBm	A.-77.0 B.-73.1 C.-73.1 D.-67.1 E.-69.2 F.-72.2 G.-56.0 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (MEDS) **Power Level -17.8 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-400MHz** **E: 407-800MHz** **F: 814-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-75.5 B.-72.3 C.-72.3 D.-67.3 E.-68.0 F.-71.8 G.-54.7 dBm	A.-75.4 B.-73.2 C.-73.2 D.-67.3 E.-67.9 F.-72.0 G.-54.2 dBm	A.-75.9 B.-72.1 C.-72.1 D.-66.5 E.-66.9 F.-71.4 G.-54.5 dBm	A.-76.3 B.-73.1 C.-73.0 D.-68.4 E.-69.2 F.-71.9 G.-55.2 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (FCC) outside MICS band edges **Power Levels: A: -14.8 dBm** **B: -19.8 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-63.6 dBm B. met at -14.8dBm	A.-64.4 dBm B. met at -14.8dBm	A.-65.1 dBm B. met at -14.8dBm	A.-62.2 dBm B. met at -14.8dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 5-8, 17 For B see Note (1)

External Mode (FCC) MICS Harmonics **Power Levels: A: -14.8 dBm** **B: -19.8 dBm**	**A: -44.2** **B: -49.2** **dBm max**	A.-76.5 dBm B. met at -14.8dBm	A.-76.5 dBm B. met at -14.8dBm	A.-76.3 dBm B. met at -14.8dBm	A.-76.4 dBm B. met at -14.8dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 20 For B see Note (2)
External Mode (FCC) outside MEDS band edges and Harmonics **All MEDS channels except ch 30.** **Power Levels: A: -17.8 dBm** **B: -22.8 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-62.8 B.-69.8 dBm	A.-64.3 B.-68.8 dBm	A.-63.2 B.-68.3 dBm	A.-64.5 B.-68.2 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (5)
External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only **Power Levels: A.-29.9 dBm** **B.-31.9 dBm**	**A.-52.2** **B.-53.7** **dBm max**	A.-62.0 dBm B.met at -29.9dBm	A.-60.0 dBm B.met at -29.9dBm	A.-62.9 dBm B.met at -29.9dBm	A.-59.0 dBm B.met at -29.9dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (6)
External Mode (ETSI) outside MICS band edges **Power Levels: A: -14.8 dBm** **B: -19.8 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-45.3 dBm B. met at -14.8dBm	A.-45.4 dBm B. met at -14.8dBm	A.-45.1 dBm B. met at -14.8dBm	A.-45.0 dBm B. met at -14.8dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 5, 9, 10, 17 For B see Note (3)
External Mode (ETSI) MICS Harmonics **Power Levels: A: -14.8 dBm** **B: -19.8 dBm**	**A: -22.85** **B: -27.85** **dBm max**	A.-76.5 dBm B. met at -14.8dBm	A.-76.5 dBm B. met at -14.8dBm	A.-76.3 dBm B. met at -14.8dBm	A.-76.4 dBm B. met at -14.8dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 20 For B see Note (4)

External Mode (ETSI) outside MEDS band edges and Harmonics **All MEDS channels** **Power Levels: A: -17.8 dBm** **B: -22.8 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-47.6 B.-52.3 dBm	A.-46.0 B-50.3 dBm	A.-46.8 B-50.8 dBm	A.-46.0 B.-50.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.5 Regulatory subtest 5, 9,10, 17, 20 For B see Note (5)

Table 38 Unwanted Emissions Including Harmonics

Impedance: 9-j20

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Implant Mode (FCC) MICS, **Power Level -5.7 to -13.7 dBm**	**-32.2 dBm max**	-51.8 dBm	-55.0 dBm	-53.5 dBm	-54.5 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.1 Regulatory subtest 2-8,16, 17-23
Implant Mode (FCC) MEDS ch11-20,21-29. **Power Level -9.6 to -17.6 dBm**	**-32.2 dBm max**	-55.6 dBm	-54.8 dBm	-55.3 dBm	-56.2 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23
Implant Mode (FCC) MEDS ch 30 **Power Level -10.3 to -18.3 dBm**	**-32.2 dBm max**	-40.5 dBm	-41.6 dBm	-40.5 dBm	-38.9 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.1 Regulatory subtest 2-5, 16, 17-23 See Note (7)
Implant Mode (ETSI) MICS, **Power Level -5.7 to -13.7 dBm**	**-10.85 dBm max or equal to average fundamental output power (which-ever is less)**	-26.6 dBm	-29.1 dBm	-27.3 dBm	-29.5 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23

Implant Mode (ETSI) MEDS, **Power Level -9.6 to -17.6 dBm**	**-10.85 dBm max or equal to peak fundamental output power (whichever is less)**	-38.7 dBm	-38.0 dBm	-38.1 dBm	-37.9 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.2 Regulatory subtest 2-5, 9-10, 16, 17-23
External Mode (MICS) **Power Level -13.7 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-401MHz** **E: 406-802MHz** **F: 812-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-75.6 B.-72.8 C.-72.8 D.-67.0 E.-67.2 F.-72.5 G.-55.4 dBm	A.-77.4 B.-72.5 C.-72.5 D.-66.3 E.-67.2 F.-72.9 G.-56.2 dBm	A.-76.2 B.-73.4 C.-73.4 D.-65.2 E.-66.6 F.-72.0 G.-55.3 dBm	A.-76.9 B.-73.4 C.-73.4 D.-66.0 E.-67.1 F.-72.9 G.-56.1 dBm	PASS	A17245 section: 2.3.4.6.4.1.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (MEDS) **Power Level -17.6 dBm and below.** **Frequency:** **A: 25-30MHz** **B: 30-88MHz** **C: 88-216MHz** **D: 216-400MHz** **E: 407-800MHz** **F: 814-960MHz** **G: 960MHz – 4.06GHz**	**A: -35.9** **B: -63.2** **C: -59.7** **D: -57.2** **E: -57.2** **F: -57.2** **G: -49.2** **dBm** **max**	A.-75.8 B.-72.6 C.-72.6 D.-67.3 E.-67.0 F.-72.5 G.-55.0 dBm	A.-76.2 B.-73.3 C.-73.3 D.-66.5 E.-68.6 F.-73.1 G.-55.1 dBm	A.-75.2 B.-72.7 C.-72.7 D.-67.2 E.-68.0 F.-71.7 G.-54.8 dBm	A.-76.5 B.-72.3 C.-72.3 D.-67.3 E.-68.3 F.-72.4 G.-55.6 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.3 Regulatory subtests A: subtest 2 B: subtest 3 C: subtest 3 D: subtest 3-4 E: subtest 18-19 F: subtest 21 G: subtest 21-23
External Mode (FCC) outside MICS band edges **Power Levels: A: -13.7 dBm** **B: -18.7 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-63.4 dBm B. met at -13.7dBm	A.-63.5 dBm B. met at -13.7dBm	A.-62.2 dBm B. met at -13.7dBm	A.-63.4 dBm B. met at -13.7dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 5-8, 17 For B see Note (1)

External Mode (FCC) MICS Harmonics **Power Levels: A: -13.7 dBm** **B: -18.7 dBm**	**A: -44.2** **B: -49.2** **dBm max**	A.-76.5 dBm B. met at -13.7dBm	A.-76.9 dBm B. met at -13.7dBm	A.-76.7 dBm B. met at -13.7dBm	A.-77.0 dBm B. met at -13.7dBm	PASS	A17245 section: 2.3.4.6.4.1.4.4 Regulatory subtest 20 For B see Note (2)
External Mode (FCC) outside MEDS band edges and Harmonics **All MEDS channels except ch 30.** **Power Levels: A: -17.6 dBm** **B: -22.6 dBm**	**A: -52.2** **B: -57.2** **dBm max**	A.-62.4 B.-67.8 dBm	A.-62.5 B.-65.3 dBm	A.-63.2 B.-68.1 dBm	A.-64.1 B.-67.3 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (5)
External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only **Power Levels: A.-29.7 dBm** **B.-31.7 dBm**	**A.-52.2** **B.-53.7** **dBm max**	A.-62.7 dBm B.met at -29.7dBm	A.-60.7 dBm B.met at -29.7dBm	A.-61.1 dBm B.met at -29.7dBm	A.-61.0 dBm B.met at -29.7dBm	PASS	A17245 section: 2.3.4.6.4.2.4.4 Regulatory subtest 5-8, 17,20 For B see Note (6)
External Mode (ETSI) outside MICS band edges **Power Levels: A: -13.7 dBm** **B: -18.7 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-45.6 dBm B. met at -13.7dBm	A.-46.0 dBm B. met at -13.7dBm	A.-45.5 dBm B. met at -13.7dBm	A.-45.7 dBm B. met at -13.7dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 5, 9, 10, 17 For B see Note (3)
External Mode (ETSI) MICS Harmonics **Power Levels: A: -13.7 dBm** **B: -18.7 dBm**	**A: -22.85** **B: -27.85** **dBm max**	A.-76.5 dBm B. met at -13.7dBm	A.-76.9 dBm B. met at -13.7dBm	A.-76.7 dBm B. met at -13.7dBm	A.-77.0 dBm B. met at -13.7dBm	PASS	A17245 section: 2.3.4.6.4.1.4.5 Regulatory subtest 20 For B see Note (4)

External Mode (ETSI) outside MEDS band edges and Harmonics **All MEDS channels** **Power Levels: A: -17.6 dBm** **B: -22.6 dBm**	**A: -30.85** **B: -35.85** **dBm max**	A.-49.0 B.-53.0 dBm	A.-47.5 B.-51.8 dBm	A.-48.2 B.-52.2 dBm	A.-47.5 B.-51.0 dBm	PASS	A17245 section: 2.3.4.6.4.2.4.5 Regulatory subtest 5, 9,10, 17, 20 For B see Note (5)

Table 39 Unwanted Emissions Including Harmonics

Notes:

(1) External Mode (FCC) outside MICS band edges - Must meet -57.2dBm with Tx Power set to min -17dBm plus relaxations depending on impedance. We are testing at -12dBm plus relaxations depending on impedance and are passing. TX ADJ/ALT-CPR, synthesizer SSB phase noise, harmonics, and general TX spurious RF outputs, will all decrease at a dB-per-dB basis or more. Therefore, testing at a 5 dB higher TX RF output power than specified, for a fixed output level TX emission is requirement, is significantly more worst-case than if tested at the originally specified TX RF output power level. This is the case with this requirement.

(2) External Mode (FCC) MICS Harmonics - Must meet -49.2dBm with Tx Power set to min -17dBm plus relaxations depending on impedance. We are testing at -12dBm plus relaxations depending on impedance and are passing. TX ADJ/ALT-CPR, synthesizer SSB phase noise, harmonics, and general TX spurious RF outputs, will all decrease at a dB-per-dB basis or more. Therefore, testing at a 5 dB higher TX RF output power than specified, for a fixed output level TX emission is requirement, is significantly more worst-case than if tested at the originally specified TX RF output power level. This is the case with this requirement.

(3) External Mode (ETSI) outside MICS band edges - Must meet -35.85dBm with Tx Power set to min -17dBm plus relaxations depending on impedance. We are testing at -12dBm plus relaxations depending on impedance and are passing. TX ADJ/ALT-CPR, synthesizer SSB phase noise, harmonics, and general TX spurious RF outputs, will all decrease at a dB-per-dB basis or more. Therefore, testing at a 5 dB higher TX RF output power than specified, for a fixed output level TX emission is requirement, is significantly more worst-case than if tested at the originally specified TX RF output power level. This is the case with this requirement.

(4) External Mode (ETSI) MICS harmonics - Must meet -27.85dBm with Tx Power set to min -17dBm plus relaxations depending on impedance. We are testing at -12dBm plus relaxations depending on impedance and are passing. TX ADJ/ALT-CPR, synthesizer SSB phase noise, harmonics, and general TX spurious RF outputs, will all decrease at a dB-per-dB basis or more. Therefore, testing at a 5 dB higher TX RF output power than specified, for a fixed output level TX emission is requirement, is significantly more worst-case than if tested at the originally specified TX RF output power level. This is the case with this requirement.

(5) External Mode (ETSI) outside MICS band edges and harmonics - Must meet -35.85dBm with Tx Power set to min -20dBm plus relaxations depending on impedance. We are testing at -19.4dBm plus relaxations depending on impedance and are passing. TX ADJ/ALT-CPR, synthesizer SSB phase noise, harmonics, and general TX spurious RF outputs, will all decrease at a dB-per-dB basis or more. Therefore, testing at a 5 dB higher TX RF output power than specified, for a fixed output level TX emission is requirement, is significantly more worst-case than if tested at the originally specified TX RF output power level. This is the case with this requirement.

(6) External Mode (FCC) outside MEDS band edges and Harmonics ch 30 only - Must meet -53.7dBm with Tx Power set to min -29.1dBm plus relaxations depending on impedance. We are testing at -27.1dBm plus relaxations depending on impedance and are passing. TX ADJ/ALT-CPR, synthesizer SSB phase noise, harmonics, and general TX spurious RF outputs, will all decrease at a dB-per-dB basis or more. Therefore, testing at a 5 dB higher TX RF output power than specified, for a fixed output level TX emission is requirement, is significantly more worst-case than if tested at the originally specified TX RF output power level. This is the case with this requirement.

(7) Implant Mode (FCC) MEDS ch 30 - Must meet -32.2dBm with Tx Power set to -7.7dBm and -15.7dBm plus relaxations depending on impedance. We are testing at -7dBm and -15dBm plus relaxations depending on impedance. TX ADJ/ALT-CPR, synthesizer SSB phase noise, harmonics, and general TX spurious RF outputs, will all decrease at a dB-per-dB basis or more. Therefore, testing at a 5 dB higher TX RF output power than specified, for a fixed output level TX emission is requirement, is significantly more worst-case than if tested at the originally specified TX RF output power level. This is the case with this requirement.

Conclusion: All results from TX Regulatory Unwanted Emissions Including Harmonics tests show compliance with the specification.

8.6 Transmitter Settling Time

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.6.6

DVT Plan: 6.3.6

Test Executive Scripts: Tx_Sc21

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Tx Power Settling Time **TelM**	**500 us max**	131 us	131us	131us	131us	PASS	Start-up sequence based on digital control.
Tx Frequency Settling Time **TelM**	**500 us max**	101 us	97us	97us	93us	PASS	
Tx Power Settling Time **TelC**	**2000 us max**	131 us	131us	127us	131us	PASS	
Tx Frequency Settling Time **TelC**	**2000 us max**	97 us	97us	97us	93us	PASS	

Table 40 Transmitter Settling Time

Conclusion: All results from Transmitter Settling Time tests show compliance with the specification.

9 Receiver Test Results

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5

9.1 Rx Filter -3dB Bandwidth

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.3

DVT Plan: 6.4.3

Test Executive Script: Rx_Sc1

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Frequency (-3dB) TelC Narrowband	**26 kHz min**	34KHz	34KHz	34KHz	33KHz	PASS	Measured using baseband, not at carrier frequency. (Spec is 52 kHz at carrier frequency)
Frequency (-3dB) TelC Wideband	**110 kHz min**	134KHz	135KHz	129KHz	134KHz	PASS	Measured using baseband, not at carrier frequency. (Spec is 220 kHz at carrier frequency)

Table 41 Rx Filter -3dB Bandwidth (TelC)

Conclusion: All results from Rx Filter -3dB Bandwidth (TelC) tests show compliance with the specification.

9.2 Sensitivity (TelM and TelC)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.2

DVT Plan: 6.4.4

Test Executive Script: Rx_Sc2 through Rx_Sc8

Impedance: 50 Ohm

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Sensitivity Level (0.01 PER) **Mode 1**	-106 dBm max	-108.50 dBm	-108.50 dBm	-108.50 dBm	-108.50 dBm	PASS	Test step resolution is .25dB but different seed levels yield results that don't fall on .25 increments.
Sensitivity Level (0.01 PER) **Mode 2**	-99.5 dBm max	-103.50 dBm	-104.00 dBm	-104.00 dBm	-103.75 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 12.6%EVM**	-98.6 dBm max	-102.85 dBm	-103.60 dBm	-103.60 dBm	-103.35 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 3**	-100.5 dBm max	-101.75 dBm	-102.00 dBm	-102.00 dBm	-102.00 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4**	-95 dBm max	-100.25 dBm	-100.50 dBm	-100.25 dBm	-100.25 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4 12.6%EVM**	-94.1 dBm max	-99.85 dBm	-100.10 dBm	-99.85 dBm	-100.35 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze Off**	-94.5 dBm max	-98.25 dBm	-98.75 dBm	-98.50 dBm	-98.50 dBm	PASS	

Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze Off**	-91.9 dBm max	-97.90 dBm	-98.15 dBm	-97.90 dBm	-98.40 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze On**	-95.4 dBm max	-98.90 dBm	-99.40 dBm	-98.90 dBm	-99.15 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze On**	-93.3 dBm max	-98.80 dBm	-99.05 dBm	-98.80 dBm	-99.05 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 6**	-103 dBm max	-106.50 dBm	-106.75 dBm	-106.50 dBm	-106.50 dBm	PASS	

Table 42 Sensitivity (TelM and TelC) for 50 Ohm

Impedance: 2:1 VSWR

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Sensitivity Level (0.01 PER) **Mode 1**	-103.9dBm max	-107.75 dBm	-107.50 dBm	-107.75 dBm	-107.50 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2**	-97.4 dBm max	-102.50 dBm	-103.00 dBm	-103.25 dBm	-103.00 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 12.6%EVM**	-97.4 dBm max	-102.10 dBm	-102.85 dBm	-102.10 dBm	-102.60 dBm	PASS	

Sensitivity Level (0.01 PER) **Mode 3**	-98.4 dBm max	-100.25 dBm	-100.50 dBm	-100.75 dBm	-100.50 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4**	-92.9 dBm max	-99.25 dBm	-99.50 dBm	-99.25 dBm	-99.50 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4 12.6%EVM**	-92.9 dBm max	-98.85 dBm	-99.10 dBm	-98.85 dBm	-99.10 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze Off**	-92.4 dBm max	-97.25 dBm	-97.50 dBm	-97.50 dBm	-97.50 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze Off**	-89.8 dBm max	-96.90 dBm	-97.40 dBm	-97.15 dBm	-97.15 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze On**	-93.3 dBm max	-97.90 dBm	-98.15 dBm	-98.15 dBm	-97.90 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze On**	-91.2 dBm max	-98.05 dBm	-98.05 dBm	-97.80 dBm	-97.80 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 6**	-100.9 dBm max	-105.50 dBm	-105.75 dBm	-105.50 dBm	-105.50 dBm	PASS	

Table 43 Sensitivity (TelM and TelC) for 2:1 VSWR

Impedance: 50– j60

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Sensitivity Level (0.01 PER) **Mode 1**	-105.4 dBm max	-107.90 dBm	-107.90 dBm	-107.90 dBm	-108.15 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2**	-98.9 dBm max	-103.15 dBm	-102.90 dBm	-103.15 dBm	-103.65 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 12.6%EVM**	-98 dBm max	-102.25 dBm	-103.00 dBm	-102.75 dBm	-103.25 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 3**	-99.9 dBm max	-101.65 dBm	-101.40 dBm	-101.65 dBm	-101.40 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4**	-94.4 dBm max	-99.40 dBm	-99.65 dBm	-99.65 dBm	-99.65 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4 12.6%EVM**	-93.5 dBm max	-98.75 dBm	-99.50 dBm	-99.00 dBm	-99.50 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze Off**	-93.9 dBm max	-97.40 dBm	-97.65 dBm	-97.65 dBm	-97.65 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze Off**	-91.3 dBm max	-97.05 dBm	-97.30 dBm	-97.05 dBm	-97.80 dBm	PASS	

Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze On**	-94.8 dBm max	-98.05 dBm	-98.30 dBm	-98.05 dBm	-98.30 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze On**	-92.7 dBm max	-97.95 dBm	-98.20 dBm	-97.70 dBm	-98.20 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 6**	-102.4 dBm max	-105.65 dBm	-105.65 dBm	-105.40 dBm	-105.65 dBm	PASS	

Table 44 Sensitivity (TelM and TelC) for 50-j60 Load Impedance

Impedance: 5 + j0

Test Parameter	**Spec**	**sn316**	**sn317**	**sn318**	**sn320**	**Compliance**	**Notes**
Sensitivity Level (0.01 PER) **Mode 1**	-104.1 dBm max	-107.35 dBm	-107.60 dBm	-107.35 dBm	-107.60 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2**	-97.6 dBm max	-102.35 dBm	-102.85 dBm	-102.85 dBm	-103.10 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 12.6%EVM**	-96.7 dBm max	-102.20 dBm	-102.45 dBm	-102.45 dBm	-102.70 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 3**	-98.6 dBm max	-102.60 dBm	-102.60 dBm	-102.85 dBm	-102.85 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4**	-93.1 dBm max	-98.60 dBm	-99.10 dBm	-98.85 dBm	-99.10 dBm	PASS	

Sensitivity Level (0.01 PER) **Mode 4 12.6%EVM**	-92.2 dBm max	-98.45 dBm	-98.70 dBm	-97.95 dBm	-99.20 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze Off**	-92.6 dBm max	-96.85 dBm	-96.85 dBm	-97.10 dBm	-97.35 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze Off**	-90 dBm max	-96.50 dBm	-96.75 dBm	-96.75 dBm	-97.00 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze On**	-93.5 dBm max	-97.25 dBm	-97.50 dBm	-97.50 dBm	-97.50 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze On**	-91.4 dBm max	-97.40 dBm	-97.40 dBm	-97.40 dBm	-97.65 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 6**	-101.1 dBm max	-104.85 dBm	-105.10 dBm	-104.85 dBm	-105.10 dBm	PASS	

Table 45 Sensitivity (TelM and TelC) for 5+j0 Load Impedance

Impedance: 5 – j60

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Sensitivity Level (0.01 PER) **Mode 1**	-102 dBm max	-103.00 dBm	-104.75 dBm	-104.75 dBm	-105.25 dBm	PASS	sn316, Ch.30, 55C affected by SAW filter roll off
Sensitivity Level (0.01 PER) **Mode 2**	-95.5 dBm max	-98.25 dBm	-100.00 dBm	-100.50 dBm	-101.00 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 12.6%EVM**	-94.6 dBm max	-97.85 dBm	-99.85 dBm	-99.85 dBm	-100.10 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 3**	-96.5 dBm max	-98.75 dBm	-101.00 dBm	-101.00 dBm	-101.00 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4**	-91 dBm max	-95.00 dBm	-96.50 dBm	-96.75 dBm	-97.00 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4 12.6%EVM**	-90.1 dBm max	-95.10 dBm	-96.35 dBm	-96.35 dBm	-96.35 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze Off**	-90.5 dBm max	-93.50 dBm	-95.00 dBm	-95.25 dBm	-95.25 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze Off**	-87.9 dBm max	-93.15 dBm	-94.90 dBm	-94.65 dBm	-95.15 dBm	PASS	

Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze On**	-91.4 dBm max	-93.90 dBm	-95.40 dBm	-95.65 dBm	-95.65 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze On**	-89.3 dBm max	-94.05 dBm	-95.30 dBm	-95.55 dBm	-95.55 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 6**	-99 dBm max	-101.50 dBm	-102.75 dBm	-103.00 dBm	-103.25 dBm	PASS	

Table 46 Sensitivity (TelM and TelC) for 5-j60 Load Impedance

Impedance: 5 – j20

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Sensitivity Level (0.01 PER) **Mode 1**	-103.2 dBm max	-105.95 dBm	-106.20 dBm	-105.95 dBm	-106.45 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2**	-96.7 dBm max	-100.70 dBm	-101.45 dBm	-101.20 dBm	-101.95 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 12.6%EVM**	-95.8 dBm max	-100.55 dBm	-101.05 dBm	-100.80 dBm	-101.55 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 3**	-97.7 dBm max	-101.70 dBm	-102.20 dBm	-101.70 dBm	-102.45 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4**	-92.2 dBm max	-97.45 dBm	-97.95 dBm	-97.45 dBm	-97.95 dBm	PASS	

Sensitivity Level (0.01 PER) **Mode 4 12.6%EVM**	-91.3 dBm max	-97.05 dBm	-97.30 dBm	-97.05 dBm	-97.80 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze Off**	-91.7 dBm max	-95.20 dBm	-95.70 dBm	-95.20 dBm	-96.20 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze Off**	-89.1 dBm max	-94.85 dBm	-95.35 dBm	-94.85 dBm	-95.85 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze On**	-92.6 dBm max	-96.10 dBm	-96.35 dBm	-95.85 dBm	-96.60 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze On**	-90.5 dBm max	-96.00 dBm	-96.25 dBm	-95.75 dBm	-96.50 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 6**	-100.2 dBm max	-103.45 dBm	-103.70 dBm	-103.20 dBm	-103.95 dBm	PASS	

Table 47 Sensitivity (TelM and TelC) for 5-j20 Load Impedance

Impedance: 9 – j20

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Sensitivity Level (0.01 PER) **Mode 1 MICS**	-104.3 dBm max	-107.55 dBm	-107.55 dBm	-107.05 dBm	-108.05 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 1 MEDS**	-103.4 dBm max	-107.15 dBm	-107.15 dBm	-106.65 dBm	-107.40 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 MICS**	-97.8 dBm max	-102.80 dBm	-102.80 dBm	-102.55 dBm	-103.55 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 MICS 12.6%EVM**	-96.9 dBm max	-102.4 dBm	-102.65 dBm	-102.15 dBm	-102.90 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 MEDS**	-96.9 dBm max	-101.90 dBm	-102.40 dBm	-101.90 dBm	-102.40 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 2 MEDS 12.6% EVM**	-96.0 dBm max	-101.75 dBm	-101.75 dBm	-101.75 dBm	-102.25 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 3**	-98.8 dBm max	-102.30 dBm	-102.30 dBm	-102.55 dBm	-102.80 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4**	-93.3 dBm max	-98.05 dBm	-98.55 dBm	-97.80 dBm	-98.80 dBm	PASS	
Sensitivity Level (0.01 PER) **Mode 4 12.6%EVM**	-92.4 dBm max	-98.15 dBm	-98.40 dBm	-98.15 dBm	-98.90 dBm	PASS	

Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze Off**	-92.8 dBm max	-96.80 dBm	-96.80 dBm	-96.30 dBm	-97.05 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze Off**	-90.2 dBm max	-96.20 dBm	-96.45 dBm	-95.95 dBm	-96.70 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **Nominal Deviation** **Freeze On**	-93.7 dBm max	-97.20 dBm	-97.20 dBm	-96.95 dBm	-97.45 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 5** **47 kHz Deviation** **Freeze On**	-91.6 dBm max	-97.35 dBm	-97.35 dBm	-96.85 dBm	-97.60 dBm	PASS	
Sensitivity Level (0.0292 PER) **Mode 6**	-101.3 dBm max	-104.80 dBm	-104.88 dBm	-104.00 dBm	-104.80 dBm	PASS	

Table 48 Sensitivity (TelM and TelC) for 9-j20 Load Impedance

Conclusion: All results from Sensitivity (TelM and TelC) tests show compliance with the specification.

9.3 Intermodulation Rejection (TelM and TelC)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.4

DVT Plan: 6.4.5

Test Executive Script: Rx_Sc2 and Rx_Sc9

This test has a deviation (Deviation #20) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
IM Rejection **Mode 1, 3, 6**	47 dB min	51.75 dB	51.75 dB	51.75 dB	51.75 dB	PASS	
IM Rejection **Mode 2**	45.3 dB min	49.75 dB	50.75 dB	50.75 dB	50.75 dB	PASS	
IM Rejection **Mode 4** **0dB attn.**	42.3 dB min	47.75 dB	47.75 dB	47.75 dB	47.75 dB	PASS	ECO 096 reduced the specification requirement with 2dB
IM Rejection **Mode 4** **0-22dB attn.**	42.3 dB min	47.05 dB	47.05 dB	47.05 dB	47.05 dB	PASS	ECO 096 reduced the specification requirement with 2dB
IM Rejection **Mode 5** **Nominal Deviation** **Freeze Off**	41.6 dB min	45.75 dB	45.75 dB	46.75 dB	46.75 dB	PASS	
IM Rejection **Mode 5** **47 kHz Deviation** **Freeze Off**	40.6 dB min	46.25 dB	45.75 dB	46.75 dB	46.5 dB	PASS	

Table 49 Intermodulation Rejection (TelM and TelC)

Conclusion: All results from Intermodulation Rejection (TelM and TelC) tests show compliance with the specification.

9.4 Adjacent Channel Rejection (TelM and TelC)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.5

DVT Plan: 6.4.6

Test Executive Script: Rx_Sc2

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Adjacent Channel Rejection **Mode 1, 2**	35 dB min	37.50 dB	38.00 dB	36.75 dB	37.75 dB	PASS	
Adjacent Channel Rejection **Mode 3-10 except 5**	40 dB min	43.75 dB	44.75 dB	45.75 dB	45.00 dB	PASS	
Adjacent Channel Rejection **Mode 5** **Nominal Deviation** **Freeze Off**	34.5 dB min	40.75 dB	40.75 dB	43.00 dB	41.00 dB	PASS	
Adjacent Channel Rejection **Mode 5** **47 kHz Deviation** **Freeze Off**	34 dB min	40.00 dB	40.50 dB	42.25 dB	40.75 dB	PASS	

Table 50 Adjacent Channel Rejection (TelM and TelC)

Conclusion: All results from Adjacent Channel Rejection (TelM and TelC) tests show compliance with the specification.

9.5 Alternate Channel Rejection (TelM and TelC)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.6

DVT Plan: 6.4.7

Test Executive Script: Rx_Sc2

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Alternate Channel Rejection **Mode 1, 2**	44 dB min	50.25 dB	50.50 dB	50.00 dB	50.25 dB	PASS	
Alternate Channel Rejection **Mode 3, 6-10**	50 dB min	58.25 dB	58.00 dB	57.50 dB	56.75 dB	PASS	
Alternate Channel Rejection **Mode 4**	48.5 dB min	53.50 dB	53.50 dB	52.75 dB	53.25 dB	PASS	
Alternate Channel Rejection **Mode 5** **Nominal Deviation** **Freeze Off**	45.5 dB min	51.75 dB	51.25 dB	51.25 dB	51.50 dB	PASS	
Alternate Channel Rejection **Mode 5** **47 kHz Deviation** **Freeze Off**	42 dB min	50.75 dB	50.25 dB	50.75 dB	51.00 dB	PASS	

Table 51 Alternate Channel Rejection (TelM and TelC)

Conclusion: All results from Alternate Channel Rejection (TelM and TelC) tests show compliance with the specification.

9.6 **External Spurious Response Rejection (TelM and TelC)**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.7

DVT Plan: 6.4.8

Beyond Alternate + X:

Test Executive Script: Spur_Sc2, Spur_Sc3, Spur_Sc4 and Spur_Sc5

Note: The DVT Plan calls for performing the coarse spurious testing (covered by Spur_Sc2, Spur_Sc3 and Spur_Sc4) at an interferer level 6dB above the specification and the fine spurious test at a level 3dB above the specification. The coarse spurious testing was actually run with an interferer level 18dB higher than specification for frequencies above 409MHz and below 398MHz, making it a much more conservative test. This "overtesting", as it is referred to, is more than sufficient to cover the extra loss (on the Spur racks) of 2.54dB at 3GHz that is above the loss at 403.5MHz, which is calibrated out. The coarse spurious testing within the frequency range of 398 to 409MHz was overtested by 5.5dB for narrowband and 2dB for wideband as agreed with Medtronic. This "inband" range is close to 403.5MHz where the calibration was performed and therefore needs no additional overtesting.
The fine spurious test was actually performed with 7dB overtesting on the spur locations discovered in the coarse spurious test below 398MHz or above 409MHz. This is more conservative than the 3dB overtest in the DVT Plan. No spurious responses were found between 1GHz and 3GHz so the fine test did not require overtesting to cover the loss at 3GHz. Within the 398 to 409MHz range the fine test was run at the spec level since the loss calibration is valid in this range and the fine test uses a CW rather than a modulated interferer.

Test Parameter	Spec	sn337	sn338	sn343	Compliance	Notes
Interferer Level	-42 dBm min	-35.25 dBm	-35.25 dBm	-27.00 dBm	PASS	.

Table 52 External Spurious Response Rejection (TelM and TelC) beyond alternate + X

Within Alternate + X:

Test Executive Script: Rx_Sc10 and Spur_Sc1

The requirements for alternate + X are to be tested on both the spurious bench and the Rx bench. The spurious bench is used at one temperature and across many channels. The Rx bench tests a few channels at the temperature extremes. (DVT Plan Section 6.4.8 Step 8).

Test Parameter	Spec	sn316 (1)	sn317 (1)	sn318 (1)	sn320 (1)	sn337 (2)	sn338 (2)	sn343 (2)	Compliance
Alternate plus X channel Rejection **Mode 1** **A: Alt+1** **B: Alt+2** **C; Alt+3** **D: Alt+4** **E: Alt+5** **F: Alt+6** **G: Alt+7** **H: Alt+8** **I: Alt+9** **J: Alt+10** **K: Alt+11**	**A: -51.5** **B: -50** **C: -49.5** **D: -48** **E: -46.5** **F: -45.5** **G: -45** **H: -44** **I: -43.5** **J: -43** **K: -42.5** **dBm min**	A.-47.00 B.-44.50 C.-42.75 D.-42.50 E.-40.25 F.-38.75 G.-38.00 H.-37.00 I.-36.25 J.-35.50 K.-34.75 dBm	A.-46.75 B.-44.00 C.-42.50 D.-41.75 E.-39.75 F.-38.50 G.-37.50 H.-36.75 I.-36.25 J.-35.00 K.-34.75 dBm	A.-46.50 B.-44.00 C.-42.25 D.-41.25 E.-40.00 F.-38.25 G-37.50 H.-36.75 I.-36.00 J.-35.00 K.-35.25 dBm	A.-46.25 B-44.00 C.-42.25 D.-42.25 E.-39.50 F.-38.25 G.-37.50 H.-36.25 I.-35.50 J.-34.75 K.-34.00 dBm	A.-47.75 B.-45.00 C.-42.75 D.-40.75 E.-39.75 F.-38.75 G.-37.50 H.-36.00 I.-35.00 J.-34.25 K.-33.50 dBm	A.-47.50 B.-44.75 C.-43.25 D.-41.75 E.-40.75 F.-39.25 G.-38.00 H.-37.75 I.-35.75 J.-35.00 K.-34.00 dBm	A.-48.00 B.-45.00 C.-42.75 D.-40.75 E.-39.75 F.-38.50 G.-36.75 H.-36.25 I.-35.50 J.-34.00 K.-33.25 dBm	PASS
Alternate plus X channel Rejection **Mode 2** **A: Alt+1** **B: Alt+2** **C; Alt+3** **D: Alt+4** **E: Alt+5** **F: Alt+6** **G: Alt+7**	**A: -49** **B: -48** **C: -48** **D: -46.5** **E: -44** **F: -43.5** **G: -43** **dBm min**	A.-44.75 B.-42.25 C.-40.75 D.-40.00 E.-38.00 F.-37.00 G.-35.75 dBm	A.-44.75 B.-42.25 C.-40.5 D.-39.25 E.-37.25 F.-36.25 G.-35.00 dBm	A.-44.75 B.-42.00 C.-40.00 D.-39.50 E.-37.75 F.-36.25 G.-35.00 dBm	A.-44.25 B-41.75 C-40.25 D.-40.25 E.-37.00 F.-36.00 G.-34.75 dBm	A.-45.25 B.-42.50 C.-40.75 D.-38.75 E.-37.25 F.-37.00 G.-35.00 dBm	A.-45.00 B.-42.75 C.-41.00 D.-39.00 E.-37.75 F.-37.50 G.-35.50 dBm	A.-46.00 B.-43.25 C.-40.75 D.-39.25 E.-37.75 F.-36.75 G.-34.75 dBm	PASS

Alternate plus 1 channel Rejection **Mode 3, 7**	**-44 dBm min**	-38.25 dBm	-38.00 dBm	-38.00 dBm	-38.00 dBm	-37.75 dBm	-38.25 dBm	-38.50 dBm	PASS
Alternate plus X channel Rejection **Mode 5 nominal deviation** **A: Alt+1** **B: Alt+2**	**A: -45.5** **B: -42.5** **dBm min**	A.-38.50 B. -35.75 dBm	A.-38.50 B.-35.25 dBm	A.-38.25 B.-35.50 dBm	A.-38.50 B-35.25 dBm	A.-38.75 B.-35.00 dBm	A.-38.75 B.-36.50 dBm	A.-39.00 B.-34.50 dBm	PASS
Alternate plus X channel Rejection **Mode 5 nominal deviation** **FREEZE** **A: Alt+1** **B: Alt+2**	**A: -45.5** **B: -42.5** **dBm min**	A.-39.25 B.-36.50 dBm	A.-39.00 B.-36.25 dBm	A.-39.25 B.-36.00 dBm	A.-38.75 B-36.00 dBm	A.-39.50 B.-35.75 dBm	A.-39.50 B.-35.75 dBm	A.-39.50 B.-35.50 dBm	PASS
Alternate plus X channel Rejection **Mode 5 47kHz deviation** **A: Alt+1** **B: Alt+2** **C; Alt+3** **D: Alt+4**	**A: -48.5** **B: -45** **C: -44** **D: -43** **dBm min**	A.-35.50 B.-32.50 C.-31.00 D.-30.00 dBm	A.-35.25 B.-32.00 C.-30.75 D.-29.75 dBm	A.-35.00 B.-32.25 C.-30.50 D.-29.75 dBm	A.-35.00 B.-32.00 C.-30.50 D.-29.25 dBm	A.-35.50 B-32.25 C.-30.00 D.-29.25 dBm	A.-43.25 B.-38.50 C.-37.50 D.-36.25 dBm	A.-35.50 B.-31.25 C.-29.75 D.-28.75 dBm	PASS

Alternate plus X channel Rejection **Mode 5 47kHz deviation** **FREEZE** **A: Alt+1** **B: Alt+2** **C; Alt+3** **D: Alt+4**	**A: -48.5** **B: -45** **C: -44** **D: -43** **dBm min**	A.-36.75 B.-33.25 C.-31.25 D.-30.50 dBm	A.-36.75 B.-33.25 C.-31.50 D.-30.25 dBm	A.-36.00 B.-33.00 C.-31.25 D.-30.00 dBm	A.-36.50 B.-33.25 C.-31.00 D.-30.00 dBm	A.-36.50 B-32.75 C.-30.50 D.-29.25 dBm	A.-42.00 B.-37.50 C.-37.25 D.-36.00 dBm	A.-36.75 B.-32.50 C.-30.50 D.-29.25 dBm	PASS
Alternate plus 1 channel Rejection **Mode 6**	**-42.5 dBm min**	-37.75 dBm	-38.00 dBm	-37.75 dBm	-37.50 dBm	-38.25 dBm	-38.50 dBm	-37.75 dBm	PASS

Table 53 External Spurious Response Rejection (TelM and TelC) within alternate + X

Note (1) : Rx bench tests a few channels at the temperature extremes.
Note (2) : The spurious bench is used at one temperature and across all channels.

Conclusion: All results from External Spurious Response Rejection for TelM and TelC, covering within and beyond alternate + X, shows compliance with the specification.

9.7 Blocking (TelM and TelC)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.7

DVT Plan: 6.4.9

Test Executive Script: Rx_Sc9

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Blocker Level **Mode 1** **Attn = 0**	**-42 dBm min**	-29.00 dBm	-29.00 dBm	-28.50 dBm	-28.50 dBm	PASS	
Blocker Level **Mode 1** **Attn = 22**	**-20 dBm min**	-15.00 dBm (1)	-15.00 dBm (1)	-15.00 dBm (1)	-15.00 dBm (1)	PASS	
Blocker Level **Mode 5** **Attn = 0**	**-42 dBm min**	-29.25 dBm	-28.75 dBm	-29.25 dBm	-28.50 dBm	PASS	
Blocker Level **Mode 5** **Attn = 22**	**-20 dBm min**	-15.00 dBm (1)	-15.00 dBm (1)	-15.00 dBm (1)	-15.00 dBm (1)	PASS	

Note: (1) The 22dB Attn results show that the spec is met but the margin from the spec is limited by the maximum signal level available from the lab equipment.

Table 54 Blocking (TelM and TelC)

Conclusion: All results from Blocking (TelM and TelC) tests show compliance with the specification.

9.8 AM Rejection (TelM and TelC)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.8

DVT Plan: 6.4.10

Test Executive Script: Rx_Sc2

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
AM Rejection	**-58 dBm min**	-50.25 dBm	-48.25 dBm	-51.50 dBm	-52.50 dBm	PASS	

Table 55 AM Rejection (TelM and TelC)

Conclusion: All results from AM Rejection (TelM and TelC) tests show compliance with the specification.

9.9 Dynamic Range (TelM and TelC)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.12

DVT Plan: 6.4.12

Test Executive Script: Rx_Sc9

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
PER Mode 2	**0.01 PER max**	.00833 PER	.00833 PER	.0083 PER	.0083 PER	PASS	PER = packet errors/total packets
PER Mode 4	**0.01 PER max**	.00317 PER	.0033 PER	.0017 PER	.0017 PER	PASS	
PER Mode 5 **Nominal Deviation** **Freeze Off**	**0.0292 PER max**	.00625 PER	.00825 PER	.004 PER	.0058 PER	PASS	
PER Mode 5 **Nominal Deviation** **Freeze On**	**0.0292 PER max**	.00325 PER	.00625 PER	.0035 PER	.0058 PER	PASS	

Table 56 Dynamic Range (TelM and TelC)

Conclusion: All results from Dynamic Range (TelM and TelC) tests show compliance with the specification.

9.10 RSSI Pre-Calibration Accuracy

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.15.2

DVT Plan: 6.4.13.1

Test Executive Script: Rx_Sc11

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Mid-scale Offset	**+/-16 dB**	2.56 dB	2.24 dB	2.56 dB	2.56 dB	PASS	**+/-16 dB from trimmed value**

Table 57 RSSI Pre-Calibration Accuracy

Conclusion: All results from RSSI Pre-Calibration Accuracy tests show compliance with the specification.

9.11 **RSSI Linearity**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.15.2

DVT Plan: 6.4.13.2

Test Executive Script: Rx_Sc11

This test has a deviation (Deviation #13) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Absolute Level (narrowband, MICS, 0dB Attenuation)	**-108 dBm min** **-104 dBm max**	-106.64 min -105.36 max dBm	-106.64 min -105.36 max dBm	-106.64 min -105.36 max dBm	-106.96 min -105.36 max dBm	PASS	
Absolute Level (narrowband, MEDS, 0dB Attenuation)	**-108 dBm min** **-104 dBm max**	-106.96 min -104.72 max dBm	-106.96 min -104.72 max dBm	-107.28 min -105.04 max dBm	-106.96 min -105.04 max dBm	PASS	
Absolute Level (narrowband, MICS, 22dB Attenuation)	**-112 dBm min** **-104 dBm max**	-107.60 min -106.32 max dBm	-106.96 min -106.00 max dBm	-106.96 min -106.00 max dBm	-107.28 min -105.68 max dBm	PASS	
Absolute Level (narrowband, MEDS, 22dB Attenuation)	**-112 dBm min** **-104 dBm max**	-108.56 min -105.68 max dBm	-108.88 min -105.68 max dBm	-108.24 min -105.68 max dBm	-108.56 min -105.36 max dBm	PASS	
Absolute Level (wideband, MICS, 0dB Attenuation)	**-103 dBm min** **-99 dBm max**	-101.64 min -100.36 max dBm	-101.64 min -100.36 max dBm	-101.64 min -100.36 max dBm	-101.96 min -100.68 max dBm	PASS	
Absolute Level (wideband, MICS, 22dB Attenuation)	**-107 dBm min** **-99 dBm max**	-101.96 min -101.00 max dBm	-101.96 min -101.00 max dBm	-101.96 min -101.00 max dBm	-102.28 min -101.00 max dBm	PASS	
Differentiable Level (MICS, -106 to -68 dBm, 0dB Attenuation, Narrowband)	**2.5 dB** (Delta of 1 code or more for every 2.5dB difference in level.)	Min Code Delta=6	Min Code Delta=4	Min Code Delta=6	Min Code Delta=6	PASS	

Differentiable Level (MICS, -68 to -55 dBm, 0dB Attenuation, Narrowband)	**4 dB** (Delta of 1 code or more for every 4 dB difference in level.)	Min Code Delta=5	Min Code Delta=5	Min Code Delta=5	Min Code Delta=5	PASS	
Differentiable Level (MEDS, -106 to -68 dBm, 0dB Attenuation, Narrowband)	**2.5 dB** (Delta of 1 code or more for every 2.5dB difference in level.)	Min Code Delta=5	Min Code Delta=3	Min Code Delta=4	Min Code Delta=5	PASS	
Differentiable Level (MEDS, -68 to -55 dBm, 0dB Attenuation, Narrowband)	**4 dB**	Min Code Delta=4	Min Code Delta=2	Min Code Delta=4	Min Code Delta=3	PASS	
Differentiable Level (MICS, -106 to -68 dBm, 22dB Attenuation, Narrowband)	**4 dB**	Min Code Delta=8	Min Code Delta=3	Min Code Delta=8	Min Code Delta=8	PASS	
Differentiable Level (MICS, -68 to -55 dBm, 22dB Attenuation, Narrowband)	**6 dB**	Min Code Delta=8	Min Code Delta=7	Min Code Delta=7	Min Code Delta=7	PASS	
Differentiable Level (MEDS, -106 to -68 dBm, 22dB Attenuation, Narrowband)	**6 dB**	Min Code Delta=10	Min Code Delta=7	Min Code Delta=11	Min Code Delta=9	PASS	
Differentiable Level (MEDS, -68 to -55 dBm, 22dB Attenuation, Narrowband)	**8 dB**	Min Code Delta=8	Min Code Delta=5	Min Code Delta=7	Min Code Delta=6	PASS	
Differentiable Level (MICS, -101 to -68 dBm, 0dB Attenuation, wideband)	**2.5 dB**	Min Code Delta=6	Min Code Delta=3	Min Code Delta=5	Min Code Delta=6	PASS	

Differentiable Level (MICS, -68 to -55 dBm, 0dB Attenuation, wideband)	**4 dB**	Min Code Delta=7	Min Code Delta=6	Min Code Delta=6	Min Code Delta=6	PASS	
Differentiable Level (MICS, -101 to -68 dBm, 22dB Attenuation, wideband)	**4 dB**	Min Code Delta=8	Min Code Delta=6	Min Code Delta=9	Min Code Delta=7	PASS	
Differentiable Level (MICS, -68 to -55 dBm, 22dB Attenuation, wideband)	**6 dB**	Min Code Delta=10	Min Code Delta=9	Min Code Delta=8	Min Code Delta=10	PASS	

Table 58 RSSI Linearity

Conclusion: All results from RSSI Linearity tests show compliance with the specification.

9.12 **RSSI Pulse Response**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.15.4

DVT Plan: 6.4.13.3

This test has a deviation (Deviation #4) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

Test Executive Script: Rx_Sc12

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
RSSI Pulse Response	**CCA Codes of margin** **Greater than 0**	11	10	10	10	PASS	

Table 59 RSSI Pulse Response

Conclusion: All results from RSSI Pulse Response tests show compliance with the specification.

9.13 **RSSI Monotonicity and Saturation without Foldback**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.15.1 and 2.3.4.5.15.3

DVT Plan: 6.4.13.4

Test Executive Script: Rx_Sc11

Note: Monotonicity and Foldback are evaluated by comparing values 2dB apart and this is done at all levels that are 0.5dB apart. For Foldback, negative steps can occur but they are all isolated from each other and all occur above -55dBm.

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Monotonicity **-109 to -50 dBm**	**Monotonic**	PASS	PASS	PASS	PASS	PASS	Pass/Fail
Foldback **-112 to -44 dBm**	**No Foldback**	PASS	PASS	PASS	PASS	PASS	Pass/Fail

Table 60 RSSI Monotonicity and Saturation without Foldback

Conclusion: All results from RSSI Monotonicity and Saturation without Foldback tests show compliance with the specification.

9.14 **Rx Conducted Spurious Emissions**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.5.16

DVT Plan: 6.4.14.1

Test Executive Script: Tx_Sc23 (test is run on the Tx DVT system)

This test has a deviation (Deviation #6) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

Note on use of Peak Detector vs CISPR Detector: The Regulatory Tx Harmonics Spurious and Rx Conducted Spurious Emissions tests have requirements to test for unwanted emissions using a CISPR detector in the spectrum analyzer. The DVT tests use a peak detector instead in order to save time. Close to the MEDS/MICS bands the CISPR detector is used but it is slow. This substitution of a peak detector for the CISPR detector is justified by lab data reported in eBinder document #139681 version 1 dated January 19, 2011. The conclusion of that document states: “The CISPR method measures a peak power that is 0.5dB lower than the approximation method. This means that the approximation method is conservative when measuring emissions and safe to use.” See Deviation #6.

Test Parameter	Spec	sn302	sn311	sn313	sn314	Compliance	Notes
Power Level from 9 kHz to 1 GHz	**-57 dBm max**	-81.6	-82.0	-81.2	-81.3	PASS	
Power Level from 1 GHz to 4.06 GHz	**-57 dBm max**	-64.7	-65.0	-64.6	-64.6	PASS	

Table 61 Rx Conducted Spurious Emissions

Conclusion: All results from Rx Conducted Spurious Emissions tests show compliance with the specification.

10 Maximum Signal Power at Antenna Test Results

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.2.1

DVT Plan: 6.4.15

Test Executive Script: Spur_Sc6 and Spur_Sc7

Test Parameter	Spec	sn346	sn347	sn348	sn352	Compliance	Notes
RSSI **Mode 4 0 dB Attenuation**	Abs level: -103 dBm min -99 dBm max Diff level: -106 to -68 dBm: 2.5dB Diff level: -68 to -55 dBm: 4 dB	-100.84 dBm min -100.84 dBm max Min code diff = 5 Min code diff = 3	-100.52 dBm min -100.52 dBm max Min code diff = 5 Min code diff = 2	-101.16 dBm min -100.84 dBm max Min code diff = 5 Min code diff = 3	-100.68 dBm min -100.68 dBm max Min code diff = 5 Min code diff = 3	PASS	Comparison was performed. Absolute numbers are worst case of both pre and post results.
Sensitivity Mode 4 0dB Attenuation	-95dBm	Delta = 0 dB, -101.25 dBm worst case	Delta = 0.25 dB, -101.5 dBm worst case	Delta = 0.25 dB, -101.25 dBm worst case	Delta = 0 dB, -101.5 dBm worst case	PASS	Deltas and absolute numbers are worst case.
Sensitivity Mode 4 2dB Attenuation	-93dBm	Delta = 0.25 dB, -100.5 dBm worst case	Delta = 0.5 dB, -100.5 dBm worst case	Delta = 0 dB, -100.75 dBm worst case	Delta = 0 dB, -100.75 dBm worst case	PASS	
Sensitivity Mode 4 22dB Attenuation	-73dBm	Delta = 0.25 dB, --81 dBm worst case	Delta = 0.5 dB, -81.5 dBm worst case	Delta = 0 dB, -81.75 dBm worst case	Delta = 0.25 dB, -81.5 dBm worst case	PASS	
AM Rejection Mode 4	-58dBm	Delta = 0.25 dB, -50 dBm	Delta = 0.75 dB, -48.25 dBm	Delta = 0.5 dB, -49.75 dBm	Delta = 0.25 dB, -49.75 dBm	PASS	
Standby Current (Disabled Static Current @25C)	50nA	Delta = 0.30 nA, 2.74 nA	Delta = 0.24 nA, 2.74 nA	Delta = 0.30nA, 2.73 nA	Delta = 0.06nA, 2.64 nA	PASS	
Tx Power	-4dBm min for MICS band	Delta = 0.08 dB, -3.58 dBm	Delta = 0.10 dB, -3.57 dBm	Delta = 0.08dB, -3.58 dBm	Delta = 0.106dB, -3.57 dBm	PASS	

Table 62 Max Power at Antenna – First four modules (tested at 64 MHz)

Test Parameter	Spec	sn353	sn354	sn355	sn356	Compliance	Notes
RSSI **Mode 4 0 dB Attenuation**	Abs level: -103 dBm min -99 dBm max Diff level: -106 to -68 dBm: 2.5dB Diff level: -68 to -55 dBm: 4 dB	-101 dBm max -101 dBm min Min code diff = 5 Min code diff = 2	-100.68 dBm max -100.68 dBm min Min code diff = 5 Min code diff = 4	-100.68 dBm max -100.68 dBm min Min code diff = 5 Min code diff = 4	-100.84 dBm max -100.84 dBm min Min code diff = 5 Min code diff = 3	PASS	Absolute numbers are worst case of both pre and post results.
Sensitivity Mode 4 0dB Attenuation	-95dBm	Delta = 0 dB, -101.25 dBm worst case	Delta = 0.25 dB, -101.25 dBm worst case	Delta = 0 dB, -101 dBm worst case	Delta = 0.25 dB, -101.25 dBm worst case	PASS	Deltas and absolute numbers are worst case.
Sensitivity Mode 4 2dB Attenuation	-93dBm	Delta = 0 dB, -100.5 dBm worst case	Delta = 0 dB, -100.5 dBm worst case	Delta = 0.25 dB, -100 dBm worst case	Delta = 0.25 dB, -100.5 dBm worst case	PASS	
Sensitivity Mode 4 22dB Attenuation	-73dBm	Delta = 0 dB, -81.75 dBm worst case	Delta = 0 dB, -81.25 dBm worst case	Delta = 0.25 dB, -81.25 dBm worst case	Delta = 0 dB, -81 dBm worst case	PASS	
AM Rejection Mode 4	-58dBm	Delta = 0.25 dB, -47.75 dBm	Delta = 0.25 dB, -46.25 dBm	Delta = 1.75 dB, -49 dBm	Delta = 0.5 dB, -49.75 dBm	PASS	
Standby Current (Disabled Static Current @25C)	50nA	Delta = 0.13 nA, 2.86 nA	Delta = 0.13 nA, 2.91 nA	Delta = 0.25 nA, 2.96 nA	Delta = 0.21 nA, 2.93 nA	PASS	
Tx Power	-4dBm min for MICS band	Delta = 0.12 dB, -3.54 dBm	Delta = 0.09 dB, -3.55 dBm	Delta = 0.10dB, -3.55 dBm	Delta = 0.10 dB, -3.58 dBm	PASS	

Table 63 Max Power at Antenna – second four modules (tested at 128 MHz)

Test Parameter	Spec	sn357	sn358	sn359	sn360	Compliance	Notes
RSSI **Mode 4 0 dB Attenuation**	Abs level: -103 dBm min -99 dBm max Diff level: -106 to -68 dBm: 2.5dB Diff level: -68 to -55 dBm: 4 dB	-100.84 dBm max -100.84 dBm min Min code diff = 5 Min code diff = 4	-100.68 dBm max -100.68 dBm min Min code diff = 5 Min code diff = 4	-100.68 dBm max -100.84 dBm min Min code diff = 5 Min code diff = 3	-100.84 dBm max -100.dBm min Min code diff = 5 Min code diff = 3	PASS	Absolute numbers are worst case of both pre and post results.
Sensitivity Mode 4 0dB Attenuation	-95dBm	Delta = 0 dB, -101.5 dBm worst case	Delta = 0.25 dB, -101.25 dBm worst case	Delta = 0.5 dB, -101.25 dBm worst case	Delta = 0.25 dB, -101.25 dBm worst case	PASS	Deltas and absolute numbers are worst case.
Sensitivity Mode 4 2dB Attenuation	-93dBm	Delta = 0.25 dB, -100.5 dBm worst case	Delta = 0.5 dB, -100.25 dBm worst case	Delta = 0.25 dB, -100 dBm worst case	Delta = 0 dB, -100.75 dBm worst case	PASS	
Sensitivity Mode 4 22dB Attenuation	-73dBm	Delta = 0.25 dB, -81.5 dBm worst case	Delta = 0.25 dB, -82.25 dBm worst case	Delta = 0 dB, -81.5 dBm worst case	Delta = 0.25 dB, -81.5 dBm worst case	PASS	
AM Rejection Mode 4	-58dBm	Delta = 0.25 dB, -47.25 dBm	Delta = 0.5 dB, -50.5 dBm	Delta = 0.75 dB, -48 dBm	Delta = 0.75 dB, -45.25 dBm	PASS	
Standby Current (Disabled Static Current @25C)	50nA	Delta = 0.12 nA, 2.78 nA	Delta = 0.12 nA, 2.92 nA	Delta = 0.14 nA, 2.66 nA	Delta = 0.22 nA, 2.74 nA	PASS	
Tx Power	-4dBm min for MICS band	Delta = 0.08 dB, -3.54 dBm	Delta = 0.09 dB, -3.56 dBm	Delta = 0.09 dB, -3.58 dBm	Delta = 0.07dB, -3.52 dBm	PASS	

Table 64 Max Power at Antenna – Third four modules (tested at 403.5 MHz)

Test Parameter	Spec	sn362	sn363	sn366	sn368	Compliance	Notes
RSSI **Mode 4 0 dB Attenuation**	Abs level: -103 dBm min -99 dBm max Diff level: -106 to -68 dBm: 2.5dB Diff level: -68 to -55 dBm: 4 dB	-100.84 dBm min -100.84 dBm max Min code diff = 5 Min code diff = 3	-101.16 dBm min -101.16 dBm max Min code diff = 5 Min code diff = 2	-100.84 dBm min -100.84 dBm max Min code diff = 5 Min code diff = 3	-101.16 dBm min -100.52 dBm max Min code diff = 5 Min code diff = 4	PASS	Absolute numbers are worst case of both pre and post results.
Sensitivity Mode 4 0dB Attenuation	-95dBm	Delta = 0.5 dB, -101 dBm worst case	Delta = 0.25 dB, -101.25 dBm worst case	Delta = 0.25 dB, -101 dBm worst case	Delta = 0.25 dB, -101.25 dBm worst case	PASS	Deltas and absolute numbers are worst case.
Sensitivity Mode 4 2dB Attenuation	-93dBm	Delta = 0.25 dB, -100.5 dBm worst case	Delta = 0.25 dB, -100.5 dBm worst case	Delta = 0.25 dB, -100.25 dBm worst case	Delta = 0.25 dB, -100.5 dBm worst case	PASS	
Sensitivity Mode 4 22dB Attenuation	-73dBm	Delta = 0 dB, -81.75 dBm worst case	Delta = 0 dB, -82 dBm worst case	Delta = 0 dB, -81.75 dBm worst case	Delta = 0.25 dB, -81.75 dBm worst case	PASS	
AM Rejection Mode 4	-58dBm	Delta = 0.5 dB, -48 dBm	Delta = 0.5 dB, -47 dBm	Delta = 0.25 dB, -48.75 dBm	Delta = 0.25 dB, -49.75 dBm	PASS	
Standby Current (Disabled Static Current @25C)	50nA	Delta = 0.25 nA, 2.77 nA	Delta = 0.19 nA, 2.75 nA	Delta = 0.27 nA, 2.75 nA	Delta = 0.14 nA, 2.65 nA	PASS	
Tx Power	-4dBm min for MICS band	Delta = 0.11 dB, -3.60 dBm	Delta = 0.09 dB, -3.56 dBm	Delta = 0.12 dB, -3.57 dBm	Delta = 0.12 dB, -3.57 dBm	PASS	

Table 65 Max Power at Antenna – Fourth four modules (tested at 450 MHz)

Conclusion: All results from Max Power at Antenna tests show compliance with the specification.

11 Wakeup Test Results

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.5 and 6.6

This test has a deviation (Deviation #12) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

11.1 Clear Channel Assessment Dwell Times

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.10

DVT Plan: 6.5.1.1

Test Executive Script: Rx_Sc12

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Dwell Times	**Nominal** **A: 0.606 ms** **B: 1.25 ms** **C: 2.5 ms** **D: 5 ms** **E: 10 ms** **F: 20 ms** **G: 40 ms** **H: 80 ms**	A: 0.630 B: 1.27 C: 2.52 D: 5.03 E: 10.03 F: 20.03 G: 40.04 H. 80.06 ms	A: 0.630 B: 1.27 C: 2.52 D: 5.03 E: 10.03 F: 20.03 G: 40.04 H: 80.06 ms	A: 0.630 B: 1.27 C: 2.52 D: 5.03 E: 10.03 F: 20.03 G: 40.04 H: 80.06 ms	A: 0.630 B: 1.27 C: 2.52 D: 5.03 E: 10.03 F: 20.03 G: 40.04 H: 80.06 ms	PASS	**No Tolerances were given in the specification.**

Table 66 Clear Channel Assessment Dwell Times

Conclusion: All results from Clear Channel Assessment Dwell Times tests show compliance with the specification.

11.2 **Wakeup Transmit Time**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.1

DVT Plan: 6.5.2.1

Test Executive Script: Rx_Sc12

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Wakeup Transmit Time **Wide**	**5 seconds max.**	4.98 s	4.98 s	4.98 s	4.98 s	PASS	Time should be close to 5 seconds without going over.
Wakeup Transmit Time **Narrow**	**5 seconds max.**	4.95 s	4.95 s	4.95 s	4.95 s	PASS	

Table 67 Wakeup Transmit Times

Conclusion: All results from Wakeup Transmit Times tests show compliance with the specification.

11.3 **Wakeup Sensitivity**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.6.1

Test Executive Script: Rx_Sc13 through Rx_Sc19 and Rx_Sc21

Impedance: 50+j0

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Narrowband** **Center Hardware Channel**	-103 dBm max	-105.75 dBm	-105.50 dBm	-105.50 dBm	-105.75 dBm	PASS	
WU Sensitivity **Narrowband** **Lower Hardware Channel**	-101.5 dBm max	-105.00 dBm	-104.75 dBm	-104.75 dBm	-105.50 dBm	PASS	

WU Sensitivity **Narrowband** **Upper Hardware Channel**	-101.5 dBm max	-104.50 dBm	-104.25 dBm	-104.00 dBm	-104.75 dBm	PASS	
WU Sensitivity **Wideband** **Center Hardware Channel**	-97.5 dBm max	-104.00 dBm	-104.25 dBm	-104.00 dBm	-104.00 dBm	PASS	
WU Sensitivity **Wideband** **Lower Hardware Channel**	-96 dBm max	-104.00 dBm	-103.50 dBm	-103.75 dBm	-103.00 dBm	PASS	
WU Sensitivity **Wideband** **Upper Hardware Channel**	-96 dBm max	-103.00 dBm	-102.50 dBm	-102.25 dBm	-101.75 dBm	PASS	Variation between Low, Upp, & Centr is within expected range. Asymmetry between Upp and Low due to Weaver filter roll off

Table 68 Wakeup Sensitivity

Impedance: VSWR 2:1

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Narrowband** **Center Hardware Channel**	-101.5 dBm max	-104.75 dBm	-104.50 dBm	-104.75 dBm	-105.00 dBm	PASS	
WU Sensitivity **Narrowband** **Lower Hardware Channel**	-100 dBm max	-105.75 dBm	-105.50 dBm	-105.25 dBm	-106.00 dBm	PASS	
WU Sensitivity **Narrowband** **Upper Hardware Channel**	-100 dBm max	-103.50 dBm	-103.50 dBm	-103.25 dBm	-103.75 dBm	PASS	

WU Sensitivity **Wideband** **Center Hardware Channel**	-96 dBm max	-103.00 dBm	-103.25 dBm	-102.75 dBm	-102.75 dBm	PASS	
WU Sensitivity **Wideband** **Lower Hardware Channel**	-94.5 dBm max	-102.25 dBm	-102.25 dBm	-102.75 dBm	-102.00 dBm	PASS	
WU Sensitivity **Wideband** **Upper Hardware Channel**	-94.5 dBm max	-101.50 dBm	-101.00 dBm	-101.00 dBm	-100.50 dBm	PASS	Variation between Low, Upp, & Centr is within expected range. Asymmetry between Upp and Low due to Weaver filter roll off

Table 69 Wakeup Sensitivity

Impedance: 50 –j60

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Narrowband** **Center Hardware Channel**	-102.4 dBm max	-104.90 dBm	-104.90 dBm	-104.40 dBm	-105.15 dBm	PASS	
WU Sensitivity **Narrowband** **Lower Hardware Channel**	-100.9 dBm max	-104.15 dBm	-104.15 dBm	-103.90 dBm	-104.65 dBm	PASS	
WU Sensitivity **Narrowband** **Upper Hardware Channel**	-100.9 dBm max	-103.90 dBm	-103.90 dBm	-103.40 dBm	-104.40 dBm	PASS	
WU Sensitivity **Wideband** **Center Hardware Channel**	-96.9 dBm max	-102.65 dBm	-103.65 dBm	-103.40 dBm	-103.90 dBm	PASS	

WU Sensitivity **Wideband** **Lower Hardware Channel**	-95.4 dBm max	-103.40 dBm	-102.90 dBm	-102.90 dBm	-102.90 dBm	PASS	
WU Sensitivity **Wideband** **Upper Hardware Channel**	-95.4 dBm max	-101.90 dBm	-100.65 dBm	-100.65 dBm	-101.65 dBm	PASS	Variation between Low, Upp, & Centr is within expected range. Asymmetry between Upp and Low due to Weaver filter roll off

Table 70 Wakeup Sensitivity

Impedance: 5 + j0

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Narrowband** **Center Hardware Channel**	-102.4 dBm max	-104.60 dBm	-104.35 dBm	-104.35 dBm	-104.85 dBm	PASS	
WU Sensitivity **Narrowband** **Lower Hardware Channel**	-100.9 dBm max	-104.10 dBm	-103.85 dBm	-103.60 dBm	-104.10 dBm	PASS	
WU Sensitivity **Narrowband** **Upper Hardware Channel**	-100.9 dBm max	-103.60 dBm	-103.35 dBm	-103.10 dBm	-103.85 dBm	PASS	
WU Sensitivity **Wideband** **Center Hardware Channel**	-96.9 dBm max	-103.35 dBm	-103.10 dBm	-102.85 dBm	-102.85 dBm	PASS	
WU Sensitivity **Wideband** **Lower Hardware Channel**	-95.4 dBm max	-102.85 dBm	-102.60 dBm	-102.85 dBm	-102.35 dBm	PASS	

WU Sensitivity **Wideband** **Upper Hardware Channel**	-95.4 dBm max	-101.60 dBm	-100.10 dBm	-101.35 dBm	-100.60 dBm	PASS	Variation between Low, Upp, & Centr is within expected range. Asymmetry between Upp and Low due to Weaver filter roll off

Table 71 Wakeup Sensitivity

Impedance: 5 –j60

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Narrowband** **Center Hardware Channel**	-99 dBm max	-101.75 dBm	-102.00 dBm	-101.50 dBm	-102.00 dBm	PASS	
WU Sensitivity **Narrowband** **Lower Hardware Channel**	-97.5 dBm max	-101.50 dBm	-101.50 dBm	-101.25 dBm	-101.75 dBm	PASS	
WU Sensitivity **Narrowband** **Upper Hardware Channel**	-97.5 dBm max	-100.75 dBm	-101.00 dBm	-100.75 dBm	-100.75 dBm	PASS	
WU Sensitivity **Wideband** **Center Hardware Channel**	-93.5 dBm max	-101.50 dBm	-100.50 dBm	-100.50 dBm	-100.75 dBm	PASS	
WU Sensitivity **Wideband** **Lower Hardware Channel**	-92 dBm max	-100.25 dBm	-99.75 dBm	-100.50 dBm	-99.25 dBm	PASS	
WU Sensitivity **Wideband** **Upper Hardware Channel**	-92 dBm max	-99.00 dBm	-98.00 dBm	-99.00 dBm	-98.50 dBm	PASS	Variation between Low, Upp, & Centr is within expected range. Asymmetry between Upp and Low due to Weaver filter roll off

Table 72 Wakeup Sensitivity

Impedance: 5 – j20

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Narrowband** **Center Hardware Channel**	-100.2 dBm max	-102.70 dBm	-103.20 dBm	-102.70 dBm	-103.45 dBm	PASS	
WU Sensitivity **Narrowband** **Lower Hardware Channel**	-98.7 dBm max	-102.20 dBm	-102.45 dBm	-101.95 dBm	-103.20 dBm	PASS	
WU Sensitivity **Narrowband** **Upper Hardware Channel**	-98.7 dBm max	-101.70 dBm	-101.95 dBm	-101.45 dBm	-102.45 dBm	PASS	
WU Sensitivity **Wideband** **Center Hardware Channel**	-94.7 dBm max	-101.95 dBm	-101.45 dBm	-101.20 dBm	-101.95 dBm	PASS	
WU Sensitivity **Wideband** **Lower Hardware Channel**	-93.2 dBm max	-101.20 dBm	-100.70 dBm	-100.45 dBm	-101.20 dBm	PASS	
WU Sensitivity **Wideband** **Upper Hardware Channel**	-93.2 dBm max	-99.20 dBm	-99.20 dBm	-98.95 dBm	-99.95 dBm	PASS	Variation between Low, Upp, & Centr is within expected range. Asymmetry between Upp and Low due to Weaver filter roll off

Table 73 Wakeup Sensitivity

Impedance: 9 – j20

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Narrowband MICS** **Center Hardware Channel**	-101.3 dBm max	-104.05 dBm	-104.05 dBm	-103.80 dBm	-105.05 dBm	PASS	
WU Sensitivity **Narrowband MICS** **Lower Hardware Channel**	-99.8 dBm max	-104.55 dBm	-104.30 dBm	-104.30 dBm	-105.30 dBm	PASS	
WU Sensitivity **Narrowband MICS** **Upper Hardware Channel**	-99.8 dBm max	-104.30 dBm	-103.80 dBm	-103.30 dBm	-104.80 dBm	PASS	
WU Sensitivity **Narrowband MEDS** **Center Hardware Channel**	-100.4 dBm max	-103.90 dBm	-103.90 dBm	-103.65 dBm	-104.65 dBm	PASS	
WU Sensitivity **Narrowband MEDS** **Lower Hardware Channel**	-98.9 dBm max	-103.40 dBm	-103.15 dBm	-102.90 dBm	-103.65 dBm	PASS	
WU Sensitivity **Narrowband MEDS** **Upper Hardware Channel**	-98.9 dBm max	-102.90 dBm	-102.65 dBm	-102.40 dBm	-103.40 dBm	PASS	
WU Sensitivity **Wideband** **Center Hardware Channel**	-95.8 dBm max	-102.45 dBm	-101.95 dBm	-101.95 dBm	-102.70 dBm	PASS	

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
WU Sensitivity **Wideband** **Lower Hardware Channel**	-94.3 dBm max	-102.30 dBm	-102.05 dBm	-102.30 dBm	-101.30 dBm	PASS	
WU Sensitivity **Wideband** **Upper Hardware Channel**	-94.3 dBm max	-101.05 dBm	-100.80 dBm	-99.55 dBm	-100.55 dBm	PASS	Variation between Low, Upp, & Centr is within expected range. Asymmetry between Upp and Low due to Weaver filter roll off

Table 74 Wakeup Sensitivity

Conclusion: All results from Wakeup Sensitivity tests show compliance with the specification.

11.4 **Wakeup Dynamic Range**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.6.1

Test Executive Script: Rx_Sc13 through Rx_Sc19 and Rx_Sc21

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Missed Detect Rate (MDR) **Narrowband** **Center Hardware Channel**	**5% MDR max**	0.33% MDR	0.67% MDR	0.00% MDR	1.00% MDR	PASS	**From sensitivity to sensitivity +3 dB.**
Missed Detect Rate (MDR) **Narrowband** **Center Hardware Channel**	**2% MDR max**	0% MDR	0% MDR	0% MDR	0% MDR	PASS	**From sensitivity +3 dB to top of range.**

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Missed Detect Rate (MDR) **Narrowband** **Side (Weaver) Hardware Channel**	**5% MDR max**	0.67% MDR	1.00% MDR	0.67% MDR	0.67% MDR	PASS	**From sensitivity to sensitivity +3 dB.**
Missed Detect Rate (MDR) **Narrowband** **Side (Weaver) Hardware Channel**	**2% MDR max**	0% MDR	0% MDR	0% MDR	0% MDR	PASS	**From sensitivity +3 dB to top of range.**
Missed Detect Rate (MDR) **Wideband** **Center Hardware Channel**	**5% MDR max**	1.33% MDR	2.67% MDR	1.00% MDR	1.33% MDR	PASS	**From sensitivity to sensitivity +3 dB.**
Missed Detect Rate (MDR) **Wideband** **Center Hardware Channel**	**2% MDR max**	0.67% MDR	0.67% MDR	0.67% MDR	1.0% MDR	PASS	**From sensitivity +3 dB to top of range.**
Missed Detect Rate (MDR) **Wideband** **Side (Weaver) Hardware Channel**	**5% MDR max**	2.00% MDR	2.67% MDR	2.00% MDR	1.67% MDR	PASS	**From sensitivity to sensitivity +3 dB**
Missed Detect Rate (MDR) **Wideband** **Side (Weaver) Hardware Channel**	**2% MDR max**	1.0% MDR	0.67% MDR	0.67% MDR	1.0% MDR	PASS	**From sensitivity +3 dB to top of range.**

Table 75 Dynamic Range

Conclusion: All results from Wakeup Dynamic Range tests show compliance with the specification.

11.5 Wakeup Intermodulation Rejection (IMR)

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.6.2

Test Executive Script: Rx_Sc21

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
IMR Narrowband Center	44 dB min	52.75 dB	51.75 dB	52.75 dB	51.75 dB	PASS	5% MDR
IMR Narrowband Lower	44 dB min	51.75 dB	52.75 dB	52.75 dB	51.75 dB	PASS	
IMR Narrowband Upper	44 dB min	51.75 dB	51.75 dB	51.75 dB	51.75 dB	PASS	
IMR Wideband Center	44 dB min	50.75 dB	50.75 dB	50.75 dB	49.75 dB	PASS	
IMR Wideband Lower	44 dB min	50.75 dB	50.75 dB	49.75 dB	50.75 dB	PASS	
IMR Wideband Upper	44 dB min	49.75 dB	48.75 dB	48.75 dB	49.75 dB	PASS	

Table 76 Wakeup Intermodulation Rejection

Conclusion: All results from Wakeup Intermodulation Rejection tests show compliance with the specification.

11.6 **Wakeup Adjacent Channel Rejection (ACR)**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.6.3

Test Executive Script: Rx_Sc21

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
ACR Narrowband Center	25 dB min	34.00 dB	33.75 dB	34.00 dB	34.50 dB	PASS	5% MDR
ACR Narrowband Lower	25 dB min	35.00 dB	34.50 dB	34.75 dB	35.00 dB	PASS	
ACR Narrowband Upper	25 dB min	33.75 dB	33.50 dB	33.75 dB	34.25 dB	PASS	
ACR Wideband Center	25 dB min	40.75 dB	40.25 dB	40.00 dB	41.50 dB	PASS	
ACR Wideband Lower	25 dB min	41.75 dB	40.75 dB	41.25 dB	41.50 dB	PASS	
ACR Wideband Upper	25 dB min	39.75 dB	40.00 dB	39.75 dB	40.50 dB	PASS	

Table 77 Wakeup Adjacent Channel Rejection

Conclusion: All results from Wakeup Adjacent Channel Rejection tests show compliance with the specification.

11.7 **Weaver Image Rejection**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.6.4

Test Executive Script: Rx_Sc21

Note: It may not be clear to the reader what the difference is between Image Rejection and Alternate Channel Rejection (AltCR) is. For any desired side-channel there are two "alternate" channels, one that is two channels higher in frequency and one that is two channels lower in frequency. In the case of Wakeup Weaver Lower Channel (one below the center channel), the Image Rejection channel is the Alternate channel on the upper side. In the case of Wakeup Weaver Upper Channel (one above the center channel), the Image Rejection channel is the Alternate channel on the lower side. The Wakeup Alternate Channel Rejection is part of Wakeup External Spurious Response and therefore reported in section 11.8 Wakeup External Spurious Response and Blocking.

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Image Rej. Narrow Lower	25 dB min	36.25 dB	39.75 dB	44.50 dB	36.75 dB	PASS	5% MDR
Image Rej. Narrow Upper	25 dB min	36.75 dB	37.75 dB	36.75 dB	34.25 dB	PASS	
Image Rej. Wide Lower	25 dB min	39.50 dB	43.00 dB	38.00 dB	39.75 dB	PASS	
Image Rej. Wide Upper	25 dB min	31.75 dB	32.75 dB	36.75 dB	29.50 dB	PASS	

Table 78 Wakeup Image Rejection Channel Rejection

Conclusion: All results from Wakeup Weaver Image Rejection tests show compliance with the specification.

11.8 **Wakeup External Spurious Response and Blocking**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.6.5

This test has a deviation (Deviation #12) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

This test has a deviation (Deviation #14) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

Test Executive Script: SpurSc2, Spur_Sc3, Spur_Sc4, Spur_Sc8, Spur_Sc9, Spur_Sc10

Note: The DVT Plan calls for an interferer at specification limit for the Wakeup spurious fine test. The spurs from the main radio coarse spurious response test (section 9.6) were tested by the Wakeup spurious fine response test with 7dB overtesting for frequencies above 409MHz and below 398MHz, making it a much more conservative test. Within the 398 to 409MHz range the Wakeup fine spurious test was run at the specified level without overtesting. The overtesting was not needed because a CW rather than a modulated interferer is used in the fine test and the loss calibration made at 403.5MHz is valid within this frequency range. Note that within the frequency range of 400.1 MHz to 406.9 MHz (which was not covered by the main radio coarse test), the Wakeup spurious coarse test was run with 7dB of overtesting (see deviation #12 for details on this).

Test Parameter	Spec	sn337	sn338	sn343	Compliance	Notes
Interferer Level **WideBand - Center Channel** **Center of Alternate channel and beyond**	-55 dBm min	-44.25 dBm	-44.25 dBm	-45.25 dBm	PASS	5% MDR
Interferer Level **NarrowBand - Center Channel** **Center of Alternate channel through Alternate channel + 2**	-60 dBm min	-53.25 dBm	-52.50 dBm	-54.50 dBm	PASS	5% MDR
Interferer Level **NarrowBand - Center Channel** **Alternate channel + 3 and beyond**	-55 dBm min	-46.50 dBm	-45.50 dBm	-48.25 dBm	PASS	5% MDR
Interferer Level relative to specified sensitivity **NB & WB - Side (Weaver) Channels** **Alternate channel + 1 and beyond except channels including 3rd harmonic of Weaver LO**	25 dB min	49.25 dB	50.00 dB	47.75 dB	PASS	5% MDR
Interferer Level relative to specified sensitivity **NB & WB - Side (Weaver) Channels** **Channels including 3rd Harmonic of Weaver LO**	20 dB min	24.75 dB	26.75 dB	26.00 dB	PASS	5% MDR Includes Alternate channel for weaver side and alternate channel + 2 for image side.

Table 79 Wakeup External Spurious Response and Blocking

Conclusion: All results from Wakeup External Spurious Response and Blocking tests show compliance with the specification.

11.9 **Wakeup Performance – Current and Latency**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.4.8.4

DVT Plan: 6.6.6

Test Executive Script: Rx_Sc20

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Standby Current	**A:100 nA max(37C)** **B:300 nA max(0-55C)**	A:41 nA B:89 nA	A:40 nA B:88 nA	A:40 nA B:87 nA	A:39 nA B:87 nA	PASS	DVT Plan 6.6.6.1
Latency (MICS) **A:25s** **B: 5s** **C: 2s**	**A: 25s max** **B: 5s max** **C: 2s max**	A: 24.80s B: 4.93s C: 1.97s	A: 24.77s B: 4.93s C: 1.97s	A: 24.77s B: 4.93s C: 1.97s	A: 24.77s B: 4.93s C: 1.97s	PASS	DVT Plan 6.6.6.2 DVT Plan 6.6.6.4 DVT Plan 6.6.6.3
Current (MICS) **25 s latency** **37C** **C1 no intfr.**	**250 nA max**	156 nA	156 nA	157 nA	156 nA	PASS	DVT Plan 6.6.6.2 DVT Plan 6.6.6.4 DVT Plan 6.6.6.3
Current (MICS) **25 s latency** **0-55C** **C1 no intfr**	**450 nA max**	210 nA	208 nA	210 nA	209 nA	PASS	DVT Plan 6.6.6.2 DVT Plan 6.6.6.4 DVT Plan 6.6.6.3
Current (MICS) **5 s latency** **37C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr.**	**1000 nA max**	A. 734 nA B. 734 nA C. 736 nA D. 628 nA	A. 734 nA B. 734 nA C. 735 nA D. 624 nA	A. 745 nA B. 745 nA C. 745 nA D. 634 nA	A. 739 nA B. 740 nA C. 739 nA D. 629 nA	PASS	

Current (MICS) **5 s latency** **0-55C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr.**	**1300 nA max**	A. 806 nA B. 807 nA C. 807 nA D. 702 nA	A. 803 nA B. 803 nA C. 804 nA D. 698 nA	A. 813 nA B. 813 nA C. 813 nA D. 707 nA	A. 809 nA B. 809 nA C. 810 nA D. 703 nA	PASS	
Current (MICS) **2 s latency** **37C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr.**	**2140 nA max**	A. 1643 nA B. 1644 nA C. 1649 nA D. 1541 nA	A. 1641 nA B. 1641 nA C. 1643 nA D. 1532 nA	A. 1668 nA B. 1668 nA C. 1668 nA D. 1557 nA	A. 1654 nA B. 1654 nA C. 1654 nA D. 1544 nA	PASS	
Current (MICS) **2 s latency** **0-55C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr.**	**2500 nA max**	A. 1759 nA B. 1760 nA C. 1759 nA D. 1655 nA	A. 1751 nA B. 1751 nA C. 1752 nA D. 1646 nA	A. 1775 nA B. 1775 nA C. 1775 nA D. 1669 nA	A. 1765 nA B. 1765 nA C. 1766 nA D. 1660 nA	PASS	
Latency (MEDS) **A: 5 seconds** **B: 2 seconds**	**A: 5 seconds max** **B: 2 seconds max**	A: 4.87 s B: 1.87 s	A: 4.87 s B: 1.87 s	A: 4.87 s B: 1.87 s	A: 4.87 s B: 1.87 s	PASS	DVT Plan 6.6.6.6 DVT Plan 6.6.6.5
Current (MEDS) **5 seconds latency** **37C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr.**	**10,000 nA max**	A. 5649 nA B. 5726 nA C. 5728 nA D. 5546 nA	A. 5623 nA B. 5736 nA C. 5755 nA D. 5518 nA	A. 5712 nA B. 5761 nA C. 5795 nA D. 5605 nA	A. 5667 nA B. 5712 nA C. 5764 nA D. 5553 nA	PASS	DVT Plan 6.6.6.6 DVT Plan 6.6.6.5

Current (MEDS) **5 seconds latency** **0-55C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr**	**11,000 nA max**	A. 5929 nA B. 5958 nA C. 5982 nA D. 5833 nA	A. 5916 nA B. 6097 nA C. 6097 nA D. 5819 nA	A. 5984 nA B. 6003 nA C. 6070 nA D. 5884 nA	A. 5957 nA B. 6060 nA C. 6095 nA D. 5860 nA	PASS	
Current (MEDS) **2 seconds latency** **37C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr**	**17,100 nA max**	A.14985 nA B.15191 nA C.15196 nA D.14891 nA	A.14914 nA B.15325 nA C.15268 nA D.14817 nA	A.15150 nA B.15282 nA C.15374 nA D.15052 nA	A.15030 nA B.15152 nA C.15292 nA D.14911 nA	PASS	
Current (MEDS) **2 seconds latency** **0-55C** **A. No intfr.** **B. 10% intfr.** **C. 100% intfr.** **D. C1 no intfr**	**18,000 nA max**	A.15664 nA B.15741 nA C.15808 nA D.15585	A.15632 nA B.16276 nA C.16119 nA D.15549	A.15812 nA B.15863 nA C.16043 nA D.15724	A.15739 nA B.16017 nA C.16113 nA D.15659	PASS	

Table 80 Wakeup Performance – Current and Latency

Conclusion: All results from Wakeup Current and Latency tests show compliance with the specification.

12 **Supply Currents Test Results**

Note: BPFLT Supply Current is a bench test.

12.1 **DVDD Supply Current**

Mozart Requirement Specification (MDT Document #A17245); rev I paragraph: 2.3.1.2

DVT Plan: 6.7.4

Test Executive Script: Rx_Sc22

This test has a deviation (Deviation #18) from the DVT Plan. See Section 6 DVT plan Deviations for more information.

Test Parameter	Spec	sn316	sn317	sn318	sn320	Compliance	Notes
Functional Static Current **Max DVDD** **Max BPFLT**	**10 uA max**	37 nA	38 nA	37 nA	37 nA	PASS	
Functional Static Current **DVDD=BPFLT + 200mV**	**10.1 uA max**	11 nA	11 nA	10 nA	11 nA	PASS	
Disabled Static Current **Max DVDD** **Max BPFLT**	**A. 50 nA (25C)** **B. 100 nA (37C)** **C. 500 nA (0-55C)** **max**	A. 13 nA B. 19 nA C. 37 nA	A. 13 nA B. 19 nA C. 37 nA	A. 13 nA B. 19 nA C. 37 nA	A. 12 nA B. 19 nA C. 36 nA	PASS	
Disabled Static Current **DVDD=BPFLT + 200mV**	**A. 150 nA (25C)** **B. 200 nA (37C)** **C. 600 nA (0-55C)** **max**	A. 1 nA B. 3 nA C. 11 nA	A. 1 nA B. 3 nA C. 11 nA	A. 1 nA B. 3 nA C. 9 nA	A. 1 nA B. 3 nA C. 10 nA	PASS	

Table 81 DVDD Supply Current

Conclusion: All results from DVDD Supply Current tests show compliance with the specification.

13 Conclusion

All results from DVT tests show compliance with the specification.

14 Appendix A – Resolution of Non-compliance

N/A

Cardiac Disease Rhythm Management **MEDTRONIC CONFIDENTIAL** This document is the property of Medtronic, Inc. and must be accounted for. Information herein is confidential. Do not reproduce it, reveal it to unauthorized persons or send it outside Medtronic without permission.	**Project DHF name:** Telemetry M	**Effectivity** Upon Approval	**Version** 2.0	**Page** Cover
	Title **Tel M FWVT Test Results and Status Phase 1 TCP-M**			
	This document is electronically controlled. Printed copies are considered uncontrolled.			

Deliverable(s):	FWVT Test Results and Status

Note: This document may be approved electronically through Documentum by the listed approvers.

Reviewed and Approved By:	**Name**	**Signatures and Date**
	Javaid Masoud	<Electronic Signature>
	Kenneth Kahle	<Electronic Signature>

Change History

Version	**Author**	**Description of Change**
2.0	Kenneth Kahle	Initial release.

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1

FWVTBatch Version: 1.0.0.0
Proctor of execution: powelm4
Configuration file: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Testmode1.xml

Total tests	Pass	Fail	Truncated	Exception	No result	No build	Pass Ratio
528	528	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2

FWVTBatch Version: 1.0.0.0
Proctor of execution: powelm4
Configuration file: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Testmode2.xml

Total tests	Pass	Fail	Truncated	Exception	No result	No build	Pass Ratio
528	528	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3

FWVTBatch Version: 1.0.0.0
Proctor of execution: powelm4
Configuration file: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Testmode3.xml

Total tests	Pass	Fail	Truncated	Exception	No result	No build	Pass Ratio
527	527	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4

FWVTBatch Version: 1.0.0.0
Proctor of execution: powelm4
Configuration file: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Testmode4.xml

Total tests	Pass	Fail	Truncated	Exception	No result	No build	Pass Ratio

527	527	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8

FWVTBatch Version: 1.0.0.0
Proctor of execution: powelm4
Configuration file: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Testmode7_8.xml

Total tests	Pass	Fail	Truncated	Exception	No result	No build	Pass Ratio
528	528	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10

FWVTBatch Version: 1.0.0.0
Proctor of execution: powelm4
Configuration file: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Testmode9_10.xml

Total tests	Pass	Fail	Truncated	Exception	No result	No build	Pass Ratio
528	528	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1

Feature	Total	Pass	Fail	NoResult	NoBuild	Truncated	Exception	Pass %
M_TCP_AntennaSelection	2	2	0	0	0	0	0	100.0%
M_TCP_Channel	61	61	0	0	0	0	0	100.0%
M_TCP_Close	20	20	0	0	0	0	0	100.0%
M_TCP_Data	132	132	0	0	0	0	0	100.0%
M_TCP_Diagnostic	36	36	0	0	0	0	0	100.0%
M_TCP_Discover	30	30	0	0	0	0	0	100.0%
M_TCP_EEPROM	7	7	0	0	0	0	0	100.0%
M_TCP_Emergency	29	29	0	0	0	0	0	100.0%
M_TCP_General	14	14	0	0	0	0	0	100.0%
M_TCP_Initialize	4	4	0	0	0	0	0	100.0%
M_TCP_MedicalEvent	16	16	0	0	0	0	0	100.0%
M_TCP_MemoryTest	5	5	0	0	0	0	0	100.0%
M_TCP_Network	28	28	0	0	0	0	0	100.0%
M_TCP_Open	28	28	0	0	0	0	0	100.0%
M_TCP_Polling	32	32	0	0	0	0	0	100.0%
M_TCP_PowerInhibit	11	11	0	0	0	0	0	100.0%
M_TCP_RawMode	12	12	0	0	0	0	0	100.0%
M_TCP_Reset	2	2	0	0	0	0	0	100.0%
M_TCP_Security	45	45	0	0	0	0	0	100.0%
M_TCP_WirelessWakeup	14	14	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2

Feature	Total	Pass	Fail	NoResult	NoBuild	Truncated	Exception	Pass %
M_TCP_AntennaSelection	2	2	0	0	0	0	0	100.0%
M_TCP_Channel	61	61	0	0	0	0	0	100.0%
M_TCP_Close	20	20	0	0	0	0	0	100.0%
M_TCP_Data	132	132	0	0	0	0	0	100.0%
M_TCP_Diagnostic	36	36	0	0	0	0	0	100.0%
M_TCP_Discover	30	30	0	0	0	0	0	100.0%
M_TCP_EEPROM	7	7	0	0	0	0	0	100.0%
M_TCP_Emergency	29	29	0	0	0	0	0	100.0%
M_TCP_General	14	14	0	0	0	0	0	100.0%
M_TCP_Initialize	4	4	0	0	0	0	0	100.0%
M_TCP_MedicalEvent	16	16	0	0	0	0	0	100.0%

M_TCP_MemoryTest	5	5	0	0	0	0	0	100.0%
M_TCP_Network	28	28	0	0	0	0	0	100.0%
M_TCP_Open	28	28	0	0	0	0	0	100.0%
M_TCP_Polling	32	32	0	0	0	0	0	100.0%
M_TCP_PowerInhibit	11	11	0	0	0	0	0	100.0%
M_TCP_RawMode	12	12	0	0	0	0	0	100.0%
M_TCP_Reset	2	2	0	0	0	0	0	100.0%
M_TCP_Security	45	45	0	0	0	0	0	100.0%
M_TCP_WirelessWakeup	14	14	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3

Feature	Total	Pass	Fail	NoResult	NoBuild	Truncated	Exception	Pass %
M_TCP_AntennaSelection	2	2	0	0	0	0	0	100.0%
M_TCP_Channel	61	61	0	0	0	0	0	100.0%
M_TCP_Close	20	20	0	0	0	0	0	100.0%
M_TCP_Data	131	131	0	0	0	0	0	100.0%
M_TCP_Diagnostic	36	36	0	0	0	0	0	100.0%
M_TCP_Discover	30	30	0	0	0	0	0	100.0%
M_TCP_EEPROM	7	7	0	0	0	0	0	100.0%
M_TCP_Emergency	29	29	0	0	0	0	0	100.0%
M_TCP_General	14	14	0	0	0	0	0	100.0%
M_TCP_Initialize	4	4	0	0	0	0	0	100.0%
M_TCP_MedicalEvent	16	16	0	0	0	0	0	100.0%
M_TCP_MemoryTest	5	5	0	0	0	0	0	100.0%
M_TCP_Network	28	28	0	0	0	0	0	100.0%
M_TCP_Open	28	28	0	0	0	0	0	100.0%
M_TCP_Polling	32	32	0	0	0	0	0	100.0%
M_TCP_PowerInhibit	11	11	0	0	0	0	0	100.0%
M_TCP_RawMode	12	12	0	0	0	0	0	100.0%
M_TCP_Reset	2	2	0	0	0	0	0	100.0%
M_TCP_Security	45	45	0	0	0	0	0	100.0%
M_TCP_WirelessWakeup	14	14	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4

Feature	Total	Pass	Fail	NoResult	NoBuild	Truncated	Exception	Pass %

M_TCP_AntennaSelection	2	2	0	0	0	0	0	100.0%
M_TCP_Channel	61	61	0	0	0	0	0	100.0%
M_TCP_Close	20	20	0	0	0	0	0	100.0%
M_TCP_Data	131	131	0	0	0	0	0	100.0%
M_TCP_Diagnostic	36	36	0	0	0	0	0	100.0%
M_TCP_Discover	30	30	0	0	0	0	0	100.0%
M_TCP_EEPROM	7	7	0	0	0	0	0	100.0%
M_TCP_Emergency	29	29	0	0	0	0	0	100.0%
M_TCP_General	14	14	0	0	0	0	0	100.0%
M_TCP_Initialize	4	4	0	0	0	0	0	100.0%
M_TCP_MedicalEvent	16	16	0	0	0	0	0	100.0%
M_TCP_MemoryTest	5	5	0	0	0	0	0	100.0%
M_TCP_Network	28	28	0	0	0	0	0	100.0%
M_TCP_Open	28	28	0	0	0	0	0	100.0%
M_TCP_Polling	32	32	0	0	0	0	0	100.0%
M_TCP_PowerInhibit	11	11	0	0	0	0	0	100.0%
M_TCP_RawMode	12	12	0	0	0	0	0	100.0%
M_TCP_Reset	2	2	0	0	0	0	0	100.0%
M_TCP_Security	45	45	0	0	0	0	0	100.0%
M_TCP_WirelessWakeup	14	14	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8

Feature	Total	Pass	Fail	NoResult	NoBuild	Truncated	Exception	Pass %
M_TCP_AntennaSelection	2	2	0	0	0	0	0	100.0%
M_TCP_Channel	61	61	0	0	0	0	0	100.0%
M_TCP_Close	20	20	0	0	0	0	0	100.0%
M_TCP_Data	132	132	0	0	0	0	0	100.0%
M_TCP_Diagnostic	36	36	0	0	0	0	0	100.0%
M_TCP_Discover	30	30	0	0	0	0	0	100.0%
M_TCP_EEPROM	7	7	0	0	0	0	0	100.0%
M_TCP_Emergency	29	29	0	0	0	0	0	100.0%
M_TCP_General	14	14	0	0	0	0	0	100.0%
M_TCP_Initialize	4	4	0	0	0	0	0	100.0%
M_TCP_MedicalEvent	16	16	0	0	0	0	0	100.0%
M_TCP_MemoryTest	5	5	0	0	0	0	0	100.0%
M_TCP_Network	28	28	0	0	0	0	0	100.0%

M_TCP_Open	28	28	0	0	0	0	0	100.0%
M_TCP_Polling	32	32	0	0	0	0	0	100.0%
M_TCP_PowerInhibit	11	11	0	0	0	0	0	100.0%
M_TCP_RawMode	12	12	0	0	0	0	0	100.0%
M_TCP_Reset	2	2	0	0	0	0	0	100.0%
M_TCP_Security	45	45	0	0	0	0	0	100.0%
M_TCP_WirelessWakeup	14	14	0	0	0	0	0	100.0%

Test Results From: \\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10

Feature	Total	Pass	Fail	NoResult	NoBuild	Truncated	Exception	Pass %
M_TCP_AntennaSelection	2	2	0	0	0	0	0	100.0%
M_TCP_Channel	61	61	0	0	0	0	0	100.0%
M_TCP_Close	20	20	0	0	0	0	0	100.0%
M_TCP_Data	132	132	0	0	0	0	0	100.0%
M_TCP_Diagnostic	36	36	0	0	0	0	0	100.0%
M_TCP_Discover	30	30	0	0	0	0	0	100.0%
M_TCP_EEPROM	7	7	0	0	0	0	0	100.0%
M_TCP_Emergency	29	29	0	0	0	0	0	100.0%
M_TCP_General	14	14	0	0	0	0	0	100.0%
M_TCP_Initialize	4	4	0	0	0	0	0	100.0%
M_TCP_MedicalEvent	16	16	0	0	0	0	0	100.0%
M_TCP_MemoryTest	5	5	0	0	0	0	0	100.0%
M_TCP_Network	28	28	0	0	0	0	0	100.0%
M_TCP_Open	28	28	0	0	0	0	0	100.0%
M_TCP_Polling	32	32	0	0	0	0	0	100.0%
M_TCP_PowerInhibit	11	11	0	0	0	0	0	100.0%
M_TCP_RawMode	12	12	0	0	0	0	0	100.0%
M_TCP_Reset	2	2	0	0	0	0	0	100.0%
M_TCP_Security	45	45	0	0	0	0	0	100.0%
M_TCP_WirelessWakeup	14	14	0	0	0	0	0	100.0%

Feature	Test	Test Path	Revision #	Revision Date	CVS Rev	CVS State	TelM Firmware Revision	TelM RAMware	Sys Init Time	Test Time (mS)	Tester	Tester Rev	Status	Warnings	Errors	Fails	Passes
M_TCP_AntennaSelection	m_Antenna001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna001\Hardware\m_Antenna001.exe	2.4	2010/06/16 12:31:48	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 07:10:19	238024	TELM-06	1	Pass	0	0	0	127
M_TCP_AntennaSelection	m_Antenna002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna002\Hardware\m_Antenna002.exe	2.0	2010/06/16 12:29:50	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 07:03:21	287177	TELM-06	1	Pass	0	0	0	99
M_TCP_Channel	m_ChannelRecovery001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery001\Hardware\m_ChannelRecovery001.exe	2.5	2010/04/08 14:25:18	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:03:43	15301	TELM-06	1	Pass	0	0	0	26
M_TCP_Channel	m_ChannelRecovery002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery002\Hardware\m_ChannelRecovery002.exe	2.2	2009/11/30 22:31:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:01:50	17560	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery003\Hardware\m_ChannelRecovery003.exe	2.2	2009/11/30 22:32:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:00:23	17550	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery004\Hardware\m_ChannelRecovery004.exe	2.2	2010/05/26 12:04:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:58:24	14525	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery005\Hardware\m_ChannelRecovery005.exe	2.5	2010/06/21 18:34:44	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:55:46	14155	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery006\Hardware\m_ChannelRecovery006.exe	2.4	2010/05/07 17:09:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:50:58	16073	TELM-06	1	Pass	0	0	0	35
M_TCP_Channel	m_ChannelRecovery007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery007\Hardware\m_ChannelRecovery007.exe	2.7	2010/05/11 11:49:24	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 20:48:03	28872	TELM-06	1	Pass	0	0	0	89
M_TCP_Channel	m_ChannelRecovery008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery008\Hardware\m_ChannelRecovery008.exe	2.6	2010/05/13 15:05:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:44:39	41291	TELM-06	1	Pass	0	0	0	574
M_TCP_Channel	m_ChannelRecovery009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery009\Hardware\m_ChannelRecovery009.exe	2.6	2010/06/22 15:16:15	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:39:56	65139	TELM-06	1	Pass	0	0	0	342
M_TCP_Channel	m_ChannelRecovery010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery010\Hardware\m_ChannelRecovery010.exe	2.8	2010/05/26 15:48:50	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 20:36:54	25296	TELM-06	1	Pass	0	0	0	132
M_TCP_Channel	m_ChannelRecovery011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery011\Hardware\m_ChannelRecovery011.exe	2.5	2010/05/13 12:31:10	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:34:55	16129	TELM-06	1	Pass	0	0	0	14
M_TCP_Channel	m_ChannelRecovery012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery012\Hardware\m_ChannelRecovery012.exe	2.6	2010/09/07 20:36:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:32:08	49065	TELM-06	1	Pass	0	0	0	193
M_TCP_Channel	m_ChannelRecovery013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery013\Hardware\m_ChannelRecovery013.exe	2.2	2010/01/05 19:17:54	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:27:03	29704	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery014\Hardware\m_ChannelRecovery014.exe	2.8	2010/05/13 13:19:47	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 20:24:49	62621	TELM-06	1	Pass	0	0	0	23
M_TCP_Channel	m_ChannelRecovery015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery015\Hardware\m_ChannelRecovery015.exe	2.6	2010/06/03 18:01:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:22:03	12529	TELM-06	1	Pass	0	0	0	88
M_TCP_Channel	m_ChannelRecovery016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery016\Hardware\m_ChannelRecovery016.exe	2.4	2010/02/11 20:10:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:19:55	38005	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery017\Hardware\m_ChannelRecovery017.exe	2.3	2010/02/11 17:38:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:18:12	10442	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery018\Hardware\m_ChannelRecovery018.exe	2.3	2010/02/01 19:41:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:14:55	18069	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery020\Hardware\m_ChannelRecovery020.exe	2.7	2010/06/23 17:42:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 20:13:26	38354	TELM-06	1	Pass	0	0	0	109
M_TCP_Channel	m_ChannelRecovery021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery021\Hardware\m_ChannelRecovery021.exe	2.3	2010/04/13 18:33:24	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:11:32	38271	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery022\Hardware\m_ChannelRecovery022.exe	2.3	2010/04/13 18:36:50	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:09:50	20011	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery023\Hardware\m_ChannelRecovery023.exe	2.5	2010/04/13 18:41:23	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:07:44	29932	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery024\Hardware\m_ChannelRecovery024.exe	2.4	2010/04/07 12:33:58	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:05:57	19918	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery025\Hardware\m_ChannelRecovery025.exe	2.2	2010/02/01 19:43:28	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:04:23	11094	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery026\Hardware\m_ChannelRecovery026.exe	2.2	2010/02/11 21:01:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:02:26	12556	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery027\Hardware\m_ChannelRecovery027.exe	2.2	2010/02/03 17:18:54	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:00:46	13664	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery028\Hardware\m_ChannelRecovery028.exe	2.2	2010/02/03 17:19:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:58:54	12575	TELM-06	1	Pass	0	0	0	10
M_TCP_Channel	m_ChannelRecovery029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery029\Hardware\m_ChannelRecovery029.exe	2.2	2009/12/18 20:13:33	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:57:15	13526	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery030\Hardware\m_ChannelRecovery030.exe	2.10	2010/06/23 21:15:31	2.10	Up-to-date	Baseline 17_1	None	09/09/2010 19:55:11	19636	TELM-06	1	Pass	0	0	0	30
M_TCP_Channel	m_ChannelRecovery031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery031\Hardware\m_ChannelRecovery031.exe	2.6	2010/04/13 18:46:58	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:53:26	18991	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery032\Hardware\m_ChannelRecovery032.exe	2.5	2010/06/23 21:50:13	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:51:17	18231	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery033\Hardware\m_ChannelRecovery033.exe	2.3	2010/04/07 17:51:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:48:21	19440	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery038\Hardware\m_ChannelRecovery038.exe	2.6	2010/05/26 15:55:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:46:58	20951	TELM-06	1	Pass	0	0	0	175
M_TCP_Channel	m_ChannelRecovery039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery039\Hardware\m_ChannelRecovery039.exe	2.3	2010/02/10 20:47:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:43:55	12348	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery040\Hardware\m_ChannelRecovery040.exe	2.5	2010/06/07 12:54:38	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:42:38	15563	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery041\Hardware\m_ChannelRecovery041.exe	2.1	2009/10/22 13:18:38	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:40:38	38568	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery042\Hardware\m_ChannelRecovery042.exe	2.3	2009/12/03 15:57:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:38:46	21980	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery043\Hardware\m_ChannelRecovery043.exe	2.6	2010/06/24 18:37:58	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:36:16	56017	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery045\Hardware\m_ChannelRecovery045.exe	2.2	2009/11/12 21:22:11	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:34:49	11701	TELM-06	1	Pass	0	0	0	44
M_TCP_Channel	m_ChannelRecovery046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery046\Hardware\m_ChannelRecovery046.exe	2.3	2009/11/16 18:40:10	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:33:17	12965	TELM-06	1	Pass	0	0	0	49
M_TCP_Channel	m_ChannelRecovery047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery047\Hardware\m_ChannelRecovery047.exe	2.1	2009/11/16 18:58:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:31:11	52924	TELM-06	1	Pass	0	0	0	95
M_TCP_Channel	m_ChannelRecovery048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery048\Hardware\m_ChannelRecovery048.exe	2.3	2010/06/24 18:46:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:29:14	64355	TELM-06	1	Pass	0	0	0	156
M_TCP_Channel	m_ChannelRecovery049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery049\Hardware\m_ChannelRecovery049.exe	2.5	2010/03/26 13:55:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:27:40	36893	TELM-06	1	Pass	0	0	0	204
M_TCP_Channel	m_ChannelRecovery050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery050\Hardware\m_ChannelRecovery050.exe	2.2	2009/11/20 13:38:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:26:19	19032	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery051\Hardware\m_ChannelRecovery051.exe	2.3	2010/06/24 19:40:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:23:53	85007	TELM-06	1	Pass	0	0	0	196
M_TCP_Channel	m_ChannelRecovery053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery053\Hardware\m_ChannelRecovery053.exe	2.3	2010/06/24 21:57:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:22:27	32186	TELM-06	1	Pass	0	0	0	96
M_TCP_Channel	m_ChannelRecovery054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery054\Hardware\m_ChannelRecovery054.exe	2.4	2010/06/24 22:31:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:21:05	10307	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery055\Hardware\m_ChannelRecovery055.exe	2.4	2010/02/11 14:12:25	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 20:05:52	2022502	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery061	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery061\Hardware\m_ChannelRecovery061.exe	2.3	2010/04/06 11:52:17	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:19:48	20866	TELM-06	1	Pass	0	0	0	12
M_TCP_Channel	m_ChannelRecovery062	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery062\Hardware\m_ChannelRecovery062.exe	2.2	2010/06/25 15:47:28	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:18:27	13203	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery063	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery063\Hardware\m_ChannelRecovery063.exe	2.9	2010/05/13 17:43:56	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 19:15:55	24739	TELM-06	1	Pass	0	0	0	39
M_TCP_Channel	m_ChannelRecovery064	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery064\Hardware\m_ChannelRecovery064.exe	2.4	2010/05/06 14:14:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:14:05	40804	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery065\Hardware\m_ChannelRecovery065.exe	2.2	2010/05/06 15:43:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:11:39	37292	TELM-06	1	Pass	0	0	0	7
M_TCP_Channel	m_ChannelRecovery066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery066\Hardware\m_ChannelRecovery066.exe	2.2	2010/05/06 15:49:13	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:09:43	37561	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery067\Hardware\m_ChannelRecovery067.exe	2.2	2010/05/06 15:58:13	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:04:29	40850	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery068\Hardware\m_ChannelRecovery068.exe	2.3	2010/06/03 18:10:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:02:35	34803	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery069	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery069\Hardware\m_ChannelRecovery069.exe	2.3	2010/05/10 15:40:47	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 12:39:18	1009467	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery070	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery070\Hardware\m_ChannelRecovery070.exe	2.1	2010/02/04 18:14:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:00:09	13041	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery071	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery071\Hardware\m_ChannelRecovery071.exe	2.1	2010/02/04 17:57:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:58:36	15728	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery072	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery072\Hardware\m_ChannelRecovery072.exe	2.3	2010/05/07 11:42:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:56:44	44191	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery073	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery073\Hardware\m_ChannelRecovery073.exe	2.3	2010/05/07 11:43:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:52:58	44246	TELM-06	1	Pass	0	0	0	8
M_TCP_Close	m_Close001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close001\Hardware\m_Close001.exe	2.6	2010/03/29 17:53:05	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:15:33	7682	TELM-06	1	Pass	0	0	0	51
M_TCP_Close	m_Close002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close002\Hardware\m_Close002.exe	2.3	2010/01/26 20:03:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:13:51	8705	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close003\Hardware\m_Close003.exe	2.3	2010/06/24 15:01:11	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:11:45	6956	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close004\Hardware\m_Close004.exe	2.5	2010/03/16 12:03:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:09:25	14834	TELM-06	1	Pass	0	0	0	149
M_TCP_Close	m_Close005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close005\Hardware\m_Close005.exe	2.5	2010/05/24 17:00:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:06:54	19295	TELM-06	1	Pass	0	0	0	33
M_TCP_Close	m_Close006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close006\Hardware\m_Close006.exe	2.2	2009/11/19 20:38:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:04:00	10956	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close007\Hardware\m_Close007.exe	2.1	2010/02/04 18:46:09	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:00:52	16371	TELM-06	1	Pass	0	0	0	22
M_TCP_Close	m_Close008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close008\Hardware\m_Close008.exe	2.4	2010/05/12 15:22:15	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:57:39	7026	TELM-06	1	Pass	0	0	0	10
M_TCP_Close	m_Close009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close009\Hardware\m_Close009.exe	2.2	2010/02/04 15:29:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:55:23	7928	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close010\Hardware\m_Close010.exe	2.0	2009/09/21 12:45:39	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:53:55	15804	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close011\Hardware\m_Close011.exe	2.0	2009/09/21 12:46:09	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:52:27	8400	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close012\Hardware\m_Close012.exe	2.0	2009/09/21 12:46:36	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:51:21	11554	TELM-06	1	Pass	0	0	0	3
M_TCP_Close	m_Close013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close013\Hardware\m_Close013.exe	2.0	2009/09/21 12:18:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:49:31	7968	TELM-06	1	Pass	0	0	0	4
M_TCP_Close	m_Close014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close014\Hardware\m_Close014.exe	2.1	2010/01/29 16:45:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:48:32	8829	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close016\Hardware\m_Close016.exe	2.2	2010/08/24 12:05:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:47:24	8162	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close017\Hardware\m_Close017.exe	2.0	2009/09/21 12:47:07	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:44:54	9456	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close019\Hardware\m_Close019.exe	2.1	2009/10/01 14:47:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:43:22	5088	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close020\Hardware\m_Close020.exe	2.2	2009/10/30 14:47:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:42:12	9659	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close021\Hardware\m_Close021.exe	2.0	2009/09/21 12:48:28	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:40:58	2798	TELM-06	1	Pass	0	0	0	1
M_TCP_Close	m_Close022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close022\Hardware\m_Close022.exe	2.2	2009/10/30 14:53:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:39:16	12524	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_ARQ001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ001\Hardware\m_ARQ001.exe	2.5	2010/06/29 15:20:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:42:47	12606	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ002\Hardware\m_ARQ002.exe	2.0	2009/09/21 18:25:53	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:41:06	8100	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ003\Hardware\m_ARQ003.exe	2.3	2010/04/05 12:22:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:39:22	8678	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_ARQ004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ004\Hardware\m_ARQ004.exe	2.1	2009/11/04 21:22:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:37:28	8387	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_ARQ005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ005\Hardware\m_ARQ005.exe	2.2	2009/12/23 13:26:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:34:16	7607	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ006\Hardware\m_ARQ006.exe	2.1	2009/09/24 16:41:52	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:31:52	9030	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_ARQ007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ007\Hardware\m_ARQ007.exe	2.0	2009/09/21 18:27:29	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:28:42	11158	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_ARQ008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ008\Hardware\m_ARQ008.exe	2.0	2009/09/21 18:28:06	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:26:44	11934	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ009\Hardware\m_ARQ009.exe	2.2	2010/02/04 13:08:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:25:27	11434	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ010\Hardware\m_ARQ010.exe	2.2	2010/02/04 13:19:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:23:29	11404	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ011\Hardware\m_ARQ011.exe	2.3	2010/02/01 13:30:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:21:21	10372	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ012\Hardware\m_ARQ012.exe	2.3	2010/02/18 15:17:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:18:42	9639	TELM-06	1	Pass	0	0	0	211
M_TCP_Data	m_ARQ013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ013\Hardware\m_ARQ013.exe	2.3	2010/02/03 16:38:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:17:06	12039	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_ARQ014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ014\Hardware\m_ARQ014.exe	2.4	2010/04/05 11:08:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:14:06	8697	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ015\Hardware\m_ARQ015.exe	2.6	2010/06/08 15:25:41	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:11:49	14204	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ016\Hardware\m_ARQ016.exe	2.3	2009/11/19 20:22:28	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:09:57	10905	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ017\Hardware\m_ARQ017.exe	2.2	2009/09/30 15:01:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:08:02	8256	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ018\Hardware\m_ARQ018.exe	2.7	2010/06/24 12:34:26	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 14:04:51	11455	TELM-06	1	Pass	0	0	0	196
M_TCP_Data	m_ARQ019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ019\Hardware\m_ARQ019.exe	2.4	2010/06/08 16:07:58	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:02:30	10396	TELM-06	1	Pass	0	0	0	68
M_TCP_Data	m_ARQ020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ020\Hardware\m_ARQ020.exe	2.2	2009/11/18 15:23:22	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:57:06	7572	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ021\Hardware\m_ARQ021.exe	2.2	2009/11/19 18:43:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:53:00	7481	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ022\Hardware\m_ARQ022.exe	2.3	2010/02/03 16:30:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:50:10	9343	TELM-06	1	Pass	0	0	0	137
M_TCP_Data	m_ARQ023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ023\Hardware\m_ARQ023.exe	2.1	2009/11/10 19:12:36	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:48:06	11415	TELM-06	1	Pass	0	0	0	81
M_TCP_Data	m_ARQ024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ024\Hardware\m_ARQ024.exe	2.3	2010/02/08 18:44:31	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:46:02	12233	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ025\Hardware\m_ARQ025.exe	2.1	2009/09/25 14:20:39	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:43:14	9482	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ026\Hardware\m_ARQ026.exe	2.1	2009/09/25 14:21:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:40:36	8981	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_ARQ027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ027\Hardware\m_ARQ027.exe	2.1	2009/09/25 14:21:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:37:43	9567	TELM-06	1	Pass	0	0	0	6

M_TCP_Data	m_ARQ028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ028\Hardware\m_ARQ028.exe	2.4	2010/04/19 14:36:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:33:41	18735	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_ARQ029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ029\Hardware\m_ARQ029.exe	2.1	2010/03/08 19:53:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:29:40	29507	TELM-06	1	Pass	0	0	0	555
M_TCP_Data	m_ARQ030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ030\Hardware\m_ARQ030.exe	2.5	2010/06/03 15:11:29	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:25:35	21523	TELM-06	1	Pass	0	0	0	355
M_TCP_Data	m_ARQ031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ031\Hardware\m_ARQ031.exe	2.3	2010/06/08 19:57:52	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:23:00	14707	TELM-06	1	Pass	0	0	0	556
M_TCP_Data	m_Data012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data012\Hardware\m_Data012.exe	2.4	2009/11/10 14:18:10	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 14:47:05	37316	TELM-06	1	Pass	0	0	0	18
M_TCP_Data	m_Data013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data013\Hardware\m_Data013.exe	2.3	2010/02/03 15:41:32	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:19:09	106600	TELM-06	1	Pass	0	0	0	335
M_TCP_Data	m_Data014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data014\Hardware\m_Data014.exe	2.3	2010/05/12 19:37:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:15:23	98782	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_Data015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data015\Hardware\m_Data015.exe	2.2	2010/02/03 17:34:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:13:06	10720	TELM-06	1	Pass	0	0	0	14
M_TCP_Data	m_Data016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data016\Hardware\m_Data016.exe	2.2	2010/02/24 20:56:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:09:31	70004	TELM-06	1	Pass	0	0	0	26
M_TCP_Data	m_Data017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data017\Hardware\m_Data017.exe	2.2	2010/02/24 22:22:41	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:06:35	67714	TELM-06	1	Pass	0	0	0	30
M_TCP_Data	m_Data018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data018\Hardware\m_Data018.exe	2.2	2010/02/03 22:48:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:01:32	59016	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_MemoryRead001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead001\Hardware\m_MemoryRead001.exe	2.2	2010/02/01 17:26:47	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:58:14	32348	TELM-06	1	Pass	0	0	0	439
M_TCP_Data	m_MemoryRead002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead002\Hardware\m_MemoryRead002.exe	2.3	2009/11/05 15:03:23	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:55:14	20661	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead003\Hardware\m_MemoryRead003.exe	2.4	2010/04/05 13:08:02	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:53:14	15603	TELM-06	1	Pass	0	0	0	320
M_TCP_Data	m_MemoryRead004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead004\Hardware\m_MemoryRead004.exe	2.3	2010/03/19 16:10:47	2.3	Up-to-date	Baseline 17_1	None	09/13/2010 07:34:02	9582	TELM-06	1	Pass	0	0	0	1
M_TCP_Data	m_MemoryRead005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead005\Hardware\m_MemoryRead005.exe	2.3	2009/12/07 19:50:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:48:00	48716	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryRead006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead006\Hardware\m_MemoryRead006.exe	2.6	2010/06/29 21:05:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:46:02	12473	TELM-06	1	Pass	0	0	0	37
M_TCP_Data	m_MemoryRead007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead007\Hardware\m_MemoryRead007.exe	2.1	2009/11/03 12:42:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:43:08	11022	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_MemoryRead008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead008\Hardware\m_MemoryRead008.exe	2.1	2009/09/30 18:09:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:40:49	8910	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead009\Hardware\m_MemoryRead009.exe	2.1	2009/12/07 19:51:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:38:29	13780	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead010\Hardware\m_MemoryRead010.exe	2.2	2009/09/30 17:31:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:34:48	7941	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead011\Hardware\m_MemoryRead011.exe	2.2	2009/09/30 17:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:31:06	7911	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead012\Hardware\m_MemoryRead012.exe	2.2	2009/09/30 17:33:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:28:14	8074	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead013\Hardware\m_MemoryRead013.exe	2.2	2009/09/30 17:33:27	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:27:02	8105	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead014\Hardware\m_MemoryRead014.exe	2.2	2009/09/30 17:33:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:25:35	7941	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead015\Hardware\m_MemoryRead015.exe	2.2	2009/09/30 17:34:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:24:14	8015	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead016\Hardware\m_MemoryRead016.exe	2.1	2009/10/19 18:24:00	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:20:17	7922	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead017\Hardware\m_MemoryRead017.exe	2.2	2009/10/19 15:52:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:19:01	8131	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead019\Hardware\m_MemoryRead019.exe	2.3	2009/12/03 13:57:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:16:26	9871	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead020\Hardware\m_MemoryRead020.exe	2.4	2010/02/09 20:49:06	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:15:02	7354	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryRead021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead021\Hardware\m_MemoryRead021.exe	2.2	2009/12/23 13:34:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:11:13	10199	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryRead030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead030\Hardware\m_MemoryRead030.exe	2.1	2009/10/16 13:54:03	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:08:56	6881	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead031\Hardware\m_MemoryRead031.exe	2.3	2009/12/28 18:23:34	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:06:37	11182	TELM-06	1	Pass	0	0	0	43
M_TCP_Data	m_MemoryRead033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead033\Hardware\m_MemoryRead033.exe	2.3	2009/11/23 18:55:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:04:45	12199	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead034\Hardware\m_MemoryRead034.exe	2.1	2009/10/16 14:15:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:02:30	8558	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead035\Hardware\m_MemoryRead035.exe	2.4	2009/11/23 18:56:23	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:57:02	18974	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead037\Hardware\m_MemoryRead037.exe	2.2	2009/11/23 18:57:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:53:41	12427	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead038\Hardware\m_MemoryRead038.exe	2.3	2010/01/19 17:36:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:50:59	14452	TELM-06	1	Pass	0	0	0	294
M_TCP_Data	m_MemoryRead039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead039\Hardware\m_MemoryRead039.exe	2.1	2009/09/28 15:09:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:46:59	8749	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead040\Hardware\m_MemoryRead040.exe	2.6	2010/01/06 14:10:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 11:44:38	32695	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryRead041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead041\Hardware\m_MemoryRead041.exe	2.4	2009/11/16 12:52:24	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:43:02	18862	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead042\Hardware\m_MemoryRead042.exe	2.3	2010/06/29 22:24:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:41:22	30996	TELM-06	1	Pass	0	0	0	41
M_TCP_Data	m_MemoryRead043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead043\Hardware\m_MemoryRead043.exe	2.2	2010/02/08 16:55:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:39:08	18068	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead045\Hardware\m_MemoryRead045.exe	2.0	2009/09/22 14:34:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:36:16	6873	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead049\Hardware\m_MemoryRead049.exe	2.3	2009/11/16 12:52:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:33:18	8901	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead051\Hardware\m_MemoryRead051.exe	2.1	2009/10/27 19:22:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:30:48	9867	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryRead052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead052\Hardware\m_MemoryRead052.exe	2.3	2010/06/29 23:18:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:29:14	11437	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite001\Hardware\m_MemoryWrite001.exe	2.2	2009/12/07 19:52:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:27:30	16048	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite002\Hardware\m_MemoryWrite002.exe	2.0	2009/09/22 14:48:30	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:26:39	10527	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite003\Hardware\m_MemoryWrite003.exe	2.1	2009/10/28 14:34:41	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:25:00	10064	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite005\Hardware\m_MemoryWrite005.exe	2.2	2009/12/15 19:05:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:23:37	8047	TELM-06	1	Pass	0	0	0	186
M_TCP_Data	m_MemoryWrite006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite006\Hardware\m_MemoryWrite006.exe	2.3	2009/11/24 17:33:37	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:21:51	13793	TELM-06	1	Pass	0	0	0	145
M_TCP_Data	m_MemoryWrite007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite007\Hardware\m_MemoryWrite007.exe	2.1	2009/10/28 15:09:57	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:20:16	8108	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite008\Hardware\m_MemoryWrite008.exe	2.1	2010/02/01 19:11:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:18:52	7940	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite009\Hardware\m_MemoryWrite009.exe	2.1	2009/10/28 15:17:22	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:17:49	7550	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite010\Hardware\m_MemoryWrite010.exe	2.1	2009/10/28 16:00:35	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:15:51	7634	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite011\Hardware\m_MemoryWrite011.exe	2.1	2009/10/28 16:07:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:12:47	7625	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite012\Hardware\m_MemoryWrite012.exe	2.2	2009/10/28 16:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:10:25	12733	TELM-06	1	Pass	0	0	0	120
M_TCP_Data	m_MemoryWrite013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite013\Hardware\m_MemoryWrite013.exe	2.1	2009/10/28 16:21:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:06:34	7629	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite014\Hardware\m_MemoryWrite014.exe	2.1	2009/10/28 16:29:59	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:02:22	7634	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite015\Hardware\m_MemoryWrite015.exe	2.1	2009/10/28 16:36:30	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:57:45	7619	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryWrite016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite016\Hardware\m_MemoryWrite016.exe	2.5	2010/02/01 21:11:41	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:54:35	9189	TELM-06	1	Pass	0	0	0	862
M_TCP_Data	m_MemoryWrite017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite017\Hardware\m_MemoryWrite017.exe	2.3	2010/01/19 17:37:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:51:07	8706	TELM-06	1	Pass	0	0	0	433
M_TCP_Data	m_MemoryWrite018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite018\Hardware\m_MemoryWrite018.exe	2.1	2009/10/29 20:06:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:45:03	6883	TELM-06	1	Pass	0	0	0	171
M_TCP_Data	m_MemoryWrite019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite019\Hardware\m_MemoryWrite019.exe	2.0	2009/09/22 15:13:38	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 10:40:05	9836	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryWrite020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite020\Hardware\m_MemoryWrite020.exe	2.2	2009/10/29 19:44:00	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:37:04	8079	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite022\Hardware\m_MemoryWrite022.exe	2.3	2009/11/03 17:28:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:35:00	10060	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite023\Hardware\m_MemoryWrite023.exe	2.4	2010/02/04 16:15:17	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:32:08	20513	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite024\Hardware\m_MemoryWrite024.exe	2.2	2009/11/24 17:35:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:30:49	7686	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_MemoryWrite025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite025\Hardware\m_MemoryWrite025.exe	2.1	2009/10/29 18:47:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:28:12	10957	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite029\Hardware\m_MemoryWrite029.exe	2.2	2010/02/01 21:00:35	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:26:34	14031	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite032\Hardware\m_MemoryWrite032.exe	2.5	2010/03/15 01:08:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:21:20	9770	TELM-06	1	Pass	0	0	0	146
M_TCP_Data	m_MemoryWrite038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite038\Hardware\m_MemoryWrite038.exe	2.1	2009/10/29 18:42:13	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:17:03	9502	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite039\Hardware\m_MemoryWrite039.exe	2.1	2010/02/10 18:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:13:03	10034	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite040\Hardware\m_MemoryWrite040.exe	2.2	2010/02/10 17:56:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:10:11	10407	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryWrite041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite041\Hardware\m_MemoryWrite041.exe	2.2	2010/02/01 20:45:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:07:21	14074	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite042\Hardware\m_MemoryWrite042.exe	2.2	2010/01/19 17:38:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:04:03	10442	TELM-06	1	Pass	0	0	0	277
M_TCP_Data	m_MemoryWrite046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite046\Hardware\m_MemoryWrite046.exe	2.3	2010/02/04 16:51:09	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:00:26	7499	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite047\Hardware\m_MemoryWrite047.exe	2.2	2009/11/11 19:12:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:55:58	9155	TELM-06	1	Pass	0	0	0	98
M_TCP_Data	m_MemoryWrite048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite048\Hardware\m_MemoryWrite048.exe	2.2	2009/11/11 18:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:53:24	9963	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite049\Hardware\m_MemoryWrite049.exe	2.5	2010/02/10 16:51:26	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 09:51:33	26715	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_MemoryWrite052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite052\Hardware\m_MemoryWrite052.exe	2.3	2009/12/23 13:35:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:46:58	11178	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite053\Hardware\m_MemoryWrite053.exe	2.2	2009/11/11 17:01:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:46:05	8621	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite054\Hardware\m_MemoryWrite054.exe	2.6	2010/06/30 17:58:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 09:44:38	11776	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryWrite055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite055\Hardware\m_MemoryWrite055.exe	2.2	2009/11/11 18:04:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:41:52	8222	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite056	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite056\Hardware\m_MemoryWrite056.exe	2.4	2010/02/10 16:32:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 09:40:21	9075	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode001\Hardware\m_TransferMode001.exe	2.8	2010/05/14 18:48:57	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 09:39:24	25076	TELM-06	1	Pass	0	0	0	4748
M_TCP_Data	m_TransferMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode002\Hardware\m_TransferMode002.exe	2.4	2010/07/02 13:00:52	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 09:38:46	10095	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_TransferMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode003\Hardware\m_TransferMode003.exe	2.0	2009/11/16 20:59:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 09:36:22	7807	TELM-06	1	Pass	0	0	0	129
M_TCP_Data	m_TransferMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode004\Hardware\m_TransferMode004.exe	2.3	2010/01/08 17:07:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:35:10	12750	TELM-06	1	Pass	0	0	0	178
M_TCP_Data	m_TransferMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode005\Hardware\m_TransferMode005.exe	2.7	2010/07/02 13:36:13	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 09:34:23	36289	TELM-06	1	Pass	0	0	0	210
M_TCP_Data	m_TransferMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode006\Hardware\m_TransferMode006.exe	2.4	2010/04/16 11:58:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 09:34:02	7601	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode007\Hardware\m_TransferMode007.exe	2.2	2009/10/12 15:46:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:33:48	12067	TELM-06	1	Pass	0	0	0	112
M_TCP_Data	m_TransferMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode008\Hardware\m_TransferMode008.exe	2.0	2009/09/09 15:28:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 09:33:36	10944	TELM-06	1	Pass	0	0	0	76
M_TCP_Data	m_TransferMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode009\Hardware\m_TransferMode009.exe	2.1	2009/10/13 16:25:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:33:22	11733	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_TransferMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode010\Hardware\m_TransferMode010.exe	2.1	2009/10/13 16:26:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:33:08	11632	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode011\Hardware\m_TransferMode011.exe	2.3	2010/02/10 13:37:08	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:32:57	8055	TELM-06	1	Pass	0	0	0	72
M_TCP_Data	m_TransferMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode012\Hardware\m_TransferMode012.exe	2.3	2010/02/10 13:38:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:32:47	8075	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_TransferMode013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode013\Hardware\m_TransferMode013.exe	2.3	2010/02/10 13:39:16	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:32:35	8054	TELM-06	1	Pass	0	0	0	44
M_TCP_Data	m_TransferMode014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode014\Hardware\m_TransferMode014.exe	2.3	2010/02/10 13:40:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:32:25	8080	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_TransferMode015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode015\Hardware\m_TransferMode015.exe	2.5	2010/05/26 14:56:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 09:32:16	7397	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_TransferMode016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode016\Hardware\m_TransferMode016.exe	2.8	2010/07/02 13:14:11	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 09:31:44	29948	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode017\Hardware\m_TransferMode017.exe	2.7	2010/05/18 14:47:09	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 09:30:54	48808	TELM-06	1	Pass	0	0	0	257
M_TCP_Data	m_TransferMode018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode018\Hardware\m_TransferMode018.exe	2.6	2010/04/16 14:06:26	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 09:30:25	26519	TELM-06	1	Pass	0	0	0	509
M_TCP_Data	m_TransferMode019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode019\Hardware\m_TransferMode019.exe	2.6	2010/03/19 18:58:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 09:29:57	22587	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics001\Hardware\m_Diagnostics001.exe	2.5	2009/12/30 17:45:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:38:15	9634	TELM-06	1	Pass	0	0	0	166
M_TCP_Diagnostic	m_Diagnostics002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics002\Hardware\m_Diagnostics002.exe	2.2	2009/11/23 20:15:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:33:01	7781	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics003\Hardware\m_Diagnostics003.exe	2.2	2010/03/14 20:01:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:32:06	6828	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics004\Hardware\m_Diagnostics004.exe	2.3	2009/12/03 13:22:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:28:28	15913	TELM-06	1	Pass	0	0	0	36
M_TCP_Diagnostic	m_Diagnostics005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics005\Hardware\m_Diagnostics005.exe	2.1	2009/10/12 14:14:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:25:37	7769	TELM-06	1	Pass	0	0	0	13
M_TCP_Diagnostic	m_Diagnostics006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics006\Hardware\m_Diagnostics006.exe	2.0	2009/09/22 17:02:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:24:32	14763	TELM-06	1	Pass	0	0	0	14

M_TCP_Diagnostic	m_Diagnostics007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics007\Hardware\m_Diagnostics007.exe	2.1	2009/09/28 13:31:55	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:23:37	15389	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics008\Hardware\m_Diagnostics008.exe	2.0	2009/09/22 17:03:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:21:54	21647	TELM-06	1	Pass	0	0	0	31
M_TCP_Diagnostic	m_Diagnostics009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics009\Hardware\m_Diagnostics009.exe	2.5	2010/06/25 17:09:14	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:20:29	16588	TELM-06	1	Pass	0	0	0	23
M_TCP_Diagnostic	m_Diagnostics010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics010\Hardware\m_Diagnostics010.exe	2.5	2010/07/20 19:18:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:19:25	10570	TELM-06	1	Pass	0	0	0	10
M_TCP_Diagnostic	m_Diagnostics011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics011\Hardware\m_Diagnostics011.exe	2.4	2009/12/23 13:29:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:17:57	29977	TELM-06	1	Pass	0	0	0	19
M_TCP_Diagnostic	m_Diagnostics012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics012\Hardware\m_Diagnostics012.exe	2.3	2010/04/16 14:20:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:16:17	17377	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics013\Hardware\m_Diagnostics013.exe	2.5	2010/06/08 17:29:50	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:14:29	20504	TELM-06	1	Pass	0	0	0	15
M_TCP_Diagnostic	m_Diagnostics014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics014\Hardware\m_Diagnostics014.exe	2.4	2010/03/14 17:27:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:12:23	18029	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics015\Hardware\m_Diagnostics015.exe	2.4	2010/01/19 17:35:47	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:09:23	25471	TELM-06	1	Pass	0	0	0	29
M_TCP_Diagnostic	m_Diagnostics016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics016\Hardware\m_Diagnostics016.exe	2.0	2009/09/22 17:09:46	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:08:07	5908	TELM-06	1	Pass	0	0	0	1
M_TCP_Diagnostic	m_Diagnostics017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics017\Hardware\m_Diagnostics017.exe	2.3	2009/11/23 20:15:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:06:07	20043	TELM-06	1	Pass	0	0	0	32
M_TCP_Diagnostic	m_Diagnostics018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics018\Hardware\m_Diagnostics018.exe	2.2	2009/12/23 13:30:36	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:04:26	11841	TELM-06	1	Pass	0	0	0	9
M_TCP_Diagnostic	m_Diagnostics019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics019\Hardware\m_Diagnostics019.exe	2.1	2009/10/13 12:56:53	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:02:42	8012	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics020\Hardware\m_Diagnostics020.exe	2.2	2009/11/23 20:16:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:00:38	10296	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics021\Hardware\m_Diagnostics021.exe	2.1	2009/10/13 13:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:59:10	6038	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics022\Hardware\m_Diagnostics022.exe	2.4	2010/02/03 17:09:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:58:05	12645	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics023\Hardware\m_Diagnostics023.exe	2.2	2010/02/03 14:24:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:54:40	39133	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics024\Hardware\m_Diagnostics024.exe	2.5	2010/02/03 14:04:47	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:51:32	15934	TELM-06	1	Pass	0	0	0	20
M_TCP_Diagnostic	m_Diagnostics026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics026\Hardware\m_Diagnostics026.exe	2.1	2009/10/15 19:32:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:50:02	6055	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics027\Hardware\m_Diagnostics027.exe	2.2	2009/11/02 13:03:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:47:40	2915	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics028\Hardware\m_Diagnostics028.exe	2.4	2009/11/16 20:37:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:42:32	11487	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics029\Hardware\m_Diagnostics029.exe	2.1	2009/10/15 19:46:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:40:20	2760	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics030\Hardware\m_Diagnostics030.exe	2.5	2010/07/20 18:09:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:36:44	7560	TELM-06	1	Pass	0	0	0	277
M_TCP_Diagnostic	m_Diagnostics031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics031\Hardware\m_Diagnostics031.exe	2.2	2009/11/11 14:33:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:34:32	10852	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics032\Hardware\m_Diagnostics032.exe	2.3	2009/12/23 13:31:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:27:02	9931	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics034\Hardware\m_Diagnostics034.exe	2.4	2009/11/16 20:30:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:23:35	9042	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics035\Hardware\m_Diagnostics035.exe	2.2	2009/12/23 13:31:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:20:59	8071	TELM-06	1	Pass	0	0	0	7
M_TCP_Diagnostic	m_Diagnostics036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics036\Hardware\m_Diagnostics036.exe	2.3	2010/03/13 01:44:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:17:46	8541	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics037\Hardware\m_Diagnostics037.exe	2.4	2010/03/13 01:23:35	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:11:20	7198	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics038\Hardware\m_Diagnostics038.exe	2.3	2010/06/07 20:16:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:05:27	49678	TELM-06	1	Pass	0	0	0	18
M_TCP_Discover	m_DiscoverFull001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull001\Hardware\m_DiscoverFull001.exe	2.8	2010/05/21 19:22:39	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 00:48:23	44695	TELM-06	1	Pass	0	0	0	150
M_TCP_Discover	m_DiscoverFull002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull002\Hardware\m_DiscoverFull002.exe	2.6	2010/03/19 18:02:47	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 00:44:19	144593	TELM-06	1	Pass	0	0	0	1815
M_TCP_Discover	m_DiscoverFull003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull003\Hardware\m_DiscoverFull003.exe	2.2	2010/02/25 20:00:01	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:39:26	12272	TELM-06	1	Pass	0	0	0	58
M_TCP_Discover	m_DiscoverFull004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull004\Hardware\m_DiscoverFull004.exe	2.4	2010/02/05 21:19:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 07:16:41	271106	TELM-06	1	Pass	0	0	0	3713
M_TCP_Discover	m_DiscoverFull005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull005\Hardware\m_DiscoverFull005.exe	2.2	2009/10/29 19:18:52	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:36:48	17488	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull006\Hardware\m_DiscoverFull006.exe	2.3	2010/02/05 21:19:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:27:44	130877	TELM-06	1	Pass	0	0	0	1727
M_TCP_Discover	m_DiscoverFull007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull007\Hardware\m_DiscoverFull007.exe	2.7	2009/12/14 21:43:07	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 00:32:34	13679	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull009\Hardware\m_DiscoverFull009.exe	2.3	2010/01/21 21:02:21	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:29:03	10978	TELM-06	1	Pass	0	0	0	65
M_TCP_Discover	m_DiscoverFull010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull010\Hardware\m_DiscoverFull010.exe	2.4	2010/05/12 17:07:31	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:22:45	7970	TELM-06	1	Pass	0	0	0	121
M_TCP_Discover	m_DiscoverFull011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull011\Hardware\m_DiscoverFull011.exe	2.3	2009/11/02 22:03:01	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:19:35	56827	TELM-06	1	Pass	0	0	0	9
M_TCP_Discover	m_DiscoverFull012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull012\Hardware\m_DiscoverFull012.exe	2.8	2010/06/29 12:07:18	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 08:18:06	366921	TELM-06	1	Pass	0	0	0	33
M_TCP_Discover	m_DiscoverFull013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull013\Hardware\m_DiscoverFull013.exe	2.2	2010/03/18 16:03:09	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:16:32	42648	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull014\Hardware\m_DiscoverFull014.exe	2.2	2010/03/18 16:03:45	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:13:01	25044	TELM-06	1	Pass	0	0	0	15
M_TCP_Discover	m_DiscoverFull015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull015\Hardware\m_DiscoverFull015.exe	2.1	2009/11/23 16:09:49	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:09:29	8083	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull016\Hardware\m_DiscoverFull016.exe	2.1	2009/11/03 20:58:07	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:06:55	8311	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull017\Hardware\m_DiscoverFull017.exe	2.1	2009/11/03 20:58:47	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:04:53	12151	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull022\Hardware\m_DiscoverFull022.exe	2.1	2009/11/23 16:11:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:58:36	8439	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull023\Hardware\m_DiscoverFull023.exe	2.1	2009/11/23 16:11:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:57:41	11502	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull024\Hardware\m_DiscoverFull024.exe	2.1	2009/11/23 16:12:04	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:54:11	12672	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull025\Hardware\m_DiscoverFull025.exe	2.1	2009/11/23 16:12:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:51:30	9031	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull029\Hardware\m_DiscoverFull029.exe	2.0	2009/11/13 21:27:59	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:49:14	12488	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull030\Hardware\m_DiscoverFull030.exe	2.3	2010/05/28 11:39:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:46:50	19313	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull031\Hardware\m_DiscoverFull031.exe	2.3	2010/05/28 12:38:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:43:35	35315	TELM-06	1	Pass	0	0	0	22
M_TCP_Discover	m_DiscoverFull032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull032\Hardware\m_DiscoverFull032.exe	2.0	2009/11/13 21:29:43	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:41:07	9026	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull033\Hardware\m_DiscoverFull033.exe	2.0	2009/11/13 21:30:18	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:38:47	9460	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull034\Hardware\m_DiscoverFull034.exe	2.2	2010/03/18 16:04:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:35:50	11069	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull035\Hardware\m_DiscoverFull035.exe	2.2	2010/02/09 20:53:53	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 13:46:16	43591	TELM-06	1	Pass	0	0	0	31
M_TCP_Discover	m_DiscoverFull036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull036\Hardware\m_DiscoverFull036.exe	2.2	2010/02/11 14:31:48	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 13:15:17	22186	TELM-06	1	Pass	0	0	0	38
M_TCP_Discover	m_DiscoverFull037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull037\Hardware\m_DiscoverFull037.exe	2.4	2010/02/03 21:18:36	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 18:19:21	35116	TELM-06	1	Pass	0	0	0	4
M_TCP_Discover	m_DiscoverFull038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull038\Hardware\m_DiscoverFull038.exe	2.1	2009/12/01 20:49:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:29:53	141922	TELM-06	1	Pass	0	0	0	7
M_TCP_EEPROM	m_Eeprom001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom001\Hardware\m_Eeprom001.exe	2.3	2009/12/23 16:23:59	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 09:59:40	11078	TELM-06	1	Pass	0	0	0	22
M_TCP_EEPROM	m_Eeprom002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom002\Hardware\m_Eeprom002.exe	2.3	2009/12/18 21:07:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:16:18	10839	TELM-06	1	Pass	0	0	0	938
M_TCP_EEPROM	m_Eeprom003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom003\Hardware\m_Eeprom003.exe	2.2	2010/01/05 18:03:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:50:10	16824	TELM-06	1	Pass	0	0	0	702
M_TCP_EEPROM	m_Eeprom004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom004\Hardware\m_Eeprom004.exe	2.2	2010/01/25 19:26:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:44:41	117190	TELM-06	1	Pass	0	0	0	583
M_TCP_EEPROM	m_Eeprom005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom005\Hardware\m_Eeprom005.exe	2.2	2010/01/19 15:32:05	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:40:33	177674	TELM-06	1	Pass	0	0	0	704
M_TCP_EEPROM	m_Eeprom006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom006\Hardware\m_Eeprom006.exe	2.2	2010/05/28 12:55:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:37:13	47881	TELM-06	1	Pass	0	0	0	6
M_TCP_EEPROM	m_Eeprom007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom007\Hardware\m_Eeprom007.exe	2.1	2010/02/18 18:38:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:35:43	5122	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency001\Hardware\m_Emergency001.exe	2.3	2009/11/03 21:54:33	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:46:29	9717	TELM-06	1	Pass	0	0	0	100
M_TCP_Emergency	m_Emergency002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency002\Hardware\m_Emergency002.exe	2.5	2010/06/25 20:19:36	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:41:26	14734	TELM-06	1	Pass	0	0	0	45
M_TCP_Emergency	m_Emergency003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency003\Hardware\m_Emergency003.exe	2.6	2010/06/29 14:19:35	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:37:58	26846	TELM-06	1	Pass	0	0	0	724
M_TCP_Emergency	m_Emergency004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency004\Hardware\m_Emergency004.exe	2.3	2010/02/05 21:22:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:33:39	30339	TELM-06	1	Pass	0	0	0	4
M_TCP_Emergency	m_Emergency005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency005\Hardware\m_Emergency005.exe	2.2	2009/11/10 00:31:59	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:30:20	8673	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency006\Hardware\m_Emergency006.exe	2.1	2009/10/16 16:44:07	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:27:09	9456	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency007\Hardware\m_Emergency007.exe	2.1	2009/10/16 16:44:28	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:21:04	9630	TELM-06	1	Pass	0	0	0	33
M_TCP_Emergency	m_Emergency009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency009\Hardware\m_Emergency009.exe	2.1	2009/10/08 19:54:42	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:17:01	6158	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency010\Hardware\m_Emergency010.exe	2.2	2010/06/08 15:56:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:12:56	13242	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency011\Hardware\m_Emergency011.exe	2.0	2009/10/21 16:25:23	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:09:58	6747	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency012\Hardware\m_Emergency012.exe	2.1	2009/11/11 13:23:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:08:00	9061	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency013\Hardware\m_Emergency013.exe	2.1	2009/11/11 14:02:24	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:04:51	14133	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency014\Hardware\m_Emergency014.exe	2.3	2009/12/23 13:33:36	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:02:15	8816	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency015\Hardware\m_Emergency015.exe	2.3	2010/06/08 15:56:40	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:59:31	9060	TELM-06	1	Pass	0	0	0	8
M_TCP_Emergency	m_Emergency016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency016\Hardware\m_Emergency016.exe	2.2	2009/12/21 21:03:44	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:57:43	15365	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency017\Hardware\m_Emergency017.exe	2.2	2009/11/16 12:31:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:54:26	11052	TELM-06	1	Pass	0	0	0	21
M_TCP_Emergency	m_Emergency018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency018\Hardware\m_Emergency018.exe	2.3	2010/06/29 15:32:40	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:50:13	8257	TELM-06	1	Pass	0	0	0	22
M_TCP_Emergency	m_Emergency019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency019\Hardware\m_Emergency019.exe	2.2	2010/01/07 19:28:32	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:48:15	11388	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency020\Hardware\m_Emergency020.exe	2.6	2010/03/18 17:47:12	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:47:02	13401	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency021\Hardware\m_Emergency021.exe	2.3	2009/12/21 19:27:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:45:14	11031	TELM-06	1	Pass	0	0	0	83
M_TCP_Emergency	m_Emergency022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency022\Hardware\m_Emergency022.exe	2.2	2010/01/07 18:54:43	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:41:51	11388	TELM-06	1	Pass	0	0	0	82
M_TCP_Emergency	m_Emergency038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency038\Hardware\m_Emergency038.exe	2.2	2010/01/07 19:42:11	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:37:22	10553	TELM-06	1	Pass	0	0	0	14
M_TCP_Emergency	m_Emergency039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency039\Hardware\m_Emergency039.exe	2.2	2010/02/03 17:51:34	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:35:12	15346	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency040\Hardware\m_Emergency040.exe	2.0	2009/09/29 14:53:27	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 05:31:07	14335	TELM-06	1	Pass	0	0	0	7
M_TCP_Emergency	m_Emergency041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency041\Hardware\m_Emergency041.exe	2.0	2009/09/29 14:54:05	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 05:29:19	4918	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency042\Hardware\m_Emergency042.exe	2.0	2009/09/29 14:54:52	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 05:28:15	4937	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency043\Hardware\m_Emergency043.exe	2.4	2010/04/21 17:13:30	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:24:18	10405	TELM-06	1	Pass	0	0	0	9
M_TCP_Emergency	m_Emergency044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency044\Hardware\m_Emergency044.exe	2.5	2009/11/10 14:19:43	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 19:41:36	37332	TELM-06	1	Pass	0	0	0	18
M_TCP_Emergency	m_Emergency045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency045\Hardware\m_Emergency045.exe	2.3	2010/05/11 12:18:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:18:18	11418	TELM-06	1	Pass	0	0	0	16
M_TCP_General	m_General003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General003\Hardware\m_General003.exe	2.2	2009/10/13 14:50:43	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:13:45	67484	TELM-06	1	Pass	0	0	0	44576
M_TCP_General	m_General005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General005\Hardware\m_General005.exe	2.4	2009/12/01 19:02:30	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:07:52	67016	TELM-06	1	Pass	0	0	0	14133
M_TCP_General	m_General015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General015\Hardware\m_General015.exe	2.3	2009/12/01 19:02:59	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:03:13	34855	TELM-06	1	Pass	0	0	0	36747
M_TCP_General	m_General016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General016\Hardware\m_General016.exe	2.4	2010/03/29 10:43:22	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:00:41	8489	TELM-06	1	Pass	0	0	0	2
M_TCP_General	m_General017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General017\Hardware\m_General017.exe	2.3	2009/12/01 19:03:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:55:47	7495	TELM-06	1	Pass	0	0	0	10
M_TCP_General	m_General018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General018\Hardware\m_General018.exe	2.2	2010/02/03 18:07:07	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:53:27	24774	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General027\Hardware\m_General027.exe	2.7	2010/06/07 11:39:47	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 11:18:05	802373	TELM-06	1	Pass	0	0	0	21
M_TCP_General	m_General028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General028\Hardware\m_General028.exe	2.5	2010/06/08 11:48:52	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 19:03:55	635895	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General029\Hardware\m_General029.exe	2.3	2010/06/07 12:27:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 18:30:33	274246	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General030\Hardware\m_General030.exe	2.1	2010/06/07 12:30:50	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:51:47	17216	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General031\Hardware\m_General031.exe	2.5	2010/05/11 12:33:53	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:49:09	58474	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General032\Hardware\m_General032.exe	2.2	2010/02/05 18:56:06	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:46:41	29598	TELM-06	1	Pass	0	0	0	11
M_TCP_General	m_General033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General033\Hardware\m_General033.exe	2.0	2010/02/04 17:42:57	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 04:44:53	20719	TELM-06	1	Pass	0	0	0	4
M_TCP_General	m_General034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General034\Hardware\m_General034.exe	2.0	2010/01/22 22:45:34	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 04:41:47	30142	TELM-06	1	Pass	0	0	0	2
M_TCP_Initialize	m_Initialize001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize001\Hardware\m_Initialize001.exe	2.8	2010/05/12 12:34:35	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 23:25:06	10560	TELM-06	1	Pass	0	0	0	9

M_TCP_Initialize	m_Initialize002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize002\Hardware\m_Initialize002.exe	2.2	2009/10/20 12:16:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:22:26	19900	TELM-06	1	Pass	0	0	0	11
M_TCP_Initialize	m_Initialize003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize003\Hardware\m_Initialize003.exe	2.4	2010/06/08 15:57:16	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:18:46	23935	TELM-06	1	Pass	0	0	0	6
M_TCP_Initialize	m_Initialize004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize004\Hardware\m_Initialize004.exe	2.4	2010/06/08 15:57:44	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:16:47	34919	TELM-06	1	Pass	0	0	0	30
M_TCP_MedicalEvent	m_MedicalEvent001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent001\Hardware\m_MedicalEvent001.exe	2.4	2010/01/18 15:42:53	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:03:01	12606	TELM-06	1	Pass	0	0	0	36
M_TCP_MedicalEvent	m_MedicalEvent002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent002\Hardware\m_MedicalEvent002.exe	2.6	2010/02/09 19:03:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:59:39	73234	TELM-06	1	Pass	0	0	0	23
M_TCP_MedicalEvent	m_MedicalEvent003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent003\Hardware\m_MedicalEvent003.exe	2.4	2010/03/11 21:16:02	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:58:13	18885	TELM-06	1	Pass	0	0	0	24
M_TCP_MedicalEvent	m_MedicalEvent004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent004\Hardware\m_MedicalEvent004.exe	2.4	2010/01/18 15:44:34	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:56:33	12047	TELM-06	1	Pass	0	0	0	12
M_TCP_MedicalEvent	m_MedicalEvent008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent008\Hardware\m_MedicalEvent008.exe	2.4	2010/01/18 15:44:59	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:55:22	8462	TELM-06	1	Pass	0	0	0	6
M_TCP_MedicalEvent	m_MedicalEvent009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent009\Hardware\m_MedicalEvent009.exe	2.2	2010/01/18 15:46:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:54:03	7659	TELM-06	1	Pass	0	0	0	4
M_TCP_MedicalEvent	m_MedicalEvent010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent010\Hardware\m_MedicalEvent010.exe	2.5	2010/01/27 14:30:44	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:52:09	16656	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent011\Hardware\m_MedicalEvent011.exe	2.2	2010/01/18 15:47:08	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:50:39	15242	TELM-06	1	Pass	0	0	0	11
M_TCP_MedicalEvent	m_MedicalEvent012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent012\Hardware\m_MedicalEvent012.exe	2.3	2010/01/18 15:47:55	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:49:36	6024	TELM-06	1	Pass	0	0	0	3
M_TCP_MedicalEvent	m_MedicalEvent014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent014\Hardware\m_MedicalEvent014.exe	2.5	2010/09/08 13:20:31	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:48:18	15285	TELM-06	1	Pass	0	0	0	18
M_TCP_MedicalEvent	m_MedicalEvent015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent015\Hardware\m_MedicalEvent015.exe	2.9	2010/09/07 20:16:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 09:26:45	56509	TELM-06	1	Pass	0	0	0	152
M_TCP_MedicalEvent	m_MedicalEvent016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent016\Hardware\m_MedicalEvent016.exe	2.9	2010/05/12 18:44:37	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 07:30:50	283136	TELM-06	1	Pass	0	0	0	374
M_TCP_MedicalEvent	m_MedicalEvent017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent017\Hardware\m_MedicalEvent017.exe	2.8	2010/09/07 20:17:30	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 09:16:34	612275	TELM-06	1	Pass	0	0	0	13
M_TCP_MedicalEvent	m_MedicalEvent018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent018\Hardware\m_MedicalEvent018.exe	2.5	2010/01/18 15:50:21	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 07:23:41	272231	TELM-06	1	Pass	0	0	0	4627
M_TCP_MedicalEvent	m_MedicalEvent019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent019\Hardware\m_MedicalEvent019.exe	2.5	2010/01/18 15:50:59	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:45:48	7358	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent020\Hardware\m_MedicalEvent020.exe	2.3	2010/01/18 15:51:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:44:23	40580	TELM-06	1	Pass	0	0	0	9
M_TCP_MemoryTest	m_MemoryTest001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest001\Hardware\m_MemoryTest001.exe	2.7	2010/06/18 12:28:23	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 14:19:39	33194	TELM-06	1	Pass	0	0	0	33
M_TCP_MemoryTest	m_MemoryTest002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest002\Hardware\m_MemoryTest002.exe	2.4	2010/05/05 19:48:21	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:36:28	240636	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest003\Hardware\m_MemoryTest003.exe	2.3	2010/05/05 20:08:58	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 09:01:03	479623	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest004\Hardware\m_MemoryTest004.exe	2.3	2010/05/05 20:09:49	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:31:17	240302	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest005\Hardware\m_MemoryTest005.exe	2.3	2010/05/05 20:10:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:50:32	478690	TELM-06	1	Pass	0	0	0	75
M_TCP_Network	m_Network001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network001\Hardware\m_Network001.exe	2.6	2010/05/21 15:45:43	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 03:56:47	24624	TELM-06	1	Pass	0	0	0	1461
M_TCP_Network	m_Network002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network002\Hardware\m_Network002.exe	2.6	2010/05/12 12:01:17	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 10:59:39	34733	TELM-06	1	Pass	0	0	0	82
M_TCP_Network	m_Network003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network003\Hardware\m_Network003.exe	2.1	2009/12/16 22:27:06	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:53:52	64765	TELM-06	1	Pass	0	0	0	86
M_TCP_Network	m_Network004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network004\Hardware\m_Network004.exe	2.1	2010/04/21 17:14:24	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:51:47	3142	TELM-06	1	Pass	0	0	0	6
M_TCP_Network	m_Network005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network005\Hardware\m_Network005.exe	2.3	2010/05/12 12:02:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:50:03	58268	TELM-06	1	Pass	0	0	0	50
M_TCP_Network	m_Network006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network006\Hardware\m_Network006.exe	2.5	2010/05/21 16:49:56	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:48:34	15386	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network007\Hardware\m_Network007.exe	2.5	2010/05/21 17:09:04	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:47:18	15844	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network008\Hardware\m_Network008.exe	2.5	2010/05/21 17:13:03	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:46:02	10404	TELM-06	1	Pass	0	0	0	1380
M_TCP_Network	m_Network009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network009\Hardware\m_Network009.exe	2.5	2010/05/21 17:21:30	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:44:42	10372	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network010\Hardware\m_Network010.exe	2.5	2010/05/21 17:34:27	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:43:06	10446	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network011\Hardware\m_Network011.exe	2.3	2010/05/12 12:03:38	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:38:33	209491	TELM-06	1	Pass	0	0	0	122
M_TCP_Network	m_Network013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network013\Hardware\m_Network013.exe	2.5	2010/07/01 13:00:48	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:34:38	141614	TELM-06	1	Pass	0	0	0	392
M_TCP_Network	m_Network014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network014\Hardware\m_Network014.exe	2.3	2010/05/12 12:06:29	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:32:56	26621	TELM-06	1	Pass	0	0	0	12
M_TCP_Network	m_Network015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network015\Hardware\m_Network015.exe	2.3	2010/05/24 13:53:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:30:24	35646	TELM-06	1	Pass	0	0	0	25
M_TCP_Network	m_Network016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network016\Hardware\m_Network016.exe	2.1	2010/05/12 12:09:43	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:28:06	55421	TELM-06	1	Pass	0	0	0	262
M_TCP_Network	m_Network017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network017\Hardware\m_Network017.exe	2.4	2010/06/07 13:19:37	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:26:46	12646	TELM-06	1	Pass	0	0	0	41
M_TCP_Network	m_Network018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network018\Hardware\m_Network018.exe	2.5	2010/06/07 13:14:52	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:25:10	21746	TELM-06	1	Pass	0	0	0	11
M_TCP_Network	m_Network019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network019\Hardware\m_Network019.exe	2.2	2010/05/12 12:12:40	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:20:29	56608	TELM-06	1	Pass	0	0	0	51
M_TCP_Network	m_Network020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network020\Hardware\m_Network020.exe	2.2	2010/05/12 12:14:53	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:18:11	60317	TELM-06	1	Pass	0	0	0	67
M_TCP_Network	m_Network021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network021\Hardware\m_Network021.exe	2.2	2010/05/12 12:16:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:15:07	60263	TELM-06	1	Pass	0	0	0	133
M_TCP_Network	m_Network022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network022\Hardware\m_Network022.exe	2.2	2010/05/12 12:17:24	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:12:49	60346	TELM-06	1	Pass	0	0	0	175
M_TCP_Network	m_Network023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network023\Hardware\m_Network023.exe	2.1	2010/05/12 12:18:23	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:06:57	19492	TELM-06	1	Pass	0	0	0	454
M_TCP_Network	m_Network024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network024\Hardware\m_Network024.exe	2.2	2010/05/24 14:41:01	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:03:39	47608	TELM-06	1	Pass	0	0	0	46
M_TCP_Network	m_Network025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network025\Hardware\m_Network025.exe	2.1	2010/05/12 12:20:01	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:58:44	19786	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network026\Hardware\m_Network026.exe	2.3	2010/05/12 12:20:56	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:55:05	20649	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network027\Hardware\m_Network027.exe	2.1	2010/05/12 12:22:02	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:53:01	20954	TELM-06	1	Pass	0	0	0	7
M_TCP_Network	m_Network028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network028\Hardware\m_Network028.exe	2.1	2010/06/07 13:22:27	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:48:48	14490	TELM-06	1	Pass	0	0	0	22
M_TCP_Network	m_Network029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network029\Hardware\m_Network029.exe	2.4	2010/02/09 21:21:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 08:40:11	271591	TELM-06	1	Pass	0	0	0	48
M_TCP_Open	m_Open001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open001\Hardware\m_Open001.exe	2.11	2010/05/20 13:50:26	2.11	Up-to-date	Baseline 17_1	None	09/09/2010 15:52:18	31711	TELM-06	1	Pass	0	0	0	335
M_TCP_Open	m_Open002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open002\Hardware\m_Open002.exe	2.3	2010/01/18 15:52:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:48:56	18666	TELM-06	1	Pass	0	0	0	31
M_TCP_Open	m_Open003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open003\Hardware\m_Open003.exe	2.0	2009/10/15 19:09:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:43:13	6520	TELM-06	1	Pass	0	0	0	3
M_TCP_Open	m_Open004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open004\Hardware\m_Open004.exe	2.6	2010/06/22 20:01:21	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:42:09	14009	TELM-06	1	Pass	0	0	0	21
M_TCP_Open	m_Open005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open005\Hardware\m_Open005.exe	2.3	2010/06/22 17:40:45	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:38:32	16841	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open006\Hardware\m_Open006.exe	2.2	2010/03/15 14:26:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:35:38	21195	TELM-06	1	Pass	0	0	0	1173
M_TCP_Open	m_Open007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open007\Hardware\m_Open007.exe	2.7	2010/03/15 14:46:20	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 15:32:17	17500	TELM-06	1	Pass	0	0	0	1181
M_TCP_Open	m_Open008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open008\Hardware\m_Open008.exe	2.1	2009/10/30 14:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:29:53	7791	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open009\Hardware\m_Open009.exe	2.7	2010/06/16 00:36:52	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 15:28:40	12371	TELM-06	1	Pass	0	0	0	1268
M_TCP_Open	m_Open010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open010\Hardware\m_Open010.exe	2.6	2010/03/15 15:05:50	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:27:00	17439	TELM-06	1	Pass	0	0	0	3961
M_TCP_Open	m_Open011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open011\Hardware\m_Open011.exe	2.2	2010/02/10 13:31:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:25:32	6242	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open012\Hardware\m_Open012.exe	2.4	2010/02/26 17:01:14	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:23:16	15659	TELM-06	1	Pass	0	0	0	10
M_TCP_Open	m_Open013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open013\Hardware\m_Open013.exe	2.5	2010/02/10 13:34:06	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:22:15	9524	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open014\Hardware\m_Open014.exe	2.6	2010/02/10 13:35:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:20:57	8590	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open015\Hardware\m_Open015.exe	2.0	2010/01/28 19:00:26	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:19:48	9405	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open030\Hardware\m_Open030.exe	2.1	2010/05/10 13:57:56	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 16:27:31	58248	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open031\Hardware\m_Open031.exe	2.7	2009/11/19 21:48:20	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 15:57:36	63823	TELM-06	1	Pass	0	0	0	13
M_TCP_Open	m_Open032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open032\Hardware\m_Open032.exe	2.3	2010/02/03 22:17:27	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 12:16:17	32558	TELM-06	1	Pass	0	0	0	18
M_TCP_Open	m_Open033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open033\Hardware\m_Open033.exe	2.3	2010/02/03 22:32:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 11:52:10	9920	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open034\Hardware\m_Open034.exe	2.2	2010/02/17 14:19:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:17:24	56963	TELM-06	1	Pass	0	0	0	42
M_TCP_Open	m_Open035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open035\Hardware\m_Open035.exe	2.2	2010/02/17 14:10:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:15:16	9741	TELM-06	1	Pass	0	0	0	32
M_TCP_Open	m_Open036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open036\Hardware\m_Open036.exe	2.1	2009/10/21 15:52:46	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 15:36:29	36275	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open037\Hardware\m_Open037.exe	2.2	2009/10/21 15:51:07	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 15:16:29	31301	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open038\Hardware\m_Open038.exe	2.7	2010/02/17 14:00:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 15:11:22	78368	TELM-06	1	Pass	0	0	0	46
M_TCP_Open	m_Open039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open039\Hardware\m_Open039.exe	2.0	2009/11/09 14:20:44	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:10:04	9099	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open040\Hardware\m_Open040.exe	2.1	2010/02/01 19:45:37	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:08:33	17044	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open041\Hardware\m_Open041.exe	2.0	2009/11/09 14:21:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 07:38:16	513670	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open042\Hardware\m_Open042.exe	2.4	2010/07/02 12:38:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:07:38	22657	TELM-06	1	Pass	0	0	0	44
M_TCP_Polling	m_Polling016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling016\Hardware\m_Polling016.exe	2.0	2009/09/23 19:25:05	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 02:44:01	49161	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling017\Hardware\m_Polling017.exe	2.1	2010/01/26 20:04:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:26:15	6214	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling018\Hardware\m_Polling018.exe	2.1	2010/01/26 20:05:23	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:41:30	81540	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling024\Hardware\m_Polling024.exe	2.5	2010/01/26 20:05:45	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:37:34	78621	TELM-06	1	Pass	0	0	0	10
M_TCP_Polling	m_Polling025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling025\Hardware\m_Polling025.exe	2.1	2009/11/24 20:51:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:34:43	46258	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling026\Hardware\m_Polling026.exe	2.2	2009/12/03 14:59:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:32:08	19545	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling040\Hardware\m_Polling040.exe	2.1	2010/01/26 20:06:14	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:29:50	6167	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling045\Hardware\m_Polling045.exe	2.1	2010/01/26 20:06:42	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:26:44	36143	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling046\Hardware\m_Polling046.exe	2.6	2010/04/16 14:21:47	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 02:22:56	47171	TELM-06	1	Pass	0	0	0	38
M_TCP_Polling	m_Polling047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling047\Hardware\m_Polling047.exe	2.1	2010/01/26 20:07:28	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:20:20	36006	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling048\Hardware\m_Polling048.exe	2.3	2009/11/24 20:53:00	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:17:25	10081	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling049\Hardware\m_Polling049.exe	2.4	2010/03/10 19:44:21	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:15:41	47298	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling050\Hardware\m_Polling050.exe	2.6	2010/06/29 17:44:57	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 02:13:37	22799	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling065\Hardware\m_Polling065.exe	2.6	2010/06/09 18:52:38	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 02:10:29	42142	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling066\Hardware\m_Polling066.exe	2.3	2009/11/09 22:57:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:07:48	52654	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling067\Hardware\m_Polling067.exe	2.1	2009/10/22 14:28:02	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:05:18	85012	TELM-06	1	Pass	0	0	0	11
M_TCP_Polling	m_Polling068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling068\Hardware\m_Polling068.exe	2.3	2009/11/05 18:23:27	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:59:39	7128	TELM-06	1	Pass	0	0	0	8
M_TCP_Polling	m_Polling076	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling076\Hardware\m_Polling076.exe	2.1	2009/11/03 15:55:29	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:58:27	10152	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling077	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling077\Hardware\m_Polling077.exe	2.1	2009/11/05 18:34:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 10:43:50	527580	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling080	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling080\Hardware\m_Polling080.exe	2.2	2009/11/05 18:24:04	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 10:32:57	277453	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling081	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling081\Hardware\m_Polling081.exe	2.3	2010/05/27 14:19:41	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:54:38	24239	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling082	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling082\Hardware\m_Polling082.exe	2.2	2009/11/02 18:25:15	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:51:48	9486	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling083	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling083\Hardware\m_Polling083.exe	2.3	2009/11/02 18:45:07	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:49:02	19511	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling085	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling085\Hardware\m_Polling085.exe	2.1	2009/10/22 13:55:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 10:14:37	529317	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling088	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling088\Hardware\m_Polling088.exe	2.5	2010/05/12 17:59:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:47:10	23611	TELM-06	1	Pass	0	0	0	27
M_TCP_Polling	m_Polling089	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling089\Hardware\m_Polling089.exe	2.7	2010/05/11 19:28:17	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 01:42:39	97111	TELM-06	1	Pass	0	0	0	26
M_TCP_Polling	m_Polling090	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling090\Hardware\m_Polling090.exe	2.6	2010/02/15 20:02:24	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:38:42	83317	TELM-06	1	Pass	0	0	0	32
M_TCP_Polling	m_Polling091	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling091\Hardware\m_Polling091.exe	2.6	2010/02/15 20:16:12	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:31:40	13197	TELM-06	1	Pass	0	0	0	13
M_TCP_Polling	m_Polling092	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling092\Hardware\m_Polling092.exe	2.1	2009/11/05 18:41:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 10:00:48	277236	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling093	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling093\Hardware\m_Polling093.exe	2.0	2009/11/09 18:30:12	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 01:28:49	16888	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling094	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling094\Hardware\m_Polling094.exe	2.0	2009/11/09 18:30:49	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 08:29:43	307501	TELM-06	1	Pass	0	0	0	2

M_TCP_Polling	m_Polling095	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling095\Hardware\m_Polling095.exe	2.0	2009/11/09 18:31:27	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 01:26:27	8356	TELM-06	1	Pass	0	0	0	2
M_TCP_PowerInhibit	m_Power001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power001\Hardware\m_Power001.exe	2.2	2010/06/07 13:10:06	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:24:43	58663	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power002\Hardware\m_Power002.exe	2.2	2010/01/05 19:15:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:21:27	10702	TELM-06	1	Pass	0	0	0	194
M_TCP_PowerInhibit	m_Power003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power003\Hardware\m_Power003.exe	2.4	2010/06/22 14:04:45	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:16:08	54558	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power004\Hardware\m_Power004.exe	2.3	2010/03/22 14:26:21	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:14:09	18198	TELM-06	1	Pass	0	0	0	46
M_TCP_PowerInhibit	m_Power005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power005\Hardware\m_Power005.exe	2.6	2010/07/02 12:34:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:56:42	299654	TELM-06	1	Pass	0	0	0	310
M_TCP_PowerInhibit	m_Power006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power006\Hardware\m_Power006.exe	2.6	2010/06/22 12:27:46	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:08:09	194744	TELM-06	1	Pass	0	0	0	2609
M_TCP_PowerInhibit	m_Power007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power007\Hardware\m_Power007.exe	2.1	2009/12/14 16:39:57	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:04:38	19408	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power008\Hardware\m_Power008.exe	2.2	2009/12/14 16:44:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:01:35	34817	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power009\Hardware\m_Power009.exe	2.4	2010/07/01 15:00:10	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:56:38	12044	TELM-06	1	Pass	0	0	0	9
M_TCP_PowerInhibit	m_Power010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power010\Hardware\m_Power010.exe	2.1	2010/02/16 22:18:45	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:54:20	11834	TELM-06	1	Pass	0	0	0	26
M_TCP_PowerInhibit	m_Power011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power011\Hardware\m_Power011.exe	2.1	2010/03/11 19:04:17	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:50:17	106275	TELM-06	1	Pass	0	0	0	116
M_TCP_RawMode	m_RawMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode001\Hardware\m_RawMode001.exe	2.4	2010/01/26 20:08:13	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:28:59	7730	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode002\Hardware\m_RawMode002.exe	2.6	2010/01/27 13:00:04	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:26:16	9657	TELM-06	1	Pass	0	0	0	87
M_TCP_RawMode	m_RawMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode003\Hardware\m_RawMode003.exe	2.5	2010/01/27 13:03:49	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:18:56	11574	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode004\Hardware\m_RawMode004.exe	2.6	2010/01/27 13:09:25	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:12:46	7527	TELM-06	1	Pass	0	0	0	2
M_TCP_RawMode	m_RawMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode005\Hardware\m_RawMode005.exe	2.3	2010/01/26 20:10:41	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:10:38	13467	TELM-06	1	Pass	0	0	0	9
M_TCP_RawMode	m_RawMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode006\Hardware\m_RawMode006.exe	2.2	2010/01/26 20:11:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:26:31	9182	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode007\Hardware\m_RawMode007.exe	2.3	2010/06/09 21:03:53	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:06:19	11340	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode008\Hardware\m_RawMode008.exe	2.4	2010/01/27 13:14:20	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:04:55	16186	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode009\Hardware\m_RawMode009.exe	2.2	2009/12/14 16:51:58	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:02:58	15773	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode010\Hardware\m_RawMode010.exe	2.1	2009/10/27 17:32:20	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:01:40	3321	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode011\Hardware\m_RawMode011.exe	2.2	2010/02/01 16:15:38	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:00:25	8799	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode012\Hardware\m_RawMode012.exe	2.3	2010/01/26 20:13:22	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:59:19	11799	TELM-06	1	Pass	0	0	0	3
M_TCP_Reset	m_Reset001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset001\Hardware\m_Reset001.exe	2.3	2010/01/06 21:25:12	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:29:50	9350	TELM-06	1	Pass	0	0	0	23
M_TCP_Reset	m_Reset002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset002\Hardware\m_Reset002.exe	2.4	2010/01/28 21:52:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:26:24	20678	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security002\Hardware\m_Security002.exe	2.9	2010/06/03 15:18:25	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 08:06:31	457327	TELM-06	1	Pass	0	0	0	154
M_TCP_Security	m_Security003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security003\Hardware\m_Security003.exe	2.4	2010/02/17 16:20:41	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:34:30	7141	TELM-06	1	Pass	0	0	0	73
M_TCP_Security	m_Security004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security004\Hardware\m_Security004.exe	2.4	2010/02/15 15:01:00	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:28:41	253912	TELM-06	1	Pass	0	0	0	112
M_TCP_Security	m_Security005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security005\Hardware\m_Security005.exe	2.2	2010/02/17 16:23:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:26:41	3998	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security006\Hardware\m_Security006.exe	2.4	2010/02/17 14:44:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:22:26	147476	TELM-06	1	Pass	0	0	0	108
M_TCP_Security	m_Security007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security007\Hardware\m_Security007.exe	2.6	2010/02/18 15:13:38	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 18:17:09	110110	TELM-06	1	Pass	0	0	0	126
M_TCP_Security	m_Security008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security008\Hardware\m_Security008.exe	2.3	2010/02/17 16:28:05	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:14:15	6759	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security009\Hardware\m_Security009.exe	2.3	2010/02/17 16:32:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:12:33	6362	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security010\Hardware\m_Security010.exe	2.3	2010/02/18 17:45:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:09:11	38532	TELM-06	1	Pass	0	0	0	113
M_TCP_Security	m_Security011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security011\Hardware\m_Security011.exe	2.5	2010/06/03 18:16:10	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 18:06:02	20121	TELM-06	1	Pass	0	0	0	58
M_TCP_Security	m_Security012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security012\Hardware\m_Security012.exe	2.9	2010/02/18 15:14:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 18:05:07	4736	TELM-06	1	Pass	0	0	0	132
M_TCP_Security	m_Security013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security013\Hardware\m_Security013.exe	2.5	2010/02/18 15:15:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 18:02:35	11859	TELM-06	1	Pass	0	0	0	140
M_TCP_Security	m_Security014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security014\Hardware\m_Security014.exe	2.2	2010/06/03 18:21:27	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 17:41:50	14900	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security015\Hardware\m_Security015.exe	2.4	2010/02/17 17:58:26	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:40:08	596225	TELM-06	1	Pass	0	0	0	34
M_TCP_Security	m_Security016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security016\Hardware\m_Security016.exe	2.2	2010/06/08 19:56:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:00:43	10779	TELM-06	1	Pass	0	0	0	146
M_TCP_Security	m_Security017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security017\Hardware\m_Security017.exe	2.2	2009/12/08 17:29:14	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:58:58	14639	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security018\Hardware\m_Security018.exe	2.4	2010/03/18 16:05:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:53:23	40567	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security019\Hardware\m_Security019.exe	2.3	2010/03/26 13:43:17	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:46:17	90409	TELM-06	1	Pass	0	0	0	1230
M_TCP_Security	m_Security020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security020\Hardware\m_Security020.exe	2.3	2010/06/03 18:22:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:44:32	22169	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security021\Hardware\m_Security021.exe	2.5	2010/06/03 18:37:52	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:41:09	40984	TELM-06	1	Pass	0	0	0	81
M_TCP_Security	m_Security022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security022\Hardware\m_Security022.exe	2.9	2010/06/18 19:05:16	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 06:48:55	290983	TELM-06	1	Pass	0	0	0	55496
M_TCP_Security	m_Security023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security023\Hardware\m_Security023.exe	2.12	2010/06/10 17:30:47	2.12	Up-to-date	Baseline 17_1	None	09/10/2010 09:34:06	51544	TELM-06	1	Pass	0	0	0	1119
M_TCP_Security	m_Security024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security024\Hardware\m_Security024.exe	2.3	2010/03/26 13:06:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:03:45	31038	TELM-06	1	Pass	0	0	0	4
M_TCP_Security	m_Security025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security025\Hardware\m_Security025.exe	2.3	2010/06/10 14:39:09	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:00:56	48457	TELM-06	1	Pass	0	0	0	158
M_TCP_Security	m_Security026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security026\Hardware\m_Security026.exe	2.3	2010/02/18 15:15:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:40:12	4205	TELM-06	1	Pass	0	0	0	60
M_TCP_Security	m_Security027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security027\Hardware\m_Security027.exe	2.1	2010/06/29 19:03:00	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:38:18	38224	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security028\Hardware\m_Security028.exe	2.1	2010/06/29 19:26:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:34:00	38151	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security029\Hardware\m_Security029.exe	2.2	2010/03/11 18:29:14	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:32:04	20544	TELM-06	1	Pass	0	0	0	46
M_TCP_Security	m_Security030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security030\Hardware\m_Security030.exe	2.2	2010/02/08 13:02:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:30:18	11778	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security031\Hardware\m_Security031.exe	2.1	2010/06/03 18:44:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:27:44	22954	TELM-06	1	Pass	0	0	0	16
M_TCP_Security	m_Security032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security032\Hardware\m_Security032.exe	2.1	2010/02/03 12:51:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:26:05	8192	TELM-06	1	Pass	0	0	0	11
M_TCP_Security	m_Security033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security033\Hardware\m_Security033.exe	2.1	2010/01/27 21:32:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:20:26	13923	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security034\Hardware\m_Security034.exe	2.3	2010/03/26 13:19:51	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:17:05	118683	TELM-06	1	Pass	0	0	0	15
M_TCP_Security	m_Security035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security035\Hardware\m_Security035.exe	2.0	2010/01/14 21:18:30	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 17:12:23	17962	TELM-06	1	Pass	0	0	0	30
M_TCP_Security	m_Security036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security036\Hardware\m_Security036.exe	2.5	2010/05/07 14:41:43	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:04:59	86723	TELM-06	1	Pass	0	0	0	10
M_TCP_Security	m_Security037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security037\Hardware\m_Security037.exe	2.6	2010/06/03 18:54:41	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 17:18:10	51134	TELM-06	1	Pass	0	0	0	119
M_TCP_Security	m_Security038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security038\Hardware\m_Security038.exe	2.9	2010/06/07 13:02:07	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 07:56:29	167622	TELM-06	1	Pass	0	0	0	170
M_TCP_Security	m_Security039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security039\Hardware\m_Security039.exe	2.2	2010/02/18 15:16:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:03:14	56942	TELM-06	1	Pass	0	0	0	55
M_TCP_Security	m_Security040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security040\Hardware\m_Security040.exe	2.3	2010/06/03 20:43:31	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:56:38	278500	TELM-06	1	Pass	0	0	0	33
M_TCP_Security	m_Security041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security041\Hardware\m_Security041.exe	2.6	2010/04/21 10:55:08	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 16:53:58	66016	TELM-06	1	Pass	0	0	0	77
M_TCP_Security	m_Security042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security042\Hardware\m_Security042.exe	2.1	2010/04/08 12:57:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:51:35	58218	TELM-06	1	Pass	0	0	0	48
M_TCP_Security	m_Security043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security043\Hardware\m_Security043.exe	2.0	2010/02/16 20:27:42	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:48:16	68755	TELM-06	1	Pass	0	0	0	56
M_TCP_Security	m_Security044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security044\Hardware\m_Security044.exe	2.2	2010/05/13 19:14:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:42:14	70729	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security045\Hardware\m_Security045.exe	2.3	2010/05/12 17:38:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:35:43	71118	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security046\Hardware\m_Security046.exe	2.2	2010/06/07 13:05:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:33:15	3514	TELM-06	1	Pass	0	0	0	30
M_TCP_WirelessWakeup	m_Wakeup001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup001\Hardware\m_Wakeup001.exe	2.3	2010/06/21 17:44:25	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 07:49:09	291447	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup002\Hardware\m_Wakeup002.exe	2.5	2010/05/12 16:16:45	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:24:08	35358	TELM-06	1	Pass	0	0	0	12
M_TCP_WirelessWakeup	m_Wakeup003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup003\Hardware\m_Wakeup003.exe	2.2	2010/03/10 17:59:29	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:22:02	14376	TELM-06	1	Pass	0	0	0	11
M_TCP_WirelessWakeup	m_Wakeup004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup004\Hardware\m_Wakeup004.exe	2.1	2009/12/08 16:08:20	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 16:56:49	19221	TELM-06	1	Pass	0	0	0	31
M_TCP_WirelessWakeup	m_Wakeup005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup005\Hardware\m_Wakeup005.exe	2.3	2010/06/08 21:32:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:20:04	20269	TELM-06	1	Pass	0	0	0	40
M_TCP_WirelessWakeup	m_Wakeup006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup006\Hardware\m_Wakeup006.exe	2.3	2010/01/27 00:35:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:17:46	10381	TELM-06	1	Pass	0	0	0	55
M_TCP_WirelessWakeup	m_Wakeup007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup007\Hardware\m_Wakeup007.exe	2.1	2009/12/04 19:28:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:15:13	12954	TELM-06	1	Pass	0	0	0	3
M_TCP_WirelessWakeup	m_Wakeup008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup008\Hardware\m_Wakeup008.exe	2.5	2010/05/07 15:41:21	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:12:50	54760	TELM-06	1	Pass	0	0	0	24
M_TCP_WirelessWakeup	m_Wakeup009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup009\Hardware\m_Wakeup009.exe	2.6	2010/05/12 16:17:49	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 16:10:38	52064	TELM-06	1	Pass	0	0	0	32
M_TCP_WirelessWakeup	m_Wakeup010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup010\Hardware\m_Wakeup010.exe	2.4	2010/05/12 16:20:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:08:38	45347	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup011\Hardware\m_Wakeup011.exe	2.3	2010/04/06 18:07:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:06:48	27976	TELM-06	1	Pass	0	0	0	25
M_TCP_WirelessWakeup	m_Wakeup012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup012\Hardware\m_Wakeup012.exe	2.6	2010/05/12 16:20:45	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 16:04:58	48697	TELM-06	1	Pass	0	0	0	45
M_TCP_WirelessWakeup	m_Wakeup013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup013\Hardware\m_Wakeup013.exe	2.2	2010/02/24 12:20:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:00:45	127498	TELM-06	1	Pass	0	0	0	2
M_TCP_WirelessWakeup	m_Wakeup014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode1\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup014\Hardware\m_Wakeup014.exe	2.3	2010/06/04 11:50:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:54:12	74754	TELM-06	1	Pass	0	0	0	30
M_TCP_AntennaSelection	m_Antenna001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna001\Hardware\m_Antenna001.exe	2.4	2010/06/16 12:31:48	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 10:52:50	238485	TELM-06	1	Pass	0	0	0	127
M_TCP_AntennaSelection	m_Antenna002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna002\Hardware\m_Antenna002.exe	2.0	2010/06/16 12:29:50	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 10:37:45	286977	TELM-06	1	Pass	0	0	0	99
M_TCP_Channel	m_ChannelRecovery001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery001\Hardware\m_ChannelRecovery001.exe	2.5	2010/04/08 14:25:18	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:29:25	15088	TELM-06	1	Pass	0	0	0	26
M_TCP_Channel	m_ChannelRecovery002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery002\Hardware\m_ChannelRecovery002.exe	2.2	2009/11/30 22:31:32	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:27:27	17487	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery003\Hardware\m_ChannelRecovery003.exe	2.2	2009/11/30 22:32:21	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:26:26	17450	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery004\Hardware\m_ChannelRecovery004.exe	2.2	2010/05/26 12:04:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:24:54	14582	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery005\Hardware\m_ChannelRecovery005.exe	2.5	2010/06/21 18:34:44	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:20:03	14296	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery006\Hardware\m_ChannelRecovery006.exe	2.4	2010/05/07 17:09:38	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:17:31	16257	TELM-06	1	Pass	0	0	0	36
M_TCP_Channel	m_ChannelRecovery007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery007\Hardware\m_ChannelRecovery007.exe	2.7	2010/05/11 11:49:24	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 03:14:37	28008	TELM-06	1	Pass	0	0	0	89
M_TCP_Channel	m_ChannelRecovery008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery008\Hardware\m_ChannelRecovery008.exe	2.6	2010/05/13 15:05:39	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 03:12:12	33985	TELM-06	1	Pass	0	0	0	574
M_TCP_Channel	m_ChannelRecovery009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery009\Hardware\m_ChannelRecovery009.exe	2.6	2010/06/22 15:16:15	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 03:05:45	63655	TELM-06	1	Pass	0	0	0	342
M_TCP_Channel	m_ChannelRecovery010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery010\Hardware\m_ChannelRecovery010.exe	2.8	2010/05/26 15:48:50	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 03:00:11	25028	TELM-06	1	Pass	0	0	0	136
M_TCP_Channel	m_ChannelRecovery011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery011\Hardware\m_ChannelRecovery011.exe	2.5	2010/05/13 12:31:10	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:56:38	15906	TELM-06	1	Pass	0	0	0	14
M_TCP_Channel	m_ChannelRecovery012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery012\Hardware\m_ChannelRecovery012.exe	2.6	2010/09/07 20:36:00	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 02:53:56	48543	TELM-06	1	Pass	0	0	0	195
M_TCP_Channel	m_ChannelRecovery013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery013\Hardware\m_ChannelRecovery013.exe	2.2	2010/01/05 19:17:54	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:52:16	24417	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery014\Hardware\m_ChannelRecovery014.exe	2.8	2010/05/13 13:19:47	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 02:46:59	56914	TELM-06	1	Pass	0	0	0	23
M_TCP_Channel	m_ChannelRecovery015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery015\Hardware\m_ChannelRecovery015.exe	2.6	2010/06/03 18:01:04	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 02:43:14	12563	TELM-06	1	Pass	0	0	0	88
M_TCP_Channel	m_ChannelRecovery016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery016\Hardware\m_ChannelRecovery016.exe	2.4	2010/02/11 20:10:03	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:40:29	37931	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery017\Hardware\m_ChannelRecovery017.exe	2.3	2010/02/11 17:38:13	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:37:22	10420	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery018\Hardware\m_ChannelRecovery018.exe	2.3	2010/02/01 19:41:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:33:12	18000	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery020\Hardware\m_ChannelRecovery020.exe	2.7	2010/06/23 17:42:40	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 02:30:57	37809	TELM-06	1	Pass	0	0	0	109
M_TCP_Channel	m_ChannelRecovery021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery021\Hardware\m_ChannelRecovery021.exe	2.3	2010/04/13 18:33:24	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:28:56	32447	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery022\Hardware\m_ChannelRecovery022.exe	2.3	2010/04/13 18:36:50	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:25:59	19838	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery023\Hardware\m_ChannelRecovery023.exe	2.5	2010/04/13 18:41:23	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:22:13	29944	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery024\Hardware\m_ChannelRecovery024.exe	2.4	2010/04/07 12:33:58	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:19:08	19945	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery025\Hardware\m_ChannelRecovery025.exe	2.2	2010/02/01 19:43:28	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:16:43	11090	TELM-06	1	Pass	0	0	0	8

M_TCP_Channel	m_ChannelRecovery026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery026\Hardware\m_ChannelRecovery026.exe	2.2	2010/02/11 21:01:06	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:14:01	11519	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery027\Hardware\m_ChannelRecovery027.exe	2.2	2010/02/03 17:18:54	2.2	Up-to-date	Baseline 17_1	None	09/13/2010 07:41:57	13574	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery028\Hardware\m_ChannelRecovery028.exe	2.2	2010/02/03 17:19:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:08:43	11561	TELM-06	1	Pass	0	0	0	10
M_TCP_Channel	m_ChannelRecovery029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery029\Hardware\m_ChannelRecovery029.exe	2.2	2009/12/18 20:13:33	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:06:45	13555	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery030\Hardware\m_ChannelRecovery030.exe	2.10	2010/06/23 21:15:31	2.10	Up-to-date	Baseline 17_1	None	09/10/2010 02:00:16	19109	TELM-06	1	Pass	0	0	0	30
M_TCP_Channel	m_ChannelRecovery031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery031\Hardware\m_ChannelRecovery031.exe	2.6	2010/04/13 18:46:58	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:58:49	18892	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery032\Hardware\m_ChannelRecovery032.exe	2.5	2010/06/23 21:50:13	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:55:21	18086	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery033\Hardware\m_ChannelRecovery033.exe	2.3	2010/04/07 17:51:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:53:55	19206	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery038\Hardware\m_ChannelRecovery038.exe	2.6	2010/05/26 15:55:12	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:48:40	20828	TELM-06	1	Pass	0	0	0	175
M_TCP_Channel	m_ChannelRecovery039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery039\Hardware\m_ChannelRecovery039.exe	2.3	2010/02/10 20:47:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:46:56	12347	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery040\Hardware\m_ChannelRecovery040.exe	2.5	2010/06/07 12:54:38	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:44:18	15573	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery041\Hardware\m_ChannelRecovery041.exe	2.1	2009/10/22 13:18:38	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:40:37	38479	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery042\Hardware\m_ChannelRecovery042.exe	2.3	2009/12/03 15:57:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:33:30	21995	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery043\Hardware\m_ChannelRecovery043.exe	2.6	2010/06/24 18:37:58	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:30:41	56022	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery045\Hardware\m_ChannelRecovery045.exe	2.2	2009/11/12 21:22:11	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:28:35	11646	TELM-06	1	Pass	0	0	0	40
M_TCP_Channel	m_ChannelRecovery046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery046\Hardware\m_ChannelRecovery046.exe	2.3	2009/11/16 18:40:10	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:26:13	12636	TELM-06	1	Pass	0	0	0	49
M_TCP_Channel	m_ChannelRecovery047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery047\Hardware\m_ChannelRecovery047.exe	2.1	2009/11/16 18:58:46	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:22:22	52466	TELM-06	1	Pass	0	0	0	95
M_TCP_Channel	m_ChannelRecovery048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery048\Hardware\m_ChannelRecovery048.exe	2.3	2010/06/24 18:46:14	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:18:49	63270	TELM-06	1	Pass	0	0	0	156
M_TCP_Channel	m_ChannelRecovery049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery049\Hardware\m_ChannelRecovery049.exe	2.5	2010/03/26 13:55:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:14:40	36508	TELM-06	1	Pass	0	0	0	207
M_TCP_Channel	m_ChannelRecovery050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery050\Hardware\m_ChannelRecovery050.exe	2.2	2009/11/20 13:38:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:12:34	18906	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery051\Hardware\m_ChannelRecovery051.exe	2.3	2010/06/24 19:40:38	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:06:23	83694	TELM-06	1	Pass	0	0	0	196
M_TCP_Channel	m_ChannelRecovery053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery053\Hardware\m_ChannelRecovery053.exe	2.3	2010/06/24 21:57:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:04:03	31591	TELM-06	1	Pass	0	0	0	96
M_TCP_Channel	m_ChannelRecovery054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery054\Hardware\m_ChannelRecovery054.exe	2.4	2010/06/24 22:31:30	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:00:41	9000	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery055\Hardware\m_ChannelRecovery055.exe	2.4	2010/02/11 14:12:25	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 03:37:30	2022332	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery061	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery061\Hardware\m_ChannelRecovery061.exe	2.3	2010/04/06 11:52:17	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:55:54	20734	TELM-06	1	Pass	0	0	0	12
M_TCP_Channel	m_ChannelRecovery062	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery062\Hardware\m_ChannelRecovery062.exe	2.2	2010/06/25 15:47:28	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:54:06	13116	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery063	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery063\Hardware\m_ChannelRecovery063.exe	2.9	2010/05/13 17:43:56	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 00:49:51	24457	TELM-06	1	Pass	0	0	0	39
M_TCP_Channel	m_ChannelRecovery064	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery064\Hardware\m_ChannelRecovery064.exe	2.4	2010/05/06 14:14:13	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:47:26	40763	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery065\Hardware\m_ChannelRecovery065.exe	2.2	2010/05/06 15:43:30	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:43:26	37288	TELM-06	1	Pass	0	0	0	7
M_TCP_Channel	m_ChannelRecovery066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery066\Hardware\m_ChannelRecovery066.exe	2.2	2010/05/06 15:49:13	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:38:33	37510	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery067\Hardware\m_ChannelRecovery067.exe	2.2	2010/05/06 15:58:13	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:36:06	40825	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery068\Hardware\m_ChannelRecovery068.exe	2.3	2010/06/03 18:10:46	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:31:57	34700	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery069	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery069\Hardware\m_ChannelRecovery069.exe	2.3	2010/05/10 15:40:47	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 00:02:47	1009344	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery070	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery070\Hardware\m_ChannelRecovery070.exe	2.1	2010/02/04 18:14:32	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:28:49	12790	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery071	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery071\Hardware\m_ChannelRecovery071.exe	2.1	2010/02/04 17:57:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:22:27	15803	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery072	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery072\Hardware\m_ChannelRecovery072.exe	2.3	2010/05/07 11:42:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:18:48	44203	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery073	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery073\Hardware\m_ChannelRecovery073.exe	2.3	2010/05/07 11:43:00	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:15:46	44180	TELM-06	1	Pass	0	0	0	8
M_TCP_Close	m_Close001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close001\Hardware\m_Close001.exe	2.6	2010/03/29 17:53:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:11:39	7568	TELM-06	1	Pass	0	0	0	51
M_TCP_Close	m_Close002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close002\Hardware\m_Close002.exe	2.3	2010/01/26 20:03:49	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:07:32	8793	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close003\Hardware\m_Close003.exe	2.3	2010/06/24 15:01:11	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:05:28	7015	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close004\Hardware\m_Close004.exe	2.5	2010/03/16 12:03:27	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:03:57	14967	TELM-06	1	Pass	0	0	0	150
M_TCP_Close	m_Close005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close005\Hardware\m_Close005.exe	2.5	2010/05/24 17:00:08	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:01:59	19416	TELM-06	1	Pass	0	0	0	61
M_TCP_Close	m_Close006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close006\Hardware\m_Close006.exe	2.2	2009/11/19 20:38:31	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:00:55	11039	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close007\Hardware\m_Close007.exe	2.1	2010/02/04 18:46:09	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:59:33	16535	TELM-06	1	Pass	0	0	0	34
M_TCP_Close	m_Close008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close008\Hardware\m_Close008.exe	2.4	2010/05/12 15:22:15	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:57:13	7081	TELM-06	1	Pass	0	0	0	10
M_TCP_Close	m_Close009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close009\Hardware\m_Close009.exe	2.2	2010/02/04 15:29:20	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:54:59	7903	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close010\Hardware\m_Close010.exe	2.0	2009/09/21 12:45:39	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:52:24	15776	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close011\Hardware\m_Close011.exe	2.0	2009/09/21 12:46:09	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:51:04	8197	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close012\Hardware\m_Close012.exe	2.0	2009/09/21 12:46:36	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:48:52	11534	TELM-06	1	Pass	0	0	0	3
M_TCP_Close	m_Close013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close013\Hardware\m_Close013.exe	2.0	2009/09/21 12:18:45	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:47:36	7995	TELM-06	1	Pass	0	0	0	4
M_TCP_Close	m_Close014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close014\Hardware\m_Close014.exe	2.1	2010/01/29 16:45:05	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:46:14	8895	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close016\Hardware\m_Close016.exe	2.2	2010/08/24 12:05:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:44:55	8137	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close017\Hardware\m_Close017.exe	2.0	2009/09/21 12:47:07	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:43:35	9549	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close019\Hardware\m_Close019.exe	2.1	2009/10/01 14:47:50	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:42:35	4923	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close020\Hardware\m_Close020.exe	2.2	2009/10/30 14:47:42	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:37:22	9555	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close021\Hardware\m_Close021.exe	2.0	2009/09/21 12:48:28	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:33:45	2726	TELM-06	1	Pass	0	0	0	1
M_TCP_Close	m_Close022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close022\Hardware\m_Close022.exe	2.2	2009/10/30 14:53:53	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:31:23	12396	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_ARQ001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ001\Hardware\m_ARQ001.exe	2.5	2010/06/29 15:20:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:53:12	14390	TELM-06	1	Pass	0	0	0	54
M_TCP_Data	m_ARQ002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ002\Hardware\m_ARQ002.exe	2.0	2009/09/21 18:25:53	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:52:08	7898	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ003\Hardware\m_ARQ003.exe	2.3	2010/04/05 12:22:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:50:49	8583	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_ARQ004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ004\Hardware\m_ARQ004.exe	2.1	2009/11/04 21:22:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:48:42	8350	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_ARQ005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ005\Hardware\m_ARQ005.exe	2.2	2009/12/23 13:26:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:47:35	7670	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ006\Hardware\m_ARQ006.exe	2.1	2009/09/24 16:41:52	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:45:06	8831	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_ARQ007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ007\Hardware\m_ARQ007.exe	2.0	2009/09/21 18:27:29	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:43:47	10882	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_ARQ008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ008\Hardware\m_ARQ008.exe	2.0	2009/09/21 18:28:06	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:42:24	11432	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ009\Hardware\m_ARQ009.exe	2.2	2010/02/04 13:08:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:41:23	9980	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ010\Hardware\m_ARQ010.exe	2.2	2010/02/04 13:19:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:40:09	10064	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ011\Hardware\m_ARQ011.exe	2.3	2010/02/01 13:30:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:38:27	9050	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ012\Hardware\m_ARQ012.exe	2.3	2010/02/18 15:17:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:33:12	8999	TELM-06	1	Pass	0	0	0	211
M_TCP_Data	m_ARQ013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ013\Hardware\m_ARQ013.exe	2.3	2010/02/03 16:38:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:32:14	11070	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_ARQ014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ014\Hardware\m_ARQ014.exe	2.4	2010/04/05 11:08:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:28:46	8072	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ015\Hardware\m_ARQ015.exe	2.6	2010/06/08 15:25:41	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:25:48	13605	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ016\Hardware\m_ARQ016.exe	2.3	2009/11/19 20:22:28	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:24:20	9916	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ017\Hardware\m_ARQ017.exe	2.2	2009/09/30 15:01:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:23:06	7437	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ018\Hardware\m_ARQ018.exe	2.7	2010/06/24 12:34:26	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 22:21:22	10942	TELM-06	1	Pass	0	0	0	326
M_TCP_Data	m_ARQ019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ019\Hardware\m_ARQ019.exe	2.4	2010/06/08 16:07:58	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:19:56	10169	TELM-06	1	Pass	0	0	0	78
M_TCP_Data	m_ARQ020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ020\Hardware\m_ARQ020.exe	2.2	2009/11/18 15:23:22	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:18:54	7480	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ021\Hardware\m_ARQ021.exe	2.2	2009/11/19 18:43:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:16:50	7487	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ022\Hardware\m_ARQ022.exe	2.3	2010/02/03 16:30:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:15:20	8902	TELM-06	1	Pass	0	0	0	137
M_TCP_Data	m_ARQ023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ023\Hardware\m_ARQ023.exe	2.1	2009/11/10 19:12:36	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:13:34	10562	TELM-06	1	Pass	0	0	0	79
M_TCP_Data	m_ARQ024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ024\Hardware\m_ARQ024.exe	2.3	2010/02/08 18:44:31	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:11:24	11055	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ025\Hardware\m_ARQ025.exe	2.1	2009/09/25 14:20:39	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:08:40	9048	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ026\Hardware\m_ARQ026.exe	2.1	2009/09/25 14:21:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:07:12	8632	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_ARQ027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ027\Hardware\m_ARQ027.exe	2.1	2009/09/25 14:21:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:04:47	9512	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ028\Hardware\m_ARQ028.exe	2.4	2010/04/19 14:36:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:03:37	18794	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_ARQ029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ029\Hardware\m_ARQ029.exe	2.1	2010/03/08 19:53:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:01:39	29519	TELM-06	1	Pass	0	0	0	555
M_TCP_Data	m_ARQ030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ030\Hardware\m_ARQ030.exe	2.5	2010/06/03 15:11:29	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:59:41	18792	TELM-06	1	Pass	0	0	0	355
M_TCP_Data	m_ARQ031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ031\Hardware\m_ARQ031.exe	2.3	2010/06/08 19:57:52	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:58:19	12576	TELM-06	1	Pass	0	0	0	556
M_TCP_Data	m_Data012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data012\Hardware\m_Data012.exe	2.4	2009/11/10 14:18:10	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 03:14:57	36902	TELM-06	1	Pass	0	0	0	18
M_TCP_Data	m_Data013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data013\Hardware\m_Data013.exe	2.3	2010/02/03 15:41:32	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:55:21	104634	TELM-06	1	Pass	0	0	0	335
M_TCP_Data	m_Data014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data014\Hardware\m_Data014.exe	2.3	2010/05/12 19:37:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:51:50	92814	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_Data015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data015\Hardware\m_Data015.exe	2.2	2010/02/03 17:34:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:50:10	9159	TELM-06	1	Pass	0	0	0	14
M_TCP_Data	m_Data016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data016\Hardware\m_Data016.exe	2.2	2010/02/24 20:56:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:47:44	70021	TELM-06	1	Pass	0	0	0	26
M_TCP_Data	m_Data017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data017\Hardware\m_Data017.exe	2.2	2010/02/24 22:22:41	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:43:07	67819	TELM-06	1	Pass	0	0	0	30
M_TCP_Data	m_Data018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data018\Hardware\m_Data018.exe	2.2	2010/02/03 22:48:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:40:57	59042	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_MemoryRead001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead001\Hardware\m_MemoryRead001.exe	2.2	2010/02/01 17:26:47	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:37:15	24620	TELM-06	1	Pass	0	0	0	439
M_TCP_Data	m_MemoryRead002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead002\Hardware\m_MemoryRead002.exe	2.3	2009/11/05 15:03:23	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:34:58	14197	TELM-06	1	Pass	0	0	0	38
M_TCP_Data	m_MemoryRead003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead003\Hardware\m_MemoryRead003.exe	2.4	2010/04/05 13:08:02	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:29:01	19971	TELM-06	1	Pass	0	0	0	320
M_TCP_Data	m_MemoryRead004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead004\Hardware\m_MemoryRead004.exe	2.3	2010/03/19 16:10:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:25:20	8485	TELM-06	1	Pass	0	0	0	1
M_TCP_Data	m_MemoryRead005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead005\Hardware\m_MemoryRead005.exe	2.3	2009/12/07 19:50:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:21:19	48651	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryRead006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead006\Hardware\m_MemoryRead006.exe	2.6	2010/06/29 21:05:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 21:19:09	12392	TELM-06	1	Pass	0	0	0	37
M_TCP_Data	m_MemoryRead007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead007\Hardware\m_MemoryRead007.exe	2.1	2009/11/03 12:42:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:12:37	10914	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_MemoryRead008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead008\Hardware\m_MemoryRead008.exe	2.1	2009/09/30 18:09:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:07:21	8327	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead009\Hardware\m_MemoryRead009.exe	2.1	2009/12/07 19:51:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:04:02	12441	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead010\Hardware\m_MemoryRead010.exe	2.2	2009/09/30 17:31:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:01:40	7738	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead011\Hardware\m_MemoryRead011.exe	2.2	2009/09/30 17:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:59:16	7759	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead012\Hardware\m_MemoryRead012.exe	2.2	2009/09/30 17:33:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:57:59	7849	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead013\Hardware\m_MemoryRead013.exe	2.2	2009/09/30 17:33:27	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:54:21	7915	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead014\Hardware\m_MemoryRead014.exe	2.2	2009/09/30 17:33:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:49:35	7979	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead015\Hardware\m_MemoryRead015.exe	2.2	2009/09/30 17:34:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:46:39	7817	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead016\Hardware\m_MemoryRead016.exe	2.1	2009/10/19 18:24:00	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:43:16	7806	TELM-06	1	Pass	0	0	0	57

M_TCP_Data	m_MemoryRead017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead017\Hardware\m_MemoryRead017.exe	2.2	2009/10/19 15:52:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:38:30	7901	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead019\Hardware\m_MemoryRead019.exe	2.3	2009/12/03 13:57:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:36:20	9623	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead020\Hardware\m_MemoryRead020.exe	2.4	2010/02/09 20:49:06	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:33:48	7113	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryRead021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead021\Hardware\m_MemoryRead021.exe	2.2	2009/12/23 13:34:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:31:56	9293	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryRead030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead030\Hardware\m_MemoryRead030.exe	2.1	2009/10/16 13:54:03	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:26:54	6611	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead031\Hardware\m_MemoryRead031.exe	2.3	2009/12/28 18:23:34	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:24:04	10531	TELM-06	1	Pass	0	0	0	43
M_TCP_Data	m_MemoryRead033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead033\Hardware\m_MemoryRead033.exe	2.3	2009/11/23 18:55:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:21:18	12011	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead034\Hardware\m_MemoryRead034.exe	2.1	2009/10/16 14:15:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:19:45	8504	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead035\Hardware\m_MemoryRead035.exe	2.4	2009/11/23 18:56:23	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:16:19	17822	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead037\Hardware\m_MemoryRead037.exe	2.2	2009/11/23 18:57:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:14:41	12455	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead038\Hardware\m_MemoryRead038.exe	2.3	2010/01/19 17:36:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:13:02	14635	TELM-06	1	Pass	0	0	0	470
M_TCP_Data	m_MemoryRead039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead039\Hardware\m_MemoryRead039.exe	2.1	2009/09/28 15:09:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:11:12	8516	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead040\Hardware\m_MemoryRead040.exe	2.6	2010/01/06 14:10:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:09:16	32367	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryRead041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead041\Hardware\m_MemoryRead041.exe	2.4	2009/11/16 12:52:24	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:07:24	18463	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead042\Hardware\m_MemoryRead042.exe	2.3	2010/06/29 22:24:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:05:08	28226	TELM-06	1	Pass	0	0	0	41
M_TCP_Data	m_MemoryRead043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead043\Hardware\m_MemoryRead043.exe	2.2	2010/02/08 16:55:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:03:33	17748	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead045\Hardware\m_MemoryRead045.exe	2.0	2009/09/22 14:34:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 20:01:43	6823	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead049\Hardware\m_MemoryRead049.exe	2.3	2009/11/16 12:52:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:00:21	8802	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead051\Hardware\m_MemoryRead051.exe	2.1	2009/10/27 19:22:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:58:26	9611	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryRead052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead052\Hardware\m_MemoryRead052.exe	2.3	2010/06/29 23:18:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:56:52	11147	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite001\Hardware\m_MemoryWrite001.exe	2.2	2009/12/07 19:52:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:54:57	12047	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite002\Hardware\m_MemoryWrite002.exe	2.0	2009/09/22 14:48:30	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:52:53	10501	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite003\Hardware\m_MemoryWrite003.exe	2.1	2009/10/28 14:34:41	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:51:05	10032	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite005\Hardware\m_MemoryWrite005.exe	2.2	2009/12/15 19:05:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:48:11	7972	TELM-06	1	Pass	0	0	0	186
M_TCP_Data	m_MemoryWrite006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite006\Hardware\m_MemoryWrite006.exe	2.3	2009/11/24 17:33:37	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:46:44	12177	TELM-06	1	Pass	0	0	0	145
M_TCP_Data	m_MemoryWrite007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite007\Hardware\m_MemoryWrite007.exe	2.1	2009/10/28 15:09:57	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:43:45	7433	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite008\Hardware\m_MemoryWrite008.exe	2.1	2010/02/01 19:11:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:42:28	7465	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite009\Hardware\m_MemoryWrite009.exe	2.1	2009/10/28 15:17:22	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:40:28	7517	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite010\Hardware\m_MemoryWrite010.exe	2.1	2009/10/28 16:00:35	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:38:36	7579	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite011\Hardware\m_MemoryWrite011.exe	2.1	2009/10/28 16:07:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:36:06	7595	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite012\Hardware\m_MemoryWrite012.exe	2.2	2009/10/28 16:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:34:35	12402	TELM-06	1	Pass	0	0	0	120
M_TCP_Data	m_MemoryWrite013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite013\Hardware\m_MemoryWrite013.exe	2.1	2009/10/28 16:21:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:33:07	7571	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite014\Hardware\m_MemoryWrite014.exe	2.1	2009/10/28 16:29:59	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:31:01	7564	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite015\Hardware\m_MemoryWrite015.exe	2.1	2009/10/28 16:36:30	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:28:49	7467	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryWrite016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite016\Hardware\m_MemoryWrite016.exe	2.5	2010/02/01 21:11:41	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:26:59	9586	TELM-06	1	Pass	0	0	0	1246
M_TCP_Data	m_MemoryWrite017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite017\Hardware\m_MemoryWrite017.exe	2.3	2010/01/19 17:37:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:25:42	8667	TELM-06	1	Pass	0	0	0	737
M_TCP_Data	m_MemoryWrite018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite018\Hardware\m_MemoryWrite018.exe	2.1	2009/10/29 20:06:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:23:34	6770	TELM-06	1	Pass	0	0	0	171
M_TCP_Data	m_MemoryWrite019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite019\Hardware\m_MemoryWrite019.exe	2.0	2009/09/22 15:13:38	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:22:05	9564	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryWrite020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite020\Hardware\m_MemoryWrite020.exe	2.2	2009/10/29 19:44:00	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:20:45	7895	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite022\Hardware\m_MemoryWrite022.exe	2.3	2009/11/03 17:28:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:19:36	10019	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite023\Hardware\m_MemoryWrite023.exe	2.4	2010/02/04 16:15:17	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:17:52	20709	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite024\Hardware\m_MemoryWrite024.exe	2.2	2009/11/24 17:35:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:15:46	7388	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_MemoryWrite025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite025\Hardware\m_MemoryWrite025.exe	2.1	2009/10/29 18:47:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:13:53	11030	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite029\Hardware\m_MemoryWrite029.exe	2.2	2010/02/01 21:00:35	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:11:02	13914	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite032\Hardware\m_MemoryWrite032.exe	2.5	2010/03/15 01:08:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:09:30	9701	TELM-06	1	Pass	0	0	0	243
M_TCP_Data	m_MemoryWrite038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite038\Hardware\m_MemoryWrite038.exe	2.1	2009/10/29 18:42:13	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:03:42	8154	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite039\Hardware\m_MemoryWrite039.exe	2.1	2010/02/10 18:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:00:36	9778	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite040\Hardware\m_MemoryWrite040.exe	2.2	2010/02/10 17:56:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:59:27	9891	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryWrite041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite041\Hardware\m_MemoryWrite041.exe	2.2	2010/02/01 20:45:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:58:06	13812	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite042\Hardware\m_MemoryWrite042.exe	2.2	2010/01/19 17:38:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:53:57	10266	TELM-06	1	Pass	0	0	0	469
M_TCP_Data	m_MemoryWrite046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite046\Hardware\m_MemoryWrite046.exe	2.3	2010/02/04 16:51:09	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:50:56	7519	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite047\Hardware\m_MemoryWrite047.exe	2.2	2009/11/11 19:12:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:49:21	8570	TELM-06	1	Pass	0	0	0	98
M_TCP_Data	m_MemoryWrite048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite048\Hardware\m_MemoryWrite048.exe	2.2	2009/11/11 18:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:43:47	9366	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite049\Hardware\m_MemoryWrite049.exe	2.5	2010/02/10 16:51:26	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 18:38:34	21482	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_MemoryWrite052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite052\Hardware\m_MemoryWrite052.exe	2.3	2009/12/23 13:35:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:36:16	11025	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite053\Hardware\m_MemoryWrite053.exe	2.2	2009/11/11 17:01:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:35:21	8575	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite054\Hardware\m_MemoryWrite054.exe	2.6	2010/06/30 17:58:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 18:34:01	10199	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryWrite055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite055\Hardware\m_MemoryWrite055.exe	2.2	2009/11/11 18:04:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:28:18	7518	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite056	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite056\Hardware\m_MemoryWrite056.exe	2.4	2010/02/10 16:32:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:26:20	8784	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode001\Hardware\m_TransferMode001.exe	2.8	2010/05/14 18:48:57	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 18:21:56	18180	TELM-06	1	Pass	0	0	0	4748
M_TCP_Data	m_TransferMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode002\Hardware\m_TransferMode002.exe	2.4	2010/07/02 13:00:52	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:16:46	10017	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_TransferMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode003\Hardware\m_TransferMode003.exe	2.0	2009/11/16 20:59:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 18:12:52	7323	TELM-06	1	Pass	0	0	0	129
M_TCP_Data	m_TransferMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode004\Hardware\m_TransferMode004.exe	2.3	2010/01/08 17:07:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:11:04	18092	TELM-06	1	Pass	0	0	0	197
M_TCP_Data	m_TransferMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode005\Hardware\m_TransferMode005.exe	2.7	2010/07/02 13:36:13	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 18:06:25	36275	TELM-06	1	Pass	0	0	0	273
M_TCP_Data	m_TransferMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode006\Hardware\m_TransferMode006.exe	2.4	2010/04/16 11:58:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:05:13	7255	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode007\Hardware\m_TransferMode007.exe	2.2	2009/10/12 15:46:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:02:49	11895	TELM-06	1	Pass	0	0	0	112
M_TCP_Data	m_TransferMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode008\Hardware\m_TransferMode008.exe	2.0	2009/09/09 15:28:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 18:00:55	10745	TELM-06	1	Pass	0	0	0	76
M_TCP_Data	m_TransferMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode009\Hardware\m_TransferMode009.exe	2.1	2009/10/13 16:25:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:58:29	11522	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_TransferMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode010\Hardware\m_TransferMode010.exe	2.1	2009/10/13 16:26:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:52:50	11342	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode011\Hardware\m_TransferMode011.exe	2.3	2010/02/10 13:37:08	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:45:31	8026	TELM-06	1	Pass	0	0	0	72
M_TCP_Data	m_TransferMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode012\Hardware\m_TransferMode012.exe	2.3	2010/02/10 13:38:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:43:57	8110	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_TransferMode013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode013\Hardware\m_TransferMode013.exe	2.3	2010/02/10 13:39:16	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:40:47	8048	TELM-06	1	Pass	0	0	0	44
M_TCP_Data	m_TransferMode014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode014\Hardware\m_TransferMode014.exe	2.3	2010/02/10 13:40:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:39:40	8204	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_TransferMode015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode015\Hardware\m_TransferMode015.exe	2.5	2010/05/26 14:56:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:38:08	7426	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_TransferMode016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode016\Hardware\m_TransferMode016.exe	2.8	2010/07/02 13:14:11	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 17:34:40	29913	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode017\Hardware\m_TransferMode017.exe	2.7	2010/05/18 14:47:09	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 17:32:26	43733	TELM-06	1	Pass	0	0	0	257
M_TCP_Data	m_TransferMode018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode018\Hardware\m_TransferMode018.exe	2.6	2010/04/16 14:06:26	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:30:33	26467	TELM-06	1	Pass	0	0	0	509
M_TCP_Data	m_TransferMode019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode019\Hardware\m_TransferMode019.exe	2.6	2010/03/19 18:58:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:28:51	22195	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics001\Hardware\m_Diagnostics001.exe	2.5	2009/12/30 17:45:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:32:55	9645	TELM-06	1	Pass	0	0	0	166
M_TCP_Diagnostic	m_Diagnostics002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics002\Hardware\m_Diagnostics002.exe	2.2	2009/11/23 20:15:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:28:25	7717	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics003\Hardware\m_Diagnostics003.exe	2.2	2010/03/14 20:01:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:27:30	6827	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics004\Hardware\m_Diagnostics004.exe	2.3	2009/12/03 13:22:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:26:06	16001	TELM-06	1	Pass	0	0	0	66
M_TCP_Diagnostic	m_Diagnostics005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics005\Hardware\m_Diagnostics005.exe	2.1	2009/10/12 14:14:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:24:36	7796	TELM-06	1	Pass	0	0	0	13
M_TCP_Diagnostic	m_Diagnostics006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics006\Hardware\m_Diagnostics006.exe	2.0	2009/09/22 17:02:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:21:05	14890	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics007\Hardware\m_Diagnostics007.exe	2.1	2009/09/28 13:31:55	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:19:26	15458	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics008\Hardware\m_Diagnostics008.exe	2.0	2009/09/22 17:03:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:16:57	21513	TELM-06	1	Pass	0	0	0	31
M_TCP_Diagnostic	m_Diagnostics009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics009\Hardware\m_Diagnostics009.exe	2.5	2010/06/25 17:09:14	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:15:25	16011	TELM-06	1	Pass	0	0	0	23
M_TCP_Diagnostic	m_Diagnostics010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics010\Hardware\m_Diagnostics010.exe	2.5	2010/07/20 19:18:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:13:10	10583	TELM-06	1	Pass	0	0	0	10
M_TCP_Diagnostic	m_Diagnostics011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics011\Hardware\m_Diagnostics011.exe	2.4	2009/12/23 13:29:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:10:11	29884	TELM-06	1	Pass	0	0	0	19
M_TCP_Diagnostic	m_Diagnostics012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics012\Hardware\m_Diagnostics012.exe	2.3	2010/04/16 14:20:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:07:58	17157	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics013\Hardware\m_Diagnostics013.exe	2.5	2010/06/08 17:29:50	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:05:51	20541	TELM-06	1	Pass	0	0	0	15
M_TCP_Diagnostic	m_Diagnostics014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics014\Hardware\m_Diagnostics014.exe	2.4	2010/03/14 17:27:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:03:57	17940	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics015\Hardware\m_Diagnostics015.exe	2.4	2010/01/19 17:35:47	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:01:59	25463	TELM-06	1	Pass	0	0	0	29
M_TCP_Diagnostic	m_Diagnostics016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics016\Hardware\m_Diagnostics016.exe	2.0	2009/09/22 17:09:46	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:56:53	5821	TELM-06	1	Pass	0	0	0	1
M_TCP_Diagnostic	m_Diagnostics017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics017\Hardware\m_Diagnostics017.exe	2.3	2009/11/23 20:15:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:53:15	24951	TELM-06	1	Pass	0	0	0	32
M_TCP_Diagnostic	m_Diagnostics018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics018\Hardware\m_Diagnostics018.exe	2.2	2009/12/23 13:30:36	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:50:45	11955	TELM-06	1	Pass	0	0	0	9
M_TCP_Diagnostic	m_Diagnostics019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics019\Hardware\m_Diagnostics019.exe	2.1	2009/10/13 12:56:53	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:46:48	7879	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics020\Hardware\m_Diagnostics020.exe	2.2	2009/11/23 20:16:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:44:26	10314	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics021\Hardware\m_Diagnostics021.exe	2.1	2009/10/13 13:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:42:53	6016	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics022\Hardware\m_Diagnostics022.exe	2.4	2010/02/03 17:09:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:41:08	12724	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics023\Hardware\m_Diagnostics023.exe	2.2	2010/02/03 14:24:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:38:28	39153	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics024\Hardware\m_Diagnostics024.exe	2.5	2010/02/03 14:04:47	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:35:58	15795	TELM-06	1	Pass	0	0	0	20
M_TCP_Diagnostic	m_Diagnostics026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics026\Hardware\m_Diagnostics026.exe	2.1	2009/10/15 19:32:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:33:10	6055	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics027\Hardware\m_Diagnostics027.exe	2.2	2009/11/02 13:03:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:31:00	3055	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics028\Hardware\m_Diagnostics028.exe	2.4	2009/11/16 20:37:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:29:51	11438	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics029\Hardware\m_Diagnostics029.exe	2.1	2009/10/15 19:46:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:28:39	2845	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics030\Hardware\m_Diagnostics030.exe	2.5	2010/07/20 18:09:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:26:52	7602	TELM-06	1	Pass	0	0	0	277
M_TCP_Diagnostic	m_Diagnostics031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics031\Hardware\m_Diagnostics031.exe	2.2	2009/11/11 14:33:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:25:35	10822	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics032\Hardware\m_Diagnostics032.exe	2.3	2009/12/23 13:31:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:24:10	9888	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics034\Hardware\m_Diagnostics034.exe	2.4	2009/11/16 20:30:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:22:24	9066	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics035\Hardware\m_Diagnostics035.exe	2.2	2009/12/23 13:31:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:21:00	8016	TELM-06	1	Pass	0	0	0	7

M_TCP_Diagnostic	m_Diagnostics036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics036\Hardware\m_Diagnostics036.exe	2.3	2010/03/13 01:44:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:19:23	8499	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics037\Hardware\m_Diagnostics037.exe	2.4	2010/03/13 01:23:35	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:18:07	7030	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics038\Hardware\m_Diagnostics038.exe	2.3	2010/06/07 20:16:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:16:15	49201	TELM-06	1	Pass	0	0	0	18
M_TCP_Discover	m_DiscoverFull001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull001\Hardware\m_DiscoverFull001.exe	2.8	2010/05/21 19:22:39	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 16:03:35	44629	TELM-06	1	Pass	0	0	0	150
M_TCP_Discover	m_DiscoverFull002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull002\Hardware\m_DiscoverFull002.exe	2.6	2010/03/19 18:02:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:55:47	144662	TELM-06	1	Pass	0	0	0	1815
M_TCP_Discover	m_DiscoverFull003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull003\Hardware\m_DiscoverFull003.exe	2.2	2010/02/25 20:00:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:53:17	12106	TELM-06	1	Pass	0	0	0	58
M_TCP_Discover	m_DiscoverFull004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull004\Hardware\m_DiscoverFull004.exe	2.4	2010/02/05 21:19:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 11:00:26	271110	TELM-06	1	Pass	0	0	0	3713
M_TCP_Discover	m_DiscoverFull005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull005\Hardware\m_DiscoverFull005.exe	2.2	2009/10/29 19:18:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:50:19	17256	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull006\Hardware\m_DiscoverFull006.exe	2.3	2010/02/05 21:19:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:42:25	130868	TELM-06	1	Pass	0	0	0	1727
M_TCP_Discover	m_DiscoverFull007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull007\Hardware\m_DiscoverFull007.exe	2.7	2009/12/14 21:43:07	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 15:48:15	13709	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull009\Hardware\m_DiscoverFull009.exe	2.3	2010/01/21 21:02:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:42:30	10997	TELM-06	1	Pass	0	0	0	65
M_TCP_Discover	m_DiscoverFull010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull010\Hardware\m_DiscoverFull010.exe	2.4	2010/05/12 17:07:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:40:22	8094	TELM-06	1	Pass	0	0	0	121
M_TCP_Discover	m_DiscoverFull011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull011\Hardware\m_DiscoverFull011.exe	2.3	2009/11/02 22:03:01	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:37:34	56722	TELM-06	1	Pass	0	0	0	9
M_TCP_Discover	m_DiscoverFull012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull012\Hardware\m_DiscoverFull012.exe	2.8	2010/06/29 12:07:18	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 12:21:41	366842	TELM-06	1	Pass	0	0	0	33
M_TCP_Discover	m_DiscoverFull013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull013\Hardware\m_DiscoverFull013.exe	2.2	2010/03/18 16:03:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:34:21	42764	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull014\Hardware\m_DiscoverFull014.exe	2.2	2010/03/18 16:03:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:31:37	24944	TELM-06	1	Pass	0	0	0	15
M_TCP_Discover	m_DiscoverFull015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull015\Hardware\m_DiscoverFull015.exe	2.1	2009/11/23 16:09:49	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:29:21	7995	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull016\Hardware\m_DiscoverFull016.exe	2.1	2009/11/03 20:58:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:28:09	8405	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull017\Hardware\m_DiscoverFull017.exe	2.1	2009/11/03 20:58:47	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:26:30	12175	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull022\Hardware\m_DiscoverFull022.exe	2.1	2009/11/23 16:11:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:25:22	8269	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull023\Hardware\m_DiscoverFull023.exe	2.1	2009/11/23 16:11:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:22:52	11491	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull024\Hardware\m_DiscoverFull024.exe	2.1	2009/11/23 16:12:04	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:21:36	12574	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull025\Hardware\m_DiscoverFull025.exe	2.1	2009/11/23 16:12:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:20:29	9056	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull029\Hardware\m_DiscoverFull029.exe	2.0	2009/11/13 21:27:59	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:19:22	12479	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull030\Hardware\m_DiscoverFull030.exe	2.3	2010/05/28 11:39:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:16:33	19316	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull031\Hardware\m_DiscoverFull031.exe	2.3	2010/05/28 12:38:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:14:40	35308	TELM-06	1	Pass	0	0	0	22
M_TCP_Discover	m_DiscoverFull032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull032\Hardware\m_DiscoverFull032.exe	2.0	2009/11/13 21:29:43	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:11:12	9172	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull033\Hardware\m_DiscoverFull033.exe	2.0	2009/11/13 21:30:18	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:09:34	9446	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull034\Hardware\m_DiscoverFull034.exe	2.2	2010/03/18 16:04:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:08:02	11166	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull035\Hardware\m_DiscoverFull035.exe	2.2	2010/02/09 20:53:53	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 23:56:25	43612	TELM-06	1	Pass	0	0	0	31
M_TCP_Discover	m_DiscoverFull036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull036\Hardware\m_DiscoverFull036.exe	2.2	2010/02/11 14:31:48	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 23:24:54	22005	TELM-06	1	Pass	0	0	0	38
M_TCP_Discover	m_DiscoverFull037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull037\Hardware\m_DiscoverFull037.exe	2.4	2010/02/03 21:18:36	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 03:08:40	35135	TELM-06	1	Pass	0	0	0	4
M_TCP_Discover	m_DiscoverFull038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull038\Hardware\m_DiscoverFull038.exe	2.1	2009/12/01 20:49:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:03:50	141702	TELM-06	1	Pass	0	0	0	7
M_TCP_EEPROM	m_Eeprom001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom001\Hardware\m_Eeprom001.exe	2.3	2009/12/23 16:23:59	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 21:32:33	11085	TELM-06	1	Pass	0	0	0	22
M_TCP_EEPROM	m_Eeprom002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom002\Hardware\m_Eeprom002.exe	2.3	2009/12/18 21:07:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 15:45:09	10794	TELM-06	1	Pass	0	0	0	938
M_TCP_EEPROM	m_Eeprom003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom003\Hardware\m_Eeprom003.exe	2.2	2010/01/05 18:03:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:59:12	16705	TELM-06	1	Pass	0	0	0	702
M_TCP_EEPROM	m_Eeprom004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom004\Hardware\m_Eeprom004.exe	2.2	2010/01/25 19:26:31	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:55:48	112602	TELM-06	1	Pass	0	0	0	583
M_TCP_EEPROM	m_Eeprom005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom005\Hardware\m_Eeprom005.exe	2.2	2010/01/19 15:32:05	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:51:29	174270	TELM-06	1	Pass	0	0	0	704
M_TCP_EEPROM	m_Eeprom006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom006\Hardware\m_Eeprom006.exe	2.2	2010/05/28 12:55:48	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:49:25	47255	TELM-06	1	Pass	0	0	0	6
M_TCP_EEPROM	m_Eeprom007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom007\Hardware\m_Eeprom007.exe	2.1	2010/02/18 18:38:33	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:48:09	4999	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency001\Hardware\m_Emergency001.exe	2.3	2009/11/03 21:54:33	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:13:15	9464	TELM-06	1	Pass	0	0	0	100
M_TCP_Emergency	m_Emergency002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency002\Hardware\m_Emergency002.exe	2.5	2010/06/25 20:19:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:11:03	18875	TELM-06	1	Pass	0	0	0	45
M_TCP_Emergency	m_Emergency003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency003\Hardware\m_Emergency003.exe	2.6	2010/06/29 14:19:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 11:07:05	28335	TELM-06	1	Pass	0	0	0	1220
M_TCP_Emergency	m_Emergency004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency004\Hardware\m_Emergency004.exe	2.3	2010/02/05 21:22:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:02:42	15022	TELM-06	1	Pass	0	0	0	4
M_TCP_Emergency	m_Emergency005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency005\Hardware\m_Emergency005.exe	2.2	2009/11/10 00:31:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:58:10	8923	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency006\Hardware\m_Emergency006.exe	2.1	2009/10/16 16:44:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:55:06	9307	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency007\Hardware\m_Emergency007.exe	2.1	2009/10/16 16:44:28	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:51:25	9557	TELM-06	1	Pass	0	0	0	33
M_TCP_Emergency	m_Emergency009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency009\Hardware\m_Emergency009.exe	2.1	2009/10/08 19:54:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:45:30	6034	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency010\Hardware\m_Emergency010.exe	2.2	2010/06/08 15:56:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:40:29	13168	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency011\Hardware\m_Emergency011.exe	2.0	2009/10/21 16:25:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 10:37:35	6743	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency012\Hardware\m_Emergency012.exe	2.1	2009/11/11 13:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:35:23	8697	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency013\Hardware\m_Emergency013.exe	2.1	2009/11/11 14:02:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:33:29	12719	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency014\Hardware\m_Emergency014.exe	2.3	2009/12/23 13:33:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:31:19	8458	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency015\Hardware\m_Emergency015.exe	2.3	2010/06/08 15:56:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:30:08	8878	TELM-06	1	Pass	0	0	0	8
M_TCP_Emergency	m_Emergency016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency016\Hardware\m_Emergency016.exe	2.2	2009/12/21 21:03:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:27:11	13725	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency017\Hardware\m_Emergency017.exe	2.2	2009/11/16 12:31:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:22:33	10318	TELM-06	1	Pass	0	0	0	21
M_TCP_Emergency	m_Emergency018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency018\Hardware\m_Emergency018.exe	2.3	2010/06/29 15:32:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:17:46	8197	TELM-06	1	Pass	0	0	0	22
M_TCP_Emergency	m_Emergency019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency019\Hardware\m_Emergency019.exe	2.2	2010/01/07 19:28:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:15:50	11236	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency020\Hardware\m_Emergency020.exe	2.6	2010/03/18 17:47:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 10:11:48	12428	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency021\Hardware\m_Emergency021.exe	2.3	2009/12/21 19:27:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:08:30	10929	TELM-06	1	Pass	0	0	0	83
M_TCP_Emergency	m_Emergency022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency022\Hardware\m_Emergency022.exe	2.2	2010/01/07 18:54:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:06:36	10833	TELM-06	1	Pass	0	0	0	82
M_TCP_Emergency	m_Emergency038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency038\Hardware\m_Emergency038.exe	2.2	2010/01/07 19:42:11	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:02:00	10687	TELM-06	1	Pass	0	0	0	14
M_TCP_Emergency	m_Emergency039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency039\Hardware\m_Emergency039.exe	2.2	2010/02/03 17:51:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:56:16	15470	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency040\Hardware\m_Emergency040.exe	2.0	2009/09/29 14:53:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 09:53:50	13974	TELM-06	1	Pass	0	0	0	7
M_TCP_Emergency	m_Emergency041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency041\Hardware\m_Emergency041.exe	2.0	2009/09/29 14:54:05	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 09:52:04	5000	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency042\Hardware\m_Emergency042.exe	2.0	2009/09/29 14:54:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 09:49:11	5003	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency043\Hardware\m_Emergency043.exe	2.4	2010/04/21 17:13:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 09:46:46	10431	TELM-06	1	Pass	0	0	0	9
M_TCP_Emergency	m_Emergency044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency044\Hardware\m_Emergency044.exe	2.5	2009/11/10 14:19:43	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 00:44:17	36899	TELM-06	1	Pass	0	0	0	18
M_TCP_Emergency	m_Emergency045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency045\Hardware\m_Emergency045.exe	2.3	2010/05/11 12:18:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:45:53	11194	TELM-06	1	Pass	0	0	0	16
M_TCP_General	m_General003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General003\Hardware\m_General003.exe	2.2	2009/10/13 14:50:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:03:21	67566	TELM-06	1	Pass	0	0	0	44576
M_TCP_General	m_General005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General005\Hardware\m_General005.exe	2.4	2009/12/01 19:02:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:59:13	67213	TELM-06	1	Pass	0	0	0	14863
M_TCP_General	m_General015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General015\Hardware\m_General015.exe	2.3	2009/12/01 19:02:59	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:56:20	35349	TELM-06	1	Pass	0	0	0	62008
M_TCP_General	m_General016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General016\Hardware\m_General016.exe	2.4	2010/03/29 10:43:22	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:54:27	8633	TELM-06	1	Pass	0	0	0	2
M_TCP_General	m_General017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General017\Hardware\m_General017.exe	2.3	2009/12/01 19:03:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:52:31	7307	TELM-06	1	Pass	0	0	0	10
M_TCP_General	m_General018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General018\Hardware\m_General018.exe	2.2	2010/02/03 18:07:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:48:51	29844	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General027\Hardware\m_General027.exe	2.7	2010/06/07 11:39:47	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 17:42:56	802189	TELM-06	1	Pass	0	0	0	21
M_TCP_General	m_General028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General028\Hardware\m_General028.exe	2.5	2010/06/08 11:48:52	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 01:06:56	635666	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General029\Hardware\m_General029.exe	2.3	2010/06/07 12:27:26	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:00:15	274278	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General030\Hardware\m_General030.exe	2.1	2010/06/07 12:30:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:46:29	17156	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General031\Hardware\m_General031.exe	2.5	2010/05/11 12:33:53	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:43:47	58081	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General032\Hardware\m_General032.exe	2.2	2010/02/05 18:56:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:41:11	29609	TELM-06	1	Pass	0	0	0	11
M_TCP_General	m_General033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General033\Hardware\m_General033.exe	2.0	2010/02/04 17:42:57	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:38:45	20787	TELM-06	1	Pass	0	0	0	4
M_TCP_General	m_General034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General034\Hardware\m_General034.exe	2.0	2010/01/22 22:45:34	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:34:59	30312	TELM-06	1	Pass	0	0	0	2
M_TCP_Initialize	m_Initialize001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize001\Hardware\m_Initialize001.exe	2.8	2010/05/12 12:34:35	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 16:43:37	10372	TELM-06	1	Pass	0	0	0	9
M_TCP_Initialize	m_Initialize002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize002\Hardware\m_Initialize002.exe	2.2	2009/10/20 12:16:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:37:14	19971	TELM-06	1	Pass	0	0	0	11
M_TCP_Initialize	m_Initialize003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize003\Hardware\m_Initialize003.exe	2.4	2010/06/08 15:57:16	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:33:35	24085	TELM-06	1	Pass	0	0	0	6
M_TCP_Initialize	m_Initialize004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize004\Hardware\m_Initialize004.exe	2.4	2010/06/08 15:57:44	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:30:59	35058	TELM-06	1	Pass	0	0	0	30
M_TCP_MedicalEvent	m_MedicalEvent001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent001\Hardware\m_MedicalEvent001.exe	2.4	2010/01/18 15:42:53	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:02:58	12569	TELM-06	1	Pass	0	0	0	36
M_TCP_MedicalEvent	m_MedicalEvent002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent002\Hardware\m_MedicalEvent002.exe	2.6	2010/02/09 19:03:39	2.6	Up-to-date	Baseline 17_1	None	09/13/2010 07:42:18	73275	TELM-06	1	Pass	0	0	0	23
M_TCP_MedicalEvent	m_MedicalEvent003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent003\Hardware\m_MedicalEvent003.exe	2.4	2010/03/11 21:16:02	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:54:26	18705	TELM-06	1	Pass	0	0	0	24
M_TCP_MedicalEvent	m_MedicalEvent004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent004\Hardware\m_MedicalEvent004.exe	2.4	2010/01/18 15:44:34	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:52:34	12149	TELM-06	1	Pass	0	0	0	12
M_TCP_MedicalEvent	m_MedicalEvent008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent008\Hardware\m_MedicalEvent008.exe	2.4	2010/01/18 15:44:59	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:50:56	8315	TELM-06	1	Pass	0	0	0	6
M_TCP_MedicalEvent	m_MedicalEvent009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent009\Hardware\m_MedicalEvent009.exe	2.2	2010/01/18 15:46:20	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:48:20	7520	TELM-06	1	Pass	0	0	0	4
M_TCP_MedicalEvent	m_MedicalEvent010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent010\Hardware\m_MedicalEvent010.exe	2.5	2010/01/27 14:30:44	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:46:12	16543	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent011\Hardware\m_MedicalEvent011.exe	2.2	2010/01/18 15:47:08	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:44:20	15138	TELM-06	1	Pass	0	0	0	11
M_TCP_MedicalEvent	m_MedicalEvent012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent012\Hardware\m_MedicalEvent012.exe	2.3	2010/01/18 15:47:55	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:41:38	6163	TELM-06	1	Pass	0	0	0	3
M_TCP_MedicalEvent	m_MedicalEvent014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent014\Hardware\m_MedicalEvent014.exe	2.5	2010/09/08 13:20:31	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:35:41	15331	TELM-06	1	Pass	0	0	0	18
M_TCP_MedicalEvent	m_MedicalEvent015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent015\Hardware\m_MedicalEvent015.exe	2.9	2010/09/07 20:16:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 09:40:54	56572	TELM-06	1	Pass	0	0	0	152
M_TCP_MedicalEvent	m_MedicalEvent016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent016\Hardware\m_MedicalEvent016.exe	2.9	2010/05/12 18:44:37	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 14:53:52	283958	TELM-06	1	Pass	0	0	0	375
M_TCP_MedicalEvent	m_MedicalEvent017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent017\Hardware\m_MedicalEvent017.exe	2.8	2010/09/07 20:17:30	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 21:07:51	612272	TELM-06	1	Pass	0	0	0	13
M_TCP_MedicalEvent	m_MedicalEvent018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent018\Hardware\m_MedicalEvent018.exe	2.5	2010/01/18 15:50:21	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 14:25:01	272211	TELM-06	1	Pass	0	0	0	4627
M_TCP_MedicalEvent	m_MedicalEvent019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent019\Hardware\m_MedicalEvent019.exe	2.5	2010/01/18 15:50:59	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:30:32	7376	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent020\Hardware\m_MedicalEvent020.exe	2.3	2010/01/18 15:51:30	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:27:14	40718	TELM-06	1	Pass	0	0	0	9
M_TCP_MemoryTest	m_MemoryTest001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest001\Hardware\m_MemoryTest001.exe	2.7	2010/06/18 12:28:23	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 00:32:24	32891	TELM-06	1	Pass	0	0	0	33
M_TCP_MemoryTest	m_MemoryTest002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest002\Hardware\m_MemoryTest002.exe	2.4	2010/05/05 19:48:21	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:20:50	240719	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest003\Hardware\m_MemoryTest003.exe	2.3	2010/05/05 20:08:58	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 13:51:36	479601	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest004\Hardware\m_MemoryTest004.exe	2.3	2010/05/05 20:09:49	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:13:58	240101	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest005\Hardware\m_MemoryTest005.exe	2.3	2010/05/05 20:10:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 13:20:15	478680	TELM-06	1	Pass	0	0	0	75
M_TCP_Network	m_Network001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network001\Hardware\m_Network001.exe	2.6	2010/05/21 15:45:43	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:21:34	24460	TELM-06	1	Pass	0	0	0	1461
M_TCP_Network	m_Network002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network002\Hardware\m_Network002.exe	2.6	2010/05/12 12:01:17	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 18:20:30	34795	TELM-06	1	Pass	0	0	0	82
M_TCP_Network	m_Network003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network003\Hardware\m_Network003.exe	2.1	2009/12/16 22:27:06	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:18:54	64777	TELM-06	1	Pass	0	0	0	86

M_TCP_Network	m_Network004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network004\Hardware\m_Network004.exe	2.1	2010/04/21 17:14:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:17:20	3053	TELM-06	1	Pass	0	0	0	6
M_TCP_Network	m_Network005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network005\Hardware\m_Network005.exe	2.3	2010/05/12 12:02:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:14:28	58136	TELM-06	1	Pass	0	0	0	50
M_TCP_Network	m_Network006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network006\Hardware\m_Network006.exe	2.5	2010/05/21 16:49:56	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:12:16	15243	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network007\Hardware\m_Network007.exe	2.5	2010/05/21 17:09:04	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:10:22	15431	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network008\Hardware\m_Network008.exe	2.5	2010/05/21 17:13:03	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:08:26	10251	TELM-06	1	Pass	0	0	0	1380
M_TCP_Network	m_Network009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network009\Hardware\m_Network009.exe	2.5	2010/05/21 17:21:30	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:05:18	10279	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network010\Hardware\m_Network010.exe	2.5	2010/05/21 17:34:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:02:54	10396	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network011\Hardware\m_Network011.exe	2.3	2010/05/12 12:03:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:57:29	209569	TELM-06	1	Pass	0	0	0	122
M_TCP_Network	m_Network013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network013\Hardware\m_Network013.exe	2.5	2010/07/01 13:00:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:53:28	141619	TELM-06	1	Pass	0	0	0	392
M_TCP_Network	m_Network014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network014\Hardware\m_Network014.exe	2.3	2010/05/12 12:06:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:50:36	26768	TELM-06	1	Pass	0	0	0	12
M_TCP_Network	m_Network015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network015\Hardware\m_Network015.exe	2.3	2010/05/24 13:53:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:48:26	35589	TELM-06	1	Pass	0	0	0	25
M_TCP_Network	m_Network016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network016\Hardware\m_Network016.exe	2.1	2010/05/12 12:09:43	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:46:22	55466	TELM-06	1	Pass	0	0	0	262
M_TCP_Network	m_Network017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network017\Hardware\m_Network017.exe	2.4	2010/06/07 13:19:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:43:38	12793	TELM-06	1	Pass	0	0	0	41
M_TCP_Network	m_Network018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network018\Hardware\m_Network018.exe	2.5	2010/06/07 13:14:52	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:40:58	21760	TELM-06	1	Pass	0	0	0	11
M_TCP_Network	m_Network019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network019\Hardware\m_Network019.exe	2.2	2010/05/12 12:12:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:38:46	56611	TELM-06	1	Pass	0	0	0	51
M_TCP_Network	m_Network020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network020\Hardware\m_Network020.exe	2.2	2010/05/12 12:14:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:35:24	60381	TELM-06	1	Pass	0	0	0	67
M_TCP_Network	m_Network021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network021\Hardware\m_Network021.exe	2.2	2010/05/12 12:16:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:31:33	60337	TELM-06	1	Pass	0	0	0	133
M_TCP_Network	m_Network022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network022\Hardware\m_Network022.exe	2.2	2010/05/12 12:17:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:26:48	60306	TELM-06	1	Pass	0	0	0	175
M_TCP_Network	m_Network023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network023\Hardware\m_Network023.exe	2.1	2010/05/12 12:18:23	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:23:38	19624	TELM-06	1	Pass	0	0	0	454
M_TCP_Network	m_Network024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network024\Hardware\m_Network024.exe	2.2	2010/05/24 14:41:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:21:06	47695	TELM-06	1	Pass	0	0	0	46
M_TCP_Network	m_Network025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network025\Hardware\m_Network025.exe	2.1	2010/05/12 12:20:01	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:17:42	19527	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network026\Hardware\m_Network026.exe	2.3	2010/05/12 12:20:56	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:13:18	20608	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network027\Hardware\m_Network027.exe	2.1	2010/05/12 12:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:10:44	20936	TELM-06	1	Pass	0	0	0	7
M_TCP_Network	m_Network028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network028\Hardware\m_Network028.exe	2.1	2010/06/07 13:22:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:07:58	14406	TELM-06	1	Pass	0	0	0	22
M_TCP_Network	m_Network029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network029\Hardware\m_Network029.exe	2.4	2010/02/09 21:21:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 11:36:22	271777	TELM-06	1	Pass	0	0	0	48
M_TCP_Open	m_Open001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open001\Hardware\m_Open001.exe	2.11	2010/05/20 13:50:26	2.11	Up-to-date	Baseline 17_1	None	09/09/2010 15:02:12	31657	TELM-06	1	Pass	0	0	0	335
M_TCP_Open	m_Open002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open002\Hardware\m_Open002.exe	2.3	2010/01/18 15:52:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:59:00	18754	TELM-06	1	Pass	0	0	0	31
M_TCP_Open	m_Open003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open003\Hardware\m_Open003.exe	2.0	2009/10/15 19:09:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:57:44	6445	TELM-06	1	Pass	0	0	0	3
M_TCP_Open	m_Open004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open004\Hardware\m_Open004.exe	2.6	2010/06/22 20:01:21	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:55:57	13310	TELM-06	1	Pass	0	0	0	21
M_TCP_Open	m_Open005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open005\Hardware\m_Open005.exe	2.3	2010/06/22 17:40:45	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:54:41	16844	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open006\Hardware\m_Open006.exe	2.2	2010/03/15 14:26:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:53:18	21151	TELM-06	1	Pass	0	0	0	1173
M_TCP_Open	m_Open007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open007\Hardware\m_Open007.exe	2.7	2010/03/15 14:46:20	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 14:51:32	17688	TELM-06	1	Pass	0	0	0	1177
M_TCP_Open	m_Open008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open008\Hardware\m_Open008.exe	2.1	2009/10/30 14:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:50:06	7821	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open009\Hardware\m_Open009.exe	2.7	2010/06/16 00:36:52	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 14:48:47	12456	TELM-06	1	Pass	0	0	0	1268
M_TCP_Open	m_Open010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open010\Hardware\m_Open010.exe	2.6	2010/03/15 15:05:50	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:46:21	17335	TELM-06	1	Pass	0	0	0	3965
M_TCP_Open	m_Open011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open011\Hardware\m_Open011.exe	2.2	2010/02/10 13:31:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:45:05	6360	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open012\Hardware\m_Open012.exe	2.4	2010/02/26 17:01:14	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:43:01	15595	TELM-06	1	Pass	0	0	0	10
M_TCP_Open	m_Open013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open013\Hardware\m_Open013.exe	2.5	2010/02/10 13:34:06	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:41:16	9370	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open014\Hardware\m_Open014.exe	2.6	2010/02/10 13:35:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:39:32	8462	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open015\Hardware\m_Open015.exe	2.0	2010/01/28 19:00:26	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:37:38	9157	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open030\Hardware\m_Open030.exe	2.1	2010/05/10 13:57:56	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 02:49:26	58213	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open031\Hardware\m_Open031.exe	2.7	2009/11/19 21:48:20	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 02:13:02	63850	TELM-06	1	Pass	0	0	0	13
M_TCP_Open	m_Open032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open032\Hardware\m_Open032.exe	2.3	2010/02/03 22:17:27	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 23:04:20	32380	TELM-06	1	Pass	0	0	0	18
M_TCP_Open	m_Open033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open033\Hardware\m_Open033.exe	2.3	2010/02/03 22:32:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 22:44:16	9947	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open034\Hardware\m_Open034.exe	2.2	2010/02/17 14:19:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:34:26	57045	TELM-06	1	Pass	0	0	0	42
M_TCP_Open	m_Open035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open035\Hardware\m_Open035.exe	2.2	2010/02/17 14:10:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:32:02	9887	TELM-06	1	Pass	0	0	0	32
M_TCP_Open	m_Open036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open036\Hardware\m_Open036.exe	2.1	2009/10/21 15:52:46	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 01:43:06	36224	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open037\Hardware\m_Open037.exe	2.2	2009/10/21 15:51:07	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 01:35:53	31278	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open038\Hardware\m_Open038.exe	2.7	2010/02/17 14:00:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 14:28:54	78218	TELM-06	1	Pass	0	0	0	46
M_TCP_Open	m_Open039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open039\Hardware\m_Open039.exe	2.0	2009/11/09 14:20:44	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:26:58	9155	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open040\Hardware\m_Open040.exe	2.1	2010/02/01 19:45:37	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:25:41	17168	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open041\Hardware\m_Open041.exe	2.0	2009/11/09 14:21:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 11:54:55	513788	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open042\Hardware\m_Open042.exe	2.4	2010/07/02 12:38:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:23:43	16719	TELM-06	1	Pass	0	0	0	60
M_TCP_Polling	m_Polling016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling016\Hardware\m_Polling016.exe	2.0	2009/09/23 19:25:05	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 00:06:04	49269	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling017\Hardware\m_Polling017.exe	2.1	2010/01/26 20:04:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:40:13	6210	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling018\Hardware\m_Polling018.exe	2.1	2010/01/26 20:05:23	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:03:10	81514	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling024\Hardware\m_Polling024.exe	2.5	2010/01/26 20:05:45	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:59:01	78580	TELM-06	1	Pass	0	0	0	10
M_TCP_Polling	m_Polling025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling025\Hardware\m_Polling025.exe	2.1	2009/11/24 20:51:39	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:56:53	46217	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling026\Hardware\m_Polling026.exe	2.2	2009/12/03 14:59:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:53:48	19513	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling040\Hardware\m_Polling040.exe	2.1	2010/01/26 20:06:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:51:40	6061	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling045\Hardware\m_Polling045.exe	2.1	2010/01/26 20:06:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:48:15	36019	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling046\Hardware\m_Polling046.exe	2.6	2010/04/16 14:21:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:45:51	47209	TELM-06	1	Pass	0	0	0	38
M_TCP_Polling	m_Polling047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling047\Hardware\m_Polling047.exe	2.1	2010/01/26 20:07:28	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:41:52	36113	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling048\Hardware\m_Polling048.exe	2.3	2009/11/24 20:53:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:39:11	9988	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling049\Hardware\m_Polling049.exe	2.4	2010/03/10 19:44:21	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:36:25	46129	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling050\Hardware\m_Polling050.exe	2.6	2010/06/29 17:44:57	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:34:51	22836	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling065\Hardware\m_Polling065.exe	2.6	2010/06/09 18:52:38	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:28:14	42134	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling066\Hardware\m_Polling066.exe	2.3	2009/11/09 22:57:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:23:23	52618	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling067\Hardware\m_Polling067.exe	2.1	2009/10/22 14:28:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:19:12	84773	TELM-06	1	Pass	0	0	0	11
M_TCP_Polling	m_Polling068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling068\Hardware\m_Polling068.exe	2.3	2009/11/05 18:23:27	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:17:24	7134	TELM-06	1	Pass	0	0	0	8
M_TCP_Polling	m_Polling076	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling076\Hardware\m_Polling076.exe	2.1	2009/11/03 15:55:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:15:44	10128	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling077	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling077\Hardware\m_Polling077.exe	2.1	2009/11/05 18:34:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 20:39:42	527722	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling080	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling080\Hardware\m_Polling080.exe	2.2	2009/11/05 18:24:04	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 19:42:17	277373	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling081	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling081\Hardware\m_Polling081.exe	2.3	2010/05/27 14:19:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:14:20	24090	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling082	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling082\Hardware\m_Polling082.exe	2.2	2009/11/02 18:25:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:12:18	9466	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling083	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling083\Hardware\m_Polling083.exe	2.3	2009/11/02 18:45:07	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:10:02	19491	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling085	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling085\Hardware\m_Polling085.exe	2.1	2009/10/22 13:55:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 19:14:33	529389	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling088	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling088\Hardware\m_Polling088.exe	2.5	2010/05/12 17:59:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:08:11	23413	TELM-06	1	Pass	0	0	0	27
M_TCP_Polling	m_Polling089	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling089\Hardware\m_Polling089.exe	2.7	2010/05/11 19:28:17	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 23:04:43	97194	TELM-06	1	Pass	0	0	0	26
M_TCP_Polling	m_Polling090	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling090\Hardware\m_Polling090.exe	2.6	2010/02/15 20:02:24	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:01:10	83158	TELM-06	1	Pass	0	0	0	32
M_TCP_Polling	m_Polling091	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling091\Hardware\m_Polling091.exe	2.6	2010/02/15 20:16:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:57:48	13086	TELM-06	1	Pass	0	0	0	13
M_TCP_Polling	m_Polling092	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling092\Hardware\m_Polling092.exe	2.1	2009/11/05 18:41:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 18:35:08	277345	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling093	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling093\Hardware\m_Polling093.exe	2.0	2009/11/09 18:30:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:55:33	16780	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling094	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling094\Hardware\m_Polling094.exe	2.0	2009/11/09 18:30:49	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 13:00:12	307368	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling095	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling095\Hardware\m_Polling095.exe	2.0	2009/11/09 18:31:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:54:14	8305	TELM-06	1	Pass	0	0	0	2
M_TCP_PowerInhibit	m_Power001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power001\Hardware\m_Power001.exe	2.2	2010/06/07 13:10:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:26:15	58701	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power002\Hardware\m_Power002.exe	2.2	2010/01/05 19:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:25:18	10748	TELM-06	1	Pass	0	0	0	284
M_TCP_PowerInhibit	m_Power003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power003\Hardware\m_Power003.exe	2.4	2010/06/22 14:04:45	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:19:21	54162	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power004\Hardware\m_Power004.exe	2.3	2010/03/22 14:26:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:15:52	18008	TELM-06	1	Pass	0	0	0	46
M_TCP_PowerInhibit	m_Power005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power005\Hardware\m_Power005.exe	2.6	2010/07/02 12:34:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 10:05:50	195830	TELM-06	1	Pass	0	0	0	310
M_TCP_PowerInhibit	m_Power006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power006\Hardware\m_Power006.exe	2.6	2010/06/22 12:27:46	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:08:59	193604	TELM-06	1	Pass	0	0	0	2684
M_TCP_PowerInhibit	m_Power007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power007\Hardware\m_Power007.exe	2.1	2009/12/14 16:39:57	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:04:28	19392	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power008\Hardware\m_Power008.exe	2.2	2009/12/14 16:44:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:02:00	34945	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power009\Hardware\m_Power009.exe	2.4	2010/07/01 15:00:10	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:56:05	12053	TELM-06	1	Pass	0	0	0	9
M_TCP_PowerInhibit	m_Power010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power010\Hardware\m_Power010.exe	2.1	2010/02/16 22:18:45	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:53:33	11880	TELM-06	1	Pass	0	0	0	26
M_TCP_PowerInhibit	m_Power011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power011\Hardware\m_Power011.exe	2.1	2010/03/11 19:04:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:49:40	97145	TELM-06	1	Pass	0	0	0	116
M_TCP_RawMode	m_RawMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode001\Hardware\m_RawMode001.exe	2.4	2010/01/26 20:08:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:28:29	7497	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode002\Hardware\m_RawMode002.exe	2.6	2010/01/27 13:00:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 16:25:43	9697	TELM-06	1	Pass	0	0	0	87
M_TCP_RawMode	m_RawMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode003\Hardware\m_RawMode003.exe	2.5	2010/01/27 13:03:49	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:23:21	11491	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode004\Hardware\m_RawMode004.exe	2.6	2010/01/27 13:09:25	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 16:21:01	7212	TELM-06	1	Pass	0	0	0	2
M_TCP_RawMode	m_RawMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode005\Hardware\m_RawMode005.exe	2.3	2010/01/26 20:10:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:19:03	13303	TELM-06	1	Pass	0	0	0	9
M_TCP_RawMode	m_RawMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode006\Hardware\m_RawMode006.exe	2.2	2010/01/26 20:11:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:39:14	8943	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode007\Hardware\m_RawMode007.exe	2.3	2010/06/09 21:03:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:16:24	11191	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode008\Hardware\m_RawMode008.exe	2.4	2010/01/27 13:14:20	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:13:56	16183	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode009\Hardware\m_RawMode009.exe	2.2	2009/12/14 16:51:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:11:33	15744	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode010\Hardware\m_RawMode010.exe	2.1	2009/10/27 17:32:20	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:09:24	3185	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode011\Hardware\m_RawMode011.exe	2.2	2010/02/01 16:15:38	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:07:27	8478	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode012\Hardware\m_RawMode012.exe	2.3	2010/01/26 20:13:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:06:03	11603	TELM-06	1	Pass	0	0	0	3
M_TCP_Reset	m_Reset001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset001\Hardware\m_Reset001.exe	2.3	2010/01/06 21:25:12	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:13:28	9246	TELM-06	1	Pass	0	0	0	23
M_TCP_Reset	m_Reset002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset002\Hardware\m_Reset002.exe	2.4	2010/01/28 21:52:07	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:09:07	20666	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security002\Hardware\m_Security002.exe	2.9	2010/06/03 15:18:25	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 17:21:52	457368	TELM-06	1	Pass	0	0	0	154

M_TCP_Security	m_Security003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security003\Hardware\m_Security003.exe	2.4	2010/02/17 16:20:41	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 10:00:05	6811	TELM-06	1	Pass	0	0	0	73
M_TCP_Security	m_Security004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security004\Hardware\m_Security004.exe	2.4	2010/02/15 15:01:00	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:50:26	253987	TELM-06	1	Pass	0	0	0	112
M_TCP_Security	m_Security005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security005\Hardware\m_Security005.exe	2.2	2010/02/17 16:23:51	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 09:35:09	4061	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security006\Hardware\m_Security006.exe	2.4	2010/02/17 14:44:38	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:27:00	153473	TELM-06	1	Pass	0	0	0	108
M_TCP_Security	m_Security007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security007\Hardware\m_Security007.exe	2.6	2010/02/18 15:13:38	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 09:09:23	109044	TELM-06	1	Pass	0	0	0	126
M_TCP_Security	m_Security008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security008\Hardware\m_Security008.exe	2.3	2010/02/17 16:28:05	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:58:43	6740	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security009\Hardware\m_Security009.exe	2.3	2010/02/17 16:32:40	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:44:59	6208	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security010\Hardware\m_Security010.exe	2.3	2010/02/18 17:45:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:34:54	38392	TELM-06	1	Pass	0	0	0	113
M_TCP_Security	m_Security011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security011\Hardware\m_Security011.exe	2.5	2010/06/03 18:16:10	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 08:24:14	20000	TELM-06	1	Pass	0	0	0	58
M_TCP_Security	m_Security012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security012\Hardware\m_Security012.exe	2.9	2010/02/18 15:14:18	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 08:16:30	4746	TELM-06	1	Pass	0	0	0	132
M_TCP_Security	m_Security013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security013\Hardware\m_Security013.exe	2.5	2010/02/18 15:15:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 08:06:17	11672	TELM-06	1	Pass	0	0	0	140
M_TCP_Security	m_Security014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security014\Hardware\m_Security014.exe	2.2	2010/06/03 18:21:27	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 03:34:49	14827	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security015\Hardware\m_Security015.exe	2.4	2010/02/17 17:58:26	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 22:07:07	596176	TELM-06	1	Pass	0	0	0	34
M_TCP_Security	m_Security016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security016\Hardware\m_Security016.exe	2.2	2010/06/08 19:56:40	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:03:33	10679	TELM-06	1	Pass	0	0	0	146
M_TCP_Security	m_Security017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security017\Hardware\m_Security017.exe	2.2	2009/12/08 17:29:14	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:01:47	14627	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security018\Hardware\m_Security018.exe	2.4	2010/03/18 16:05:37	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 07:59:19	40624	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security019\Hardware\m_Security019.exe	2.3	2010/03/26 13:43:17	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 07:54:03	90395	TELM-06	1	Pass	0	0	0	1226
M_TCP_Security	m_Security020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security020\Hardware\m_Security020.exe	2.3	2010/06/03 18:22:30	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 07:46:53	22013	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security021\Hardware\m_Security021.exe	2.5	2010/06/03 18:37:52	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 07:30:08	40997	TELM-06	1	Pass	0	0	0	81
M_TCP_Security	m_Security022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security022\Hardware\m_Security022.exe	2.9	2010/06/18 19:05:16	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 10:23:47	290940	TELM-06	1	Pass	0	0	0	55505
M_TCP_Security	m_Security023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security023\Hardware\m_Security023.exe	2.12	2010/06/10 17:30:47	2.12	Up-to-date	Baseline 17_1	None	09/10/2010 21:51:39	51681	TELM-06	1	Pass	0	0	0	1119
M_TCP_Security	m_Security024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security024\Hardware\m_Security024.exe	2.3	2010/03/26 13:06:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 17:02:04	30694	TELM-06	1	Pass	0	0	0	4
M_TCP_Security	m_Security025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security025\Hardware\m_Security025.exe	2.3	2010/06/10 14:39:09	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 16:37:09	48466	TELM-06	1	Pass	0	0	0	158
M_TCP_Security	m_Security026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security026\Hardware\m_Security026.exe	2.3	2010/02/18 15:15:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 07:28:14	4194	TELM-06	1	Pass	0	0	0	60
M_TCP_Security	m_Security027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security027\Hardware\m_Security027.exe	2.1	2010/06/29 19:03:00	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 07:21:15	38007	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security028\Hardware\m_Security028.exe	2.1	2010/06/29 19:26:29	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 07:14:19	42255	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security029\Hardware\m_Security029.exe	2.2	2010/03/11 18:29:14	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 07:08:11	20448	TELM-06	1	Pass	0	0	0	46
M_TCP_Security	m_Security030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security030\Hardware\m_Security030.exe	2.2	2010/02/08 13:02:02	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:56:27	11966	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security031\Hardware\m_Security031.exe	2.1	2010/06/03 18:44:12	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:53:51	23087	TELM-06	1	Pass	0	0	0	16
M_TCP_Security	m_Security032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security032\Hardware\m_Security032.exe	2.1	2010/02/03 12:51:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:46:41	8142	TELM-06	1	Pass	0	0	0	11
M_TCP_Security	m_Security033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security033\Hardware\m_Security033.exe	2.1	2010/01/27 21:32:15	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:41:42	13895	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security034\Hardware\m_Security034.exe	2.3	2010/03/26 13:19:51	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:35:57	118781	TELM-06	1	Pass	0	0	0	15
M_TCP_Security	m_Security035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security035\Hardware\m_Security035.exe	2.0	2010/01/14 21:18:30	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:34:11	17849	TELM-06	1	Pass	0	0	0	30
M_TCP_Security	m_Security036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security036\Hardware\m_Security036.exe	2.5	2010/05/07 14:41:43	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:30:30	86752	TELM-06	1	Pass	0	0	0	10
M_TCP_Security	m_Security037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security037\Hardware\m_Security037.exe	2.6	2010/06/03 18:54:41	2.6	Up-to-date	Baseline 17_1	None	09/11/2010 03:31:17	51152	TELM-06	1	Pass	0	0	0	119
M_TCP_Security	m_Security038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security038\Hardware\m_Security038.exe	2.9	2010/06/07 13:02:07	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 16:06:43	167541	TELM-06	1	Pass	0	0	0	170
M_TCP_Security	m_Security039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security039\Hardware\m_Security039.exe	2.2	2010/02/18 15:16:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:27:21	56653	TELM-06	1	Pass	0	0	0	55
M_TCP_Security	m_Security040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security040\Hardware\m_Security040.exe	2.3	2010/06/03 20:43:31	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:21:16	278511	TELM-06	1	Pass	0	0	0	33
M_TCP_Security	m_Security041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security041\Hardware\m_Security041.exe	2.6	2010/04/21 10:55:08	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:17:09	65808	TELM-06	1	Pass	0	0	0	77
M_TCP_Security	m_Security042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security042\Hardware\m_Security042.exe	2.1	2010/04/08 12:57:18	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:13:11	58145	TELM-06	1	Pass	0	0	0	48
M_TCP_Security	m_Security043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security043\Hardware\m_Security043.exe	2.0	2010/02/16 20:27:42	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:10:06	68922	TELM-06	1	Pass	0	0	0	56
M_TCP_Security	m_Security044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security044\Hardware\m_Security044.exe	2.2	2010/05/13 19:14:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:06:47	70705	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security045\Hardware\m_Security045.exe	2.3	2010/05/12 17:38:53	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:03:39	71181	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security046\Hardware\m_Security046.exe	2.2	2010/06/07 13:05:30	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:02:25	3559	TELM-06	1	Pass	0	0	0	30
M_TCP_WirelessWakeup	m_Wakeup001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup001\Hardware\m_Wakeup001.exe	2.3	2010/06/21 17:44:25	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 15:22:11	291386	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup002\Hardware\m_Wakeup002.exe	2.5	2010/05/12 16:16:45	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:46:25	35400	TELM-06	1	Pass	0	0	0	12
M_TCP_WirelessWakeup	m_Wakeup003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup003\Hardware\m_Wakeup003.exe	2.2	2010/03/10 17:59:29	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:44:57	14343	TELM-06	1	Pass	0	0	0	11
M_TCP_WirelessWakeup	m_Wakeup004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup004\Hardware\m_Wakeup004.exe	2.1	2009/12/08 16:08:20	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 03:27:30	19177	TELM-06	1	Pass	0	0	0	31
M_TCP_WirelessWakeup	m_Wakeup005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup005\Hardware\m_Wakeup005.exe	2.3	2010/06/08 21:32:21	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:42:03	20413	TELM-06	1	Pass	0	0	0	40
M_TCP_WirelessWakeup	m_Wakeup006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup006\Hardware\m_Wakeup006.exe	2.3	2010/01/27 00:35:22	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:37:34	10249	TELM-06	1	Pass	0	0	0	55
M_TCP_WirelessWakeup	m_Wakeup007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup007\Hardware\m_Wakeup007.exe	2.1	2009/12/04 19:28:48	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:35:36	12950	TELM-06	1	Pass	0	0	0	3
M_TCP_WirelessWakeup	m_Wakeup008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup008\Hardware\m_Wakeup008.exe	2.5	2010/05/07 15:41:21	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:31:23	54608	TELM-06	1	Pass	0	0	0	24
M_TCP_WirelessWakeup	m_Wakeup009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup009\Hardware\m_Wakeup009.exe	2.6	2010/05/12 16:17:49	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:29:25	52064	TELM-06	1	Pass	0	0	0	32
M_TCP_WirelessWakeup	m_Wakeup010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup010\Hardware\m_Wakeup010.exe	2.4	2010/05/12 16:20:12	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:26:31	45348	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup011\Hardware\m_Wakeup011.exe	2.3	2010/04/06 18:07:38	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:23:27	27969	TELM-06	1	Pass	0	0	0	25
M_TCP_WirelessWakeup	m_Wakeup012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup012\Hardware\m_Wakeup012.exe	2.6	2010/05/12 16:20:45	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:17:28	48825	TELM-06	1	Pass	0	0	0	45
M_TCP_WirelessWakeup	m_Wakeup013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup013\Hardware\m_Wakeup013.exe	2.2	2010/02/24 12:20:40	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:11:16	127637	TELM-06	1	Pass	0	0	0	2
M_TCP_WirelessWakeup	m_Wakeup014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode2\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup014\Hardware\m_Wakeup014.exe	2.3	2010/06/04 11:50:36	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:05:59	74654	TELM-06	1	Pass	0	0	0	30
M_TCP_AntennaSelection	m_Antenna001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna001\Hardware\m_Antenna001.exe	2.4	2010/06/16 12:31:48	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 14:07:38	237449	TELM-06	1	Pass	0	0	0	127
M_TCP_AntennaSelection	m_Antenna002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna002\Hardware\m_Antenna002.exe	2.0	2010/06/16 12:29:50	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 13:33:26	286976	TELM-06	1	Pass	0	0	0	99
M_TCP_Channel	m_ChannelRecovery001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery001\Hardware\m_ChannelRecovery001.exe	2.5	2010/04/08 14:25:18	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:14:25	14941	TELM-06	1	Pass	0	0	0	26
M_TCP_Channel	m_ChannelRecovery002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery002\Hardware\m_ChannelRecovery002.exe	2.2	2009/11/30 22:31:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:12:27	17500	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery003\Hardware\m_ChannelRecovery003.exe	2.2	2009/11/30 22:32:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:10:34	17473	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery004\Hardware\m_ChannelRecovery004.exe	2.2	2010/05/26 12:04:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:08:38	14512	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery005\Hardware\m_ChannelRecovery005.exe	2.5	2010/06/21 18:34:44	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:07:08	14186	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery006\Hardware\m_ChannelRecovery006.exe	2.4	2010/05/07 17:09:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:05:39	15731	TELM-06	1	Pass	0	0	0	36
M_TCP_Channel	m_ChannelRecovery007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery007\Hardware\m_ChannelRecovery007.exe	2.7	2010/05/11 11:49:24	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 20:03:53	27871	TELM-06	1	Pass	0	0	0	89
M_TCP_Channel	m_ChannelRecovery008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery008\Hardware\m_ChannelRecovery008.exe	2.6	2010/05/13 15:05:39	2.6	Up-to-date	Baseline 17_1	None	09/13/2010 08:00:28	32704	TELM-06	1	Pass	0	0	0	575
M_TCP_Channel	m_ChannelRecovery009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery009\Hardware\m_ChannelRecovery009.exe	2.6	2010/06/22 15:16:15	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:59:21	57689	TELM-06	1	Pass	0	0	0	342
M_TCP_Channel	m_ChannelRecovery010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery010\Hardware\m_ChannelRecovery010.exe	2.8	2010/05/26 15:48:50	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 19:58:00	25100	TELM-06	1	Pass	0	0	0	142
M_TCP_Channel	m_ChannelRecovery011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery011\Hardware\m_ChannelRecovery011.exe	2.5	2010/05/13 12:31:10	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:56:34	15711	TELM-06	1	Pass	0	0	0	14
M_TCP_Channel	m_ChannelRecovery012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery012\Hardware\m_ChannelRecovery012.exe	2.6	2010/09/07 20:36:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:54:06	48115	TELM-06	1	Pass	0	0	0	199
M_TCP_Channel	m_ChannelRecovery013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery013\Hardware\m_ChannelRecovery013.exe	2.2	2010/01/05 19:17:54	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:52:27	24231	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery014\Hardware\m_ChannelRecovery014.exe	2.8	2010/05/13 13:19:47	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 19:49:12	51455	TELM-06	1	Pass	0	0	0	23
M_TCP_Channel	m_ChannelRecovery015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery015\Hardware\m_ChannelRecovery015.exe	2.6	2010/06/03 18:01:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:47:57	12555	TELM-06	1	Pass	0	0	0	88
M_TCP_Channel	m_ChannelRecovery016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery016\Hardware\m_ChannelRecovery016.exe	2.4	2010/02/11 20:10:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:44:43	38139	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery017\Hardware\m_ChannelRecovery017.exe	2.3	2010/02/11 17:38:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:43:21	10364	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery018\Hardware\m_ChannelRecovery018.exe	2.3	2010/02/01 19:41:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:41:49	18054	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery020\Hardware\m_ChannelRecovery020.exe	2.7	2010/06/23 17:42:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 19:39:49	37573	TELM-06	1	Pass	0	0	0	109
M_TCP_Channel	m_ChannelRecovery021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery021\Hardware\m_ChannelRecovery021.exe	2.3	2010/04/13 18:33:24	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:38:06	27230	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery022\Hardware\m_ChannelRecovery022.exe	2.3	2010/04/13 18:36:50	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:35:44	19784	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery023\Hardware\m_ChannelRecovery023.exe	2.5	2010/04/13 18:41:23	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:34:02	29873	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery024\Hardware\m_ChannelRecovery024.exe	2.4	2010/04/07 12:33:58	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:32:45	19920	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery025\Hardware\m_ChannelRecovery025.exe	2.2	2010/02/01 19:43:28	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:30:49	11188	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery026\Hardware\m_ChannelRecovery026.exe	2.2	2010/02/11 21:01:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:28:59	11541	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery027\Hardware\m_ChannelRecovery027.exe	2.2	2010/02/03 17:18:54	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:27:24	13564	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery028\Hardware\m_ChannelRecovery028.exe	2.2	2010/02/03 17:19:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:26:04	11560	TELM-06	1	Pass	0	0	0	10
M_TCP_Channel	m_ChannelRecovery029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery029\Hardware\m_ChannelRecovery029.exe	2.2	2009/12/18 20:13:33	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:23:18	13550	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery030\Hardware\m_ChannelRecovery030.exe	2.10	2010/06/23 21:15:31	2.10	Up-to-date	Baseline 17_1	None	09/09/2010 19:21:44	18622	TELM-06	1	Pass	0	0	0	30
M_TCP_Channel	m_ChannelRecovery031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery031\Hardware\m_ChannelRecovery031.exe	2.6	2010/04/13 18:46:58	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:20:25	18792	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery032\Hardware\m_ChannelRecovery032.exe	2.5	2010/06/23 21:50:13	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:18:51	18118	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery033\Hardware\m_ChannelRecovery033.exe	2.3	2010/04/07 17:51:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:16:31	19164	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery038\Hardware\m_ChannelRecovery038.exe	2.6	2010/05/26 15:55:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:15:00	20829	TELM-06	1	Pass	0	0	0	175
M_TCP_Channel	m_ChannelRecovery039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery039\Hardware\m_ChannelRecovery039.exe	2.3	2010/02/10 20:47:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:12:28	12347	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery040\Hardware\m_ChannelRecovery040.exe	2.5	2010/06/07 12:54:38	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:10:32	15515	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery041\Hardware\m_ChannelRecovery041.exe	2.1	2009/10/22 13:18:38	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:05:23	38545	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery042\Hardware\m_ChannelRecovery042.exe	2.3	2009/12/03 15:57:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:03:52	22004	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery043\Hardware\m_ChannelRecovery043.exe	2.6	2010/06/24 18:37:58	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:01:36	55900	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery045\Hardware\m_ChannelRecovery045.exe	2.2	2009/11/12 21:22:11	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:59:53	11403	TELM-06	1	Pass	0	0	0	44
M_TCP_Channel	m_ChannelRecovery046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery046\Hardware\m_ChannelRecovery046.exe	2.3	2009/11/16 18:40:10	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:58:22	12700	TELM-06	1	Pass	0	0	0	49
M_TCP_Channel	m_ChannelRecovery047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery047\Hardware\m_ChannelRecovery047.exe	2.1	2009/11/16 18:58:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:55:50	52164	TELM-06	1	Pass	0	0	0	95
M_TCP_Channel	m_ChannelRecovery048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery048\Hardware\m_ChannelRecovery048.exe	2.3	2010/06/24 18:46:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:51:53	62623	TELM-06	1	Pass	0	0	0	156
M_TCP_Channel	m_ChannelRecovery049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery049\Hardware\m_ChannelRecovery049.exe	2.5	2010/03/26 13:55:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 18:49:31	36313	TELM-06	1	Pass	0	0	0	207
M_TCP_Channel	m_ChannelRecovery050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery050\Hardware\m_ChannelRecovery050.exe	2.2	2009/11/20 13:38:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:44:09	18591	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery051\Hardware\m_ChannelRecovery051.exe	2.3	2010/06/24 19:40:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:38:58	83014	TELM-06	1	Pass	0	0	0	196
M_TCP_Channel	m_ChannelRecovery053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery053\Hardware\m_ChannelRecovery053.exe	2.3	2010/06/24 21:57:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:36:28	31118	TELM-06	1	Pass	0	0	0	96
M_TCP_Channel	m_ChannelRecovery054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery054\Hardware\m_ChannelRecovery054.exe	2.4	2010/06/24 22:31:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:35:10	9100	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery055\Hardware\m_ChannelRecovery055.exe	2.4	2010/02/11 14:12:25	2.4	Up-to-date	Baseline 17_1	None	09/13/2010 08:01:03	2022132	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery061	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery061\Hardware\m_ChannelRecovery061.exe	2.3	2010/04/06 11:52:17	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:33:39	20766	TELM-06	1	Pass	0	0	0	12
M_TCP_Channel	m_ChannelRecovery062	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery062\Hardware\m_ChannelRecovery062.exe	2.2	2010/06/25 15:47:28	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:28:04	13146	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery063	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery063\Hardware\m_ChannelRecovery063.exe	2.9	2010/05/13 17:43:56	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 18:25:54	24631	TELM-06	1	Pass	0	0	0	39

M_TCP_Channel	m_ChannelRecovery064	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery064\Hardware\m_ChannelRecovery064.exe	2.4	2010/05/06 14:14:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:21:13	40775	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery065\Hardware\m_ChannelRecovery065.exe	2.2	2010/05/06 15:43:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:16:08	37347	TELM-06	1	Pass	0	0	0	7
M_TCP_Channel	m_ChannelRecovery066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery066\Hardware\m_ChannelRecovery066.exe	2.2	2010/05/06 15:49:13	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:13:10	37415	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery067\Hardware\m_ChannelRecovery067.exe	2.2	2010/05/06 15:58:13	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:11:39	40752	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery068\Hardware\m_ChannelRecovery068.exe	2.3	2010/06/03 18:10:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:08:24	34645	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery069	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery069\Hardware\m_ChannelRecovery069.exe	2.3	2010/05/10 15:40:47	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 02:51:48	1009315	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery070	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery070\Hardware\m_ChannelRecovery070.exe	2.1	2010/02/04 18:14:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:05:36	12727	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery071	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery071\Hardware\m_ChannelRecovery071.exe	2.1	2010/02/04 17:57:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:03:38	15900	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery072	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery072\Hardware\m_ChannelRecovery072.exe	2.3	2010/05/07 11:42:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:01:38	44221	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery073	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery073\Hardware\m_ChannelRecovery073.exe	2.3	2010/05/07 11:43:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:59:36	44262	TELM-06	1	Pass	0	0	0	8
M_TCP_Close	m_Close001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close001\Hardware\m_Close001.exe	2.6	2010/03/29 17:53:05	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:23:06	7423	TELM-06	1	Pass	0	0	0	51
M_TCP_Close	m_Close002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close002\Hardware\m_Close002.exe	2.3	2010/01/26 20:03:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:21:50	8788	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close003\Hardware\m_Close003.exe	2.3	2010/06/24 15:01:11	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:20:39	6977	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close004\Hardware\m_Close004.exe	2.5	2010/03/16 12:03:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:19:05	14979	TELM-06	1	Pass	0	0	0	165
M_TCP_Close	m_Close005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close005\Hardware\m_Close005.exe	2.5	2010/05/24 17:00:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:16:54	19440	TELM-06	1	Pass	0	0	0	118
M_TCP_Close	m_Close006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close006\Hardware\m_Close006.exe	2.2	2009/11/19 20:38:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:14:26	11112	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close007\Hardware\m_Close007.exe	2.1	2010/02/04 18:46:09	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:10:53	16394	TELM-06	1	Pass	0	0	0	66
M_TCP_Close	m_Close008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close008\Hardware\m_Close008.exe	2.4	2010/05/12 15:22:15	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:09:26	7054	TELM-06	1	Pass	0	0	0	10
M_TCP_Close	m_Close009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close009\Hardware\m_Close009.exe	2.2	2010/02/04 15:29:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:08:14	7712	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close010\Hardware\m_Close010.exe	2.0	2009/09/21 12:45:39	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:06:13	15767	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close011\Hardware\m_Close011.exe	2.0	2009/09/21 12:46:09	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:02:02	8121	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close012\Hardware\m_Close012.exe	2.0	2009/09/21 12:46:36	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:59:20	11427	TELM-06	1	Pass	0	0	0	3
M_TCP_Close	m_Close013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close013\Hardware\m_Close013.exe	2.0	2009/09/21 12:18:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:57:53	7865	TELM-06	1	Pass	0	0	0	4
M_TCP_Close	m_Close014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close014\Hardware\m_Close014.exe	2.1	2010/01/29 16:45:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:56:11	8856	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close016\Hardware\m_Close016.exe	2.2	2010/08/24 12:05:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:54:59	7996	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close017\Hardware\m_Close017.exe	2.0	2009/09/21 12:47:07	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:53:40	9572	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close019\Hardware\m_Close019.exe	2.1	2009/10/01 14:47:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:51:52	4997	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close020\Hardware\m_Close020.exe	2.2	2009/10/30 14:47:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:50:16	9423	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close021\Hardware\m_Close021.exe	2.0	2009/09/21 12:48:28	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:49:01	2710	TELM-06	1	Pass	0	0	0	1
M_TCP_Close	m_Close022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close022\Hardware\m_Close022.exe	2.2	2009/10/30 14:53:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:46:41	12339	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_ARQ001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ001\Hardware\m_ARQ001.exe	2.5	2010/06/29 15:20:36	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 07:03:07	12973	TELM-06	1	Pass	0	0	0	78
M_TCP_Data	m_ARQ002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ002\Hardware\m_ARQ002.exe	2.0	2009/09/21 18:25:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:56:17	7946	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ003\Hardware\m_ARQ003.exe	2.3	2010/04/05 12:22:02	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:48:45	8386	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_ARQ004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ004\Hardware\m_ARQ004.exe	2.1	2009/11/04 21:22:33	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:46:19	8343	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_ARQ005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ005\Hardware\m_ARQ005.exe	2.2	2009/12/23 13:26:32	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:41:16	7706	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ006\Hardware\m_ARQ006.exe	2.1	2009/09/24 16:41:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:35:47	8665	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_ARQ007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ007\Hardware\m_ARQ007.exe	2.0	2009/09/21 18:27:29	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:33:26	10701	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_ARQ008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ008\Hardware\m_ARQ008.exe	2.0	2009/09/21 18:28:06	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:30:07	10935	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ009\Hardware\m_ARQ009.exe	2.2	2010/02/04 13:08:24	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:26:59	9049	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ010\Hardware\m_ARQ010.exe	2.2	2010/02/04 13:19:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:20:54	9037	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ011\Hardware\m_ARQ011.exe	2.3	2010/02/01 13:30:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:16:51	8470	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ012\Hardware\m_ARQ012.exe	2.3	2010/02/18 15:17:44	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:12:46	8767	TELM-06	1	Pass	0	0	0	211
M_TCP_Data	m_ARQ013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ013\Hardware\m_ARQ013.exe	2.3	2010/02/03 16:38:29	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:09:45	10724	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_ARQ014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ014\Hardware\m_ARQ014.exe	2.4	2010/04/05 11:08:56	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 06:06:37	8673	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ015\Hardware\m_ARQ015.exe	2.6	2010/06/08 15:25:41	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:03:22	13233	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ016\Hardware\m_ARQ016.exe	2.3	2009/11/19 20:22:28	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:02:02	9446	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ017\Hardware\m_ARQ017.exe	2.2	2009/09/30 15:01:03	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:59:04	6812	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ018\Hardware\m_ARQ018.exe	2.7	2010/06/24 12:34:26	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 05:55:35	10996	TELM-06	1	Pass	0	0	0	586
M_TCP_Data	m_ARQ019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ019\Hardware\m_ARQ019.exe	2.4	2010/06/08 16:07:58	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:51:17	10001	TELM-06	1	Pass	0	0	0	103
M_TCP_Data	m_ARQ020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ020\Hardware\m_ARQ020.exe	2.2	2009/11/18 15:23:22	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:49:14	7580	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ021\Hardware\m_ARQ021.exe	2.2	2009/11/19 18:43:52	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:47:59	7466	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ022\Hardware\m_ARQ022.exe	2.3	2010/02/03 16:30:46	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:45:56	8716	TELM-06	1	Pass	0	0	0	137
M_TCP_Data	m_ARQ023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ023\Hardware\m_ARQ023.exe	2.1	2009/11/10 19:12:36	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:42:52	9969	TELM-06	1	Pass	0	0	0	73
M_TCP_Data	m_ARQ024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ024\Hardware\m_ARQ024.exe	2.3	2010/02/08 18:44:31	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:38:43	10288	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ025\Hardware\m_ARQ025.exe	2.1	2009/09/25 14:20:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:35:59	8767	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ026\Hardware\m_ARQ026.exe	2.1	2009/09/25 14:21:05	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:34:03	8623	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_ARQ027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ027\Hardware\m_ARQ027.exe	2.1	2009/09/25 14:21:27	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:30:57	9061	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ028\Hardware\m_ARQ028.exe	2.4	2010/04/19 14:36:56	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:28:44	18626	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_ARQ029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ029\Hardware\m_ARQ029.exe	2.1	2010/03/08 19:53:24	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:25:59	29531	TELM-06	1	Pass	0	0	0	555
M_TCP_Data	m_ARQ031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ031\Hardware\m_ARQ031.exe	2.3	2010/06/08 19:57:52	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:22:52	11485	TELM-06	1	Pass	0	0	0	596
M_TCP_Data	m_Data012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data012\Hardware\m_Data012.exe	2.4	2009/11/10 14:18:10	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 05:40:16	37780	TELM-06	1	Pass	0	0	0	18
M_TCP_Data	m_Data013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data013\Hardware\m_Data013.exe	2.3	2010/02/03 15:41:32	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:15:42	103233	TELM-06	1	Pass	0	0	0	335
M_TCP_Data	m_Data014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data014\Hardware\m_Data014.exe	2.3	2010/05/12 19:37:35	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:09:35	97777	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_Data015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data015\Hardware\m_Data015.exe	2.2	2010/02/03 17:34:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:05:48	8541	TELM-06	1	Pass	0	0	0	14
M_TCP_Data	m_Data016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data016\Hardware\m_Data016.exe	2.2	2010/02/24 20:56:20	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:01:46	69951	TELM-06	1	Pass	0	0	0	26
M_TCP_Data	m_Data017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data017\Hardware\m_Data017.exe	2.2	2010/02/24 22:22:41	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:59:18	67691	TELM-06	1	Pass	0	0	0	30
M_TCP_Data	m_Data018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data018\Hardware\m_Data018.exe	2.2	2010/02/03 22:48:42	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:54:46	59151	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_MemoryRead001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead001\Hardware\m_MemoryRead001.exe	2.2	2010/02/01 17:26:47	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:52:49	19863	TELM-06	1	Pass	0	0	0	439
M_TCP_Data	m_MemoryRead002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead002\Hardware\m_MemoryRead002.exe	2.3	2009/11/05 15:03:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:51:07	18961	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_MemoryRead003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead003\Hardware\m_MemoryRead003.exe	2.4	2010/04/05 13:08:02	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:48:30	15411	TELM-06	1	Pass	0	0	0	320
M_TCP_Data	m_MemoryRead004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead004\Hardware\m_MemoryRead004.exe	2.3	2010/03/19 16:10:47	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:46:30	8355	TELM-06	1	Pass	0	0	0	1
M_TCP_Data	m_MemoryRead005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead005\Hardware\m_MemoryRead005.exe	2.3	2009/12/07 19:50:30	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:43:30	48562	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryRead006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead006\Hardware\m_MemoryRead006.exe	2.6	2010/06/29 21:05:35	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:41:24	12203	TELM-06	1	Pass	0	0	0	37
M_TCP_Data	m_MemoryRead007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead007\Hardware\m_MemoryRead007.exe	2.1	2009/11/03 12:42:15	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:35:29	10811	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_MemoryRead008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead008\Hardware\m_MemoryRead008.exe	2.1	2009/09/30 18:09:14	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:30:22	7757	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead009\Hardware\m_MemoryRead009.exe	2.1	2009/12/07 19:51:11	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:26:59	11446	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead010\Hardware\m_MemoryRead010.exe	2.2	2009/09/30 17:31:58	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:25:40	7699	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead011\Hardware\m_MemoryRead011.exe	2.2	2009/09/30 17:32:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:17:59	7636	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead012\Hardware\m_MemoryRead012.exe	2.2	2009/09/30 17:33:03	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:11:49	7566	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead013\Hardware\m_MemoryRead013.exe	2.2	2009/09/30 17:33:27	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:08:01	7687	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead014\Hardware\m_MemoryRead014.exe	2.2	2009/09/30 17:33:51	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:05:51	7718	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead015\Hardware\m_MemoryRead015.exe	2.2	2009/09/30 17:34:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:04:34	7696	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead016\Hardware\m_MemoryRead016.exe	2.1	2009/10/19 18:24:00	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:02:38	7691	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead017\Hardware\m_MemoryRead017.exe	2.2	2009/10/19 15:52:59	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:01:20	7729	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead019\Hardware\m_MemoryRead019.exe	2.3	2009/12/03 13:57:22	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:00:02	9269	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead020\Hardware\m_MemoryRead020.exe	2.4	2010/02/09 20:49:06	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:58:20	7011	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryRead021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead021\Hardware\m_MemoryRead021.exe	2.2	2009/12/23 13:34:31	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:55:54	8581	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryRead030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead030\Hardware\m_MemoryRead030.exe	2.1	2009/10/16 13:54:03	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:53:09	6507	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead031\Hardware\m_MemoryRead031.exe	2.3	2009/12/28 18:23:34	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:51:24	10027	TELM-06	1	Pass	0	0	0	43
M_TCP_Data	m_MemoryRead033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead033\Hardware\m_MemoryRead033.exe	2.3	2009/11/23 18:55:48	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:49:16	11958	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead034\Hardware\m_MemoryRead034.exe	2.1	2009/10/16 14:15:21	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:48:01	8058	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead035\Hardware\m_MemoryRead035.exe	2.4	2009/11/23 18:56:23	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:46:35	16900	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead037\Hardware\m_MemoryRead037.exe	2.2	2009/11/23 18:57:01	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:45:19	12306	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead038\Hardware\m_MemoryRead038.exe	2.3	2010/01/19 17:36:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:43:47	14671	TELM-06	1	Pass	0	0	0	982
M_TCP_Data	m_MemoryRead039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead039\Hardware\m_MemoryRead039.exe	2.1	2009/09/28 15:09:29	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:42:56	8723	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead040\Hardware\m_MemoryRead040.exe	2.6	2010/01/06 14:10:47	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 03:37:48	32278	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryRead041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead041\Hardware\m_MemoryRead041.exe	2.4	2009/11/16 12:52:24	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:34:10	18237	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead042\Hardware\m_MemoryRead042.exe	2.3	2010/06/29 22:24:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:31:37	26221	TELM-06	1	Pass	0	0	0	41
M_TCP_Data	m_MemoryRead043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead043\Hardware\m_MemoryRead043.exe	2.2	2010/02/08 16:55:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:29:52	17603	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead045\Hardware\m_MemoryRead045.exe	2.0	2009/09/22 14:34:52	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:27:58	6769	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead049\Hardware\m_MemoryRead049.exe	2.3	2009/11/16 12:52:57	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:26:03	8686	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead051\Hardware\m_MemoryRead051.exe	2.1	2009/10/27 19:22:11	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:24:31	9657	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryRead052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead052\Hardware\m_MemoryRead052.exe	2.3	2010/06/29 23:18:49	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:19:38	11148	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite001\Hardware\m_MemoryWrite001.exe	2.2	2009/12/07 19:52:07	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:17:08	8963	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite002\Hardware\m_MemoryWrite002.exe	2.0	2009/09/22 14:48:30	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:14:24	10554	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite003\Hardware\m_MemoryWrite003.exe	2.1	2009/10/28 14:34:41	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:12:00	10263	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite005\Hardware\m_MemoryWrite005.exe	2.2	2009/12/15 19:05:34	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:05:24	7862	TELM-06	1	Pass	0	0	0	186
M_TCP_Data	m_MemoryWrite006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite006\Hardware\m_MemoryWrite006.exe	2.3	2009/11/24 17:33:37	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:03:27	11282	TELM-06	1	Pass	0	0	0	145
M_TCP_Data	m_MemoryWrite007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite007\Hardware\m_MemoryWrite007.exe	2.1	2009/10/28 15:09:57	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:58:35	6980	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite008\Hardware\m_MemoryWrite008.exe	2.1	2010/02/01 19:11:34	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:54:55	7406	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite009\Hardware\m_MemoryWrite009.exe	2.1	2009/10/28 15:17:22	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:52:42	7512	TELM-06	1	Pass	0	0	0	23

M_TCP_Data	m_MemoryWrite010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite010\Hardware\m_MemoryWrite010.exe	2.1	2009/10/28 16:00:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:47:57	7560	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite011\Hardware\m_MemoryWrite011.exe	2.1	2009/10/28 16:07:34	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:43:28	7509	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite012\Hardware\m_MemoryWrite012.exe	2.2	2009/10/28 16:15:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:41:10	12268	TELM-06	1	Pass	0	0	0	120
M_TCP_Data	m_MemoryWrite013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite013\Hardware\m_MemoryWrite013.exe	2.1	2009/10/28 16:21:17	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:37:11	7526	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite014\Hardware\m_MemoryWrite014.exe	2.1	2009/10/28 16:29:59	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:33:33	7555	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite015\Hardware\m_MemoryWrite015.exe	2.1	2009/10/28 16:36:30	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:31:38	7581	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryWrite016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite016\Hardware\m_MemoryWrite016.exe	2.5	2010/02/01 21:11:41	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:28:43	9551	TELM-06	1	Pass	0	0	0	2286
M_TCP_Data	m_MemoryWrite017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite017\Hardware\m_MemoryWrite017.exe	2.3	2010/01/19 17:37:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:25:49	8703	TELM-06	1	Pass	0	0	0	1553
M_TCP_Data	m_MemoryWrite018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite018\Hardware\m_MemoryWrite018.exe	2.1	2009/10/29 20:06:15	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:22:05	6671	TELM-06	1	Pass	0	0	0	171
M_TCP_Data	m_MemoryWrite019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite019\Hardware\m_MemoryWrite019.exe	2.0	2009/09/22 15:13:38	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 02:18:13	9553	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryWrite020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite020\Hardware\m_MemoryWrite020.exe	2.2	2009/10/29 19:44:00	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:16:55	7889	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite022\Hardware\m_MemoryWrite022.exe	2.3	2009/11/03 17:28:46	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:15:17	9891	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite023\Hardware\m_MemoryWrite023.exe	2.4	2010/02/04 16:15:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:13:04	20500	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite024\Hardware\m_MemoryWrite024.exe	2.2	2009/11/24 17:35:21	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:10:06	7527	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_MemoryWrite025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite025\Hardware\m_MemoryWrite025.exe	2.1	2009/10/29 18:47:12	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:07:25	10885	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite029\Hardware\m_MemoryWrite029.exe	2.2	2010/02/01 21:00:35	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:04:52	14037	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite032\Hardware\m_MemoryWrite032.exe	2.5	2010/03/15 01:08:48	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:59:16	9682	TELM-06	1	Pass	0	0	0	505
M_TCP_Data	m_MemoryWrite038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite038\Hardware\m_MemoryWrite038.exe	2.1	2009/10/29 18:42:13	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:55:41	7230	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite039\Hardware\m_MemoryWrite039.exe	2.1	2010/02/10 18:22:02	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:54:16	9694	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite040\Hardware\m_MemoryWrite040.exe	2.2	2010/02/10 17:56:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:49:24	9517	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryWrite041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite041\Hardware\m_MemoryWrite041.exe	2.2	2010/02/01 20:45:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:47:36	13738	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite042\Hardware\m_MemoryWrite042.exe	2.2	2010/01/19 17:38:15	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:42:27	10220	TELM-06	1	Pass	0	0	0	997
M_TCP_Data	m_MemoryWrite046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite046\Hardware\m_MemoryWrite046.exe	2.3	2010/02/04 16:51:09	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:40:07	7264	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite047\Hardware\m_MemoryWrite047.exe	2.2	2009/11/11 19:12:51	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:33:13	8238	TELM-06	1	Pass	0	0	0	98
M_TCP_Data	m_MemoryWrite048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite048\Hardware\m_MemoryWrite048.exe	2.2	2009/11/11 18:32:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:29:07	9043	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite049\Hardware\m_MemoryWrite049.exe	2.5	2010/02/10 16:51:26	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:27:48	21187	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_MemoryWrite052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite052\Hardware\m_MemoryWrite052.exe	2.3	2009/12/23 13:35:47	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:25:44	11110	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite053\Hardware\m_MemoryWrite053.exe	2.2	2009/11/11 17:01:43	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:21:17	8457	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite054\Hardware\m_MemoryWrite054.exe	2.6	2010/06/30 17:58:00	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:17:04	10268	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryWrite055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite055\Hardware\m_MemoryWrite055.exe	2.2	2009/11/11 18:04:03	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:14:30	7356	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite056	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite056\Hardware\m_MemoryWrite056.exe	2.4	2010/02/10 16:32:12	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:11:41	8646	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode001\Hardware\m_TransferMode001.exe	2.8	2010/05/14 18:48:57	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 01:06:05	17062	TELM-06	1	Pass	0	0	0	4748
M_TCP_Data	m_TransferMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode002\Hardware\m_TransferMode002.exe	2.4	2010/07/02 13:00:52	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:03:07	9940	TELM-06	1	Pass	0	0	0	167
M_TCP_Data	m_TransferMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode003\Hardware\m_TransferMode003.exe	2.0	2009/11/16 20:59:23	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 01:00:28	7081	TELM-06	1	Pass	0	0	0	129
M_TCP_Data	m_TransferMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode004\Hardware\m_TransferMode004.exe	2.3	2010/01/08 17:07:47	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:55:33	17974	TELM-06	1	Pass	0	0	0	312
M_TCP_Data	m_TransferMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode005\Hardware\m_TransferMode005.exe	2.7	2010/07/02 13:36:13	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 00:53:27	36355	TELM-06	1	Pass	0	0	0	464
M_TCP_Data	m_TransferMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode006\Hardware\m_TransferMode006.exe	2.4	2010/04/16 11:58:13	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:49:42	6946	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode007\Hardware\m_TransferMode007.exe	2.2	2009/10/12 15:46:45	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:48:08	11838	TELM-06	1	Pass	0	0	0	112
M_TCP_Data	m_TransferMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode008\Hardware\m_TransferMode008.exe	2.0	2009/09/09 15:28:45	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 00:44:07	10651	TELM-06	1	Pass	0	0	0	76
M_TCP_Data	m_TransferMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode009\Hardware\m_TransferMode009.exe	2.1	2009/10/13 16:25:11	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:39:12	11369	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_TransferMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode010\Hardware\m_TransferMode010.exe	2.1	2009/10/13 16:26:21	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:35:52	11373	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode011\Hardware\m_TransferMode011.exe	2.3	2010/02/10 13:37:08	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:31:47	8294	TELM-06	1	Pass	0	0	0	72
M_TCP_Data	m_TransferMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode012\Hardware\m_TransferMode012.exe	2.3	2010/02/10 13:38:00	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:28:38	8121	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_TransferMode013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode013\Hardware\m_TransferMode013.exe	2.3	2010/02/10 13:39:16	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:22:17	8267	TELM-06	1	Pass	0	0	0	44
M_TCP_Data	m_TransferMode014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode014\Hardware\m_TransferMode014.exe	2.3	2010/02/10 13:40:35	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:18:38	8117	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_TransferMode015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode015\Hardware\m_TransferMode015.exe	2.5	2010/05/26 14:56:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 00:15:35	7373	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_TransferMode016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode016\Hardware\m_TransferMode016.exe	2.8	2010/07/02 13:14:11	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 00:12:29	29890	TELM-06	1	Pass	0	0	0	47
M_TCP_Data	m_TransferMode017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode017\Hardware\m_TransferMode017.exe	2.7	2010/05/18 14:47:09	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 00:09:39	36016	TELM-06	1	Pass	0	0	0	257
M_TCP_Data	m_TransferMode018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode018\Hardware\m_TransferMode018.exe	2.6	2010/04/16 14:06:26	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 00:05:36	26544	TELM-06	1	Pass	0	0	0	509
M_TCP_Data	m_TransferMode019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode019\Hardware\m_TransferMode019.exe	2.6	2010/03/19 18:58:04	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 00:04:32	17327	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics001\Hardware\m_Diagnostics001.exe	2.5	2009/12/30 17:45:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:12:11	9547	TELM-06	1	Pass	0	0	0	166
M_TCP_Diagnostic	m_Diagnostics002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics002\Hardware\m_Diagnostics002.exe	2.2	2009/11/23 20:15:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:04:49	7870	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics003\Hardware\m_Diagnostics003.exe	2.2	2010/03/14 20:01:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:02:37	6786	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics004\Hardware\m_Diagnostics004.exe	2.3	2009/12/03 13:22:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:56:19	16010	TELM-06	1	Pass	0	0	0	128
M_TCP_Diagnostic	m_Diagnostics005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics005\Hardware\m_Diagnostics005.exe	2.1	2009/10/12 14:14:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:53:47	7749	TELM-06	1	Pass	0	0	0	13
M_TCP_Diagnostic	m_Diagnostics006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics006\Hardware\m_Diagnostics006.exe	2.0	2009/09/22 17:02:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:51:19	14791	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics007\Hardware\m_Diagnostics007.exe	2.1	2009/09/28 13:31:55	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:43:50	14993	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics008\Hardware\m_Diagnostics008.exe	2.0	2009/09/22 17:03:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:37:36	21314	TELM-06	1	Pass	0	0	0	31
M_TCP_Diagnostic	m_Diagnostics009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics009\Hardware\m_Diagnostics009.exe	2.5	2010/06/25 17:09:14	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:34:02	15864	TELM-06	1	Pass	0	0	0	23
M_TCP_Diagnostic	m_Diagnostics010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics010\Hardware\m_Diagnostics010.exe	2.5	2010/07/20 19:18:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:31:36	10511	TELM-06	1	Pass	0	0	0	10
M_TCP_Diagnostic	m_Diagnostics011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics011\Hardware\m_Diagnostics011.exe	2.4	2009/12/23 13:29:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:29:18	30001	TELM-06	1	Pass	0	0	0	19
M_TCP_Diagnostic	m_Diagnostics012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics012\Hardware\m_Diagnostics012.exe	2.3	2010/04/16 14:20:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:25:55	17182	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics013\Hardware\m_Diagnostics013.exe	2.5	2010/06/08 17:29:50	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:23:35	20545	TELM-06	1	Pass	0	0	0	15
M_TCP_Diagnostic	m_Diagnostics014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics014\Hardware\m_Diagnostics014.exe	2.4	2010/03/14 17:27:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:21:09	17743	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics015\Hardware\m_Diagnostics015.exe	2.4	2010/01/19 17:35:47	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:19:35	25478	TELM-06	1	Pass	0	0	0	29
M_TCP_Diagnostic	m_Diagnostics016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics016\Hardware\m_Diagnostics016.exe	2.0	2009/09/22 17:09:46	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:16:37	5861	TELM-06	1	Pass	0	0	0	1
M_TCP_Diagnostic	m_Diagnostics017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics017\Hardware\m_Diagnostics017.exe	2.3	2009/11/23 20:15:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:14:47	24893	TELM-06	1	Pass	0	0	0	32
M_TCP_Diagnostic	m_Diagnostics018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics018\Hardware\m_Diagnostics018.exe	2.2	2009/12/23 13:30:36	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:12:36	11799	TELM-06	1	Pass	0	0	0	9
M_TCP_Diagnostic	m_Diagnostics019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics019\Hardware\m_Diagnostics019.exe	2.1	2009/10/13 12:56:53	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:10:15	7429	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics020\Hardware\m_Diagnostics020.exe	2.2	2009/11/23 20:16:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:08:26	10249	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics021\Hardware\m_Diagnostics021.exe	2.1	2009/10/13 13:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:07:18	6019	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics022\Hardware\m_Diagnostics022.exe	2.4	2010/02/03 17:09:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:05:49	12695	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics023\Hardware\m_Diagnostics023.exe	2.2	2010/02/03 14:24:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:02:54	39085	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics024\Hardware\m_Diagnostics024.exe	2.5	2010/02/03 14:04:47	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:55:29	15648	TELM-06	1	Pass	0	0	0	20
M_TCP_Diagnostic	m_Diagnostics026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics026\Hardware\m_Diagnostics026.exe	2.1	2009/10/15 19:32:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:53:09	6200	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics027\Hardware\m_Diagnostics027.exe	2.2	2009/11/02 13:03:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:50:14	2922	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics028\Hardware\m_Diagnostics028.exe	2.4	2009/11/16 20:37:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:48:01	11448	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics029\Hardware\m_Diagnostics029.exe	2.1	2009/10/15 19:46:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:42:26	2746	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics030\Hardware\m_Diagnostics030.exe	2.5	2010/07/20 18:09:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:40:14	7305	TELM-06	1	Pass	0	0	0	277
M_TCP_Diagnostic	m_Diagnostics031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics031\Hardware\m_Diagnostics031.exe	2.2	2009/11/11 14:33:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:37:22	10676	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics032\Hardware\m_Diagnostics032.exe	2.3	2009/12/23 13:31:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:34:09	9761	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics034\Hardware\m_Diagnostics034.exe	2.4	2009/11/16 20:30:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:31:26	9026	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics035\Hardware\m_Diagnostics035.exe	2.2	2009/12/23 13:31:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:29:10	7802	TELM-06	1	Pass	0	0	0	7
M_TCP_Diagnostic	m_Diagnostics036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics036\Hardware\m_Diagnostics036.exe	2.3	2010/03/13 01:44:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:27:59	8664	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics037\Hardware\m_Diagnostics037.exe	2.4	2010/03/13 01:23:35	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:26:21	7051	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics038\Hardware\m_Diagnostics038.exe	2.3	2010/06/07 20:16:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:24:32	48684	TELM-06	1	Pass	0	0	0	18
M_TCP_Discover	m_DiscoverFull001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull001\Hardware\m_DiscoverFull001.exe	2.8	2010/05/21 19:22:39	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 13:28:54	44318	TELM-06	1	Pass	0	0	0	150
M_TCP_Discover	m_DiscoverFull002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull002\Hardware\m_DiscoverFull002.exe	2.6	2010/03/19 18:02:47	2.6	Up-to-date	Baseline 17_1	None	09/13/2010 07:58:01	144656	TELM-06	1	Pass	0	0	0	1815
M_TCP_Discover	m_DiscoverFull003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull003\Hardware\m_DiscoverFull003.exe	2.2	2010/02/25 20:00:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:22:45	12278	TELM-06	1	Pass	0	0	0	58
M_TCP_Discover	m_DiscoverFull004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull004\Hardware\m_DiscoverFull004.exe	2.4	2010/02/05 21:19:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 14:42:32	270973	TELM-06	1	Pass	0	0	0	3713
M_TCP_Discover	m_DiscoverFull005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull005\Hardware\m_DiscoverFull005.exe	2.2	2009/10/29 19:18:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:18:51	17360	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull006\Hardware\m_DiscoverFull006.exe	2.3	2010/02/05 21:19:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:13:37	130947	TELM-06	1	Pass	0	0	0	1727
M_TCP_Discover	m_DiscoverFull007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull007\Hardware\m_DiscoverFull007.exe	2.7	2009/12/14 21:43:07	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 13:15:06	13692	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull009\Hardware\m_DiscoverFull009.exe	2.3	2010/01/21 21:02:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:12:54	11070	TELM-06	1	Pass	0	0	0	65
M_TCP_Discover	m_DiscoverFull010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull010\Hardware\m_DiscoverFull010.exe	2.4	2010/05/12 17:07:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:09:21	8029	TELM-06	1	Pass	0	0	0	121
M_TCP_Discover	m_DiscoverFull011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull011\Hardware\m_DiscoverFull011.exe	2.3	2009/11/02 22:03:01	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:05:37	56737	TELM-06	1	Pass	0	0	0	9
M_TCP_Discover	m_DiscoverFull012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull012\Hardware\m_DiscoverFull012.exe	2.8	2010/06/29 12:07:18	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 15:51:26	366832	TELM-06	1	Pass	0	0	0	33
M_TCP_Discover	m_DiscoverFull013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull013\Hardware\m_DiscoverFull013.exe	2.2	2010/03/18 16:03:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:00:31	42502	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull014\Hardware\m_DiscoverFull014.exe	2.2	2010/03/18 16:03:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:57:48	24829	TELM-06	1	Pass	0	0	0	15
M_TCP_Discover	m_DiscoverFull015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull015\Hardware\m_DiscoverFull015.exe	2.1	2009/11/23 16:09:49	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:55:04	7809	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull016\Hardware\m_DiscoverFull016.exe	2.1	2009/11/03 20:58:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:52:50	8453	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull017\Hardware\m_DiscoverFull017.exe	2.1	2009/11/03 20:58:47	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:49:43	12251	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull022\Hardware\m_DiscoverFull022.exe	2.1	2009/11/23 16:11:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:46:55	8301	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull023\Hardware\m_DiscoverFull023.exe	2.1	2009/11/23 16:11:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:45:02	11485	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull024\Hardware\m_DiscoverFull024.exe	2.1	2009/11/23 16:12:04	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:41:44	12571	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull025\Hardware\m_DiscoverFull025.exe	2.1	2009/11/23 16:12:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:40:20	9087	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull029\Hardware\m_DiscoverFull029.exe	2.0	2009/11/13 21:27:59	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:38:14	12340	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull030\Hardware\m_DiscoverFull030.exe	2.3	2010/05/28 11:39:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:33:11	19157	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull031\Hardware\m_DiscoverFull031.exe	2.3	2010/05/28 12:38:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:30:03	35432	TELM-06	1	Pass	0	0	0	22
M_TCP_Discover	m_DiscoverFull032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull032\Hardware\m_DiscoverFull032.exe	2.0	2009/11/13 21:29:43	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:28:00	9237	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull033\Hardware\m_DiscoverFull033.exe	2.0	2009/11/13 21:30:18	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:26:47	9579	TELM-06	1	Pass	0	0	0	2

M_TCP_Discover	m_DiscoverFull034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull034\Hardware\m_DiscoverFull034.exe	2.2	2010/03/18 16:04:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:25:20	11266	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull035\Hardware\m_DiscoverFull035.exe	2.2	2010/02/09 20:53:53	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 02:01:37	43778	TELM-06	1	Pass	0	0	0	31
M_TCP_Discover	m_DiscoverFull036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull036\Hardware\m_DiscoverFull036.exe	2.2	2010/02/11 14:31:48	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 01:42:42	22074	TELM-06	1	Pass	0	0	0	38
M_TCP_Discover	m_DiscoverFull037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull037\Hardware\m_DiscoverFull037.exe	2.4	2010/02/03 21:18:36	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 03:35:47	35196	TELM-06	1	Pass	0	0	0	4
M_TCP_Discover	m_DiscoverFull038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull038\Hardware\m_DiscoverFull038.exe	2.1	2009/12/01 20:49:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:21:47	141690	TELM-06	1	Pass	0	0	0	7
M_TCP_EEPROM	m_Eeprom001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom001\Hardware\m_Eeprom001.exe	2.3	2009/12/23 16:23:59	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 00:02:00	11105	TELM-06	1	Pass	0	0	0	22
M_TCP_EEPROM	m_Eeprom002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom002\Hardware\m_Eeprom002.exe	2.3	2009/12/18 21:07:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 18:30:20	10794	TELM-06	1	Pass	0	0	0	938
M_TCP_EEPROM	m_Eeprom003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom003\Hardware\m_Eeprom003.exe	2.2	2010/01/05 18:03:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:58:58	16726	TELM-06	1	Pass	0	0	0	702
M_TCP_EEPROM	m_Eeprom004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom004\Hardware\m_Eeprom004.exe	2.2	2010/01/25 19:26:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:56:09	109621	TELM-06	1	Pass	0	0	0	583
M_TCP_EEPROM	m_Eeprom005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom005\Hardware\m_Eeprom005.exe	2.2	2010/01/19 15:32:05	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:51:28	172234	TELM-06	1	Pass	0	0	0	704
M_TCP_EEPROM	m_Eeprom006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom006\Hardware\m_Eeprom006.exe	2.2	2010/05/28 12:55:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:48:46	46885	TELM-06	1	Pass	0	0	0	6
M_TCP_EEPROM	m_Eeprom007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom007\Hardware\m_Eeprom007.exe	2.1	2010/02/18 18:38:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:46:32	5090	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency001\Hardware\m_Emergency001.exe	2.3	2009/11/03 21:54:33	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 12:16:04	9474	TELM-06	1	Pass	0	0	0	100
M_TCP_Emergency	m_Emergency002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency002\Hardware\m_Emergency002.exe	2.5	2010/06/25 20:19:36	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 11:47:05	13036	TELM-06	1	Pass	0	0	0	45
M_TCP_Emergency	m_Emergency003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency003\Hardware\m_Emergency003.exe	2.6	2010/06/29 14:19:35	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 11:12:39	28332	TELM-06	1	Pass	0	0	0	2564
M_TCP_Emergency	m_Emergency004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency004\Hardware\m_Emergency004.exe	2.3	2010/02/05 21:22:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 10:57:24	11821	TELM-06	1	Pass	0	0	0	4
M_TCP_Emergency	m_Emergency005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency005\Hardware\m_Emergency005.exe	2.2	2009/11/10 00:31:59	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 10:43:26	8148	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency006\Hardware\m_Emergency006.exe	2.1	2009/10/16 16:44:07	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 10:31:32	9310	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency007\Hardware\m_Emergency007.exe	2.1	2009/10/16 16:44:28	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 10:14:04	9537	TELM-06	1	Pass	0	0	0	33
M_TCP_Emergency	m_Emergency009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency009\Hardware\m_Emergency009.exe	2.1	2009/10/08 19:54:42	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 10:00:39	6003	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency010\Hardware\m_Emergency010.exe	2.2	2010/06/08 15:56:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 09:59:15	13158	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency011\Hardware\m_Emergency011.exe	2.0	2009/10/21 16:25:23	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 09:39:49	6739	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency012\Hardware\m_Emergency012.exe	2.1	2009/11/11 13:23:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 09:33:37	8294	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency013\Hardware\m_Emergency013.exe	2.1	2009/11/11 14:02:24	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 09:16:20	11805	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency014\Hardware\m_Emergency014.exe	2.3	2009/12/23 13:33:36	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 09:00:45	8180	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency015\Hardware\m_Emergency015.exe	2.3	2010/06/08 15:56:40	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:50:03	8998	TELM-06	1	Pass	0	0	0	8
M_TCP_Emergency	m_Emergency016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency016\Hardware\m_Emergency016.exe	2.2	2009/12/21 21:03:44	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:39:46	12569	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency017\Hardware\m_Emergency017.exe	2.2	2009/11/16 12:31:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:29:31	9870	TELM-06	1	Pass	0	0	0	21
M_TCP_Emergency	m_Emergency018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency018\Hardware\m_Emergency018.exe	2.3	2010/06/29 15:32:40	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:17:39	8054	TELM-06	1	Pass	0	0	0	22
M_TCP_Emergency	m_Emergency019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency019\Hardware\m_Emergency019.exe	2.2	2010/01/07 19:28:32	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:15:45	11145	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency020\Hardware\m_Emergency020.exe	2.6	2010/03/18 17:47:12	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 08:06:04	11447	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency021\Hardware\m_Emergency021.exe	2.3	2009/12/21 19:27:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:03:04	10487	TELM-06	1	Pass	0	0	0	83
M_TCP_Emergency	m_Emergency022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency022\Hardware\m_Emergency022.exe	2.2	2010/01/07 18:54:43	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:00:28	10360	TELM-06	1	Pass	0	0	0	82
M_TCP_Emergency	m_Emergency038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency038\Hardware\m_Emergency038.exe	2.2	2010/01/07 19:42:11	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 07:56:17	10412	TELM-06	1	Pass	0	0	0	14
M_TCP_Emergency	m_Emergency039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency039\Hardware\m_Emergency039.exe	2.2	2010/02/03 17:51:34	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 07:48:51	15425	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency040\Hardware\m_Emergency040.exe	2.0	2009/09/29 14:53:27	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 07:38:00	13949	TELM-06	1	Pass	0	0	0	7
M_TCP_Emergency	m_Emergency041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency041\Hardware\m_Emergency041.exe	2.0	2009/09/29 14:54:05	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 07:30:02	4972	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency042\Hardware\m_Emergency042.exe	2.0	2009/09/29 14:54:52	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 07:23:34	5054	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency043\Hardware\m_Emergency043.exe	2.4	2010/04/21 17:13:30	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 07:16:29	10387	TELM-06	1	Pass	0	0	0	9
M_TCP_Emergency	m_Emergency044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency044\Hardware\m_Emergency044.exe	2.5	2009/11/10 14:19:43	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 05:40:57	37654	TELM-06	1	Pass	0	0	0	18
M_TCP_Emergency	m_Emergency045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency045\Hardware\m_Emergency045.exe	2.3	2010/05/11 12:18:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 07:10:07	11106	TELM-06	1	Pass	0	0	0	16
M_TCP_General	m_General003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General003\Hardware\m_General003.exe	2.2	2009/10/13 14:50:43	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:00:22	67610	TELM-06	1	Pass	0	0	0	44646
M_TCP_General	m_General005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General005\Hardware\m_General005.exe	2.4	2009/12/01 19:02:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:55:10	67645	TELM-06	1	Pass	0	0	0	16811
M_TCP_General	m_General015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General015\Hardware\m_General015.exe	2.3	2009/12/01 19:02:59	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:53:06	36440	TELM-06	1	Pass	0	0	0	130639
M_TCP_General	m_General016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General016\Hardware\m_General016.exe	2.4	2010/03/29 10:43:22	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:49:53	8578	TELM-06	1	Pass	0	0	0	2
M_TCP_General	m_General017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General017\Hardware\m_General017.exe	2.3	2009/12/01 19:03:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:48:07	7334	TELM-06	1	Pass	0	0	0	10
M_TCP_General	m_General018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General018\Hardware\m_General018.exe	2.2	2010/02/03 18:07:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:45:19	29674	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General027\Hardware\m_General027.exe	2.7	2010/06/07 11:39:47	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 00:46:50	802135	TELM-06	1	Pass	0	0	0	21
M_TCP_General	m_General028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General028\Hardware\m_General028.exe	2.5	2010/06/08 11:48:52	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 04:55:51	636573	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General029\Hardware\m_General029.exe	2.3	2010/06/07 12:27:26	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 04:17:04	274170	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General030\Hardware\m_General030.exe	2.1	2010/06/07 12:30:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:43:17	17093	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General031\Hardware\m_General031.exe	2.5	2010/05/11 12:33:53	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:40:07	57928	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General032\Hardware\m_General032.exe	2.2	2010/02/05 18:56:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:38:04	29704	TELM-06	1	Pass	0	0	0	11
M_TCP_General	m_General033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General033\Hardware\m_General033.exe	2.0	2010/02/04 17:42:57	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:35:16	20666	TELM-06	1	Pass	0	0	0	4
M_TCP_General	m_General034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General034\Hardware\m_General034.exe	2.0	2010/01/22 22:45:34	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:28:58	30103	TELM-06	1	Pass	0	0	0	2
M_TCP_Initialize	m_Initialize001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize001\Hardware\m_Initialize001.exe	2.8	2010/05/12 12:34:35	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 14:45:13	10359	TELM-06	1	Pass	0	0	0	9
M_TCP_Initialize	m_Initialize002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize002\Hardware\m_Initialize002.exe	2.2	2009/10/20 12:16:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:43:20	19827	TELM-06	1	Pass	0	0	0	11
M_TCP_Initialize	m_Initialize003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize003\Hardware\m_Initialize003.exe	2.4	2010/06/08 15:57:16	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:41:28	23954	TELM-06	1	Pass	0	0	0	6
M_TCP_Initialize	m_Initialize004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize004\Hardware\m_Initialize004.exe	2.4	2010/06/08 15:57:44	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:39:42	40008	TELM-06	1	Pass	0	0	0	30
M_TCP_MedicalEvent	m_MedicalEvent001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent001\Hardware\m_MedicalEvent001.exe	2.4	2010/01/18 15:42:53	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:44:56	12692	TELM-06	1	Pass	0	0	0	36
M_TCP_MedicalEvent	m_MedicalEvent002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent002\Hardware\m_MedicalEvent002.exe	2.6	2010/02/09 19:03:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:42:04	73364	TELM-06	1	Pass	0	0	0	23
M_TCP_MedicalEvent	m_MedicalEvent003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent003\Hardware\m_MedicalEvent003.exe	2.4	2010/03/11 21:16:02	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:40:18	18627	TELM-06	1	Pass	0	0	0	24
M_TCP_MedicalEvent	m_MedicalEvent004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent004\Hardware\m_MedicalEvent004.exe	2.4	2010/01/18 15:44:34	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:39:11	12200	TELM-06	1	Pass	0	0	0	12
M_TCP_MedicalEvent	m_MedicalEvent008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent008\Hardware\m_MedicalEvent008.exe	2.4	2010/01/18 15:44:59	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:35:23	8443	TELM-06	1	Pass	0	0	0	6
M_TCP_MedicalEvent	m_MedicalEvent009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent009\Hardware\m_MedicalEvent009.exe	2.2	2010/01/18 15:46:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:33:23	7630	TELM-06	1	Pass	0	0	0	4
M_TCP_MedicalEvent	m_MedicalEvent010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent010\Hardware\m_MedicalEvent010.exe	2.5	2010/01/27 14:30:44	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:31:11	16607	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent011\Hardware\m_MedicalEvent011.exe	2.2	2010/01/18 15:47:08	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:29:26	15248	TELM-06	1	Pass	0	0	0	11
M_TCP_MedicalEvent	m_MedicalEvent012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent012\Hardware\m_MedicalEvent012.exe	2.3	2010/01/18 15:47:55	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:27:26	6190	TELM-06	1	Pass	0	0	0	3
M_TCP_MedicalEvent	m_MedicalEvent014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent014\Hardware\m_MedicalEvent014.exe	2.5	2010/09/08 13:20:31	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:25:30	15202	TELM-06	1	Pass	0	0	0	18
M_TCP_MedicalEvent	m_MedicalEvent015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent015\Hardware\m_MedicalEvent015.exe	2.9	2010/09/07 20:16:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 10:10:49	56459	TELM-06	1	Pass	0	0	0	152
M_TCP_MedicalEvent	m_MedicalEvent016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent016\Hardware\m_MedicalEvent016.exe	2.9	2010/05/12 18:44:37	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 17:13:24	283968	TELM-06	1	Pass	0	0	0	375
M_TCP_MedicalEvent	m_MedicalEvent017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent017\Hardware\m_MedicalEvent017.exe	2.8	2010/09/07 20:17:30	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 23:46:10	612173	TELM-06	1	Pass	0	0	0	13
M_TCP_MedicalEvent	m_MedicalEvent018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent018\Hardware\m_MedicalEvent018.exe	2.5	2010/01/18 15:50:21	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 16:51:25	272303	TELM-06	1	Pass	0	0	0	4627
M_TCP_MedicalEvent	m_MedicalEvent019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent019\Hardware\m_MedicalEvent019.exe	2.5	2010/01/18 15:50:59	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:20:18	7315	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent020\Hardware\m_MedicalEvent020.exe	2.3	2010/01/18 15:51:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:16:12	40575	TELM-06	1	Pass	0	0	0	9
M_TCP_MemoryTest	m_MemoryTest001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest001\Hardware\m_MemoryTest001.exe	2.7	2010/06/18 12:28:23	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 02:48:52	32962	TELM-06	1	Pass	0	0	0	33
M_TCP_MemoryTest	m_MemoryTest002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest002\Hardware\m_MemoryTest002.exe	2.4	2010/05/05 19:48:21	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:54:16	240655	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest003\Hardware\m_MemoryTest003.exe	2.3	2010/05/05 20:08:58	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 18:11:21	479563	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest004\Hardware\m_MemoryTest004.exe	2.3	2010/05/05 20:09:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:48:41	240123	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest005\Hardware\m_MemoryTest005.exe	2.3	2010/05/05 20:10:26	2.3	Up-to-date	Baseline 17_1	None	09/13/2010 07:49:37	478617	TELM-06	1	Pass	0	0	0	75
M_TCP_Network	m_Network001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network001\Hardware\m_Network001.exe	2.6	2010/05/21 15:45:43	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:20:28	24495	TELM-06	1	Pass	0	0	0	1461
M_TCP_Network	m_Network002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network002\Hardware\m_Network002.exe	2.6	2010/05/12 12:01:17	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 21:51:02	34690	TELM-06	1	Pass	0	0	0	82
M_TCP_Network	m_Network003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network003\Hardware\m_Network003.exe	2.1	2009/12/16 22:27:06	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:17:51	64847	TELM-06	1	Pass	0	0	0	86
M_TCP_Network	m_Network004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network004\Hardware\m_Network004.exe	2.1	2010/04/21 17:14:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:16:20	3100	TELM-06	1	Pass	0	0	0	6
M_TCP_Network	m_Network005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network005\Hardware\m_Network005.exe	2.3	2010/05/12 12:02:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:13:55	58249	TELM-06	1	Pass	0	0	0	50
M_TCP_Network	m_Network006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network006\Hardware\m_Network006.exe	2.5	2010/05/21 16:49:56	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:11:37	15258	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network007\Hardware\m_Network007.exe	2.5	2010/05/21 17:09:04	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:09:16	15472	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network008\Hardware\m_Network008.exe	2.5	2010/05/21 17:13:03	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:06:54	10551	TELM-06	1	Pass	0	0	0	1380
M_TCP_Network	m_Network009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network009\Hardware\m_Network009.exe	2.5	2010/05/21 17:21:30	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:05:03	10427	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network010\Hardware\m_Network010.exe	2.5	2010/05/21 17:34:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:03:05	10446	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network011\Hardware\m_Network011.exe	2.3	2010/05/12 12:03:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:57:50	209544	TELM-06	1	Pass	0	0	0	122
M_TCP_Network	m_Network013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network013\Hardware\m_Network013.exe	2.5	2010/07/01 13:00:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:54:04	141734	TELM-06	1	Pass	0	0	0	392
M_TCP_Network	m_Network014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network014\Hardware\m_Network014.exe	2.3	2010/05/12 12:06:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:51:16	26677	TELM-06	1	Pass	0	0	0	12
M_TCP_Network	m_Network015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network015\Hardware\m_Network015.exe	2.3	2010/05/24 13:53:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:47:13	35646	TELM-06	1	Pass	0	0	0	25
M_TCP_Network	m_Network016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network016\Hardware\m_Network016.exe	2.1	2010/05/12 12:09:43	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:45:13	55530	TELM-06	1	Pass	0	0	0	262
M_TCP_Network	m_Network017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network017\Hardware\m_Network017.exe	2.4	2010/06/07 13:19:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:43:23	12740	TELM-06	1	Pass	0	0	0	41
M_TCP_Network	m_Network018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network018\Hardware\m_Network018.exe	2.5	2010/06/07 13:14:52	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:41:54	21789	TELM-06	1	Pass	0	0	0	11
M_TCP_Network	m_Network019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network019\Hardware\m_Network019.exe	2.2	2010/05/12 12:12:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:39:29	56450	TELM-06	1	Pass	0	0	0	51
M_TCP_Network	m_Network020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network020\Hardware\m_Network020.exe	2.2	2010/05/12 12:14:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:36:24	60264	TELM-06	1	Pass	0	0	0	67
M_TCP_Network	m_Network021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network021\Hardware\m_Network021.exe	2.2	2010/05/12 12:16:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:33:28	60339	TELM-06	1	Pass	0	0	0	133
M_TCP_Network	m_Network022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network022\Hardware\m_Network022.exe	2.2	2010/05/12 12:17:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:31:04	60277	TELM-06	1	Pass	0	0	0	175
M_TCP_Network	m_Network023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network023\Hardware\m_Network023.exe	2.1	2010/05/12 12:18:23	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:29:28	19521	TELM-06	1	Pass	0	0	0	454
M_TCP_Network	m_Network024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network024\Hardware\m_Network024.exe	2.2	2010/05/24 14:41:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:27:49	47719	TELM-06	1	Pass	0	0	0	46
M_TCP_Network	m_Network025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network025\Hardware\m_Network025.exe	2.1	2010/05/12 12:20:01	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:26:17	19511	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network026\Hardware\m_Network026.exe	2.3	2010/05/12 12:20:56	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:25:13	20539	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network027\Hardware\m_Network027.exe	2.1	2010/05/12 12:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:23:47	21134	TELM-06	1	Pass	0	0	0	7
M_TCP_Network	m_Network028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network028\Hardware\m_Network028.exe	2.1	2010/06/07 13:22:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:22:07	14363	TELM-06	1	Pass	0	0	0	22
M_TCP_Network	m_Network029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network029\Hardware\m_Network029.exe	2.4	2010/02/09 21:21:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 15:27:18	271740	TELM-06	1	Pass	0	0	0	48
M_TCP_Open	m_Open001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open001\Hardware\m_Open001.exe	2.11	2010/05/20 13:50:26	2.11	Up-to-date	Baseline 17_1	None	09/09/2010 11:20:26	31704	TELM-06	1	Pass	0	0	0	335
M_TCP_Open	m_Open002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open002\Hardware\m_Open002.exe	2.3	2010/01/18 15:52:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:19:02	18626	TELM-06	1	Pass	0	0	0	31
M_TCP_Open	m_Open003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open003\Hardware\m_Open003.exe	2.0	2009/10/15 19:09:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:17:59	6424	TELM-06	1	Pass	0	0	0	3

M_TCP_Open	m_Open004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open004\Hardware\m_Open004.exe	2.6	2010/06/22 20:01:21	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 11:16:01	13174	TELM-06	1	Pass	0	0	0	21
M_TCP_Open	m_Open005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open005\Hardware\m_Open005.exe	2.3	2010/06/22 17:40:45	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:12:57	16887	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open006\Hardware\m_Open006.exe	2.2	2010/03/15 14:26:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:10:39	21283	TELM-06	1	Pass	0	0	0	1173
M_TCP_Open	m_Open007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open007\Hardware\m_Open007.exe	2.7	2010/03/15 14:46:20	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 11:06:45	17576	TELM-06	1	Pass	0	0	0	1181
M_TCP_Open	m_Open008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open008\Hardware\m_Open008.exe	2.1	2009/10/30 14:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:02:32	7857	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open009\Hardware\m_Open009.exe	2.7	2010/06/16 00:36:52	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 10:57:55	12442	TELM-06	1	Pass	0	0	0	1268
M_TCP_Open	m_Open010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open010\Hardware\m_Open010.exe	2.6	2010/03/15 15:05:50	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 10:54:45	17503	TELM-06	1	Pass	0	0	0	3965
M_TCP_Open	m_Open011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open011\Hardware\m_Open011.exe	2.2	2010/02/10 13:31:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:51:17	6434	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open012\Hardware\m_Open012.exe	2.4	2010/02/26 17:01:14	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:45:12	15608	TELM-06	1	Pass	0	0	0	10
M_TCP_Open	m_Open013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open013\Hardware\m_Open013.exe	2.5	2010/02/10 13:34:06	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:40:17	9500	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open014\Hardware\m_Open014.exe	2.6	2010/02/10 13:35:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 10:37:25	8444	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open015\Hardware\m_Open015.exe	2.0	2010/01/28 19:00:26	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 10:35:13	8910	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open030\Hardware\m_Open030.exe	2.1	2010/05/10 13:57:56	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 03:33:10	57984	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open031\Hardware\m_Open031.exe	2.7	2009/11/19 21:48:20	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 03:28:31	63789	TELM-06	1	Pass	0	0	0	13
M_TCP_Open	m_Open032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open032\Hardware\m_Open032.exe	2.3	2010/02/03 22:17:27	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:35:19	32332	TELM-06	1	Pass	0	0	0	18
M_TCP_Open	m_Open033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open033\Hardware\m_Open033.exe	2.3	2010/02/03 22:32:54	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:06:44	9941	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open034\Hardware\m_Open034.exe	2.2	2010/02/17 14:19:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:32:31	56986	TELM-06	1	Pass	0	0	0	42
M_TCP_Open	m_Open035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open035\Hardware\m_Open035.exe	2.2	2010/02/17 14:10:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:31:07	9757	TELM-06	1	Pass	0	0	0	32
M_TCP_Open	m_Open036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open036\Hardware\m_Open036.exe	2.1	2009/10/21 15:52:46	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 03:26:13	36305	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open037\Hardware\m_Open037.exe	2.2	2009/10/21 15:51:07	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 03:13:52	31336	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open038\Hardware\m_Open038.exe	2.7	2010/02/17 14:00:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 10:28:43	78270	TELM-06	1	Pass	0	0	0	46
M_TCP_Open	m_Open039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open039\Hardware\m_Open039.exe	2.0	2009/11/09 14:20:44	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 10:27:01	9094	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open040\Hardware\m_Open040.exe	2.1	2010/02/01 19:45:37	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:22:14	17037	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open041\Hardware\m_Open041.exe	2.0	2009/11/09 14:21:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 14:59:14	513795	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open042\Hardware\m_Open042.exe	2.4	2010/07/02 12:38:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:17:28	15930	TELM-06	1	Pass	0	0	0	52
M_TCP_Polling	m_Polling016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling016\Hardware\m_Polling016.exe	2.0	2009/09/23 19:25:05	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:37:48	49140	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling017\Hardware\m_Polling017.exe	2.1	2010/01/26 20:04:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:06:28	6223	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling018\Hardware\m_Polling018.exe	2.1	2010/01/26 20:05:23	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:35:25	81494	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling024\Hardware\m_Polling024.exe	2.5	2010/01/26 20:05:45	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:32:14	78591	TELM-06	1	Pass	0	0	0	10
M_TCP_Polling	m_Polling025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling025\Hardware\m_Polling025.exe	2.1	2009/11/24 20:51:39	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:30:15	46139	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling026\Hardware\m_Polling026.exe	2.2	2009/12/03 14:59:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:27:09	19449	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling040\Hardware\m_Polling040.exe	2.1	2010/01/26 20:06:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:25:59	6044	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling045\Hardware\m_Polling045.exe	2.1	2010/01/26 20:06:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:24:02	35996	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling046\Hardware\m_Polling046.exe	2.6	2010/04/16 14:21:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:22:12	47148	TELM-06	1	Pass	0	0	0	38
M_TCP_Polling	m_Polling047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling047\Hardware\m_Polling047.exe	2.1	2010/01/26 20:07:28	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:20:14	36178	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling048\Hardware\m_Polling048.exe	2.3	2009/11/24 20:53:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:17:37	9964	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling049\Hardware\m_Polling049.exe	2.4	2010/03/10 19:44:21	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:16:03	46193	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling050\Hardware\m_Polling050.exe	2.6	2010/06/29 17:44:57	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:12:42	22757	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling065\Hardware\m_Polling065.exe	2.6	2010/06/09 18:52:38	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:11:05	42375	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling066\Hardware\m_Polling066.exe	2.3	2009/11/09 22:57:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:09:03	52626	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling067\Hardware\m_Polling067.exe	2.1	2009/10/22 14:28:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:06:35	84810	TELM-06	1	Pass	0	0	0	11
M_TCP_Polling	m_Polling068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling068\Hardware\m_Polling068.exe	2.3	2009/11/05 18:23:27	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:04:43	7055	TELM-06	1	Pass	0	0	0	8
M_TCP_Polling	m_Polling076	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling076\Hardware\m_Polling076.exe	2.1	2009/11/03 15:55:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:02:17	10093	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling077	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling077\Hardware\m_Polling077.exe	2.1	2009/11/05 18:34:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 23:06:05	527412	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling080	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling080\Hardware\m_Polling080.exe	2.2	2009/11/05 18:24:04	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 22:58:46	277402	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling081	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling081\Hardware\m_Polling081.exe	2.3	2010/05/27 14:19:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:56:40	24160	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling082	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling082\Hardware\m_Polling082.exe	2.2	2009/11/02 18:25:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:52:48	9483	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling083	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling083\Hardware\m_Polling083.exe	2.3	2009/11/02 18:45:07	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:49:47	19464	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling085	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling085\Hardware\m_Polling085.exe	2.1	2009/10/22 13:55:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 22:35:25	529324	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling088	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling088\Hardware\m_Polling088.exe	2.5	2010/05/12 17:59:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:47:39	23416	TELM-06	1	Pass	0	0	0	27
M_TCP_Polling	m_Polling089	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling089\Hardware\m_Polling089.exe	2.7	2010/05/11 19:28:17	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 13:44:23	97045	TELM-06	1	Pass	0	0	0	26
M_TCP_Polling	m_Polling090	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling090\Hardware\m_Polling090.exe	2.6	2010/02/15 20:02:24	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 13:41:55	77565	TELM-06	1	Pass	0	0	0	32
M_TCP_Polling	m_Polling091	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling091\Hardware\m_Polling091.exe	2.6	2010/02/15 20:16:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 13:40:21	13039	TELM-06	1	Pass	0	0	0	13
M_TCP_Polling	m_Polling092	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling092\Hardware\m_Polling092.exe	2.1	2009/11/05 18:41:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 22:02:27	277220	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling093	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling093\Hardware\m_Polling093.exe	2.0	2009/11/09 18:30:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:37:08	16786	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling094	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling094\Hardware\m_Polling094.exe	2.0	2009/11/09 18:30:49	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 16:19:32	307312	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling095	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling095\Hardware\m_Polling095.exe	2.0	2009/11/09 18:31:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:33:30	8406	TELM-06	1	Pass	0	0	0	2
M_TCP_PowerInhibit	m_Power001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power001\Hardware\m_Power001.exe	2.2	2010/06/07 13:10:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:42:16	58686	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power002\Hardware\m_Power002.exe	2.2	2010/01/05 19:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:38:17	10791	TELM-06	1	Pass	0	0	0	549
M_TCP_PowerInhibit	m_Power003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power003\Hardware\m_Power003.exe	2.4	2010/06/22 14:04:45	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:35:23	54050	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power004\Hardware\m_Power004.exe	2.3	2010/03/22 14:26:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:33:28	18116	TELM-06	1	Pass	0	0	0	46
M_TCP_PowerInhibit	m_Power005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power005\Hardware\m_Power005.exe	2.6	2010/07/02 12:34:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 12:36:59	135748	TELM-06	1	Pass	0	0	0	310
M_TCP_PowerInhibit	m_Power006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power006\Hardware\m_Power006.exe	2.6	2010/06/22 12:27:46	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:28:18	192896	TELM-06	1	Pass	0	0	0	2904
M_TCP_PowerInhibit	m_Power007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power007\Hardware\m_Power007.exe	2.1	2009/12/14 16:39:57	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:26:12	19244	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power008\Hardware\m_Power008.exe	2.2	2009/12/14 16:44:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:23:28	34721	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power009\Hardware\m_Power009.exe	2.4	2010/07/01 15:00:10	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:21:32	11986	TELM-06	1	Pass	0	0	0	9
M_TCP_PowerInhibit	m_Power010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power010\Hardware\m_Power010.exe	2.1	2010/02/16 22:18:45	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:18:56	11859	TELM-06	1	Pass	0	0	0	26
M_TCP_PowerInhibit	m_Power011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power011\Hardware\m_Power011.exe	2.1	2010/03/11 19:04:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:16:39	91211	TELM-06	1	Pass	0	0	0	116
M_TCP_RawMode	m_RawMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode001\Hardware\m_RawMode001.exe	2.4	2010/01/26 20:08:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:54:24	7493	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode002\Hardware\m_RawMode002.exe	2.6	2010/01/27 13:00:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:53:00	9501	TELM-06	1	Pass	0	0	0	87
M_TCP_RawMode	m_RawMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode003\Hardware\m_RawMode003.exe	2.5	2010/01/27 13:03:49	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:51:46	11287	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode004\Hardware\m_RawMode004.exe	2.6	2010/01/27 13:09:25	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:50:40	7233	TELM-06	1	Pass	0	0	0	2
M_TCP_RawMode	m_RawMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode005\Hardware\m_RawMode005.exe	2.3	2010/01/26 20:10:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:48:53	13472	TELM-06	1	Pass	0	0	0	9
M_TCP_RawMode	m_RawMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode006\Hardware\m_RawMode006.exe	2.2	2010/01/26 20:11:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:08:19	8985	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode007\Hardware\m_RawMode007.exe	2.3	2010/06/09 21:03:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:47:55	11439	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode008\Hardware\m_RawMode008.exe	2.4	2010/01/27 13:14:20	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:45:35	16183	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode009\Hardware\m_RawMode009.exe	2.2	2009/12/14 16:51:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:44:11	15764	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode010\Hardware\m_RawMode010.exe	2.1	2009/10/27 17:32:20	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:43:05	3187	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode011\Hardware\m_RawMode011.exe	2.2	2010/02/01 16:15:38	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:41:52	8553	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode012\Hardware\m_RawMode012.exe	2.3	2010/01/26 20:13:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:40:32	11605	TELM-06	1	Pass	0	0	0	3
M_TCP_Reset	m_Reset001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset001\Hardware\m_Reset001.exe	2.3	2010/01/06 21:25:12	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:03:33	8972	TELM-06	1	Pass	0	0	0	23
M_TCP_Reset	m_Reset002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset002\Hardware\m_Reset002.exe	2.4	2010/01/28 21:52:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:01:17	20418	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security002\Hardware\m_Security002.exe	2.9	2010/06/03 15:18:25	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 20:59:36	457425	TELM-06	1	Pass	0	0	0	154
M_TCP_Security	m_Security003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security003\Hardware\m_Security003.exe	2.4	2010/02/17 16:20:41	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:38:56	6887	TELM-06	1	Pass	0	0	0	73
M_TCP_Security	m_Security004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security004\Hardware\m_Security004.exe	2.4	2010/02/15 15:01:00	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:33:47	253972	TELM-06	1	Pass	0	0	0	112
M_TCP_Security	m_Security005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security005\Hardware\m_Security005.exe	2.2	2010/02/17 16:23:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:32:39	4140	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security006\Hardware\m_Security006.exe	2.4	2010/02/17 14:44:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:29:06	152688	TELM-06	1	Pass	0	0	0	108
M_TCP_Security	m_Security007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security007\Hardware\m_Security007.exe	2.6	2010/02/18 15:13:38	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:26:16	97899	TELM-06	1	Pass	0	0	0	126
M_TCP_Security	m_Security008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security008\Hardware\m_Security008.exe	2.3	2010/02/17 16:28:05	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:25:01	6825	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security009\Hardware\m_Security009.exe	2.3	2010/02/17 16:32:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:23:55	6330	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security010\Hardware\m_Security010.exe	2.3	2010/02/18 17:45:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:22:19	38597	TELM-06	1	Pass	0	0	0	113
M_TCP_Security	m_Security011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security011\Hardware\m_Security011.exe	2.5	2010/06/03 18:16:10	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:20:47	20209	TELM-06	1	Pass	0	0	0	58
M_TCP_Security	m_Security012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security012\Hardware\m_Security012.exe	2.9	2010/02/18 15:14:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 22:19:37	4811	TELM-06	1	Pass	0	0	0	132
M_TCP_Security	m_Security013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security013\Hardware\m_Security013.exe	2.5	2010/02/18 15:15:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:18:30	11526	TELM-06	1	Pass	0	0	0	140
M_TCP_Security	m_Security014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security014\Hardware\m_Security014.exe	2.2	2010/06/03 18:21:27	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 04:13:59	15000	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security015\Hardware\m_Security015.exe	2.4	2010/02/17 17:58:26	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 00:34:17	596190	TELM-06	1	Pass	0	0	0	34
M_TCP_Security	m_Security016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security016\Hardware\m_Security016.exe	2.2	2010/06/08 19:56:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:16:38	10646	TELM-06	1	Pass	0	0	0	146
M_TCP_Security	m_Security017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security017\Hardware\m_Security017.exe	2.2	2009/12/08 17:29:14	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:14:52	14594	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security018\Hardware\m_Security018.exe	2.4	2010/03/18 16:05:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:12:44	40597	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security019\Hardware\m_Security019.exe	2.3	2010/03/26 13:43:17	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:09:51	90420	TELM-06	1	Pass	0	0	0	1230
M_TCP_Security	m_Security020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security020\Hardware\m_Security020.exe	2.3	2010/06/03 18:22:30	2.3	Up-to-date	Baseline 17_1	None	09/13/2010 09:13:01	22043	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security021\Hardware\m_Security021.exe	2.5	2010/06/03 18:37:52	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:06:29	40981	TELM-06	1	Pass	0	0	0	81
M_TCP_Security	m_Security022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security022\Hardware\m_Security022.exe	2.9	2010/06/18 19:05:16	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 13:10:21	290972	TELM-06	1	Pass	0	0	0	55496
M_TCP_Security	m_Security023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security023\Hardware\m_Security023.exe	2.12	2010/06/10 17:30:47	2.12	Up-to-date	Baseline 17_1	None	09/11/2010 00:30:53	51625	TELM-06	1	Pass	0	0	0	1119
M_TCP_Security	m_Security024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security024\Hardware\m_Security024.exe	2.3	2010/03/26 13:06:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 20:05:20	30725	TELM-06	1	Pass	0	0	0	4
M_TCP_Security	m_Security025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security025\Hardware\m_Security025.exe	2.3	2010/06/10 14:39:09	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 19:40:45	48592	TELM-06	1	Pass	0	0	0	158
M_TCP_Security	m_Security026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security026\Hardware\m_Security026.exe	2.3	2010/02/18 15:15:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:04:40	4329	TELM-06	1	Pass	0	0	0	60
M_TCP_Security	m_Security027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security027\Hardware\m_Security027.exe	2.1	2010/06/29 19:03:00	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:02:52	42393	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security028\Hardware\m_Security028.exe	2.1	2010/06/29 19:26:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:00:50	47226	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security029\Hardware\m_Security029.exe	2.2	2010/03/11 18:29:14	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:59:19	20429	TELM-06	1	Pass	0	0	0	46
M_TCP_Security	m_Security030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security030\Hardware\m_Security030.exe	2.2	2010/02/08 13:02:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:57:51	11759	TELM-06	1	Pass	0	0	0	44

M_TCP_Security	m_Security031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security031\Hardware\m_Security031.exe	2.1	2010/06/03 18:44:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:54:16	22951	TELM-06	1	Pass	0	0	0	16
M_TCP_Security	m_Security032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security032\Hardware\m_Security032.exe	2.1	2010/02/03 12:51:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:51:21	8317	TELM-06	1	Pass	0	0	0	11
M_TCP_Security	m_Security033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security033\Hardware\m_Security033.exe	2.1	2010/01/27 21:32:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:49:45	14033	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security034\Hardware\m_Security034.exe	2.3	2010/03/26 13:19:51	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:45:39	118775	TELM-06	1	Pass	0	0	0	15
M_TCP_Security	m_Security035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security035\Hardware\m_Security035.exe	2.0	2010/01/14 21:18:30	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 21:42:12	17687	TELM-06	1	Pass	0	0	0	30
M_TCP_Security	m_Security036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security036\Hardware\m_Security036.exe	2.5	2010/05/07 14:41:43	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:38:52	86746	TELM-06	1	Pass	0	0	0	10
M_TCP_Security	m_Security037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security037\Hardware\m_Security037.exe	2.6	2010/06/03 18:54:41	2.6	Up-to-date	Baseline 17_1	None	09/11/2010 04:12:06	51037	TELM-06	1	Pass	0	0	0	119
M_TCP_Security	m_Security038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security038\Hardware\m_Security038.exe	2.9	2010/06/07 13:02:07	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 19:01:04	167401	TELM-06	1	Pass	0	0	0	170
M_TCP_Security	m_Security039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security039\Hardware\m_Security039.exe	2.2	2010/02/18 15:16:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:35:45	56645	TELM-06	1	Pass	0	0	0	55
M_TCP_Security	m_Security040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security040\Hardware\m_Security040.exe	2.3	2010/06/03 20:43:31	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:29:52	278601	TELM-06	1	Pass	0	0	0	33
M_TCP_Security	m_Security041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security041\Hardware\m_Security041.exe	2.6	2010/04/21 10:55:08	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 21:25:53	65816	TELM-06	1	Pass	0	0	0	77
M_TCP_Security	m_Security042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security042\Hardware\m_Security042.exe	2.1	2010/04/08 12:57:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:22:35	58241	TELM-06	1	Pass	0	0	0	48
M_TCP_Security	m_Security043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security043\Hardware\m_Security043.exe	2.0	2010/02/16 20:27:42	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 21:19:48	68945	TELM-06	1	Pass	0	0	0	56
M_TCP_Security	m_Security044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security044\Hardware\m_Security044.exe	2.2	2010/05/13 19:14:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:16:33	70744	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security045\Hardware\m_Security045.exe	2.3	2010/05/12 17:38:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:10:08	71169	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security046\Hardware\m_Security046.exe	2.2	2010/06/07 13:05:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:05:21	3332	TELM-06	1	Pass	0	0	0	30
M_TCP_WirelessWakeup	m_Wakeup001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup001\Hardware\m_Wakeup001.exe	2.3	2010/06/21 17:44:25	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 21:22:59	291397	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup002\Hardware\m_Wakeup002.exe	2.5	2010/05/12 16:16:45	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:24:17	35420	TELM-06	1	Pass	0	0	0	12
M_TCP_WirelessWakeup	m_Wakeup003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup003\Hardware\m_Wakeup003.exe	2.2	2010/03/10 17:59:29	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:20:39	14370	TELM-06	1	Pass	0	0	0	11
M_TCP_WirelessWakeup	m_Wakeup004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup004\Hardware\m_Wakeup004.exe	2.1	2009/12/08 16:08:20	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 04:15:00	19126	TELM-06	1	Pass	0	0	0	31
M_TCP_WirelessWakeup	m_Wakeup005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup005\Hardware\m_Wakeup005.exe	2.3	2010/06/08 21:32:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:17:32	20349	TELM-06	1	Pass	0	0	0	40
M_TCP_WirelessWakeup	m_Wakeup006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup006\Hardware\m_Wakeup006.exe	2.3	2010/01/27 00:35:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:15:56	10212	TELM-06	1	Pass	0	0	0	55
M_TCP_WirelessWakeup	m_Wakeup007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup007\Hardware\m_Wakeup007.exe	2.1	2009/12/04 19:28:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:14:54	12775	TELM-06	1	Pass	0	0	0	3
M_TCP_WirelessWakeup	m_Wakeup008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup008\Hardware\m_Wakeup008.exe	2.5	2010/05/07 15:41:21	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:12:42	54725	TELM-06	1	Pass	0	0	0	24
M_TCP_WirelessWakeup	m_Wakeup009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup009\Hardware\m_Wakeup009.exe	2.6	2010/05/12 16:17:49	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:10:36	52093	TELM-06	1	Pass	0	0	0	32
M_TCP_WirelessWakeup	m_Wakeup010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup010\Hardware\m_Wakeup010.exe	2.4	2010/05/12 16:20:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:07:25	45176	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup011\Hardware\m_Wakeup011.exe	2.3	2010/04/06 18:07:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:04:12	27904	TELM-06	1	Pass	0	0	0	25
M_TCP_WirelessWakeup	m_Wakeup012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup012\Hardware\m_Wakeup012.exe	2.6	2010/05/12 16:20:45	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:02:35	48713	TELM-06	1	Pass	0	0	0	45
M_TCP_WirelessWakeup	m_Wakeup013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup013\Hardware\m_Wakeup013.exe	2.2	2010/02/24 12:20:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:58:02	127480	TELM-06	1	Pass	0	0	0	2
M_TCP_WirelessWakeup	m_Wakeup014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode3\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup014\Hardware\m_Wakeup014.exe	2.3	2010/06/04 11:50:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:55:52	74549	TELM-06	1	Pass	0	0	0	30
M_TCP_AntennaSelection	m_Antenna001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna001\Hardware\m_Antenna001.exe	2.4	2010/06/16 12:31:48	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:29:36	238627	TELM-06	1	Pass	0	0	0	127
M_TCP_AntennaSelection	m_Antenna002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna002\Hardware\m_Antenna002.exe	2.0	2010/06/16 12:29:50	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 09:11:14	287108	TELM-06	1	Pass	0	0	0	99
M_TCP_Channel	m_ChannelRecovery001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery001\Hardware\m_ChannelRecovery001.exe	2.5	2010/04/08 14:25:18	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:26:44	14994	TELM-06	1	Pass	0	0	0	26
M_TCP_Channel	m_ChannelRecovery002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery002\Hardware\m_ChannelRecovery002.exe	2.2	2009/11/30 22:31:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:25:49	17596	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery003\Hardware\m_ChannelRecovery003.exe	2.2	2009/11/30 22:32:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:24:11	17489	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery004\Hardware\m_ChannelRecovery004.exe	2.2	2010/05/26 12:04:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:22:36	14577	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery005\Hardware\m_ChannelRecovery005.exe	2.5	2010/06/21 18:34:44	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:21:20	14383	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery006\Hardware\m_ChannelRecovery006.exe	2.4	2010/05/07 17:09:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:20:10	15770	TELM-06	1	Pass	0	0	0	36
M_TCP_Channel	m_ChannelRecovery007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery007\Hardware\m_ChannelRecovery007.exe	2.7	2010/05/11 11:49:24	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 15:18:35	27520	TELM-06	1	Pass	0	0	0	89
M_TCP_Channel	m_ChannelRecovery008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery008\Hardware\m_ChannelRecovery008.exe	2.6	2010/05/13 15:05:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:16:05	26923	TELM-06	1	Pass	0	0	0	574
M_TCP_Channel	m_ChannelRecovery009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery009\Hardware\m_ChannelRecovery009.exe	2.6	2010/06/22 15:16:15	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:12:43	51091	TELM-06	1	Pass	0	0	0	342
M_TCP_Channel	m_ChannelRecovery010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery010\Hardware\m_ChannelRecovery010.exe	2.8	2010/05/26 15:48:50	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 15:10:15	25121	TELM-06	1	Pass	0	0	0	159
M_TCP_Channel	m_ChannelRecovery011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery011\Hardware\m_ChannelRecovery011.exe	2.5	2010/05/13 12:31:10	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:08:51	15975	TELM-06	1	Pass	0	0	0	15
M_TCP_Channel	m_ChannelRecovery012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery012\Hardware\m_ChannelRecovery012.exe	2.6	2010/09/07 20:36:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:06:35	48015	TELM-06	1	Pass	0	0	0	220
M_TCP_Channel	m_ChannelRecovery013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery013\Hardware\m_ChannelRecovery013.exe	2.2	2010/01/05 19:17:54	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:03:15	24074	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery014\Hardware\m_ChannelRecovery014.exe	2.8	2010/05/13 13:19:47	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 15:00:53	53411	TELM-06	1	Pass	0	0	0	23
M_TCP_Channel	m_ChannelRecovery015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery015\Hardware\m_ChannelRecovery015.exe	2.6	2010/06/03 18:01:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:58:33	12475	TELM-06	1	Pass	0	0	0	88
M_TCP_Channel	m_ChannelRecovery016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery016\Hardware\m_ChannelRecovery016.exe	2.4	2010/02/11 20:10:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:56:48	37916	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery017\Hardware\m_ChannelRecovery017.exe	2.3	2010/02/11 17:38:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:55:32	10403	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery018\Hardware\m_ChannelRecovery018.exe	2.3	2010/02/01 19:41:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:54:21	18002	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery020\Hardware\m_ChannelRecovery020.exe	2.7	2010/06/23 17:42:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 14:52:37	37596	TELM-06	1	Pass	0	0	0	109
M_TCP_Channel	m_ChannelRecovery021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery021\Hardware\m_ChannelRecovery021.exe	2.3	2010/04/13 18:33:24	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:50:55	25170	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery022\Hardware\m_ChannelRecovery022.exe	2.3	2010/04/13 18:36:50	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:49:44	19765	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery023\Hardware\m_ChannelRecovery023.exe	2.5	2010/04/13 18:41:23	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:47:46	29961	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery024\Hardware\m_ChannelRecovery024.exe	2.4	2010/04/07 12:33:58	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:45:26	19840	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery025\Hardware\m_ChannelRecovery025.exe	2.2	2010/02/01 19:43:28	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:43:42	11145	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery026\Hardware\m_ChannelRecovery026.exe	2.2	2010/02/11 21:01:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:41:54	11511	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery027\Hardware\m_ChannelRecovery027.exe	2.2	2010/02/03 17:18:54	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:40:25	13679	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery028\Hardware\m_ChannelRecovery028.exe	2.2	2010/02/03 17:19:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:38:39	11582	TELM-06	1	Pass	0	0	0	10
M_TCP_Channel	m_ChannelRecovery029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery029\Hardware\m_ChannelRecovery029.exe	2.2	2009/12/18 20:13:33	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:36:47	13580	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery030\Hardware\m_ChannelRecovery030.exe	2.10	2010/06/23 21:15:31	2.10	Up-to-date	Baseline 17_1	None	09/09/2010 14:33:35	18452	TELM-06	1	Pass	0	0	0	30
M_TCP_Channel	m_ChannelRecovery031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery031\Hardware\m_ChannelRecovery031.exe	2.6	2010/04/13 18:46:58	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:31:03	18847	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery032\Hardware\m_ChannelRecovery032.exe	2.5	2010/06/23 21:50:13	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:28:07	18017	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery033\Hardware\m_ChannelRecovery033.exe	2.3	2010/04/07 17:51:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:26:08	19181	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery038\Hardware\m_ChannelRecovery038.exe	2.6	2010/05/26 15:55:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:24:54	20842	TELM-06	1	Pass	0	0	0	175
M_TCP_Channel	m_ChannelRecovery039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery039\Hardware\m_ChannelRecovery039.exe	2.3	2010/02/10 20:47:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:23:01	12273	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery040\Hardware\m_ChannelRecovery040.exe	2.5	2010/06/07 12:54:38	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:20:53	15501	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery041\Hardware\m_ChannelRecovery041.exe	2.1	2009/10/22 13:18:38	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:17:49	38494	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery042\Hardware\m_ChannelRecovery042.exe	2.3	2009/12/03 15:57:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:15:39	21921	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery043\Hardware\m_ChannelRecovery043.exe	2.6	2010/06/24 18:37:58	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:13:07	55948	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery045\Hardware\m_ChannelRecovery045.exe	2.2	2009/11/12 21:22:11	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:10:50	11397	TELM-06	1	Pass	0	0	0	43
M_TCP_Channel	m_ChannelRecovery046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery046\Hardware\m_ChannelRecovery046.exe	2.3	2009/11/16 18:40:10	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:08:49	12645	TELM-06	1	Pass	0	0	0	49
M_TCP_Channel	m_ChannelRecovery047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery047\Hardware\m_ChannelRecovery047.exe	2.1	2009/11/16 18:58:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:05:40	52199	TELM-06	1	Pass	0	0	0	95
M_TCP_Channel	m_ChannelRecovery048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery048\Hardware\m_ChannelRecovery048.exe	2.3	2010/06/24 18:46:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:03:06	62420	TELM-06	1	Pass	0	0	0	156
M_TCP_Channel	m_ChannelRecovery049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery049\Hardware\m_ChannelRecovery049.exe	2.5	2010/03/26 13:55:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:01:00	36225	TELM-06	1	Pass	0	0	0	231
M_TCP_Channel	m_ChannelRecovery050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery050\Hardware\m_ChannelRecovery050.exe	2.2	2009/11/20 13:38:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:55:51	18609	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery051\Hardware\m_ChannelRecovery051.exe	2.3	2010/06/24 19:40:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:51:05	82847	TELM-06	1	Pass	0	0	0	196
M_TCP_Channel	m_ChannelRecovery053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery053\Hardware\m_ChannelRecovery053.exe	2.3	2010/06/24 21:57:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:49:15	31002	TELM-06	1	Pass	0	0	0	96
M_TCP_Channel	m_ChannelRecovery054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery054\Hardware\m_ChannelRecovery054.exe	2.4	2010/06/24 22:31:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:47:29	9005	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery055\Hardware\m_ChannelRecovery055.exe	2.4	2010/02/11 14:12:25	2.4	Up-to-date	Baseline 17_1	None	09/13/2010 08:34:45	2022139	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery061	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery061\Hardware\m_ChannelRecovery061.exe	2.3	2010/04/06 11:52:17	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:44:00	20726	TELM-06	1	Pass	0	0	0	12
M_TCP_Channel	m_ChannelRecovery062	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery062\Hardware\m_ChannelRecovery062.exe	2.2	2010/06/25 15:47:28	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:41:41	13160	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery063	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery063\Hardware\m_ChannelRecovery063.exe	2.9	2010/05/13 17:43:56	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 13:39:45	24326	TELM-06	1	Pass	0	0	0	39
M_TCP_Channel	m_ChannelRecovery064	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery064\Hardware\m_ChannelRecovery064.exe	2.4	2010/05/06 14:14:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:36:26	40815	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery065\Hardware\m_ChannelRecovery065.exe	2.2	2010/05/06 15:43:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:32:36	37425	TELM-06	1	Pass	0	0	0	7
M_TCP_Channel	m_ChannelRecovery066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery066\Hardware\m_ChannelRecovery066.exe	2.2	2010/05/06 15:49:13	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:27:51	37484	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery067\Hardware\m_ChannelRecovery067.exe	2.2	2010/05/06 15:58:13	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:24:00	40832	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery068\Hardware\m_ChannelRecovery068.exe	2.3	2010/06/03 18:10:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:21:57	34982	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery069	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery069\Hardware\m_ChannelRecovery069.exe	2.3	2010/05/10 15:40:47	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 19:23:37	1009433	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery070	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery070\Hardware\m_ChannelRecovery070.exe	2.1	2010/02/04 18:14:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:18:04	12714	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery071	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery071\Hardware\m_ChannelRecovery071.exe	2.1	2010/02/04 17:57:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:14:48	15738	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery072	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery072\Hardware\m_ChannelRecovery072.exe	2.3	2010/05/07 11:42:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:12:07	44209	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery073	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery073\Hardware\m_ChannelRecovery073.exe	2.3	2010/05/07 11:43:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:08:35	44322	TELM-06	1	Pass	0	0	0	8
M_TCP_Close	m_Close001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close001\Hardware\m_Close001.exe	2.6	2010/03/29 17:53:05	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:53:46	7391	TELM-06	1	Pass	0	0	0	51
M_TCP_Close	m_Close002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close002\Hardware\m_Close002.exe	2.3	2010/01/26 20:03:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:52:06	8970	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close003\Hardware\m_Close003.exe	2.3	2010/06/24 15:01:11	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:49:04	6979	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close004\Hardware\m_Close004.exe	2.5	2010/03/16 12:03:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:47:29	15027	TELM-06	1	Pass	0	0	0	152
M_TCP_Close	m_Close005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close005\Hardware\m_Close005.exe	2.5	2010/05/24 17:00:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:44:20	19416	TELM-06	1	Pass	0	0	0	219
M_TCP_Close	m_Close006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close006\Hardware\m_Close006.exe	2.2	2009/11/19 20:38:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:43:08	11104	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close007\Hardware\m_Close007.exe	2.1	2010/02/04 18:46:09	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:41:31	16390	TELM-06	1	Pass	0	0	0	105
M_TCP_Close	m_Close008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close008\Hardware\m_Close008.exe	2.4	2010/05/12 15:22:15	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:39:23	6891	TELM-06	1	Pass	0	0	0	10
M_TCP_Close	m_Close009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close009\Hardware\m_Close009.exe	2.2	2010/02/04 15:29:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:37:27	7722	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close010\Hardware\m_Close010.exe	2.0	2009/09/21 12:45:39	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:35:15	15727	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close011\Hardware\m_Close011.exe	2.0	2009/09/21 12:46:09	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:33:42	8084	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close012\Hardware\m_Close012.exe	2.0	2009/09/21 12:46:36	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:32:20	11518	TELM-06	1	Pass	0	0	0	3
M_TCP_Close	m_Close013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close013\Hardware\m_Close013.exe	2.0	2009/09/21 12:18:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:30:28	8005	TELM-06	1	Pass	0	0	0	4
M_TCP_Close	m_Close014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close014\Hardware\m_Close014.exe	2.1	2010/01/29 16:45:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:28:29	8889	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close016\Hardware\m_Close016.exe	2.2	2010/08/24 12:05:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:26:49	8153	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close017\Hardware\m_Close017.exe	2.0	2009/09/21 12:47:07	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:25:30	9589	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close019\Hardware\m_Close019.exe	2.1	2009/10/01 14:47:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:23:12	5038	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close020\Hardware\m_Close020.exe	2.2	2009/10/30 14:47:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:21:32	9578	TELM-06	1	Pass	0	0	0	2

M_TCP_Close	m_Close021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close021\Hardware\m_Close021.exe	2.0	2009/09/21 12:48:28	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:20:21	2830	TELM-06	1	Pass	0	0	0	1
M_TCP_Close	m_Close022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close022\Hardware\m_Close022.exe	2.2	2009/10/30 14:53:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:19:11	12344	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_ARQ001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ001\Hardware\m_ARQ001.exe	2.5	2010/06/29 15:20:36	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:06:01	16964	TELM-06	1	Pass	0	0	0	78
M_TCP_Data	m_ARQ002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ002\Hardware\m_ARQ002.exe	2.0	2009/09/21 18:25:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 04:04:44	7927	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ003\Hardware\m_ARQ003.exe	2.3	2010/04/05 12:22:02	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:02:48	8529	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_ARQ004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ004\Hardware\m_ARQ004.exe	2.1	2009/11/04 21:22:33	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:01:30	8182	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_ARQ005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ005\Hardware\m_ARQ005.exe	2.2	2009/12/23 13:26:32	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:00:14	7520	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ006\Hardware\m_ARQ006.exe	2.1	2009/09/24 16:41:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:58:28	8634	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_ARQ007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ007\Hardware\m_ARQ007.exe	2.0	2009/09/21 18:27:29	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:56:04	10629	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_ARQ008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ008\Hardware\m_ARQ008.exe	2.0	2009/09/21 18:28:06	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:53:18	10782	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ009\Hardware\m_ARQ009.exe	2.2	2010/02/04 13:08:24	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:51:37	8699	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ010\Hardware\m_ARQ010.exe	2.2	2010/02/04 13:19:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:49:31	8681	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ011\Hardware\m_ARQ011.exe	2.3	2010/02/01 13:30:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:48:11	8115	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ012\Hardware\m_ARQ012.exe	2.3	2010/02/18 15:17:44	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:46:53	8480	TELM-06	1	Pass	0	0	0	211
M_TCP_Data	m_ARQ013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ013\Hardware\m_ARQ013.exe	2.3	2010/02/03 16:38:29	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:45:33	10326	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_ARQ014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ014\Hardware\m_ARQ014.exe	2.4	2010/04/05 11:08:56	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:44:03	8193	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ015\Hardware\m_ARQ015.exe	2.6	2010/06/08 15:25:41	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 03:42:42	12798	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ016\Hardware\m_ARQ016.exe	2.3	2009/11/19 20:22:28	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:38:23	9242	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ017\Hardware\m_ARQ017.exe	2.2	2009/09/30 15:01:03	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:34:30	6613	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ018\Hardware\m_ARQ018.exe	2.7	2010/06/24 12:34:26	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 03:32:44	10827	TELM-06	1	Pass	0	0	0	992
M_TCP_Data	m_ARQ019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ019\Hardware\m_ARQ019.exe	2.4	2010/06/08 16:07:58	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:30:12	9988	TELM-06	1	Pass	0	0	0	132
M_TCP_Data	m_ARQ020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ020\Hardware\m_ARQ020.exe	2.2	2009/11/18 15:23:22	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:27:47	7497	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ021\Hardware\m_ARQ021.exe	2.2	2009/11/19 18:43:52	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:25:53	7492	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ022\Hardware\m_ARQ022.exe	2.3	2010/02/03 16:30:46	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:24:44	8429	TELM-06	1	Pass	0	0	0	137
M_TCP_Data	m_ARQ023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ023\Hardware\m_ARQ023.exe	2.1	2009/11/10 19:12:36	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:19:51	9746	TELM-06	1	Pass	0	0	0	63
M_TCP_Data	m_ARQ024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ024\Hardware\m_ARQ024.exe	2.3	2010/02/08 18:44:31	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:17:18	9980	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ025\Hardware\m_ARQ025.exe	2.1	2009/09/25 14:20:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:14:14	8816	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ026\Hardware\m_ARQ026.exe	2.1	2009/09/25 14:21:05	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:11:50	8576	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_ARQ027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ027\Hardware\m_ARQ027.exe	2.1	2009/09/25 14:21:27	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:05:34	8932	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ028\Hardware\m_ARQ028.exe	2.4	2010/04/19 14:36:56	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:00:37	18576	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_ARQ029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ029\Hardware\m_ARQ029.exe	2.1	2010/03/08 19:53:24	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:56:06	29576	TELM-06	1	Pass	0	0	0	555
M_TCP_Data	m_ARQ031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ031\Hardware\m_ARQ031.exe	2.3	2010/06/08 19:57:52	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:53:43	11065	TELM-06	1	Pass	0	0	0	646
M_TCP_Data	m_Data012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data012\Hardware\m_Data012.exe	2.4	2009/11/10 14:18:10	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 00:45:42	42889	TELM-06	1	Pass	0	0	0	18
M_TCP_Data	m_Data013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data013\Hardware\m_Data013.exe	2.3	2010/02/03 15:41:32	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:50:29	102872	TELM-06	1	Pass	0	0	0	335
M_TCP_Data	m_Data014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data014\Hardware\m_Data014.exe	2.3	2010/05/12 19:37:35	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:45:24	92220	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_Data015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data015\Hardware\m_Data015.exe	2.2	2010/02/03 17:34:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:43:03	8203	TELM-06	1	Pass	0	0	0	14
M_TCP_Data	m_Data016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data016\Hardware\m_Data016.exe	2.2	2010/02/24 20:56:20	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:39:17	70034	TELM-06	1	Pass	0	0	0	26
M_TCP_Data	m_Data017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data017\Hardware\m_Data017.exe	2.2	2010/02/24 22:22:41	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:36:01	67644	TELM-06	1	Pass	0	0	0	30
M_TCP_Data	m_Data018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data018\Hardware\m_Data018.exe	2.2	2010/02/03 22:48:42	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:33:43	59138	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_MemoryRead001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead001\Hardware\m_MemoryRead001.exe	2.2	2010/02/01 17:26:47	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:31:48	17780	TELM-06	1	Pass	0	0	0	439
M_TCP_Data	m_MemoryRead002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead002\Hardware\m_MemoryRead002.exe	2.3	2009/11/05 15:03:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:29:30	18654	TELM-06	1	Pass	0	0	0	31
M_TCP_Data	m_MemoryRead003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead003\Hardware\m_MemoryRead003.exe	2.4	2010/04/05 13:08:02	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:26:22	20056	TELM-06	1	Pass	0	0	0	320
M_TCP_Data	m_MemoryRead004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead004\Hardware\m_MemoryRead004.exe	2.3	2010/03/19 16:10:47	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:22:45	8382	TELM-06	1	Pass	0	0	0	1
M_TCP_Data	m_MemoryRead005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead005\Hardware\m_MemoryRead005.exe	2.3	2009/12/07 19:50:30	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:19:30	48511	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryRead006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead006\Hardware\m_MemoryRead006.exe	2.6	2010/06/29 21:05:35	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 02:17:11	12123	TELM-06	1	Pass	0	0	0	37
M_TCP_Data	m_MemoryRead007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead007\Hardware\m_MemoryRead007.exe	2.1	2009/11/03 12:42:15	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:15:29	10764	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_MemoryRead008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead008\Hardware\m_MemoryRead008.exe	2.1	2009/09/30 18:09:14	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:13:27	7759	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead009\Hardware\m_MemoryRead009.exe	2.1	2009/12/07 19:51:11	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:10:16	11057	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead010\Hardware\m_MemoryRead010.exe	2.2	2009/09/30 17:31:58	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:07:38	7520	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead011\Hardware\m_MemoryRead011.exe	2.2	2009/09/30 17:32:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:05:08	7457	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead012\Hardware\m_MemoryRead012.exe	2.2	2009/09/30 17:33:03	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:59:28	7454	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead013\Hardware\m_MemoryRead013.exe	2.2	2009/09/30 17:33:27	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:58:17	7639	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead014\Hardware\m_MemoryRead014.exe	2.2	2009/09/30 17:33:51	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:54:28	7472	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead015\Hardware\m_MemoryRead015.exe	2.2	2009/09/30 17:34:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:49:37	7475	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead016\Hardware\m_MemoryRead016.exe	2.1	2009/10/19 18:24:00	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:48:01	7723	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead017\Hardware\m_MemoryRead017.exe	2.2	2009/10/19 15:52:59	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:45:16	7619	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead019\Hardware\m_MemoryRead019.exe	2.3	2009/12/03 13:57:22	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:41:18	9153	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead020\Hardware\m_MemoryRead020.exe	2.4	2010/02/09 20:49:06	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:33:54	6828	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryRead021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead021\Hardware\m_MemoryRead021.exe	2.2	2009/12/23 13:34:31	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:31:54	8377	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryRead030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead030\Hardware\m_MemoryRead030.exe	2.1	2009/10/16 13:54:03	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:29:18	6761	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead031\Hardware\m_MemoryRead031.exe	2.3	2009/12/28 18:23:34	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:28:10	10062	TELM-06	1	Pass	0	0	0	43
M_TCP_Data	m_MemoryRead033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead033\Hardware\m_MemoryRead033.exe	2.3	2009/11/23 18:55:48	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:24:29	12075	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead034\Hardware\m_MemoryRead034.exe	2.1	2009/10/16 14:15:21	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:21:07	8073	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead035\Hardware\m_MemoryRead035.exe	2.4	2009/11/23 18:56:23	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:15:49	16660	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead037\Hardware\m_MemoryRead037.exe	2.2	2009/11/23 18:57:01	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:13:55	12349	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead038\Hardware\m_MemoryRead038.exe	2.3	2010/01/19 17:36:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:11:25	14742	TELM-06	1	Pass	0	0	0	1590
M_TCP_Data	m_MemoryRead039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead039\Hardware\m_MemoryRead039.exe	2.1	2009/09/28 15:09:29	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:04:58	8669	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead040\Hardware\m_MemoryRead040.exe	2.6	2010/01/06 14:10:47	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:02:32	32261	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryRead041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead041\Hardware\m_MemoryRead041.exe	2.4	2009/11/16 12:52:24	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:56:53	18140	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead042\Hardware\m_MemoryRead042.exe	2.3	2010/06/29 22:24:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:54:49	22416	TELM-06	1	Pass	0	0	0	41
M_TCP_Data	m_MemoryRead043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead043\Hardware\m_MemoryRead043.exe	2.2	2010/02/08 16:55:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:52:30	17564	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead045\Hardware\m_MemoryRead045.exe	2.0	2009/09/22 14:34:52	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 00:49:10	6936	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead049\Hardware\m_MemoryRead049.exe	2.3	2009/11/16 12:52:57	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:46:45	8619	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead051\Hardware\m_MemoryRead051.exe	2.1	2009/10/27 19:22:11	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:41:41	9650	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryRead052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead052\Hardware\m_MemoryRead052.exe	2.3	2010/06/29 23:18:49	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:37:29	10980	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite001\Hardware\m_MemoryWrite001.exe	2.2	2009/12/07 19:52:07	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:34:19	7839	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite002\Hardware\m_MemoryWrite002.exe	2.0	2009/09/22 14:48:30	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 00:30:14	10405	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite003\Hardware\m_MemoryWrite003.exe	2.1	2009/10/28 14:34:41	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:27:38	9909	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite005\Hardware\m_MemoryWrite005.exe	2.2	2009/12/15 19:05:34	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:20:33	8171	TELM-06	1	Pass	0	0	0	186
M_TCP_Data	m_MemoryWrite006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite006\Hardware\m_MemoryWrite006.exe	2.3	2009/11/24 17:33:37	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:17:17	10946	TELM-06	1	Pass	0	0	0	145
M_TCP_Data	m_MemoryWrite007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite007\Hardware\m_MemoryWrite007.exe	2.1	2009/10/28 15:09:57	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:13:38	6656	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite008\Hardware\m_MemoryWrite008.exe	2.1	2010/02/01 19:11:34	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:10:18	7452	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite009\Hardware\m_MemoryWrite009.exe	2.1	2009/10/28 15:17:22	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:08:18	7477	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite010\Hardware\m_MemoryWrite010.exe	2.1	2009/10/28 16:00:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:05:13	7543	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite011\Hardware\m_MemoryWrite011.exe	2.1	2009/10/28 16:07:34	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:02:11	7520	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite012\Hardware\m_MemoryWrite012.exe	2.2	2009/10/28 16:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:58:47	12152	TELM-06	1	Pass	0	0	0	120
M_TCP_Data	m_MemoryWrite013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite013\Hardware\m_MemoryWrite013.exe	2.1	2009/10/28 16:21:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:56:20	7674	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite014\Hardware\m_MemoryWrite014.exe	2.1	2009/10/28 16:29:59	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:52:01	7716	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite015\Hardware\m_MemoryWrite015.exe	2.1	2009/10/28 16:36:30	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:49:43	7487	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryWrite016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite016\Hardware\m_MemoryWrite016.exe	2.5	2010/02/01 21:11:41	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:49:02	9408	TELM-06	1	Pass	0	0	0	3470
M_TCP_Data	m_MemoryWrite017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite017\Hardware\m_MemoryWrite017.exe	2.3	2010/01/19 17:37:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:46:39	8676	TELM-06	1	Pass	0	0	0	2465
M_TCP_Data	m_MemoryWrite018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite018\Hardware\m_MemoryWrite018.exe	2.1	2009/10/29 20:06:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:41:44	6703	TELM-06	1	Pass	0	0	0	171
M_TCP_Data	m_MemoryWrite019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite019\Hardware\m_MemoryWrite019.exe	2.0	2009/09/22 15:13:38	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:39:55	9489	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryWrite020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite020\Hardware\m_MemoryWrite020.exe	2.2	2009/10/29 19:44:00	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:38:36	7807	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite022\Hardware\m_MemoryWrite022.exe	2.3	2009/11/03 17:28:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:35:38	9816	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite023\Hardware\m_MemoryWrite023.exe	2.4	2010/02/04 16:15:17	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:29:31	20488	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite024\Hardware\m_MemoryWrite024.exe	2.2	2009/11/24 17:35:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:24:56	7475	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_MemoryWrite025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite025\Hardware\m_MemoryWrite025.exe	2.1	2009/10/29 18:47:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:20:55	10956	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite029\Hardware\m_MemoryWrite029.exe	2.2	2010/02/01 21:00:35	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:18:09	13985	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite032\Hardware\m_MemoryWrite032.exe	2.5	2010/03/15 01:08:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:16:09	9679	TELM-06	1	Pass	0	0	0	804
M_TCP_Data	m_MemoryWrite038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite038\Hardware\m_MemoryWrite038.exe	2.1	2009/10/29 18:42:13	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:14:46	6956	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite039\Hardware\m_MemoryWrite039.exe	2.1	2010/02/10 18:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:12:30	9587	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite040\Hardware\m_MemoryWrite040.exe	2.2	2010/02/10 17:56:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:10:25	9605	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryWrite041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite041\Hardware\m_MemoryWrite041.exe	2.2	2010/02/01 20:45:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:08:38	13698	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite042\Hardware\m_MemoryWrite042.exe	2.2	2010/01/19 17:38:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:06:22	10251	TELM-06	1	Pass	0	0	0	1589
M_TCP_Data	m_MemoryWrite046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite046\Hardware\m_MemoryWrite046.exe	2.3	2010/02/04 16:51:09	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:03:25	7424	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite047\Hardware\m_MemoryWrite047.exe	2.2	2009/11/11 19:12:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:00:11	8037	TELM-06	1	Pass	0	0	0	96
M_TCP_Data	m_MemoryWrite048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite048\Hardware\m_MemoryWrite048.exe	2.2	2009/11/11 18:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:57:08	8886	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite049\Hardware\m_MemoryWrite049.exe	2.5	2010/02/10 16:51:26	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:54:34	21279	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_MemoryWrite052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite052\Hardware\m_MemoryWrite052.exe	2.3	2009/12/23 13:35:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:52:48	11121	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite053\Hardware\m_MemoryWrite053.exe	2.2	2009/11/11 17:01:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:51:58	8413	TELM-06	1	Pass	0	0	0	9

M_TCP_Data	m_MemoryWrite054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite054\Hardware\m_MemoryWrite054.exe	2.6	2010/06/30 17:58:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:50:59	10030	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryWrite055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite055\Hardware\m_MemoryWrite055.exe	2.2	2009/11/11 18:04:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:49:09	7312	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite056	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite056\Hardware\m_MemoryWrite056.exe	2.4	2010/02/10 16:32:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:47:45	8649	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode001\Hardware\m_TransferMode001.exe	2.8	2010/05/14 18:48:57	2.8	Up-to-date	Baseline 17_1	None	09/13/2010 09:08:29	16448	TELM-06	1	Pass	0	0	0	4748
M_TCP_Data	m_TransferMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode002\Hardware\m_TransferMode002.exe	2.4	2010/07/02 13:00:52	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:43:59	10063	TELM-06	1	Pass	0	0	0	166
M_TCP_Data	m_TransferMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode003\Hardware\m_TransferMode003.exe	2.0	2009/11/16 20:59:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:42:57	7002	TELM-06	1	Pass	0	0	0	129
M_TCP_Data	m_TransferMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode004\Hardware\m_TransferMode004.exe	2.3	2010/01/08 17:07:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:41:03	18424	TELM-06	1	Pass	0	0	0	532
M_TCP_Data	m_TransferMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode005\Hardware\m_TransferMode005.exe	2.7	2010/07/02 13:36:13	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 22:39:31	36238	TELM-06	1	Pass	0	0	0	659
M_TCP_Data	m_TransferMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode006\Hardware\m_TransferMode006.exe	2.4	2010/04/16 11:58:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:38:37	6944	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode007\Hardware\m_TransferMode007.exe	2.2	2009/10/12 15:46:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:33:22	11673	TELM-06	1	Pass	0	0	0	112
M_TCP_Data	m_TransferMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode008\Hardware\m_TransferMode008.exe	2.0	2009/09/09 15:28:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:31:54	10625	TELM-06	1	Pass	0	0	0	76
M_TCP_Data	m_TransferMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode009\Hardware\m_TransferMode009.exe	2.1	2009/10/13 16:25:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:28:13	11301	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_TransferMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode010\Hardware\m_TransferMode010.exe	2.1	2009/10/13 16:26:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:25:25	11313	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode011\Hardware\m_TransferMode011.exe	2.3	2010/02/10 13:37:08	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:24:09	8112	TELM-06	1	Pass	0	0	0	72
M_TCP_Data	m_TransferMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode012\Hardware\m_TransferMode012.exe	2.3	2010/02/10 13:38:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:23:16	8226	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_TransferMode013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode013\Hardware\m_TransferMode013.exe	2.3	2010/02/10 13:39:16	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:21:44	8172	TELM-06	1	Pass	0	0	0	44
M_TCP_Data	m_TransferMode014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode014\Hardware\m_TransferMode014.exe	2.3	2010/02/10 13:40:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:20:18	8351	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_TransferMode015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode015\Hardware\m_TransferMode015.exe	2.5	2010/05/26 14:56:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:19:15	7428	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_TransferMode016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode016\Hardware\m_TransferMode016.exe	2.8	2010/07/02 13:14:11	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 22:17:25	29918	TELM-06	1	Pass	0	0	0	47
M_TCP_Data	m_TransferMode017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode017\Hardware\m_TransferMode017.exe	2.7	2010/05/18 14:47:09	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 22:15:41	33460	TELM-06	1	Pass	0	0	0	257
M_TCP_Data	m_TransferMode018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode018\Hardware\m_TransferMode018.exe	2.6	2010/04/16 14:06:26	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:14:01	26459	TELM-06	1	Pass	0	0	0	509
M_TCP_Data	m_TransferMode019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode019\Hardware\m_TransferMode019.exe	2.6	2010/03/19 18:58:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:12:05	16728	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics001\Hardware\m_Diagnostics001.exe	2.5	2009/12/30 17:45:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:30:54	9524	TELM-06	1	Pass	0	0	0	166
M_TCP_Diagnostic	m_Diagnostics002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics002\Hardware\m_Diagnostics002.exe	2.2	2009/11/23 20:15:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:27:50	7959	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics003\Hardware\m_Diagnostics003.exe	2.2	2010/03/14 20:01:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:26:36	6736	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics004\Hardware\m_Diagnostics004.exe	2.3	2009/12/03 13:22:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:25:02	15978	TELM-06	1	Pass	0	0	0	238
M_TCP_Diagnostic	m_Diagnostics005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics005\Hardware\m_Diagnostics005.exe	2.1	2009/10/12 14:14:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:21:37	7905	TELM-06	1	Pass	0	0	0	13
M_TCP_Diagnostic	m_Diagnostics006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics006\Hardware\m_Diagnostics006.exe	2.0	2009/09/22 17:02:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:20:01	14634	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics007\Hardware\m_Diagnostics007.exe	2.1	2009/09/28 13:31:55	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:17:35	14820	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics008\Hardware\m_Diagnostics008.exe	2.0	2009/09/22 17:03:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:15:55	21156	TELM-06	1	Pass	0	0	0	31
M_TCP_Diagnostic	m_Diagnostics009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics009\Hardware\m_Diagnostics009.exe	2.5	2010/06/25 17:09:14	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:13:24	15872	TELM-06	1	Pass	0	0	0	23
M_TCP_Diagnostic	m_Diagnostics010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics010\Hardware\m_Diagnostics010.exe	2.5	2010/07/20 19:18:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:10:49	10492	TELM-06	1	Pass	0	0	0	10
M_TCP_Diagnostic	m_Diagnostics011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics011\Hardware\m_Diagnostics011.exe	2.4	2009/12/23 13:29:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:08:22	29460	TELM-06	1	Pass	0	0	0	19
M_TCP_Diagnostic	m_Diagnostics012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics012\Hardware\m_Diagnostics012.exe	2.3	2010/04/16 14:20:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:06:15	17088	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics013\Hardware\m_Diagnostics013.exe	2.5	2010/06/08 17:29:50	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:04:19	20425	TELM-06	1	Pass	0	0	0	15
M_TCP_Diagnostic	m_Diagnostics014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics014\Hardware\m_Diagnostics014.exe	2.4	2010/03/14 17:27:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:02:44	17727	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics015\Hardware\m_Diagnostics015.exe	2.4	2010/01/19 17:35:47	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:57:22	25614	TELM-06	1	Pass	0	0	0	29
M_TCP_Diagnostic	m_Diagnostics016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics016\Hardware\m_Diagnostics016.exe	2.0	2009/09/22 17:09:46	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:53:56	5963	TELM-06	1	Pass	0	0	0	1
M_TCP_Diagnostic	m_Diagnostics017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics017\Hardware\m_Diagnostics017.exe	2.3	2009/11/23 20:15:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:51:44	24734	TELM-06	1	Pass	0	0	0	32
M_TCP_Diagnostic	m_Diagnostics018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics018\Hardware\m_Diagnostics018.exe	2.2	2009/12/23 13:30:36	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:47:52	11865	TELM-06	1	Pass	0	0	0	9
M_TCP_Diagnostic	m_Diagnostics019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics019\Hardware\m_Diagnostics019.exe	2.1	2009/10/13 12:56:53	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:46:09	7186	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics020\Hardware\m_Diagnostics020.exe	2.2	2009/11/23 20:16:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:43:37	10955	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics021\Hardware\m_Diagnostics021.exe	2.1	2009/10/13 13:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:42:19	6231	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics022\Hardware\m_Diagnostics022.exe	2.4	2010/02/03 17:09:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:40:27	12601	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics023\Hardware\m_Diagnostics023.exe	2.2	2010/02/03 14:24:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:37:27	39218	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics024\Hardware\m_Diagnostics024.exe	2.5	2010/02/03 14:04:47	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:34:31	15532	TELM-06	1	Pass	0	0	0	20
M_TCP_Diagnostic	m_Diagnostics026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics026\Hardware\m_Diagnostics026.exe	2.1	2009/10/15 19:32:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:32:07	6131	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics027\Hardware\m_Diagnostics027.exe	2.2	2009/11/02 13:03:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:30:05	2893	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics028\Hardware\m_Diagnostics028.exe	2.4	2009/11/16 20:37:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:28:43	11412	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics029\Hardware\m_Diagnostics029.exe	2.1	2009/10/15 19:46:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:27:03	2861	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics030\Hardware\m_Diagnostics030.exe	2.5	2010/07/20 18:09:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:25:47	7393	TELM-06	1	Pass	0	0	0	277
M_TCP_Diagnostic	m_Diagnostics031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics031\Hardware\m_Diagnostics031.exe	2.2	2009/11/11 14:33:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:24:22	9883	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics032\Hardware\m_Diagnostics032.exe	2.3	2009/12/23 13:31:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:22:34	9708	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics034\Hardware\m_Diagnostics034.exe	2.4	2009/11/16 20:30:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:21:10	8911	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics035\Hardware\m_Diagnostics035.exe	2.2	2009/12/23 13:31:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:19:33	7822	TELM-06	1	Pass	0	0	0	7
M_TCP_Diagnostic	m_Diagnostics036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics036\Hardware\m_Diagnostics036.exe	2.3	2010/03/13 01:44:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:18:15	8485	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics037\Hardware\m_Diagnostics037.exe	2.4	2010/03/13 01:23:35	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:17:06	6852	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics038\Hardware\m_Diagnostics038.exe	2.3	2010/06/07 20:16:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:13:27	48480	TELM-06	1	Pass	0	0	0	18
M_TCP_Discover	m_DiscoverFull001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull001\Hardware\m_DiscoverFull001.exe	2.8	2010/05/21 19:22:39	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 18:51:07	44643	TELM-06	1	Pass	0	0	0	150
M_TCP_Discover	m_DiscoverFull002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull002\Hardware\m_DiscoverFull002.exe	2.6	2010/03/19 18:02:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 18:46:54	144963	TELM-06	1	Pass	0	0	0	1815
M_TCP_Discover	m_DiscoverFull003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull003\Hardware\m_DiscoverFull003.exe	2.2	2010/02/25 20:00:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:43:32	12152	TELM-06	1	Pass	0	0	0	58
M_TCP_Discover	m_DiscoverFull004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull004\Hardware\m_DiscoverFull004.exe	2.4	2010/02/05 21:19:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:35:15	270966	TELM-06	1	Pass	0	0	0	3713
M_TCP_Discover	m_DiscoverFull005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull005\Hardware\m_DiscoverFull005.exe	2.2	2009/10/29 19:18:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:38:03	17757	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull006\Hardware\m_DiscoverFull006.exe	2.3	2010/02/05 21:19:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:36:33	130849	TELM-06	1	Pass	0	0	0	1727
M_TCP_Discover	m_DiscoverFull007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull007\Hardware\m_DiscoverFull007.exe	2.7	2009/12/14 21:43:07	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 18:35:49	14035	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull009\Hardware\m_DiscoverFull009.exe	2.3	2010/01/21 21:02:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:34:38	11006	TELM-06	1	Pass	0	0	0	65
M_TCP_Discover	m_DiscoverFull010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull010\Hardware\m_DiscoverFull010.exe	2.4	2010/05/12 17:07:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:32:57	7995	TELM-06	1	Pass	0	0	0	121
M_TCP_Discover	m_DiscoverFull011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull011\Hardware\m_DiscoverFull011.exe	2.3	2009/11/02 22:03:01	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:26:47	56732	TELM-06	1	Pass	0	0	0	9
M_TCP_Discover	m_DiscoverFull012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull012\Hardware\m_DiscoverFull012.exe	2.8	2010/06/29 12:07:18	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 11:40:55	366851	TELM-06	1	Pass	0	0	0	33
M_TCP_Discover	m_DiscoverFull013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull013\Hardware\m_DiscoverFull013.exe	2.2	2010/03/18 16:03:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:24:55	42438	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull014\Hardware\m_DiscoverFull014.exe	2.2	2010/03/18 16:03:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:20:47	24895	TELM-06	1	Pass	0	0	0	15
M_TCP_Discover	m_DiscoverFull015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull015\Hardware\m_DiscoverFull015.exe	2.1	2009/11/23 16:09:49	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:15:58	7883	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull016\Hardware\m_DiscoverFull016.exe	2.1	2009/11/03 20:58:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:13:00	8252	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull017\Hardware\m_DiscoverFull017.exe	2.1	2009/11/03 20:58:47	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:11:24	12217	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull022\Hardware\m_DiscoverFull022.exe	2.1	2009/11/23 16:11:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:08:14	8461	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull023\Hardware\m_DiscoverFull023.exe	2.1	2009/11/23 16:11:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:05:22	11500	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull024\Hardware\m_DiscoverFull024.exe	2.1	2009/11/23 16:12:04	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:03:03	12598	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull025\Hardware\m_DiscoverFull025.exe	2.1	2009/11/23 16:12:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:01:08	9039	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull029\Hardware\m_DiscoverFull029.exe	2.0	2009/11/13 21:27:59	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 17:58:44	12487	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull030\Hardware\m_DiscoverFull030.exe	2.3	2010/05/28 11:39:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:53:03	19182	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull031\Hardware\m_DiscoverFull031.exe	2.3	2010/05/28 12:38:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:45:41	35373	TELM-06	1	Pass	0	0	0	22
M_TCP_Discover	m_DiscoverFull032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull032\Hardware\m_DiscoverFull032.exe	2.0	2009/11/13 21:29:43	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 17:44:07	9136	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull033\Hardware\m_DiscoverFull033.exe	2.0	2009/11/13 21:30:18	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 17:40:57	9513	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull034\Hardware\m_DiscoverFull034.exe	2.2	2010/03/18 16:04:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:39:50	11144	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull035\Hardware\m_DiscoverFull035.exe	2.2	2010/02/09 20:53:53	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 20:58:49	43619	TELM-06	1	Pass	0	0	0	31
M_TCP_Discover	m_DiscoverFull036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull036\Hardware\m_DiscoverFull036.exe	2.2	2010/02/11 14:31:48	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 19:51:31	22045	TELM-06	1	Pass	0	0	0	38
M_TCP_Discover	m_DiscoverFull037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull037\Hardware\m_DiscoverFull037.exe	2.4	2010/02/03 21:18:36	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 23:35:18	35104	TELM-06	1	Pass	0	0	0	4
M_TCP_Discover	m_DiscoverFull038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull038\Hardware\m_DiscoverFull038.exe	2.1	2009/12/01 20:49:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:35:45	141700	TELM-06	1	Pass	0	0	0	7
M_TCP_EEPROM	m_Eeprom001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom001\Hardware\m_Eeprom001.exe	2.3	2009/12/23 16:23:59	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 15:27:05	11097	TELM-06	1	Pass	0	0	0	22
M_TCP_EEPROM	m_Eeprom002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom002\Hardware\m_Eeprom002.exe	2.3	2009/12/18 21:07:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 10:00:13	10789	TELM-06	1	Pass	0	0	0	938
M_TCP_EEPROM	m_Eeprom003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom003\Hardware\m_Eeprom003.exe	2.2	2010/01/05 18:03:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:11:24	16672	TELM-06	1	Pass	0	0	0	702
M_TCP_EEPROM	m_Eeprom004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom004\Hardware\m_Eeprom004.exe	2.2	2010/01/25 19:26:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:07:35	108583	TELM-06	1	Pass	0	0	0	583
M_TCP_EEPROM	m_Eeprom005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom005\Hardware\m_Eeprom005.exe	2.2	2010/01/19 15:32:05	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:02:59	171454	TELM-06	1	Pass	0	0	0	704
M_TCP_EEPROM	m_Eeprom006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom006\Hardware\m_Eeprom006.exe	2.2	2010/05/28 12:55:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:58:20	46893	TELM-06	1	Pass	0	0	0	6
M_TCP_EEPROM	m_Eeprom007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom007\Hardware\m_Eeprom007.exe	2.1	2010/02/18 18:38:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:57:25	5046	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency001\Hardware\m_Emergency001.exe	2.3	2009/11/03 21:54:33	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:09:11	9759	TELM-06	1	Pass	0	0	0	100
M_TCP_Emergency	m_Emergency002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency002\Hardware\m_Emergency002.exe	2.5	2010/06/25 20:19:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:07:53	12815	TELM-06	1	Pass	0	0	0	45
M_TCP_Emergency	m_Emergency003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency003\Hardware\m_Emergency003.exe	2.6	2010/06/29 14:19:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:05:30	34367	TELM-06	1	Pass	0	0	0	4180
M_TCP_Emergency	m_Emergency004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency004\Hardware\m_Emergency004.exe	2.3	2010/02/05 21:22:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:04:12	11394	TELM-06	1	Pass	0	0	0	4
M_TCP_Emergency	m_Emergency005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency005\Hardware\m_Emergency005.exe	2.2	2009/11/10 00:31:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:02:32	7922	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency006\Hardware\m_Emergency006.exe	2.1	2009/10/16 16:44:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:00:12	9389	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency007\Hardware\m_Emergency007.exe	2.1	2009/10/16 16:44:28	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:58:46	9325	TELM-06	1	Pass	0	0	0	33
M_TCP_Emergency	m_Emergency009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency009\Hardware\m_Emergency009.exe	2.1	2009/10/08 19:54:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:57:42	6141	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency010\Hardware\m_Emergency010.exe	2.2	2010/06/08 15:56:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:54:02	13102	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency011\Hardware\m_Emergency011.exe	2.0	2009/10/21 16:25:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 21:51:13	6739	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency012\Hardware\m_Emergency012.exe	2.1	2009/11/11 13:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:49:35	7990	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency013\Hardware\m_Emergency013.exe	2.1	2009/11/11 14:02:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:45:25	11411	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency014\Hardware\m_Emergency014.exe	2.3	2009/12/23 13:33:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:42:02	8170	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency015\Hardware\m_Emergency015.exe	2.3	2010/06/08 15:56:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:38:41	8848	TELM-06	1	Pass	0	0	0	8
M_TCP_Emergency	m_Emergency016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency016\Hardware\m_Emergency016.exe	2.2	2009/12/21 21:03:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:35:31	11944	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency017\Hardware\m_Emergency017.exe	2.2	2009/11/16 12:31:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:29:40	9596	TELM-06	1	Pass	0	0	0	21

M_TCP_Emergency	m_Emergency018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency018\Hardware\m_Emergency018.exe	2.3	2010/06/29 15:32:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:25:43	7860	TELM-06	1	Pass	0	0	0	22
M_TCP_Emergency	m_Emergency019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency019\Hardware\m_Emergency019.exe	2.2	2010/01/07 19:28:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:22:22	10883	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency020\Hardware\m_Emergency020.exe	2.6	2010/03/18 17:47:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 21:19:35	10833	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency021\Hardware\m_Emergency021.exe	2.3	2009/12/21 19:27:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:16:21	10357	TELM-06	1	Pass	0	0	0	83
M_TCP_Emergency	m_Emergency022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency022\Hardware\m_Emergency022.exe	2.2	2010/01/07 18:54:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:09:55	10075	TELM-06	1	Pass	0	0	0	82
M_TCP_Emergency	m_Emergency038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency038\Hardware\m_Emergency038.exe	2.2	2010/01/07 19:42:11	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:05:09	10565	TELM-06	1	Pass	0	0	0	14
M_TCP_Emergency	m_Emergency039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency039\Hardware\m_Emergency039.exe	2.2	2010/02/03 17:51:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:03:07	15296	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency040\Hardware\m_Emergency040.exe	2.0	2009/09/29 14:53:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 21:01:01	14005	TELM-06	1	Pass	0	0	0	7
M_TCP_Emergency	m_Emergency041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency041\Hardware\m_Emergency041.exe	2.0	2009/09/29 14:54:05	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 20:58:51	4891	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency042\Hardware\m_Emergency042.exe	2.0	2009/09/29 14:54:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 20:56:02	4927	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency043\Hardware\m_Emergency043.exe	2.4	2010/04/21 17:13:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:51:16	10326	TELM-06	1	Pass	0	0	0	9
M_TCP_Emergency	m_Emergency044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency044\Hardware\m_Emergency044.exe	2.5	2009/11/10 14:19:43	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 00:32:58	42816	TELM-06	1	Pass	0	0	0	18
M_TCP_Emergency	m_Emergency045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency045\Hardware\m_Emergency045.exe	2.3	2010/05/11 12:18:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:48:34	10807	TELM-06	1	Pass	0	0	0	16
M_TCP_General	m_General003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General003\Hardware\m_General003.exe	2.2	2009/10/13 14:50:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:45:22	67655	TELM-06	1	Pass	0	0	0	44744
M_TCP_General	m_General005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General005\Hardware\m_General005.exe	2.4	2009/12/01 19:02:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:41:03	68285	TELM-06	1	Pass	0	0	0	19027
M_TCP_General	m_General015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General015\Hardware\m_General015.exe	2.3	2009/12/01 19:02:59	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:37:33	37323	TELM-06	1	Pass	0	0	0	208790
M_TCP_General	m_General016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General016\Hardware\m_General016.exe	2.4	2010/03/29 10:43:22	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:35:13	8566	TELM-06	1	Pass	0	0	0	2
M_TCP_General	m_General017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General017\Hardware\m_General017.exe	2.3	2009/12/01 19:03:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:33:09	16507	TELM-06	1	Pass	0	0	0	10
M_TCP_General	m_General018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General018\Hardware\m_General018.exe	2.2	2010/02/03 18:07:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:27:45	29733	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General027\Hardware\m_General027.exe	2.7	2010/06/07 11:39:47	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 16:38:00	802172	TELM-06	1	Pass	0	0	0	21
M_TCP_General	m_General028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General028\Hardware\m_General028.exe	2.5	2010/06/08 11:48:52	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 00:20:16	635763	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General029\Hardware\m_General029.exe	2.3	2010/06/07 12:27:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 23:57:24	274293	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General030\Hardware\m_General030.exe	2.1	2010/06/07 12:30:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:25:54	17223	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General031\Hardware\m_General031.exe	2.5	2010/05/11 12:33:53	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:22:27	57884	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General032\Hardware\m_General032.exe	2.2	2010/02/05 18:56:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:20:46	29743	TELM-06	1	Pass	0	0	0	11
M_TCP_General	m_General033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General033\Hardware\m_General033.exe	2.0	2010/02/04 17:42:57	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 20:18:34	20836	TELM-06	1	Pass	0	0	0	4
M_TCP_General	m_General034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General034\Hardware\m_General034.exe	2.0	2010/01/22 22:45:34	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 20:15:26	30285	TELM-06	1	Pass	0	0	0	2
M_TCP_Initialize	m_Initialize001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize001\Hardware\m_Initialize001.exe	2.8	2010/05/12 12:34:35	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 17:33:47	10525	TELM-06	1	Pass	0	0	0	9
M_TCP_Initialize	m_Initialize002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize002\Hardware\m_Initialize002.exe	2.2	2009/10/20 12:16:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:31:42	19865	TELM-06	1	Pass	0	0	0	11
M_TCP_Initialize	m_Initialize003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize003\Hardware\m_Initialize003.exe	2.4	2010/06/08 15:57:16	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:29:52	23996	TELM-06	1	Pass	0	0	0	6
M_TCP_Initialize	m_Initialize004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize004\Hardware\m_Initialize004.exe	2.4	2010/06/08 15:57:44	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:28:09	40022	TELM-06	1	Pass	0	0	0	30
M_TCP_MedicalEvent	m_MedicalEvent001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent001\Hardware\m_MedicalEvent001.exe	2.4	2010/01/18 15:42:53	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 06:02:54	12586	TELM-06	1	Pass	0	0	0	36
M_TCP_MedicalEvent	m_MedicalEvent002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent002\Hardware\m_MedicalEvent002.exe	2.6	2010/02/09 19:03:39	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:00:31	72357	TELM-06	1	Pass	0	0	0	23
M_TCP_MedicalEvent	m_MedicalEvent003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent003\Hardware\m_MedicalEvent003.exe	2.4	2010/03/11 21:16:02	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:58:23	18706	TELM-06	1	Pass	0	0	0	24
M_TCP_MedicalEvent	m_MedicalEvent004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent004\Hardware\m_MedicalEvent004.exe	2.4	2010/01/18 15:44:34	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:55:01	12043	TELM-06	1	Pass	0	0	0	12
M_TCP_MedicalEvent	m_MedicalEvent008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent008\Hardware\m_MedicalEvent008.exe	2.4	2010/01/18 15:44:59	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:50:46	8306	TELM-06	1	Pass	0	0	0	6
M_TCP_MedicalEvent	m_MedicalEvent009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent009\Hardware\m_MedicalEvent009.exe	2.2	2010/01/18 15:46:20	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:48:44	7517	TELM-06	1	Pass	0	0	0	4
M_TCP_MedicalEvent	m_MedicalEvent010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent010\Hardware\m_MedicalEvent010.exe	2.5	2010/01/27 14:30:44	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:47:18	16638	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent011\Hardware\m_MedicalEvent011.exe	2.2	2010/01/18 15:47:08	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:45:40	15222	TELM-06	1	Pass	0	0	0	11
M_TCP_MedicalEvent	m_MedicalEvent012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent012\Hardware\m_MedicalEvent012.exe	2.3	2010/01/18 15:47:55	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:42:25	6095	TELM-06	1	Pass	0	0	0	3
M_TCP_MedicalEvent	m_MedicalEvent014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent014\Hardware\m_MedicalEvent014.exe	2.5	2010/09/08 13:20:31	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:37:46	15209	TELM-06	1	Pass	0	0	0	18
M_TCP_MedicalEvent	m_MedicalEvent015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent015\Hardware\m_MedicalEvent015.exe	2.9	2010/09/07 20:16:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 09:35:24	56535	TELM-06	1	Pass	0	0	0	152
M_TCP_MedicalEvent	m_MedicalEvent016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent016\Hardware\m_MedicalEvent016.exe	2.9	2010/05/12 18:44:37	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 13:05:23	284059	TELM-06	1	Pass	0	0	0	375
M_TCP_MedicalEvent	m_MedicalEvent017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent017\Hardware\m_MedicalEvent017.exe	2.8	2010/09/07 20:17:30	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 17:02:36	612345	TELM-06	1	Pass	0	0	0	13
M_TCP_MedicalEvent	m_MedicalEvent018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent018\Hardware\m_MedicalEvent018.exe	2.5	2010/01/18 15:50:21	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 12:27:50	272170	TELM-06	1	Pass	0	0	0	4627
M_TCP_MedicalEvent	m_MedicalEvent019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent019\Hardware\m_MedicalEvent019.exe	2.5	2010/01/18 15:50:59	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:35:50	7269	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent020\Hardware\m_MedicalEvent020.exe	2.3	2010/01/18 15:51:30	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:33:20	40559	TELM-06	1	Pass	0	0	0	9
M_TCP_MemoryTest	m_MemoryTest001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest001\Hardware\m_MemoryTest001.exe	2.7	2010/06/18 12:28:23	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 21:57:13	32905	TELM-06	1	Pass	0	0	0	33
M_TCP_MemoryTest	m_MemoryTest002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest002\Hardware\m_MemoryTest002.exe	2.4	2010/05/05 19:48:21	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 08:35:45	240603	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest003\Hardware\m_MemoryTest003.exe	2.3	2010/05/05 20:08:58	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 14:29:36	480014	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest004\Hardware\m_MemoryTest004.exe	2.3	2010/05/05 20:09:49	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:24:47	240230	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest005\Hardware\m_MemoryTest005.exe	2.3	2010/05/05 20:10:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 13:59:37	478661	TELM-06	1	Pass	0	0	0	75
M_TCP_Network	m_Network001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network001\Hardware\m_Network001.exe	2.6	2010/05/21 15:45:43	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 10:52:54	24619	TELM-06	1	Pass	0	0	0	1461
M_TCP_Network	m_Network002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network002\Hardware\m_Network002.exe	2.6	2010/05/12 12:01:17	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 14:58:38	34731	TELM-06	1	Pass	0	0	0	82
M_TCP_Network	m_Network003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network003\Hardware\m_Network003.exe	2.1	2009/12/16 22:27:06	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:49:40	64697	TELM-06	1	Pass	0	0	0	86
M_TCP_Network	m_Network004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network004\Hardware\m_Network004.exe	2.1	2010/04/21 17:14:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:40:43	2977	TELM-06	1	Pass	0	0	0	6
M_TCP_Network	m_Network005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network005\Hardware\m_Network005.exe	2.3	2010/05/12 12:02:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:37:43	58140	TELM-06	1	Pass	0	0	0	50
M_TCP_Network	m_Network006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network006\Hardware\m_Network006.exe	2.5	2010/05/21 16:49:56	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:36:46	15450	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network007\Hardware\m_Network007.exe	2.5	2010/05/21 17:09:04	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:33:43	15314	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network008\Hardware\m_Network008.exe	2.5	2010/05/21 17:13:03	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:31:30	10304	TELM-06	1	Pass	0	0	0	1380
M_TCP_Network	m_Network009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network009\Hardware\m_Network009.exe	2.5	2010/05/21 17:21:30	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:30:18	10289	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network010\Hardware\m_Network010.exe	2.5	2010/05/21 17:34:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:27:28	10268	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network011\Hardware\m_Network011.exe	2.3	2010/05/12 12:03:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:22:45	209509	TELM-06	1	Pass	0	0	0	122
M_TCP_Network	m_Network013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network013\Hardware\m_Network013.exe	2.5	2010/07/01 13:00:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:17:56	141725	TELM-06	1	Pass	0	0	0	392
M_TCP_Network	m_Network014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network014\Hardware\m_Network014.exe	2.3	2010/05/12 12:06:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:16:03	26677	TELM-06	1	Pass	0	0	0	12
M_TCP_Network	m_Network015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network015\Hardware\m_Network015.exe	2.3	2010/05/24 13:53:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:12:02	35777	TELM-06	1	Pass	0	0	0	25
M_TCP_Network	m_Network016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network016\Hardware\m_Network016.exe	2.1	2010/05/12 12:09:43	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:08:42	55444	TELM-06	1	Pass	0	0	0	262
M_TCP_Network	m_Network017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network017\Hardware\m_Network017.exe	2.4	2010/06/07 13:19:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:06:48	12738	TELM-06	1	Pass	0	0	0	41
M_TCP_Network	m_Network018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network018\Hardware\m_Network018.exe	2.5	2010/06/07 13:14:52	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:01:35	21761	TELM-06	1	Pass	0	0	0	11
M_TCP_Network	m_Network019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network019\Hardware\m_Network019.exe	2.2	2010/05/12 12:12:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:56:35	56700	TELM-06	1	Pass	0	0	0	51
M_TCP_Network	m_Network020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network020\Hardware\m_Network020.exe	2.2	2010/05/12 12:14:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:54:07	60484	TELM-06	1	Pass	0	0	0	67
M_TCP_Network	m_Network021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network021\Hardware\m_Network021.exe	2.2	2010/05/12 12:16:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:52:13	60290	TELM-06	1	Pass	0	0	0	133
M_TCP_Network	m_Network022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network022\Hardware\m_Network022.exe	2.2	2010/05/12 12:17:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:48:09	60297	TELM-06	1	Pass	0	0	0	175
M_TCP_Network	m_Network023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network023\Hardware\m_Network023.exe	2.1	2010/05/12 12:18:23	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:46:23	19524	TELM-06	1	Pass	0	0	0	454
M_TCP_Network	m_Network024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network024\Hardware\m_Network024.exe	2.2	2010/05/24 14:41:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:45:02	47789	TELM-06	1	Pass	0	0	0	46
M_TCP_Network	m_Network025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network025\Hardware\m_Network025.exe	2.1	2010/05/12 12:20:01	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:42:02	19646	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network026\Hardware\m_Network026.exe	2.3	2010/05/12 12:20:56	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:40:32	20693	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network027\Hardware\m_Network027.exe	2.1	2010/05/12 12:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:39:51	21178	TELM-06	1	Pass	0	0	0	7
M_TCP_Network	m_Network028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network028\Hardware\m_Network028.exe	2.1	2010/06/07 13:22:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:38:58	14321	TELM-06	1	Pass	0	0	0	22
M_TCP_Network	m_Network029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network029\Hardware\m_Network029.exe	2.4	2010/02/09 21:21:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:54:41	271601	TELM-06	1	Pass	0	0	0	48
M_TCP_Open	m_Open001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open001\Hardware\m_Open001.exe	2.11	2010/05/20 13:50:26	2.11	Up-to-date	Baseline 17_1	None	09/10/2010 05:25:25	31718	TELM-06	1	Pass	0	0	0	335
M_TCP_Open	m_Open002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open002\Hardware\m_Open002.exe	2.3	2010/01/18 15:52:02	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:23:07	18640	TELM-06	1	Pass	0	0	0	31
M_TCP_Open	m_Open003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open003\Hardware\m_Open003.exe	2.0	2009/10/15 19:09:20	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 05:15:34	6440	TELM-06	1	Pass	0	0	0	3
M_TCP_Open	m_Open004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open004\Hardware\m_Open004.exe	2.6	2010/06/22 20:01:21	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:09:21	13160	TELM-06	1	Pass	0	0	0	21
M_TCP_Open	m_Open005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open005\Hardware\m_Open005.exe	2.3	2010/06/22 17:40:45	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:05:30	16782	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open006\Hardware\m_Open006.exe	2.2	2010/03/15 14:26:09	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:01:23	21298	TELM-06	1	Pass	0	0	0	1173
M_TCP_Open	m_Open007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open007\Hardware\m_Open007.exe	2.7	2010/03/15 14:46:20	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 04:57:43	17439	TELM-06	1	Pass	0	0	0	1181
M_TCP_Open	m_Open008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open008\Hardware\m_Open008.exe	2.1	2009/10/30 14:23:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:54:16	7714	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open009\Hardware\m_Open009.exe	2.7	2010/06/16 00:36:52	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 04:52:20	12290	TELM-06	1	Pass	0	0	0	1268
M_TCP_Open	m_Open010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open010\Hardware\m_Open010.exe	2.6	2010/03/15 15:05:50	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:50:36	17321	TELM-06	1	Pass	0	0	0	3965
M_TCP_Open	m_Open011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open011\Hardware\m_Open011.exe	2.2	2010/02/10 13:31:30	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:48:12	6505	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open012\Hardware\m_Open012.exe	2.4	2010/02/26 17:01:14	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:45:54	15439	TELM-06	1	Pass	0	0	0	10
M_TCP_Open	m_Open013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open013\Hardware\m_Open013.exe	2.5	2010/02/10 13:34:06	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:43:18	9370	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open014\Hardware\m_Open014.exe	2.6	2010/02/10 13:35:39	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:41:14	8548	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open015\Hardware\m_Open015.exe	2.0	2010/01/28 19:00:26	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 04:35:18	8877	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open030\Hardware\m_Open030.exe	2.1	2010/05/10 13:57:56	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 02:00:37	58114	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open031\Hardware\m_Open031.exe	2.7	2009/11/19 21:48:20	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 01:37:00	63772	TELM-06	1	Pass	0	0	0	13
M_TCP_Open	m_Open032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open032\Hardware\m_Open032.exe	2.3	2010/02/03 22:17:27	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 21:41:36	32340	TELM-06	1	Pass	0	0	0	18
M_TCP_Open	m_Open033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open033\Hardware\m_Open033.exe	2.3	2010/02/03 22:32:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 21:22:47	9989	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open034\Hardware\m_Open034.exe	2.2	2010/02/17 14:19:09	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:29:09	56852	TELM-06	1	Pass	0	0	0	42
M_TCP_Open	m_Open035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open035\Hardware\m_Open035.exe	2.2	2010/02/17 14:10:57	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:26:29	9889	TELM-06	1	Pass	0	0	0	32
M_TCP_Open	m_Open036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open036\Hardware\m_Open036.exe	2.1	2009/10/21 15:52:46	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 01:17:46	36230	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open037\Hardware\m_Open037.exe	2.2	2009/10/21 15:51:07	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 01:05:25	31228	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open038\Hardware\m_Open038.exe	2.7	2010/02/17 14:00:40	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 04:19:29	78398	TELM-06	1	Pass	0	0	0	46
M_TCP_Open	m_Open039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open039\Hardware\m_Open039.exe	2.0	2009/11/09 14:20:44	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 04:13:48	9205	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open040\Hardware\m_Open040.exe	2.1	2010/02/01 19:45:37	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:11:21	17146	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open041\Hardware\m_Open041.exe	2.0	2009/11/09 14:21:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 12:03:31	513504	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open042\Hardware\m_Open042.exe	2.4	2010/07/02 12:38:07	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:07:42	15572	TELM-06	1	Pass	0	0	0	55
M_TCP_Polling	m_Polling016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling016\Hardware\m_Polling016.exe	2.0	2009/09/23 19:25:05	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 08:16:47	49149	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling017\Hardware\m_Polling017.exe	2.1	2010/01/26 20:04:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:35:01	6310	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling018\Hardware\m_Polling018.exe	2.1	2010/01/26 20:05:23	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 08:14:22	81425	TELM-06	1	Pass	0	0	0	4

M_TCP_Polling	m_Polling024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling024\Hardware\m_Polling024.exe	2.5	2010/01/26 20:05:45	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 08:04:23	78537	TELM-06	1	Pass	0	0	0	10
M_TCP_Polling	m_Polling025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling025\Hardware\m_Polling025.exe	2.1	2009/11/24 20:51:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 08:02:15	46092	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling026\Hardware\m_Polling026.exe	2.2	2009/12/03 14:59:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:00:05	19476	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling040\Hardware\m_Polling040.exe	2.1	2010/01/26 20:06:14	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 07:55:50	6019	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling045\Hardware\m_Polling045.exe	2.1	2010/01/26 20:06:42	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 07:47:44	35156	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling046\Hardware\m_Polling046.exe	2.6	2010/04/16 14:21:47	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 07:36:43	47269	TELM-06	1	Pass	0	0	0	38
M_TCP_Polling	m_Polling047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling047\Hardware\m_Polling047.exe	2.1	2010/01/26 20:07:28	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 07:28:25	34998	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling048\Hardware\m_Polling048.exe	2.3	2009/11/24 20:53:00	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 07:22:56	10156	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling049\Hardware\m_Polling049.exe	2.4	2010/03/10 19:44:21	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 07:15:22	46154	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling050\Hardware\m_Polling050.exe	2.6	2010/06/29 17:44:57	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 07:08:52	22836	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling065\Hardware\m_Polling065.exe	2.6	2010/06/09 18:52:38	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 07:01:56	42326	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling066\Hardware\m_Polling066.exe	2.3	2009/11/09 22:57:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:54:28	52721	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling067\Hardware\m_Polling067.exe	2.1	2009/10/22 14:28:02	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:47:04	85015	TELM-06	1	Pass	0	0	0	11
M_TCP_Polling	m_Polling068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling068\Hardware\m_Polling068.exe	2.3	2009/11/05 18:23:27	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:45:30	6847	TELM-06	1	Pass	0	0	0	8
M_TCP_Polling	m_Polling076	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling076\Hardware\m_Polling076.exe	2.1	2009/11/03 15:55:29	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:40:51	10147	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling077	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling077\Hardware\m_Polling077.exe	2.1	2009/11/05 18:34:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 18:47:29	527402	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling080	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling080\Hardware\m_Polling080.exe	2.2	2009/11/05 18:24:04	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 18:21:07	277478	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling081	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling081\Hardware\m_Polling081.exe	2.3	2010/05/27 14:19:41	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:35:00	24078	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling082	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling082\Hardware\m_Polling082.exe	2.2	2009/11/02 18:25:15	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:32:36	9473	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling083	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling083\Hardware\m_Polling083.exe	2.3	2009/11/02 18:45:07	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:29:17	19478	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling085	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling085\Hardware\m_Polling085.exe	2.1	2009/10/22 13:55:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 17:56:21	529517	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling088	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling088\Hardware\m_Polling088.exe	2.5	2010/05/12 17:59:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:26:10	23459	TELM-06	1	Pass	0	0	0	27
M_TCP_Polling	m_Polling089	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling089\Hardware\m_Polling089.exe	2.7	2010/05/11 19:28:17	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 06:18:42	96924	TELM-06	1	Pass	0	0	0	26
M_TCP_Polling	m_Polling090	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling090\Hardware\m_Polling090.exe	2.6	2010/02/15 20:02:24	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:15:10	77375	TELM-06	1	Pass	0	0	0	32
M_TCP_Polling	m_Polling091	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling091\Hardware\m_Polling091.exe	2.6	2010/02/15 20:16:12	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:12:19	13091	TELM-06	1	Pass	0	0	0	13
M_TCP_Polling	m_Polling092	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling092\Hardware\m_Polling092.exe	2.1	2009/11/05 18:41:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 17:29:31	277362	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling093	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling093\Hardware\m_Polling093.exe	2.0	2009/11/09 18:30:12	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:09:13	16759	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling094	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling094\Hardware\m_Polling094.exe	2.0	2009/11/09 18:30:49	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 13:28:16	307337	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling095	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling095\Hardware\m_Polling095.exe	2.0	2009/11/09 18:31:27	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 06:06:10	8388	TELM-06	1	Pass	0	0	0	2
M_TCP_PowerInhibit	m_Power001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power001\Hardware\m_Power001.exe	2.2	2010/06/07 13:10:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:16:52	58756	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power002\Hardware\m_Power002.exe	2.2	2010/01/05 19:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:15:24	10627	TELM-06	1	Pass	0	0	0	884
M_TCP_PowerInhibit	m_Power003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power003\Hardware\m_Power003.exe	2.4	2010/06/22 14:04:45	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:12:42	54233	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power004\Hardware\m_Power004.exe	2.3	2010/03/22 14:26:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:11:19	17815	TELM-06	1	Pass	0	0	0	46
M_TCP_PowerInhibit	m_Power005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power005\Hardware\m_Power005.exe	2.6	2010/07/02 12:34:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 08:58:52	109129	TELM-06	1	Pass	0	0	0	310
M_TCP_PowerInhibit	m_Power006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power006\Hardware\m_Power006.exe	2.6	2010/06/22 12:27:46	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:06:12	192754	TELM-06	1	Pass	0	0	0	3124
M_TCP_PowerInhibit	m_Power007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power007\Hardware\m_Power007.exe	2.1	2009/12/14 16:39:57	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:03:22	19182	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power008\Hardware\m_Power008.exe	2.2	2009/12/14 16:44:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:01:00	34705	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power009\Hardware\m_Power009.exe	2.4	2010/07/01 15:00:10	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:59:39	12179	TELM-06	1	Pass	0	0	0	9
M_TCP_PowerInhibit	m_Power010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power010\Hardware\m_Power010.exe	2.1	2010/02/16 22:18:45	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:57:41	12017	TELM-06	1	Pass	0	0	0	26
M_TCP_PowerInhibit	m_Power011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power011\Hardware\m_Power011.exe	2.1	2010/03/11 19:04:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:54:19	88835	TELM-06	1	Pass	0	0	0	116
M_TCP_RawMode	m_RawMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode001\Hardware\m_RawMode001.exe	2.4	2010/01/26 20:08:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:14:07	7483	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode002\Hardware\m_RawMode002.exe	2.6	2010/01/27 13:00:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:12:13	9642	TELM-06	1	Pass	0	0	0	87
M_TCP_RawMode	m_RawMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode003\Hardware\m_RawMode003.exe	2.5	2010/01/27 13:03:49	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:10:21	11293	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode004\Hardware\m_RawMode004.exe	2.6	2010/01/27 13:09:25	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:08:29	7143	TELM-06	1	Pass	0	0	0	2
M_TCP_RawMode	m_RawMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode005\Hardware\m_RawMode005.exe	2.3	2010/01/26 20:10:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:06:33	13334	TELM-06	1	Pass	0	0	0	9
M_TCP_RawMode	m_RawMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode006\Hardware\m_RawMode006.exe	2.2	2010/01/26 20:11:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:34:13	9145	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode007\Hardware\m_RawMode007.exe	2.3	2010/06/09 21:03:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:04:46	11179	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode008\Hardware\m_RawMode008.exe	2.4	2010/01/27 13:14:20	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:03:00	16167	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode009\Hardware\m_RawMode009.exe	2.2	2009/12/14 16:51:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:01:04	15774	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode010\Hardware\m_RawMode010.exe	2.1	2009/10/27 17:32:20	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:59:09	3241	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode011\Hardware\m_RawMode011.exe	2.2	2010/02/01 16:15:38	2.2	Up-to-date	Baseline 17_1	None	09/13/2010 09:08:47	8678	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode012\Hardware\m_RawMode012.exe	2.3	2010/01/26 20:13:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:56:08	11586	TELM-06	1	Pass	0	0	0	3
M_TCP_Reset	m_Reset001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset001\Hardware\m_Reset001.exe	2.3	2010/01/06 21:25:12	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:30:46	8956	TELM-06	1	Pass	0	0	0	23
M_TCP_Reset	m_Reset002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset002\Hardware\m_Reset002.exe	2.4	2010/01/28 21:52:07	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:28:21	20387	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security002\Hardware\m_Security002.exe	2.9	2010/06/03 15:18:25	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 11:05:00	457358	TELM-06	1	Pass	0	0	0	154
M_TCP_Security	m_Security003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security003\Hardware\m_Security003.exe	2.4	2010/02/17 16:20:41	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:25:57	6780	TELM-06	1	Pass	0	0	0	73
M_TCP_Security	m_Security004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security004\Hardware\m_Security004.exe	2.4	2010/02/15 15:01:00	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:20:52	253979	TELM-06	1	Pass	0	0	0	112
M_TCP_Security	m_Security005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security005\Hardware\m_Security005.exe	2.2	2010/02/17 16:23:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:19:05	4038	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security006\Hardware\m_Security006.exe	2.4	2010/02/17 14:44:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:13:16	144490	TELM-06	1	Pass	0	0	0	108
M_TCP_Security	m_Security007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security007\Hardware\m_Security007.exe	2.6	2010/02/18 15:13:38	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:07:10	97654	TELM-06	1	Pass	0	0	0	126
M_TCP_Security	m_Security008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security008\Hardware\m_Security008.exe	2.3	2010/02/17 16:28:05	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:04:12	6765	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security009\Hardware\m_Security009.exe	2.3	2010/02/17 16:32:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:01:42	6178	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security010\Hardware\m_Security010.exe	2.3	2010/02/18 17:45:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:55:06	38512	TELM-06	1	Pass	0	0	0	113
M_TCP_Security	m_Security011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security011\Hardware\m_Security011.exe	2.5	2010/06/03 18:16:10	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:52:59	20116	TELM-06	1	Pass	0	0	0	58
M_TCP_Security	m_Security012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security012\Hardware\m_Security012.exe	2.9	2010/02/18 15:14:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 16:48:09	4695	TELM-06	1	Pass	0	0	0	132
M_TCP_Security	m_Security013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security013\Hardware\m_Security013.exe	2.5	2010/02/18 15:15:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:37:58	11698	TELM-06	1	Pass	0	0	0	140
M_TCP_Security	m_Security014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security014\Hardware\m_Security014.exe	2.2	2010/06/03 18:21:27	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 23:05:49	14909	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security015\Hardware\m_Security015.exe	2.4	2010/02/17 17:58:26	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 16:09:32	596313	TELM-06	1	Pass	0	0	0	34
M_TCP_Security	m_Security016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security016\Hardware\m_Security016.exe	2.2	2010/06/08 19:56:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:35:31	10649	TELM-06	1	Pass	0	0	0	146
M_TCP_Security	m_Security017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security017\Hardware\m_Security017.exe	2.2	2009/12/08 17:29:14	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:32:59	14534	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security018\Hardware\m_Security018.exe	2.4	2010/03/18 16:05:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:30:00	40643	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security019\Hardware\m_Security019.exe	2.3	2010/03/26 13:43:17	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:26:46	90586	TELM-06	1	Pass	0	0	0	1220
M_TCP_Security	m_Security020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security020\Hardware\m_Security020.exe	2.3	2010/06/03 18:22:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:24:44	22009	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security021\Hardware\m_Security021.exe	2.5	2010/06/03 18:37:52	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:22:38	40932	TELM-06	1	Pass	0	0	0	81
M_TCP_Security	m_Security022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security022\Hardware\m_Security022.exe	2.9	2010/06/18 19:05:16	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 08:45:07	290902	TELM-06	1	Pass	0	0	0	55496
M_TCP_Security	m_Security023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security023\Hardware\m_Security023.exe	2.12	2010/06/10 17:30:47	2.12	Up-to-date	Baseline 17_1	None	09/10/2010 15:45:21	51610	TELM-06	1	Pass	0	0	0	1119
M_TCP_Security	m_Security024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security024\Hardware\m_Security024.exe	2.3	2010/03/26 13:06:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 10:56:51	30652	TELM-06	1	Pass	0	0	0	4
M_TCP_Security	m_Security025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security025\Hardware\m_Security025.exe	2.3	2010/06/10 14:39:09	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 10:42:35	48415	TELM-06	1	Pass	0	0	0	158
M_TCP_Security	m_Security026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security026\Hardware\m_Security026.exe	2.3	2010/02/18 15:15:54	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:20:26	4275	TELM-06	1	Pass	0	0	0	60
M_TCP_Security	m_Security027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security027\Hardware\m_Security027.exe	2.1	2010/06/29 19:03:00	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:17:58	42221	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security028\Hardware\m_Security028.exe	2.1	2010/06/29 19:26:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:15:28	47101	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security029\Hardware\m_Security029.exe	2.2	2010/03/11 18:29:14	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:12:13	20519	TELM-06	1	Pass	0	0	0	46
M_TCP_Security	m_Security030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security030\Hardware\m_Security030.exe	2.2	2010/02/08 13:02:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:10:01	11944	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security031\Hardware\m_Security031.exe	2.1	2010/06/03 18:44:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:07:49	23228	TELM-06	1	Pass	0	0	0	16
M_TCP_Security	m_Security032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security032\Hardware\m_Security032.exe	2.1	2010/02/03 12:51:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:06:27	8266	TELM-06	1	Pass	0	0	0	11
M_TCP_Security	m_Security033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security033\Hardware\m_Security033.exe	2.1	2010/01/27 21:32:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:04:30	14068	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security034\Hardware\m_Security034.exe	2.3	2010/03/26 13:19:51	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:58:14	118541	TELM-06	1	Pass	0	0	0	15
M_TCP_Security	m_Security035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security035\Hardware\m_Security035.exe	2.0	2010/01/14 21:18:30	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:53:31	17561	TELM-06	1	Pass	0	0	0	30
M_TCP_Security	m_Security036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security036\Hardware\m_Security036.exe	2.5	2010/05/07 14:41:43	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:50:49	86781	TELM-06	1	Pass	0	0	0	10
M_TCP_Security	m_Security037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security037\Hardware\m_Security037.exe	2.6	2010/06/03 18:54:41	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 22:57:53	51184	TELM-06	1	Pass	0	0	0	119
M_TCP_Security	m_Security038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security038\Hardware\m_Security038.exe	2.9	2010/06/07 13:02:07	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 10:28:43	167454	TELM-06	1	Pass	0	0	0	170
M_TCP_Security	m_Security039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security039\Hardware\m_Security039.exe	2.2	2010/02/18 15:16:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:49:16	56429	TELM-06	1	Pass	0	0	0	55
M_TCP_Security	m_Security040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security040\Hardware\m_Security040.exe	2.3	2010/06/03 20:43:31	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:43:21	278351	TELM-06	1	Pass	0	0	0	33
M_TCP_Security	m_Security041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security041\Hardware\m_Security041.exe	2.6	2010/04/21 10:55:08	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 15:41:01	65776	TELM-06	1	Pass	0	0	0	77
M_TCP_Security	m_Security042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security042\Hardware\m_Security042.exe	2.1	2010/04/08 12:57:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:39:03	58388	TELM-06	1	Pass	0	0	0	48
M_TCP_Security	m_Security043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security043\Hardware\m_Security043.exe	2.0	2010/02/16 20:27:42	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:36:01	68761	TELM-06	1	Pass	0	0	0	56
M_TCP_Security	m_Security044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security044\Hardware\m_Security044.exe	2.2	2010/05/13 19:14:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:32:37	70720	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security045\Hardware\m_Security045.exe	2.3	2010/05/12 17:38:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:30:14	71374	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security046\Hardware\m_Security046.exe	2.2	2010/06/07 13:05:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:28:54	3397	TELM-06	1	Pass	0	0	0	30
M_TCP_WirelessWakeup	m_Wakeup001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup001\Hardware\m_Wakeup001.exe	2.3	2010/06/21 17:44:25	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 10:09:10	291342	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup002\Hardware\m_Wakeup002.exe	2.5	2010/05/12 16:16:45	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:05:00	35313	TELM-06	1	Pass	0	0	0	12
M_TCP_WirelessWakeup	m_Wakeup003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup003\Hardware\m_Wakeup003.exe	2.2	2010/03/10 17:59:29	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:01:16	14383	TELM-06	1	Pass	0	0	0	11
M_TCP_WirelessWakeup	m_Wakeup004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup004\Hardware\m_Wakeup004.exe	2.1	2009/12/08 16:08:20	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 22:25:55	19178	TELM-06	1	Pass	0	0	0	31
M_TCP_WirelessWakeup	m_Wakeup005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup005\Hardware\m_Wakeup005.exe	2.3	2010/06/08 21:32:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:57:25	20375	TELM-06	1	Pass	0	0	0	40
M_TCP_WirelessWakeup	m_Wakeup006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup006\Hardware\m_Wakeup006.exe	2.3	2010/01/27 00:35:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:54:51	10369	TELM-06	1	Pass	0	0	0	55
M_TCP_WirelessWakeup	m_Wakeup007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup007\Hardware\m_Wakeup007.exe	2.1	2009/12/04 19:28:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:53:00	12849	TELM-06	1	Pass	0	0	0	3
M_TCP_WirelessWakeup	m_Wakeup008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup008\Hardware\m_Wakeup008.exe	2.5	2010/05/07 15:41:21	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:49:58	54619	TELM-06	1	Pass	0	0	0	24
M_TCP_WirelessWakeup	m_Wakeup009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup009\Hardware\m_Wakeup009.exe	2.6	2010/05/12 16:17:49	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:47:06	51995	TELM-06	1	Pass	0	0	0	32
M_TCP_WirelessWakeup	m_Wakeup010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup010\Hardware\m_Wakeup010.exe	2.4	2010/05/12 16:20:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:45:16	45163	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup011\Hardware\m_Wakeup011.exe	2.3	2010/04/06 18:07:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:41:58	28016	TELM-06	1	Pass	0	0	0	25
M_TCP_WirelessWakeup	m_Wakeup012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup012\Hardware\m_Wakeup012.exe	2.6	2010/05/12 16:20:45	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:39:30	48767	TELM-06	1	Pass	0	0	0	45

M_TCP_WirelessWakeup	m_Wakeup013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup013\Hardware\m_Wakeup013.exe	2.2	2010/02/24 12:20:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:35:57	127557	TELM-06	1	Pass	0	0	0	2
M_TCP_WirelessWakeup	m_Wakeup014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode4\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup014\Hardware\m_Wakeup014.exe	2.3	2010/06/04 11:50:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:33:32	74529	TELM-06	1	Pass	0	0	0	30
M_TCP_AntennaSelection	m_Antenna001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna001\Hardware\m_Antenna001.exe	2.4	2010/06/16 12:31:48	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 12:56:10	239252	TELM-06	1	Pass	0	0	0	127
M_TCP_AntennaSelection	m_Antenna002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna002\Hardware\m_Antenna002.exe	2.0	2010/06/16 12:29:50	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 12:16:51	287084	TELM-06	1	Pass	0	0	0	99
M_TCP_Channel	m_ChannelRecovery001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery001\Hardware\m_ChannelRecovery001.exe	2.5	2010/04/08 14:25:18	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:53:11	15080	TELM-06	1	Pass	0	0	0	26
M_TCP_Channel	m_ChannelRecovery002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery002\Hardware\m_ChannelRecovery002.exe	2.2	2009/11/30 22:31:32	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:51:27	17552	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery003\Hardware\m_ChannelRecovery003.exe	2.2	2009/11/30 22:32:21	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:48:49	17538	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery004\Hardware\m_ChannelRecovery004.exe	2.2	2010/05/26 12:04:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:47:13	14429	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery005\Hardware\m_ChannelRecovery005.exe	2.5	2010/06/21 18:34:44	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:44:37	14279	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery006\Hardware\m_ChannelRecovery006.exe	2.4	2010/05/07 17:09:38	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:42:19	15882	TELM-06	1	Pass	0	0	0	35
M_TCP_Channel	m_ChannelRecovery007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery007\Hardware\m_ChannelRecovery007.exe	2.7	2010/05/11 11:49:24	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 04:35:57	27822	TELM-06	1	Pass	0	0	0	89
M_TCP_Channel	m_ChannelRecovery008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery008\Hardware\m_ChannelRecovery008.exe	2.6	2010/05/13 15:05:39	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:30:42	32926	TELM-06	1	Pass	0	0	0	575
M_TCP_Channel	m_ChannelRecovery009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery009\Hardware\m_ChannelRecovery009.exe	2.6	2010/06/22 15:16:15	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:27:56	60221	TELM-06	1	Pass	0	0	0	342
M_TCP_Channel	m_ChannelRecovery010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery010\Hardware\m_ChannelRecovery010.exe	2.8	2010/05/26 15:48:50	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 04:25:50	25032	TELM-06	1	Pass	0	0	0	138
M_TCP_Channel	m_ChannelRecovery011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery011\Hardware\m_ChannelRecovery011.exe	2.5	2010/05/13 12:31:10	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 04:19:10	15676	TELM-06	1	Pass	0	0	0	14
M_TCP_Channel	m_ChannelRecovery012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery012\Hardware\m_ChannelRecovery012.exe	2.6	2010/09/07 20:36:00	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:12:56	48395	TELM-06	1	Pass	0	0	0	195
M_TCP_Channel	m_ChannelRecovery013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery013\Hardware\m_ChannelRecovery013.exe	2.2	2010/01/05 19:17:54	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:10:54	24270	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery014\Hardware\m_ChannelRecovery014.exe	2.8	2010/05/13 13:19:47	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 04:06:32	57504	TELM-06	1	Pass	0	0	0	23
M_TCP_Channel	m_ChannelRecovery015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery015\Hardware\m_ChannelRecovery015.exe	2.6	2010/06/03 18:01:04	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:05:14	12561	TELM-06	1	Pass	0	0	0	88
M_TCP_Channel	m_ChannelRecovery016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery016\Hardware\m_ChannelRecovery016.exe	2.4	2010/02/11 20:10:03	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:03:17	37997	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery017\Hardware\m_ChannelRecovery017.exe	2.3	2010/02/11 17:38:13	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:01:47	10351	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery018\Hardware\m_ChannelRecovery018.exe	2.3	2010/02/01 19:41:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:00:35	17986	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery020\Hardware\m_ChannelRecovery020.exe	2.7	2010/06/23 17:42:40	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 03:58:39	37789	TELM-06	1	Pass	0	0	0	109
M_TCP_Channel	m_ChannelRecovery021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery021\Hardware\m_ChannelRecovery021.exe	2.3	2010/04/13 18:33:24	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:56:16	29246	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery022\Hardware\m_ChannelRecovery022.exe	2.3	2010/04/13 18:36:50	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:53:30	19857	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery023\Hardware\m_ChannelRecovery023.exe	2.5	2010/04/13 18:41:23	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:51:51	29849	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery024\Hardware\m_ChannelRecovery024.exe	2.4	2010/04/07 12:33:58	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:49:41	19873	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery025\Hardware\m_ChannelRecovery025.exe	2.2	2010/02/01 19:43:28	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:48:21	11135	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery026\Hardware\m_ChannelRecovery026.exe	2.2	2010/02/11 21:01:06	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:47:03	11551	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery027\Hardware\m_ChannelRecovery027.exe	2.2	2010/02/03 17:18:54	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:45:46	13542	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery028\Hardware\m_ChannelRecovery028.exe	2.2	2010/02/03 17:19:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:44:28	11496	TELM-06	1	Pass	0	0	0	10
M_TCP_Channel	m_ChannelRecovery029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery029\Hardware\m_ChannelRecovery029.exe	2.2	2009/12/18 20:13:33	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:43:18	13606	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery030\Hardware\m_ChannelRecovery030.exe	2.10	2010/06/23 21:15:31	2.10	Up-to-date	Baseline 17_1	None	09/10/2010 03:42:05	18614	TELM-06	1	Pass	0	0	0	30
M_TCP_Channel	m_ChannelRecovery031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery031\Hardware\m_ChannelRecovery031.exe	2.6	2010/04/13 18:46:58	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 03:37:01	18926	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery032\Hardware\m_ChannelRecovery032.exe	2.5	2010/06/23 21:50:13	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:33:25	18240	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery033\Hardware\m_ChannelRecovery033.exe	2.3	2010/04/07 17:51:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:31:03	19248	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery038\Hardware\m_ChannelRecovery038.exe	2.6	2010/05/26 15:55:12	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 03:29:03	20902	TELM-06	1	Pass	0	0	0	175
M_TCP_Channel	m_ChannelRecovery039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery039\Hardware\m_ChannelRecovery039.exe	2.3	2010/02/10 20:47:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:27:00	12288	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery040\Hardware\m_ChannelRecovery040.exe	2.5	2010/06/07 12:54:38	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 03:25:35	15578	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery041\Hardware\m_ChannelRecovery041.exe	2.1	2009/10/22 13:18:38	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:21:28	38449	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery042\Hardware\m_ChannelRecovery042.exe	2.3	2009/12/03 15:57:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:19:14	22027	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery043\Hardware\m_ChannelRecovery043.exe	2.6	2010/06/24 18:37:58	2.6	Up-to-date	Baseline 17_1	None	09/13/2010 09:08:57	56053	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery045\Hardware\m_ChannelRecovery045.exe	2.2	2009/11/12 21:22:11	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:13:51	11524	TELM-06	1	Pass	0	0	0	42
M_TCP_Channel	m_ChannelRecovery046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery046\Hardware\m_ChannelRecovery046.exe	2.3	2009/11/16 18:40:10	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:07:20	12605	TELM-06	1	Pass	0	0	0	49
M_TCP_Channel	m_ChannelRecovery047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery047\Hardware\m_ChannelRecovery047.exe	2.1	2009/11/16 18:58:46	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:04:29	52517	TELM-06	1	Pass	0	0	0	95
M_TCP_Channel	m_ChannelRecovery048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery048\Hardware\m_ChannelRecovery048.exe	2.3	2010/06/24 18:46:14	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:59:06	62725	TELM-06	1	Pass	0	0	0	156
M_TCP_Channel	m_ChannelRecovery049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery049\Hardware\m_ChannelRecovery049.exe	2.5	2010/03/26 13:55:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:55:27	36668	TELM-06	1	Pass	0	0	0	204
M_TCP_Channel	m_ChannelRecovery050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery050\Hardware\m_ChannelRecovery050.exe	2.2	2009/11/20 13:38:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:53:23	18837	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery051\Hardware\m_ChannelRecovery051.exe	2.3	2010/06/24 19:40:38	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:49:04	82860	TELM-06	1	Pass	0	0	0	196
M_TCP_Channel	m_ChannelRecovery053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery053\Hardware\m_ChannelRecovery053.exe	2.3	2010/06/24 21:57:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:44:51	31301	TELM-06	1	Pass	0	0	0	96
M_TCP_Channel	m_ChannelRecovery054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery054\Hardware\m_ChannelRecovery054.exe	2.4	2010/06/24 22:31:30	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:42:53	9016	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery055\Hardware\m_ChannelRecovery055.exe	2.4	2010/02/11 14:12:25	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 05:06:31	2022352	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery061	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery061\Hardware\m_ChannelRecovery061.exe	2.3	2010/04/06 11:52:17	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:38:54	20741	TELM-06	1	Pass	0	0	0	12
M_TCP_Channel	m_ChannelRecovery062	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery062\Hardware\m_ChannelRecovery062.exe	2.2	2010/06/25 15:47:28	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:35:32	13122	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery063	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery063\Hardware\m_ChannelRecovery063.exe	2.9	2010/05/13 17:43:56	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 02:32:31	24547	TELM-06	1	Pass	0	0	0	39
M_TCP_Channel	m_ChannelRecovery064	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery064\Hardware\m_ChannelRecovery064.exe	2.4	2010/05/06 14:14:13	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:29:59	40910	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery065\Hardware\m_ChannelRecovery065.exe	2.2	2010/05/06 15:43:30	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:27:22	37314	TELM-06	1	Pass	0	0	0	7
M_TCP_Channel	m_ChannelRecovery066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery066\Hardware\m_ChannelRecovery066.exe	2.2	2010/05/06 15:49:13	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:23:44	37418	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery067\Hardware\m_ChannelRecovery067.exe	2.2	2010/05/06 15:58:13	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:20:58	40746	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery068\Hardware\m_ChannelRecovery068.exe	2.3	2010/06/03 18:10:46	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:17:37	34695	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery069	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery069\Hardware\m_ChannelRecovery069.exe	2.3	2010/05/10 15:40:47	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:43:45	1009298	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery070	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery070\Hardware\m_ChannelRecovery070.exe	2.1	2010/02/04 18:14:32	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:16:29	12782	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery071	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery071\Hardware\m_ChannelRecovery071.exe	2.1	2010/02/04 17:57:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:14:15	15792	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery072	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery072\Hardware\m_ChannelRecovery072.exe	2.3	2010/05/07 11:42:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:11:29	44162	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery073	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery073\Hardware\m_ChannelRecovery073.exe	2.3	2010/05/07 11:43:00	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:09:19	44170	TELM-06	1	Pass	0	0	0	8
M_TCP_Close	m_Close001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close001\Hardware\m_Close001.exe	2.6	2010/03/29 17:53:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 11:00:15	7427	TELM-06	1	Pass	0	0	0	51
M_TCP_Close	m_Close002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close002\Hardware\m_Close002.exe	2.3	2010/01/26 20:03:49	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 10:52:39	8701	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close003\Hardware\m_Close003.exe	2.3	2010/06/24 15:01:11	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 10:37:37	6976	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close004\Hardware\m_Close004.exe	2.5	2010/03/16 12:03:27	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 10:23:29	14977	TELM-06	1	Pass	0	0	0	142
M_TCP_Close	m_Close005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close005\Hardware\m_Close005.exe	2.5	2010/05/24 17:00:08	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 10:05:28	19655	TELM-06	1	Pass	0	0	0	104
M_TCP_Close	m_Close006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close006\Hardware\m_Close006.exe	2.2	2009/11/19 20:38:31	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 09:59:52	11290	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close007\Hardware\m_Close007.exe	2.1	2010/02/04 18:46:09	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 09:50:07	16396	TELM-06	1	Pass	0	0	0	44
M_TCP_Close	m_Close008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close008\Hardware\m_Close008.exe	2.4	2010/05/12 15:22:15	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:35:00	6965	TELM-06	1	Pass	0	0	0	10
M_TCP_Close	m_Close009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close009\Hardware\m_Close009.exe	2.2	2010/02/04 15:29:20	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 09:26:50	7789	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close010\Hardware\m_Close010.exe	2.0	2009/09/21 12:45:39	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 09:09:04	15917	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close011\Hardware\m_Close011.exe	2.0	2009/09/21 12:46:09	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 08:58:33	8158	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close012\Hardware\m_Close012.exe	2.0	2009/09/21 12:46:36	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 08:44:45	11506	TELM-06	1	Pass	0	0	0	3
M_TCP_Close	m_Close013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close013\Hardware\m_Close013.exe	2.0	2009/09/21 12:18:45	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 08:35:34	7909	TELM-06	1	Pass	0	0	0	4
M_TCP_Close	m_Close014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close014\Hardware\m_Close014.exe	2.1	2010/01/29 16:45:05	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 08:24:36	8827	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close016\Hardware\m_Close016.exe	2.2	2010/08/24 12:05:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:16:36	8001	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close017\Hardware\m_Close017.exe	2.0	2009/09/21 12:47:07	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 08:14:10	9555	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close019\Hardware\m_Close019.exe	2.1	2009/10/01 14:47:50	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 08:04:17	5010	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close020\Hardware\m_Close020.exe	2.2	2009/10/30 14:47:42	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:02:03	9479	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close021\Hardware\m_Close021.exe	2.0	2009/09/21 12:48:28	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 08:00:01	2741	TELM-06	1	Pass	0	0	0	1
M_TCP_Close	m_Close022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close022\Hardware\m_Close022.exe	2.2	2009/10/30 14:53:53	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 07:55:36	12356	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_ARQ001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ001\Hardware\m_ARQ001.exe	2.5	2010/06/29 15:20:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:41:35	14085	TELM-06	1	Pass	0	0	0	54
M_TCP_Data	m_ARQ002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ002\Hardware\m_ARQ002.exe	2.0	2009/09/21 18:25:53	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:40:22	7998	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ003\Hardware\m_ARQ003.exe	2.3	2010/04/05 12:22:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:38:45	8507	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_ARQ004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ004\Hardware\m_ARQ004.exe	2.1	2009/11/04 21:22:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:33:36	8198	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_ARQ005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ005\Hardware\m_ARQ005.exe	2.2	2009/12/23 13:26:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:32:29	7555	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ006\Hardware\m_ARQ006.exe	2.1	2009/09/24 16:41:52	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:28:56	8619	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_ARQ007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ007\Hardware\m_ARQ007.exe	2.0	2009/09/21 18:27:29	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:26:04	10678	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_ARQ008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ008\Hardware\m_ARQ008.exe	2.0	2009/09/21 18:28:06	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:24:48	10868	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ009\Hardware\m_ARQ009.exe	2.2	2010/02/04 13:08:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:23:26	9234	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ010\Hardware\m_ARQ010.exe	2.2	2010/02/04 13:19:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:21:34	8984	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ011\Hardware\m_ARQ011.exe	2.3	2010/02/01 13:30:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:20:08	8672	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ012\Hardware\m_ARQ012.exe	2.3	2010/02/18 15:17:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:19:04	8837	TELM-06	1	Pass	0	0	0	211
M_TCP_Data	m_ARQ013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ013\Hardware\m_ARQ013.exe	2.3	2010/02/03 16:38:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:17:13	10713	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_ARQ014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ014\Hardware\m_ARQ014.exe	2.4	2010/04/05 11:08:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:15:31	8577	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ015\Hardware\m_ARQ015.exe	2.6	2010/06/08 15:25:41	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:13:47	13035	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ016\Hardware\m_ARQ016.exe	2.3	2009/11/19 20:22:28	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:11:53	9583	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ017\Hardware\m_ARQ017.exe	2.2	2009/09/30 15:01:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:09:02	7034	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ018\Hardware\m_ARQ018.exe	2.7	2010/06/24 12:34:26	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 22:07:22	10845	TELM-06	1	Pass	0	0	0	370
M_TCP_Data	m_ARQ019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ019\Hardware\m_ARQ019.exe	2.4	2010/06/08 16:07:58	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:04:59	10238	TELM-06	1	Pass	0	0	0	86
M_TCP_Data	m_ARQ020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ020\Hardware\m_ARQ020.exe	2.2	2009/11/18 15:23:22	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:03:57	7613	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ021\Hardware\m_ARQ021.exe	2.2	2009/11/19 18:43:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:02:11	7589	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ022\Hardware\m_ARQ022.exe	2.3	2010/02/03 16:30:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:00:01	8614	TELM-06	1	Pass	0	0	0	137
M_TCP_Data	m_ARQ023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ023\Hardware\m_ARQ023.exe	2.1	2009/11/10 19:12:36	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:58:33	9956	TELM-06	1	Pass	0	0	0	73
M_TCP_Data	m_ARQ024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ024\Hardware\m_ARQ024.exe	2.3	2010/02/08 18:44:31	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:57:08	10268	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ025\Hardware\m_ARQ025.exe	2.1	2009/09/25 14:20:39	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:53:25	8775	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ026\Hardware\m_ARQ026.exe	2.1	2009/09/25 14:21:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:50:20	8702	TELM-06	1	Pass	0	0	0	4

M_TCP_Data	m_ARQ027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ027\Hardware\m_ARQ027.exe	2.1	2009/09/25 14:21:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:48:57	9201	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ028\Hardware\m_ARQ028.exe	2.4	2010/04/19 14:36:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:42:47	18733	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_ARQ029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ029\Hardware\m_ARQ029.exe	2.1	2010/03/08 19:53:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:40:25	29548	TELM-06	1	Pass	0	0	0	555
M_TCP_Data	m_ARQ030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ030\Hardware\m_ARQ030.exe	2.5	2010/06/03 15:11:29	2.5	Up-to-date	Baseline 17_1	None	09/13/2010 09:09:56	17765	TELM-06	1	Pass	0	0	0	355
M_TCP_Data	m_ARQ031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ031\Hardware\m_ARQ031.exe	2.3	2010/06/08 19:57:52	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:34:44	11816	TELM-06	1	Pass	0	0	0	560
M_TCP_Data	m_Data012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data012\Hardware\m_Data012.exe	2.4	2009/11/10 14:18:10	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 03:27:50	37796	TELM-06	1	Pass	0	0	0	18
M_TCP_Data	m_Data013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data013\Hardware\m_Data013.exe	2.3	2010/02/03 15:41:32	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:27:14	103890	TELM-06	1	Pass	0	0	0	335
M_TCP_Data	m_Data014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data014\Hardware\m_Data014.exe	2.3	2010/05/12 19:37:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:23:46	92624	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_Data015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data015\Hardware\m_Data015.exe	2.2	2010/02/03 17:34:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:21:09	8632	TELM-06	1	Pass	0	0	0	14
M_TCP_Data	m_Data016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data016\Hardware\m_Data016.exe	2.2	2010/02/24 20:56:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:17:56	70167	TELM-06	1	Pass	0	0	0	26
M_TCP_Data	m_Data017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data017\Hardware\m_Data017.exe	2.2	2010/02/24 22:22:41	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:11:29	67700	TELM-06	1	Pass	0	0	0	30
M_TCP_Data	m_Data018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data018\Hardware\m_Data018.exe	2.2	2010/02/03 22:48:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:06:20	58968	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_MemoryRead001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead001\Hardware\m_MemoryRead001.exe	2.2	2010/02/01 17:26:47	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:04:44	21388	TELM-06	1	Pass	0	0	0	439
M_TCP_Data	m_MemoryRead002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead002\Hardware\m_MemoryRead002.exe	2.3	2009/11/05 15:03:23	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:02:51	14018	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_MemoryRead003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead003\Hardware\m_MemoryRead003.exe	2.4	2010/04/05 13:08:02	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:00:43	15514	TELM-06	1	Pass	0	0	0	320
M_TCP_Data	m_MemoryRead004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead004\Hardware\m_MemoryRead004.exe	2.3	2010/03/19 16:10:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:58:41	8430	TELM-06	1	Pass	0	0	0	1
M_TCP_Data	m_MemoryRead005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead005\Hardware\m_MemoryRead005.exe	2.3	2009/12/07 19:50:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:54:56	48639	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryRead006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead006\Hardware\m_MemoryRead006.exe	2.6	2010/06/29 21:05:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:50:43	12380	TELM-06	1	Pass	0	0	0	37
M_TCP_Data	m_MemoryRead007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead007\Hardware\m_MemoryRead007.exe	2.1	2009/11/03 12:42:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:47:51	10807	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_MemoryRead008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead008\Hardware\m_MemoryRead008.exe	2.1	2009/09/30 18:09:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:44:29	8012	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead009\Hardware\m_MemoryRead009.exe	2.1	2009/12/07 19:51:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:39:42	11809	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead010\Hardware\m_MemoryRead010.exe	2.2	2009/09/30 17:31:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:37:22	7643	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead011\Hardware\m_MemoryRead011.exe	2.2	2009/09/30 17:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:34:44	7568	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead012\Hardware\m_MemoryRead012.exe	2.2	2009/09/30 17:33:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:32:59	8158	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead013\Hardware\m_MemoryRead013.exe	2.2	2009/09/30 17:33:27	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:27:35	7783	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead014\Hardware\m_MemoryRead014.exe	2.2	2009/09/30 17:33:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:24:39	7756	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead015\Hardware\m_MemoryRead015.exe	2.2	2009/09/30 17:34:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:22:17	7753	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead016\Hardware\m_MemoryRead016.exe	2.1	2009/10/19 18:24:00	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:20:35	7780	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead017\Hardware\m_MemoryRead017.exe	2.2	2009/10/19 15:52:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:18:24	7805	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead019\Hardware\m_MemoryRead019.exe	2.3	2009/12/03 13:57:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:15:16	9427	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead020\Hardware\m_MemoryRead020.exe	2.4	2010/02/09 20:49:06	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:13:18	6897	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryRead021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead021\Hardware\m_MemoryRead021.exe	2.2	2009/12/23 13:34:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:11:22	8665	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryRead030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead030\Hardware\m_MemoryRead030.exe	2.1	2009/10/16 13:54:03	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:10:13	6837	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead031\Hardware\m_MemoryRead031.exe	2.3	2009/12/28 18:23:34	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:08:17	10503	TELM-06	1	Pass	0	0	0	43
M_TCP_Data	m_MemoryRead033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead033\Hardware\m_MemoryRead033.exe	2.3	2009/11/23 18:55:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:06:19	12079	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead034\Hardware\m_MemoryRead034.exe	2.1	2009/10/16 14:15:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:04:35	8221	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead035\Hardware\m_MemoryRead035.exe	2.4	2009/11/23 18:56:23	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:02:40	17346	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead037\Hardware\m_MemoryRead037.exe	2.2	2009/11/23 18:57:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:00:32	12206	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead038\Hardware\m_MemoryRead038.exe	2.3	2010/01/19 17:36:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:58:38	14512	TELM-06	1	Pass	0	0	0	630
M_TCP_Data	m_MemoryRead039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead039\Hardware\m_MemoryRead039.exe	2.1	2009/09/28 15:09:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:57:05	8528	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead040\Hardware\m_MemoryRead040.exe	2.6	2010/01/06 14:10:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:55:33	32486	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryRead041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead041\Hardware\m_MemoryRead041.exe	2.4	2009/11/16 12:52:24	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:53:05	18317	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead042\Hardware\m_MemoryRead042.exe	2.3	2010/06/29 22:24:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:51:38	27175	TELM-06	1	Pass	0	0	0	41
M_TCP_Data	m_MemoryRead043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead043\Hardware\m_MemoryRead043.exe	2.2	2010/02/08 16:55:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:48:44	17737	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead045\Hardware\m_MemoryRead045.exe	2.0	2009/09/22 14:34:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:47:20	6784	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead049\Hardware\m_MemoryRead049.exe	2.3	2009/11/16 12:52:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:44:10	8683	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead051\Hardware\m_MemoryRead051.exe	2.1	2009/10/27 19:22:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:42:56	9759	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryRead052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead052\Hardware\m_MemoryRead052.exe	2.3	2010/06/29 23:18:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:41:19	11118	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite001\Hardware\m_MemoryWrite001.exe	2.2	2009/12/07 19:52:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:39:11	10337	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite002\Hardware\m_MemoryWrite002.exe	2.0	2009/09/22 14:48:30	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:37:15	10594	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite003\Hardware\m_MemoryWrite003.exe	2.1	2009/10/28 14:34:41	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:35:03	10070	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite005\Hardware\m_MemoryWrite005.exe	2.2	2009/12/15 19:05:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:33:32	7847	TELM-06	1	Pass	0	0	0	186
M_TCP_Data	m_MemoryWrite006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite006\Hardware\m_MemoryWrite006.exe	2.3	2009/11/24 17:33:37	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:32:06	11778	TELM-06	1	Pass	0	0	0	145
M_TCP_Data	m_MemoryWrite007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite007\Hardware\m_MemoryWrite007.exe	2.1	2009/10/28 15:09:57	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:30:20	7064	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite008\Hardware\m_MemoryWrite008.exe	2.1	2010/02/01 19:11:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:28:19	7410	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite009\Hardware\m_MemoryWrite009.exe	2.1	2009/10/28 15:17:22	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:26:39	7591	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite010\Hardware\m_MemoryWrite010.exe	2.1	2009/10/28 16:00:35	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:25:19	7520	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite011\Hardware\m_MemoryWrite011.exe	2.1	2009/10/28 16:07:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:23:01	7387	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite012\Hardware\m_MemoryWrite012.exe	2.2	2009/10/28 16:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:21:18	12277	TELM-06	1	Pass	0	0	0	120
M_TCP_Data	m_MemoryWrite013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite013\Hardware\m_MemoryWrite013.exe	2.1	2009/10/28 16:21:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:20:10	7561	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite014\Hardware\m_MemoryWrite014.exe	2.1	2009/10/28 16:29:59	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:18:41	7556	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite015\Hardware\m_MemoryWrite015.exe	2.1	2009/10/28 16:36:30	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:16:21	7614	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryWrite016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite016\Hardware\m_MemoryWrite016.exe	2.5	2010/02/01 21:11:41	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:14:47	9665	TELM-06	1	Pass	0	0	0	1566
M_TCP_Data	m_MemoryWrite017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite017\Hardware\m_MemoryWrite017.exe	2.3	2010/01/19 17:37:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:12:17	8850	TELM-06	1	Pass	0	0	0	977
M_TCP_Data	m_MemoryWrite018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite018\Hardware\m_MemoryWrite018.exe	2.1	2009/10/29 20:06:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:10:24	7002	TELM-06	1	Pass	0	0	0	171
M_TCP_Data	m_MemoryWrite019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite019\Hardware\m_MemoryWrite019.exe	2.0	2009/09/22 15:13:38	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:05:11	9621	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryWrite020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite020\Hardware\m_MemoryWrite020.exe	2.2	2009/10/29 19:44:00	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:03:11	7836	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite022\Hardware\m_MemoryWrite022.exe	2.3	2009/11/03 17:28:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:00:23	9863	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite023\Hardware\m_MemoryWrite023.exe	2.4	2010/02/04 16:15:17	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:58:54	20522	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite024\Hardware\m_MemoryWrite024.exe	2.2	2009/11/24 17:35:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:57:31	7521	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_MemoryWrite025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite025\Hardware\m_MemoryWrite025.exe	2.1	2009/10/29 18:47:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:53:44	10954	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite029\Hardware\m_MemoryWrite029.exe	2.2	2010/02/01 21:00:35	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:50:28	14040	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite032\Hardware\m_MemoryWrite032.exe	2.5	2010/03/15 01:08:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 18:44:29	9639	TELM-06	1	Pass	0	0	0	329
M_TCP_Data	m_MemoryWrite038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite038\Hardware\m_MemoryWrite038.exe	2.1	2009/10/29 18:42:13	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:40:23	7632	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite039\Hardware\m_MemoryWrite039.exe	2.1	2010/02/10 18:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:37:00	9580	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite040\Hardware\m_MemoryWrite040.exe	2.2	2010/02/10 17:56:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:35:31	9707	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryWrite041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite041\Hardware\m_MemoryWrite041.exe	2.2	2010/02/01 20:45:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:34:14	13666	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite042\Hardware\m_MemoryWrite042.exe	2.2	2010/01/19 17:38:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:28:28	10083	TELM-06	1	Pass	0	0	0	629
M_TCP_Data	m_MemoryWrite046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite046\Hardware\m_MemoryWrite046.exe	2.3	2010/02/04 16:51:09	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:26:31	7487	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite047\Hardware\m_MemoryWrite047.exe	2.2	2009/11/11 19:12:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:22:16	8475	TELM-06	1	Pass	0	0	0	98
M_TCP_Data	m_MemoryWrite048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite048\Hardware\m_MemoryWrite048.exe	2.2	2009/11/11 18:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:16:59	9213	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite049\Hardware\m_MemoryWrite049.exe	2.5	2010/02/10 16:51:26	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 18:13:50	21276	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_MemoryWrite052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite052\Hardware\m_MemoryWrite052.exe	2.3	2009/12/23 13:35:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:12:21	11172	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite053\Hardware\m_MemoryWrite053.exe	2.2	2009/11/11 17:01:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:09:00	8568	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite054\Hardware\m_MemoryWrite054.exe	2.6	2010/06/30 17:58:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 18:05:50	10231	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryWrite055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite055\Hardware\m_MemoryWrite055.exe	2.2	2009/11/11 18:04:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:03:56	7384	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite056	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite056\Hardware\m_MemoryWrite056.exe	2.4	2010/02/10 16:32:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:02:25	8805	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode001\Hardware\m_TransferMode001.exe	2.8	2010/05/14 18:48:57	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 18:00:23	17444	TELM-06	1	Pass	0	0	0	4820
M_TCP_Data	m_TransferMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode002\Hardware\m_TransferMode002.exe	2.4	2010/07/02 13:00:52	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:58:17	10004	TELM-06	1	Pass	0	0	0	166
M_TCP_Data	m_TransferMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode003\Hardware\m_TransferMode003.exe	2.0	2009/11/16 20:59:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 17:52:42	7084	TELM-06	1	Pass	0	0	0	129
M_TCP_Data	m_TransferMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode004\Hardware\m_TransferMode004.exe	2.3	2010/01/08 17:07:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:45:10	18177	TELM-06	1	Pass	0	0	0	210
M_TCP_Data	m_TransferMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode005\Hardware\m_TransferMode005.exe	2.7	2010/07/02 13:36:13	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 17:43:19	36277	TELM-06	1	Pass	0	0	0	345
M_TCP_Data	m_TransferMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode006\Hardware\m_TransferMode006.exe	2.4	2010/04/16 11:58:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:40:39	7107	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode007\Hardware\m_TransferMode007.exe	2.2	2009/10/12 15:46:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:39:25	11960	TELM-06	1	Pass	0	0	0	112
M_TCP_Data	m_TransferMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode008\Hardware\m_TransferMode008.exe	2.0	2009/09/09 15:28:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 17:35:33	10651	TELM-06	1	Pass	0	0	0	76
M_TCP_Data	m_TransferMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode009\Hardware\m_TransferMode009.exe	2.1	2009/10/13 16:25:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:33:33	11483	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_TransferMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode010\Hardware\m_TransferMode010.exe	2.1	2009/10/13 16:26:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:31:29	11346	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode011\Hardware\m_TransferMode011.exe	2.3	2010/02/10 13:37:08	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:29:42	8111	TELM-06	1	Pass	0	0	0	72
M_TCP_Data	m_TransferMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode012\Hardware\m_TransferMode012.exe	2.3	2010/02/10 13:38:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:27:34	8183	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_TransferMode013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode013\Hardware\m_TransferMode013.exe	2.3	2010/02/10 13:39:16	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:25:46	8121	TELM-06	1	Pass	0	0	0	44
M_TCP_Data	m_TransferMode014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode014\Hardware\m_TransferMode014.exe	2.3	2010/02/10 13:40:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:20:42	8247	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_TransferMode015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode015\Hardware\m_TransferMode015.exe	2.5	2010/05/26 14:56:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 17:16:54	7376	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_TransferMode016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode016\Hardware\m_TransferMode016.exe	2.8	2010/07/02 13:14:11	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 17:12:44	29965	TELM-06	1	Pass	0	0	0	47
M_TCP_Data	m_TransferMode017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode017\Hardware\m_TransferMode017.exe	2.7	2010/05/18 14:47:09	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 17:06:28	39950	TELM-06	1	Pass	0	0	0	257
M_TCP_Data	m_TransferMode018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode018\Hardware\m_TransferMode018.exe	2.6	2010/04/16 14:06:26	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:02:45	26517	TELM-06	1	Pass	0	0	0	509
M_TCP_Data	m_TransferMode019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode019\Hardware\m_TransferMode019.exe	2.6	2010/03/19 18:58:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:01:18	22681	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics001\Hardware\m_Diagnostics001.exe	2.5	2009/12/30 17:45:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:07:37	9603	TELM-06	1	Pass	0	0	0	166
M_TCP_Diagnostic	m_Diagnostics002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics002\Hardware\m_Diagnostics002.exe	2.2	2009/11/23 20:15:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:06:17	7860	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics003\Hardware\m_Diagnostics003.exe	2.2	2010/03/14 20:01:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:04:21	6654	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics004\Hardware\m_Diagnostics004.exe	2.3	2009/12/03 13:22:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:00:15	16001	TELM-06	1	Pass	0	0	0	112
M_TCP_Diagnostic	m_Diagnostics005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics005\Hardware\m_Diagnostics005.exe	2.1	2009/10/12 14:14:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:54:02	7903	TELM-06	1	Pass	0	0	0	13

M_TCP_Diagnostic	m_Diagnostics006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics006\Hardware\m_Diagnostics006.exe	2.0	2009/09/22 17:02:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:52:52	14799	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics007\Hardware\m_Diagnostics007.exe	2.1	2009/09/28 13:31:55	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:48:31	15117	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics008\Hardware\m_Diagnostics008.exe	2.0	2009/09/22 17:03:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:42:42	21391	TELM-06	1	Pass	0	0	0	31
M_TCP_Diagnostic	m_Diagnostics009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics009\Hardware\m_Diagnostics009.exe	2.5	2010/06/25 17:09:14	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:40:43	15765	TELM-06	1	Pass	0	0	0	23
M_TCP_Diagnostic	m_Diagnostics010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics010\Hardware\m_Diagnostics010.exe	2.5	2010/07/20 19:18:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:38:51	10510	TELM-06	1	Pass	0	0	0	10
M_TCP_Diagnostic	m_Diagnostics011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics011\Hardware\m_Diagnostics011.exe	2.4	2009/12/23 13:29:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:35:06	29903	TELM-06	1	Pass	0	0	0	19
M_TCP_Diagnostic	m_Diagnostics012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics012\Hardware\m_Diagnostics012.exe	2.3	2010/04/16 14:20:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:33:51	17260	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics013\Hardware\m_Diagnostics013.exe	2.5	2010/06/08 17:29:50	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:29:31	20540	TELM-06	1	Pass	0	0	0	15
M_TCP_Diagnostic	m_Diagnostics014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics014\Hardware\m_Diagnostics014.exe	2.4	2010/03/14 17:27:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:28:19	17855	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics015\Hardware\m_Diagnostics015.exe	2.4	2010/01/19 17:35:47	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:27:31	25625	TELM-06	1	Pass	0	0	0	29
M_TCP_Diagnostic	m_Diagnostics016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics016\Hardware\m_Diagnostics016.exe	2.0	2009/09/22 17:09:46	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:25:41	5820	TELM-06	1	Pass	0	0	0	1
M_TCP_Diagnostic	m_Diagnostics017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics017\Hardware\m_Diagnostics017.exe	2.3	2009/11/23 20:15:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:23:35	25074	TELM-06	1	Pass	0	0	0	32
M_TCP_Diagnostic	m_Diagnostics018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics018\Hardware\m_Diagnostics018.exe	2.2	2009/12/23 13:30:36	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:22:01	11920	TELM-06	1	Pass	0	0	0	9
M_TCP_Diagnostic	m_Diagnostics019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics019\Hardware\m_Diagnostics019.exe	2.1	2009/10/13 12:56:53	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:20:47	7683	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics020\Hardware\m_Diagnostics020.exe	2.2	2009/11/23 20:16:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:19:36	10291	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics021\Hardware\m_Diagnostics021.exe	2.1	2009/10/13 13:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:17:16	6223	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics022\Hardware\m_Diagnostics022.exe	2.4	2010/02/03 17:09:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:15:29	12645	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics023\Hardware\m_Diagnostics023.exe	2.2	2010/02/03 14:24:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:13:35	39171	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics024\Hardware\m_Diagnostics024.exe	2.5	2010/02/03 14:04:47	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:09:46	15625	TELM-06	1	Pass	0	0	0	20
M_TCP_Diagnostic	m_Diagnostics026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics026\Hardware\m_Diagnostics026.exe	2.1	2009/10/15 19:32:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:08:25	6091	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics027\Hardware\m_Diagnostics027.exe	2.2	2009/11/02 13:03:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:06:31	2926	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics028\Hardware\m_Diagnostics028.exe	2.4	2009/11/16 20:37:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:02:47	11591	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics029\Hardware\m_Diagnostics029.exe	2.1	2009/10/15 19:46:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:59:34	2734	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics030\Hardware\m_Diagnostics030.exe	2.5	2010/07/20 18:09:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:58:03	7445	TELM-06	1	Pass	0	0	0	277
M_TCP_Diagnostic	m_Diagnostics031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics031\Hardware\m_Diagnostics031.exe	2.2	2009/11/11 14:33:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:56:21	10890	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics032\Hardware\m_Diagnostics032.exe	2.3	2009/12/23 13:31:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:55:10	9822	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics034\Hardware\m_Diagnostics034.exe	2.4	2009/11/16 20:30:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:53:52	9056	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics035\Hardware\m_Diagnostics035.exe	2.2	2009/12/23 13:31:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:51:59	7998	TELM-06	1	Pass	0	0	0	7
M_TCP_Diagnostic	m_Diagnostics036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics036\Hardware\m_Diagnostics036.exe	2.3	2010/03/13 01:44:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:50:29	8508	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics037\Hardware\m_Diagnostics037.exe	2.4	2010/03/13 01:23:35	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:49:05	6885	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics038\Hardware\m_Diagnostics038.exe	2.3	2010/06/07 20:16:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:46:55	48896	TELM-06	1	Pass	0	0	0	18
M_TCP_Discover	m_DiscoverFull001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull001\Hardware\m_DiscoverFull001.exe	2.8	2010/05/21 19:22:39	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 11:52:29	44483	TELM-06	1	Pass	0	0	0	150
M_TCP_Discover	m_DiscoverFull002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull002\Hardware\m_DiscoverFull002.exe	2.6	2010/03/19 18:02:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 11:48:18	144672	TELM-06	1	Pass	0	0	0	1815
M_TCP_Discover	m_DiscoverFull003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull003\Hardware\m_DiscoverFull003.exe	2.2	2010/02/25 20:00:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:46:29	12109	TELM-06	1	Pass	0	0	0	58
M_TCP_Discover	m_DiscoverFull004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull004\Hardware\m_DiscoverFull004.exe	2.4	2010/02/05 21:19:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 13:15:41	270966	TELM-06	1	Pass	0	0	0	3713
M_TCP_Discover	m_DiscoverFull005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull005\Hardware\m_DiscoverFull005.exe	2.2	2009/10/29 19:18:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:44:08	17341	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull006\Hardware\m_DiscoverFull006.exe	2.3	2010/02/05 21:19:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:04:15	130996	TELM-06	1	Pass	0	0	0	1727
M_TCP_Discover	m_DiscoverFull007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull007\Hardware\m_DiscoverFull007.exe	2.7	2009/12/14 21:43:07	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 11:42:37	13715	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull009\Hardware\m_DiscoverFull009.exe	2.3	2010/01/21 21:02:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:40:56	11054	TELM-06	1	Pass	0	0	0	65
M_TCP_Discover	m_DiscoverFull010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull010\Hardware\m_DiscoverFull010.exe	2.4	2010/05/12 17:07:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:38:18	7973	TELM-06	1	Pass	0	0	0	121
M_TCP_Discover	m_DiscoverFull011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull011\Hardware\m_DiscoverFull011.exe	2.3	2009/11/02 22:03:01	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:34:59	56802	TELM-06	1	Pass	0	0	0	9
M_TCP_Discover	m_DiscoverFull012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull012\Hardware\m_DiscoverFull012.exe	2.8	2010/06/29 12:07:18	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 14:47:44	366828	TELM-06	1	Pass	0	0	0	33
M_TCP_Discover	m_DiscoverFull013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull013\Hardware\m_DiscoverFull013.exe	2.2	2010/03/18 16:03:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:32:26	42577	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull014\Hardware\m_DiscoverFull014.exe	2.2	2010/03/18 16:03:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:30:22	24541	TELM-06	1	Pass	0	0	0	15
M_TCP_Discover	m_DiscoverFull015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull015\Hardware\m_DiscoverFull015.exe	2.1	2009/11/23 16:09:49	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:29:04	7877	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull016\Hardware\m_DiscoverFull016.exe	2.1	2009/11/03 20:58:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:27:20	8245	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull017\Hardware\m_DiscoverFull017.exe	2.1	2009/11/03 20:58:47	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:26:02	12119	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull022\Hardware\m_DiscoverFull022.exe	2.1	2009/11/23 16:11:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:24:49	8407	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull023\Hardware\m_DiscoverFull023.exe	2.1	2009/11/23 16:11:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:23:23	11414	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull024\Hardware\m_DiscoverFull024.exe	2.1	2009/11/23 16:12:04	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:21:37	12592	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull025\Hardware\m_DiscoverFull025.exe	2.1	2009/11/23 16:12:32	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:20:05	9090	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull029\Hardware\m_DiscoverFull029.exe	2.0	2009/11/13 21:27:59	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:18:38	12360	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull030\Hardware\m_DiscoverFull030.exe	2.3	2010/05/28 11:39:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:17:28	19191	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull031\Hardware\m_DiscoverFull031.exe	2.3	2010/05/28 12:38:20	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:15:14	35314	TELM-06	1	Pass	0	0	0	22
M_TCP_Discover	m_DiscoverFull032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull032\Hardware\m_DiscoverFull032.exe	2.0	2009/11/13 21:29:43	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:12:36	9202	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull033\Hardware\m_DiscoverFull033.exe	2.0	2009/11/13 21:30:18	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:10:13	9438	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull034\Hardware\m_DiscoverFull034.exe	2.2	2010/03/18 16:04:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:06:22	11086	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull035\Hardware\m_DiscoverFull035.exe	2.2	2010/02/09 20:53:53	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 00:44:55	43666	TELM-06	1	Pass	0	0	0	31
M_TCP_Discover	m_DiscoverFull036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull036\Hardware\m_DiscoverFull036.exe	2.2	2010/02/11 14:31:48	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 00:32:00	22022	TELM-06	1	Pass	0	0	0	38
M_TCP_Discover	m_DiscoverFull037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull037\Hardware\m_DiscoverFull037.exe	2.4	2010/02/03 21:18:36	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 02:14:08	35116	TELM-06	1	Pass	0	0	0	4
M_TCP_Discover	m_DiscoverFull038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull038\Hardware\m_DiscoverFull038.exe	2.1	2009/12/01 20:49:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:59:59	141757	TELM-06	1	Pass	0	0	0	7
M_TCP_EEPROM	m_Eeprom001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom001\Hardware\m_Eeprom001.exe	2.3	2009/12/23 16:23:59	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 23:57:12	11077	TELM-06	1	Pass	0	0	0	22
M_TCP_EEPROM	m_Eeprom002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom002\Hardware\m_Eeprom002.exe	2.3	2009/12/18 21:07:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 19:23:25	10787	TELM-06	1	Pass	0	0	0	938
M_TCP_EEPROM	m_Eeprom003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom003\Hardware\m_Eeprom003.exe	2.2	2010/01/05 18:03:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:46:07	16769	TELM-06	1	Pass	0	0	0	702
M_TCP_EEPROM	m_Eeprom004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom004\Hardware\m_Eeprom004.exe	2.2	2010/01/25 19:26:31	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:43:04	110561	TELM-06	1	Pass	0	0	0	583
M_TCP_EEPROM	m_Eeprom005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom005\Hardware\m_Eeprom005.exe	2.2	2010/01/19 15:32:05	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:38:56	173076	TELM-06	1	Pass	0	0	0	704
M_TCP_EEPROM	m_Eeprom006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom006\Hardware\m_Eeprom006.exe	2.2	2010/05/28 12:55:48	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:36:33	47136	TELM-06	1	Pass	0	0	0	6
M_TCP_EEPROM	m_Eeprom007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom007\Hardware\m_Eeprom007.exe	2.1	2010/02/18 18:38:33	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:35:30	5083	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency001\Hardware\m_Emergency001.exe	2.3	2009/11/03 21:54:33	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:45:58	9519	TELM-06	1	Pass	0	0	0	100
M_TCP_Emergency	m_Emergency002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency002\Hardware\m_Emergency002.exe	2.5	2010/06/25 20:19:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:44:07	13462	TELM-06	1	Pass	0	0	0	45
M_TCP_Emergency	m_Emergency003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency003\Hardware\m_Emergency003.exe	2.6	2010/06/29 14:19:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:42:17	28086	TELM-06	1	Pass	0	0	0	1620
M_TCP_Emergency	m_Emergency004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency004\Hardware\m_Emergency004.exe	2.3	2010/02/05 21:22:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:40:51	13256	TELM-06	1	Pass	0	0	0	4
M_TCP_Emergency	m_Emergency005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency005\Hardware\m_Emergency005.exe	2.2	2009/11/10 00:31:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:39:12	8745	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency006\Hardware\m_Emergency006.exe	2.1	2009/10/16 16:44:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:37:16	9371	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency007\Hardware\m_Emergency007.exe	2.1	2009/10/16 16:44:28	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:34:04	9556	TELM-06	1	Pass	0	0	0	33
M_TCP_Emergency	m_Emergency009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency009\Hardware\m_Emergency009.exe	2.1	2009/10/08 19:54:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:31:44	6121	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency010\Hardware\m_Emergency010.exe	2.2	2010/06/08 15:56:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:28:28	13083	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency011\Hardware\m_Emergency011.exe	2.0	2009/10/21 16:25:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:26:36	6735	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency012\Hardware\m_Emergency012.exe	2.1	2009/11/11 13:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:25:17	8468	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency013\Hardware\m_Emergency013.exe	2.1	2009/11/11 14:02:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:23:15	12122	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency014\Hardware\m_Emergency014.exe	2.3	2009/12/23 13:33:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:21:11	8303	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency015\Hardware\m_Emergency015.exe	2.3	2010/06/08 15:56:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:18:29	8989	TELM-06	1	Pass	0	0	0	8
M_TCP_Emergency	m_Emergency016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency016\Hardware\m_Emergency016.exe	2.2	2009/12/21 21:03:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:16:52	12907	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency017\Hardware\m_Emergency017.exe	2.2	2009/11/16 12:31:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:14:16	9845	TELM-06	1	Pass	0	0	0	21
M_TCP_Emergency	m_Emergency018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency018\Hardware\m_Emergency018.exe	2.3	2010/06/29 15:32:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:12:06	8018	TELM-06	1	Pass	0	0	0	22
M_TCP_Emergency	m_Emergency019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency019\Hardware\m_Emergency019.exe	2.2	2010/01/07 19:28:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:10:10	10967	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency020\Hardware\m_Emergency020.exe	2.6	2010/03/18 17:47:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:08:12	11581	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency021\Hardware\m_Emergency021.exe	2.3	2009/12/21 19:27:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:05:06	10692	TELM-06	1	Pass	0	0	0	83
M_TCP_Emergency	m_Emergency022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency022\Hardware\m_Emergency022.exe	2.2	2010/01/07 18:54:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:02:42	10431	TELM-06	1	Pass	0	0	0	82
M_TCP_Emergency	m_Emergency038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency038\Hardware\m_Emergency038.exe	2.2	2010/01/07 19:42:11	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:57:17	10446	TELM-06	1	Pass	0	0	0	14
M_TCP_Emergency	m_Emergency039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency039\Hardware\m_Emergency039.exe	2.2	2010/02/03 17:51:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:53:10	15403	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency040\Hardware\m_Emergency040.exe	2.0	2009/09/29 14:53:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:50:20	14071	TELM-06	1	Pass	0	0	0	7
M_TCP_Emergency	m_Emergency041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency041\Hardware\m_Emergency041.exe	2.0	2009/09/29 14:54:05	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:48:20	5088	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency042\Hardware\m_Emergency042.exe	2.0	2009/09/29 14:54:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:46:16	5283	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency043\Hardware\m_Emergency043.exe	2.4	2010/04/21 17:13:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:43:26	10256	TELM-06	1	Pass	0	0	0	9
M_TCP_Emergency	m_Emergency044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency044\Hardware\m_Emergency044.exe	2.5	2009/11/10 14:19:43	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 02:50:27	37902	TELM-06	1	Pass	0	0	0	18
M_TCP_Emergency	m_Emergency045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency045\Hardware\m_Emergency045.exe	2.3	2010/05/11 12:18:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:40:46	11235	TELM-06	1	Pass	0	0	0	16
M_TCP_General	m_General003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General003\Hardware\m_General003.exe	2.2	2009/10/13 14:50:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:34:20	67756	TELM-06	1	Pass	0	0	0	44576
M_TCP_General	m_General005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General005\Hardware\m_General005.exe	2.4	2009/12/01 19:02:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:31:48	67379	TELM-06	1	Pass	0	0	0	15471
M_TCP_General	m_General015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General015\Hardware\m_General015.exe	2.3	2009/12/01 19:02:59	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:28:39	35628	TELM-06	1	Pass	0	0	0	83475
M_TCP_General	m_General016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General016\Hardware\m_General016.exe	2.4	2010/03/29 10:43:22	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:26:14	8532	TELM-06	1	Pass	0	0	0	2
M_TCP_General	m_General017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General017\Hardware\m_General017.exe	2.3	2009/12/01 19:03:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:23:58	7464	TELM-06	1	Pass	0	0	0	10
M_TCP_General	m_General018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General018\Hardware\m_General018.exe	2.2	2010/02/03 18:07:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:21:30	29784	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General027\Hardware\m_General027.exe	2.7	2010/06/07 11:39:47	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 22:44:29	802166	TELM-06	1	Pass	0	0	0	21
M_TCP_General	m_General028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General028\Hardware\m_General028.exe	2.5	2010/06/08 11:48:52	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 03:15:35	635673	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General029\Hardware\m_General029.exe	2.3	2010/06/07 12:27:26	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 03:09:16	274290	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General030\Hardware\m_General030.exe	2.1	2010/06/07 12:30:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:19:17	17122	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General031\Hardware\m_General031.exe	2.5	2010/05/11 12:33:53	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 16:16:45	57957	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General032\Hardware\m_General032.exe	2.2	2010/02/05 18:56:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:14:15	29672	TELM-06	1	Pass	0	0	0	11
M_TCP_General	m_General033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General033\Hardware\m_General033.exe	2.0	2010/02/04 17:42:57	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:11:51	20585	TELM-06	1	Pass	0	0	0	4
M_TCP_General	m_General034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General034\Hardware\m_General034.exe	2.0	2010/01/22 22:45:34	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:09:29	30123	TELM-06	1	Pass	0	0	0	2

M_TCP_Initialize	m_Initialize001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize001\Hardware\m_Initialize001.exe	2.8	2010/05/12 12:34:35	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 10:17:15	10382	TELM-06	1	Pass	0	0	0	9
M_TCP_Initialize	m_Initialize002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize002\Hardware\m_Initialize002.exe	2.2	2009/10/20 12:16:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:13:15	19918	TELM-06	1	Pass	0	0	0	11
M_TCP_Initialize	m_Initialize003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize003\Hardware\m_Initialize003.exe	2.4	2010/06/08 15:57:16	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:10:23	24126	TELM-06	1	Pass	0	0	0	6
M_TCP_Initialize	m_Initialize004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize004\Hardware\m_Initialize004.exe	2.4	2010/06/08 15:57:44	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:07:37	40192	TELM-06	1	Pass	0	0	0	30
M_TCP_MedicalEvent	m_MedicalEvent001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent001\Hardware\m_MedicalEvent001.exe	2.4	2010/01/18 15:42:53	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:57:31	12589	TELM-06	1	Pass	0	0	0	36
M_TCP_MedicalEvent	m_MedicalEvent002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent002\Hardware\m_MedicalEvent002.exe	2.6	2010/02/09 19:03:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 10:53:21	73371	TELM-06	1	Pass	0	0	0	23
M_TCP_MedicalEvent	m_MedicalEvent003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent003\Hardware\m_MedicalEvent003.exe	2.4	2010/03/11 21:16:02	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:50:46	18753	TELM-06	1	Pass	0	0	0	24
M_TCP_MedicalEvent	m_MedicalEvent004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent004\Hardware\m_MedicalEvent004.exe	2.4	2010/01/18 15:44:34	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:44:49	12033	TELM-06	1	Pass	0	0	0	12
M_TCP_MedicalEvent	m_MedicalEvent008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent008\Hardware\m_MedicalEvent008.exe	2.4	2010/01/18 15:44:59	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:39:54	8299	TELM-06	1	Pass	0	0	0	6
M_TCP_MedicalEvent	m_MedicalEvent009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent009\Hardware\m_MedicalEvent009.exe	2.2	2010/01/18 15:46:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:37:14	7707	TELM-06	1	Pass	0	0	0	4
M_TCP_MedicalEvent	m_MedicalEvent010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent010\Hardware\m_MedicalEvent010.exe	2.5	2010/01/27 14:30:44	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:34:42	16716	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent011\Hardware\m_MedicalEvent011.exe	2.2	2010/01/18 15:47:08	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:31:52	15196	TELM-06	1	Pass	0	0	0	11
M_TCP_MedicalEvent	m_MedicalEvent012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent012\Hardware\m_MedicalEvent012.exe	2.3	2010/01/18 15:47:55	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:30:59	6166	TELM-06	1	Pass	0	0	0	3
M_TCP_MedicalEvent	m_MedicalEvent014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent014\Hardware\m_MedicalEvent014.exe	2.5	2010/09/08 13:20:31	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:28:25	15263	TELM-06	1	Pass	0	0	0	18
M_TCP_MedicalEvent	m_MedicalEvent015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent015\Hardware\m_MedicalEvent015.exe	2.9	2010/09/07 20:16:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 10:00:37	56476	TELM-06	1	Pass	0	0	0	152
M_TCP_MedicalEvent	m_MedicalEvent016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent016\Hardware\m_MedicalEvent016.exe	2.9	2010/05/12 18:44:37	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 14:20:15	283898	TELM-06	1	Pass	0	0	0	375
M_TCP_MedicalEvent	m_MedicalEvent017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent017\Hardware\m_MedicalEvent017.exe	2.8	2010/09/07 20:17:30	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 20:48:33	612642	TELM-06	1	Pass	0	0	0	13
M_TCP_MedicalEvent	m_MedicalEvent018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent018\Hardware\m_MedicalEvent018.exe	2.5	2010/01/18 15:50:21	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 13:47:03	272164	TELM-06	1	Pass	0	0	0	4627
M_TCP_MedicalEvent	m_MedicalEvent019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent019\Hardware\m_MedicalEvent019.exe	2.5	2010/01/18 15:50:59	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:26:51	7476	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent020\Hardware\m_MedicalEvent020.exe	2.3	2010/01/18 15:51:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:21:32	40717	TELM-06	1	Pass	0	0	0	9
M_TCP_MemoryTest	m_MemoryTest001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest001\Hardware\m_MemoryTest001.exe	2.7	2010/06/18 12:28:23	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 01:04:51	32980	TELM-06	1	Pass	0	0	0	33
M_TCP_MemoryTest	m_MemoryTest002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest002\Hardware\m_MemoryTest002.exe	2.4	2010/05/05 19:48:21	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:44:06	240725	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest003\Hardware\m_MemoryTest003.exe	2.3	2010/05/05 20:08:58	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 15:58:42	479602	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest004\Hardware\m_MemoryTest004.exe	2.3	2010/05/05 20:09:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:38:13	240134	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest005\Hardware\m_MemoryTest005.exe	2.3	2010/05/05 20:10:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 15:37:08	478655	TELM-06	1	Pass	0	0	0	75
M_TCP_Network	m_Network001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network001\Hardware\m_Network001.exe	2.6	2010/05/21 15:45:43	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 07:47:18	24633	TELM-06	1	Pass	0	0	0	1461
M_TCP_Network	m_Network002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network002\Hardware\m_Network002.exe	2.6	2010/05/12 12:01:17	2.6	Up-to-date	Baseline 17_1	None	09/11/2010 00:19:39	34802	TELM-06	1	Pass	0	0	0	82
M_TCP_Network	m_Network003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network003\Hardware\m_Network003.exe	2.1	2009/12/16 22:27:06	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 07:35:36	64790	TELM-06	1	Pass	0	0	0	86
M_TCP_Network	m_Network004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network004\Hardware\m_Network004.exe	2.1	2010/04/21 17:14:24	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 07:28:20	2901	TELM-06	1	Pass	0	0	0	6
M_TCP_Network	m_Network005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network005\Hardware\m_Network005.exe	2.3	2010/05/12 12:02:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 07:21:55	58315	TELM-06	1	Pass	0	0	0	50
M_TCP_Network	m_Network006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network006\Hardware\m_Network006.exe	2.5	2010/05/21 16:49:56	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 07:15:03	15414	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network007\Hardware\m_Network007.exe	2.5	2010/05/21 17:09:04	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 07:08:34	15633	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network008\Hardware\m_Network008.exe	2.5	2010/05/21 17:13:03	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 07:01:43	10466	TELM-06	1	Pass	0	0	0	1380
M_TCP_Network	m_Network009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network009\Hardware\m_Network009.exe	2.5	2010/05/21 17:21:30	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:54:16	10414	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network010\Hardware\m_Network010.exe	2.5	2010/05/21 17:34:27	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:46:52	10388	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network011\Hardware\m_Network011.exe	2.3	2010/05/12 12:03:38	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:41:58	209534	TELM-06	1	Pass	0	0	0	122
M_TCP_Network	m_Network013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network013\Hardware\m_Network013.exe	2.5	2010/07/01 13:00:48	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:38:26	141629	TELM-06	1	Pass	0	0	0	392
M_TCP_Network	m_Network014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network014\Hardware\m_Network014.exe	2.3	2010/05/12 12:06:29	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:34:32	26840	TELM-06	1	Pass	0	0	0	12
M_TCP_Network	m_Network015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network015\Hardware\m_Network015.exe	2.3	2010/05/24 13:53:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:31:59	35718	TELM-06	1	Pass	0	0	0	25
M_TCP_Network	m_Network016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network016\Hardware\m_Network016.exe	2.1	2010/05/12 12:09:43	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:28:20	55464	TELM-06	1	Pass	0	0	0	262
M_TCP_Network	m_Network017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network017\Hardware\m_Network017.exe	2.4	2010/06/07 13:19:37	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 06:25:56	12606	TELM-06	1	Pass	0	0	0	41
M_TCP_Network	m_Network018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network018\Hardware\m_Network018.exe	2.5	2010/06/07 13:14:52	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:18:18	21781	TELM-06	1	Pass	0	0	0	11
M_TCP_Network	m_Network019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network019\Hardware\m_Network019.exe	2.2	2010/05/12 12:12:40	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:14:11	56452	TELM-06	1	Pass	0	0	0	51
M_TCP_Network	m_Network020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network020\Hardware\m_Network020.exe	2.2	2010/05/12 12:14:53	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:11:17	60277	TELM-06	1	Pass	0	0	0	67
M_TCP_Network	m_Network021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network021\Hardware\m_Network021.exe	2.2	2010/05/12 12:16:17	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:08:10	60352	TELM-06	1	Pass	0	0	0	133
M_TCP_Network	m_Network022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network022\Hardware\m_Network022.exe	2.2	2010/05/12 12:17:24	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:05:08	60295	TELM-06	1	Pass	0	0	0	175
M_TCP_Network	m_Network023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network023\Hardware\m_Network023.exe	2.1	2010/05/12 12:18:23	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 06:02:31	19519	TELM-06	1	Pass	0	0	0	454
M_TCP_Network	m_Network024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network024\Hardware\m_Network024.exe	2.2	2010/05/24 14:41:01	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:59:41	47557	TELM-06	1	Pass	0	0	0	46
M_TCP_Network	m_Network025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network025\Hardware\m_Network025.exe	2.1	2010/05/12 12:20:01	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:58:01	19649	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network026\Hardware\m_Network026.exe	2.3	2010/05/12 12:20:56	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:54:38	20676	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network027\Hardware\m_Network027.exe	2.1	2010/05/12 12:22:02	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:50:23	21110	TELM-06	1	Pass	0	0	0	7
M_TCP_Network	m_Network028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network028\Hardware\m_Network028.exe	2.1	2010/06/07 13:22:27	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:48:27	14445	TELM-06	1	Pass	0	0	0	22
M_TCP_Network	m_Network029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network029\Hardware\m_Network029.exe	2.4	2010/02/09 21:21:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 19:46:57	271612	TELM-06	1	Pass	0	0	0	48
M_TCP_Open	m_Open001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open001\Hardware\m_Open001.exe	2.11	2010/05/20 13:50:26	2.11	Up-to-date	Baseline 17_1	None	09/09/2010 23:22:48	31719	TELM-06	1	Pass	0	0	0	335
M_TCP_Open	m_Open002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open002\Hardware\m_Open002.exe	2.3	2010/01/18 15:52:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:18:25	18693	TELM-06	1	Pass	0	0	0	31
M_TCP_Open	m_Open003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open003\Hardware\m_Open003.exe	2.0	2009/10/15 19:09:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:16:39	6417	TELM-06	1	Pass	0	0	0	3
M_TCP_Open	m_Open004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open004\Hardware\m_Open004.exe	2.6	2010/06/22 20:01:21	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:15:19	13002	TELM-06	1	Pass	0	0	0	21
M_TCP_Open	m_Open005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open005\Hardware\m_Open005.exe	2.3	2010/06/22 17:40:45	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:14:01	16756	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open006\Hardware\m_Open006.exe	2.2	2010/03/15 14:26:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:11:53	21177	TELM-06	1	Pass	0	0	0	1173
M_TCP_Open	m_Open007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open007\Hardware\m_Open007.exe	2.7	2010/03/15 14:46:20	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 23:09:41	17509	TELM-06	1	Pass	0	0	0	1181
M_TCP_Open	m_Open008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open008\Hardware\m_Open008.exe	2.1	2009/10/30 14:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:07:15	7868	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open009\Hardware\m_Open009.exe	2.7	2010/06/16 00:36:52	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 23:03:46	12459	TELM-06	1	Pass	0	0	0	1268
M_TCP_Open	m_Open010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open010\Hardware\m_Open010.exe	2.6	2010/03/15 15:05:50	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:00:31	17326	TELM-06	1	Pass	0	0	0	3965
M_TCP_Open	m_Open011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open011\Hardware\m_Open011.exe	2.2	2010/02/10 13:31:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:57:31	6420	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open012\Hardware\m_Open012.exe	2.4	2010/02/26 17:01:14	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:55:07	15445	TELM-06	1	Pass	0	0	0	10
M_TCP_Open	m_Open013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open013\Hardware\m_Open013.exe	2.5	2010/02/10 13:34:06	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:53:43	9423	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open014\Hardware\m_Open014.exe	2.6	2010/02/10 13:35:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:52:37	8458	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open015\Hardware\m_Open015.exe	2.0	2010/01/28 19:00:26	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:51:36	8926	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open030\Hardware\m_Open030.exe	2.1	2010/05/10 13:57:56	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 04:12:59	58132	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open031\Hardware\m_Open031.exe	2.7	2009/11/19 21:48:20	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 03:36:23	63956	TELM-06	1	Pass	0	0	0	13
M_TCP_Open	m_Open032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open032\Hardware\m_Open032.exe	2.3	2010/02/03 22:17:27	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:36:25	32280	TELM-06	1	Pass	0	0	0	18
M_TCP_Open	m_Open033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open033\Hardware\m_Open033.exe	2.3	2010/02/03 22:32:54	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:17:34	10044	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open034\Hardware\m_Open034.exe	2.2	2010/02/17 14:19:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:49:42	56855	TELM-06	1	Pass	0	0	0	42
M_TCP_Open	m_Open035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open035\Hardware\m_Open035.exe	2.2	2010/02/17 14:10:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:48:20	9921	TELM-06	1	Pass	0	0	0	32
M_TCP_Open	m_Open036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open036\Hardware\m_Open036.exe	2.1	2009/10/21 15:52:46	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 03:34:11	36392	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open037\Hardware\m_Open037.exe	2.2	2009/10/21 15:51:07	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 03:30:44	31216	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open038\Hardware\m_Open038.exe	2.7	2010/02/17 14:00:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 22:46:04	78284	TELM-06	1	Pass	0	0	0	46
M_TCP_Open	m_Open039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open039\Hardware\m_Open039.exe	2.0	2009/11/09 14:20:44	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:44:44	9177	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open040\Hardware\m_Open040.exe	2.1	2010/02/01 19:45:37	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:43:28	17230	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open041\Hardware\m_Open041.exe	2.0	2009/11/09 14:21:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 16:28:32	513908	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open042\Hardware\m_Open042.exe	2.4	2010/07/02 12:38:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:42:39	16301	TELM-06	1	Pass	0	0	0	52
M_TCP_Polling	m_Polling016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling016\Hardware\m_Polling016.exe	2.0	2009/09/23 19:25:05	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:37:56	49189	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling017\Hardware\m_Polling017.exe	2.1	2010/01/26 20:04:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:56:08	6202	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling018\Hardware\m_Polling018.exe	2.1	2010/01/26 20:05:23	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:34:01	81486	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling024\Hardware\m_Polling024.exe	2.5	2010/01/26 20:05:45	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:30:12	78657	TELM-06	1	Pass	0	0	0	10
M_TCP_Polling	m_Polling025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling025\Hardware\m_Polling025.exe	2.1	2009/11/24 20:51:39	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:26:00	46141	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling026\Hardware\m_Polling026.exe	2.2	2009/12/03 14:59:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:23:16	19487	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling040\Hardware\m_Polling040.exe	2.1	2010/01/26 20:06:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:20:58	6215	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling045\Hardware\m_Polling045.exe	2.1	2010/01/26 20:06:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:17:04	36156	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling046\Hardware\m_Polling046.exe	2.6	2010/04/16 14:21:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 13:13:41	47156	TELM-06	1	Pass	0	0	0	38
M_TCP_Polling	m_Polling047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling047\Hardware\m_Polling047.exe	2.1	2010/01/26 20:07:28	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:11:06	36017	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling048\Hardware\m_Polling048.exe	2.3	2009/11/24 20:53:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:07:46	10051	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling049\Hardware\m_Polling049.exe	2.4	2010/03/10 19:44:21	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:02:33	46133	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling050\Hardware\m_Polling050.exe	2.6	2010/06/29 17:44:57	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:58:48	22906	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling065\Hardware\m_Polling065.exe	2.6	2010/06/09 18:52:38	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:55:36	42120	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling066\Hardware\m_Polling066.exe	2.3	2009/11/09 22:57:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:53:32	52755	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling067\Hardware\m_Polling067.exe	2.1	2009/10/22 14:28:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:51:05	84824	TELM-06	1	Pass	0	0	0	11
M_TCP_Polling	m_Polling068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling068\Hardware\m_Polling068.exe	2.3	2009/11/05 18:23:27	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:49:23	7089	TELM-06	1	Pass	0	0	0	8
M_TCP_Polling	m_Polling076	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling076\Hardware\m_Polling076.exe	2.1	2009/11/03 15:55:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:46:17	10132	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling077	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling077\Hardware\m_Polling077.exe	2.1	2009/11/05 18:34:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 22:17:06	527442	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling080	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling080\Hardware\m_Polling080.exe	2.2	2009/11/05 18:24:04	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 21:52:33	277436	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling081	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling081\Hardware\m_Polling081.exe	2.3	2010/05/27 14:19:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:43:21	24158	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling082	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling082\Hardware\m_Polling082.exe	2.2	2009/11/02 18:25:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:40:59	9445	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling083	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling083\Hardware\m_Polling083.exe	2.3	2009/11/02 18:45:07	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:39:07	19425	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling085	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling085\Hardware\m_Polling085.exe	2.1	2009/10/22 13:55:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 21:32:45	529495	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling088	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling088\Hardware\m_Polling088.exe	2.5	2010/05/12 17:59:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:35:31	23469	TELM-06	1	Pass	0	0	0	27
M_TCP_Polling	m_Polling089	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling089\Hardware\m_Polling089.exe	2.7	2010/05/11 19:28:17	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 12:31:16	97139	TELM-06	1	Pass	0	0	0	26
M_TCP_Polling	m_Polling090	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling090\Hardware\m_Polling090.exe	2.6	2010/02/15 20:02:24	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:28:36	83044	TELM-06	1	Pass	0	0	0	32
M_TCP_Polling	m_Polling091	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling091\Hardware\m_Polling091.exe	2.6	2010/02/15 20:16:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:27:13	13224	TELM-06	1	Pass	0	0	0	13
M_TCP_Polling	m_Polling092	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling092\Hardware\m_Polling092.exe	2.1	2009/11/05 18:41:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 21:18:07	277218	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling093	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling093\Hardware\m_Polling093.exe	2.0	2009/11/09 18:30:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:25:45	16911	TELM-06	1	Pass	0	0	0	1

M_TCP_Polling	m_Polling094	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling094\Hardware\m_Polling094.exe	2.0	2009/11/09 18:30:49	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 15:17:01	307319	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling095	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling095\Hardware\m_Polling095.exe	2.0	2009/11/09 18:31:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 12:24:24	8437	TELM-06	1	Pass	0	0	0	2
M_TCP_PowerInhibit	m_Power001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power001\Hardware\m_Power001.exe	2.2	2010/06/07 13:10:06	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:32:20	58694	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power002\Hardware\m_Power002.exe	2.2	2010/01/05 19:15:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:30:20	10758	TELM-06	1	Pass	0	0	0	394
M_TCP_PowerInhibit	m_Power003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power003\Hardware\m_Power003.exe	2.4	2010/06/22 14:04:45	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:27:18	54002	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power004\Hardware\m_Power004.exe	2.3	2010/03/22 14:26:21	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:23:57	17800	TELM-06	1	Pass	0	0	0	46
M_TCP_PowerInhibit	m_Power005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power005\Hardware\m_Power005.exe	2.6	2010/07/02 12:34:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 11:52:22	148979	TELM-06	1	Pass	0	0	0	310
M_TCP_PowerInhibit	m_Power006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power006\Hardware\m_Power006.exe	2.6	2010/06/22 12:27:46	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:18:32	193253	TELM-06	1	Pass	0	0	0	2744
M_TCP_PowerInhibit	m_Power007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power007\Hardware\m_Power007.exe	2.1	2009/12/14 16:39:57	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:13:25	19241	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power008\Hardware\m_Power008.exe	2.2	2009/12/14 16:44:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:07:15	34837	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power009\Hardware\m_Power009.exe	2.4	2010/07/01 15:00:10	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 05:03:51	12118	TELM-06	1	Pass	0	0	0	9
M_TCP_PowerInhibit	m_Power010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power010\Hardware\m_Power010.exe	2.1	2010/02/16 22:18:45	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:00:26	11873	TELM-06	1	Pass	0	0	0	26
M_TCP_PowerInhibit	m_Power011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power011\Hardware\m_Power011.exe	2.1	2010/03/11 19:04:17	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:55:58	93234	TELM-06	1	Pass	0	0	0	116
M_TCP_RawMode	m_RawMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode001\Hardware\m_RawMode001.exe	2.4	2010/01/26 20:08:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:20:55	7616	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode002\Hardware\m_RawMode002.exe	2.6	2010/01/27 13:00:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:19:12	9551	TELM-06	1	Pass	0	0	0	87
M_TCP_RawMode	m_RawMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode003\Hardware\m_RawMode003.exe	2.5	2010/01/27 13:03:49	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:16:39	11458	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode004\Hardware\m_RawMode004.exe	2.6	2010/01/27 13:09:25	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:15:14	7342	TELM-06	1	Pass	0	0	0	2
M_TCP_RawMode	m_RawMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode005\Hardware\m_RawMode005.exe	2.3	2010/01/26 20:10:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:12:52	13380	TELM-06	1	Pass	0	0	0	9
M_TCP_RawMode	m_RawMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode006\Hardware\m_RawMode006.exe	2.2	2010/01/26 20:11:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:53:38	9112	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode007\Hardware\m_RawMode007.exe	2.3	2010/06/09 21:03:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:09:49	11364	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode008\Hardware\m_RawMode008.exe	2.4	2010/01/27 13:14:20	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:07:32	16147	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode009\Hardware\m_RawMode009.exe	2.2	2009/12/14 16:51:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:05:31	15728	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode010\Hardware\m_RawMode010.exe	2.1	2009/10/27 17:32:20	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:03:46	3333	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode011\Hardware\m_RawMode011.exe	2.2	2010/02/01 16:15:38	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:01:48	8596	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode012\Hardware\m_RawMode012.exe	2.3	2010/01/26 20:13:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:56:39	11660	TELM-06	1	Pass	0	0	0	3
M_TCP_Reset	m_Reset001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset001\Hardware\m_Reset001.exe	2.3	2010/01/06 21:25:12	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:55:47	9203	TELM-06	1	Pass	0	0	0	23
M_TCP_Reset	m_Reset002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset002\Hardware\m_Reset002.exe	2.4	2010/01/28 21:52:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:52:36	20469	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security002\Hardware\m_Security002.exe	2.9	2010/06/03 15:18:25	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 18:39:48	457347	TELM-06	1	Pass	0	0	0	154
M_TCP_Security	m_Security003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security003\Hardware\m_Security003.exe	2.4	2010/02/17 16:20:41	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:07:17	6797	TELM-06	1	Pass	0	0	0	73
M_TCP_Security	m_Security004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security004\Hardware\m_Security004.exe	2.4	2010/02/15 15:01:00	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:00:36	253943	TELM-06	1	Pass	0	0	0	112
M_TCP_Security	m_Security005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security005\Hardware\m_Security005.exe	2.2	2010/02/17 16:23:51	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:59:10	4199	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security006\Hardware\m_Security006.exe	2.4	2010/02/17 14:44:38	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:55:49	145508	TELM-06	1	Pass	0	0	0	108
M_TCP_Security	m_Security007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security007\Hardware\m_Security007.exe	2.6	2010/02/18 15:13:38	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:52:00	113194	TELM-06	1	Pass	0	0	0	126
M_TCP_Security	m_Security008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security008\Hardware\m_Security008.exe	2.3	2010/02/17 16:28:05	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:48:31	6706	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security009\Hardware\m_Security009.exe	2.3	2010/02/17 16:32:40	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:47:53	6323	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security010\Hardware\m_Security010.exe	2.3	2010/02/18 17:45:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:44:36	38608	TELM-06	1	Pass	0	0	0	113
M_TCP_Security	m_Security011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security011\Hardware\m_Security011.exe	2.5	2010/06/03 18:16:10	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:40:15	20029	TELM-06	1	Pass	0	0	0	58
M_TCP_Security	m_Security012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security012\Hardware\m_Security012.exe	2.9	2010/02/18 15:14:18	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 01:33:23	4660	TELM-06	1	Pass	0	0	0	132
M_TCP_Security	m_Security013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security013\Hardware\m_Security013.exe	2.5	2010/02/18 15:15:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:30:27	11771	TELM-06	1	Pass	0	0	0	140
M_TCP_Security	m_Security014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security014\Hardware\m_Security014.exe	2.2	2010/06/03 18:21:27	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 04:55:32	15976	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security015\Hardware\m_Security015.exe	2.4	2010/02/17 17:58:26	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 23:25:18	596197	TELM-06	1	Pass	0	0	0	34
M_TCP_Security	m_Security016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security016\Hardware\m_Security016.exe	2.2	2010/06/08 19:56:40	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:28:22	10648	TELM-06	1	Pass	0	0	0	146
M_TCP_Security	m_Security017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security017\Hardware\m_Security017.exe	2.2	2009/12/08 17:29:14	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:25:56	14809	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security018\Hardware\m_Security018.exe	2.4	2010/03/18 16:05:37	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:21:39	40556	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security019\Hardware\m_Security019.exe	2.3	2010/03/26 13:43:17	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:17:17	90648	TELM-06	1	Pass	0	0	0	1226
M_TCP_Security	m_Security020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security020\Hardware\m_Security020.exe	2.3	2010/06/03 18:22:30	2.3	Up-to-date	Baseline 17_1	None	09/13/2010 09:10:16	22108	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security021\Hardware\m_Security021.exe	2.5	2010/06/03 18:37:52	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:11:51	40918	TELM-06	1	Pass	0	0	0	81
M_TCP_Security	m_Security022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security022\Hardware\m_Security022.exe	2.9	2010/06/18 19:05:16	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 11:31:28	290901	TELM-06	1	Pass	0	0	0	55495
M_TCP_Security	m_Security023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security023\Hardware\m_Security023.exe	2.12	2010/06/10 17:30:47	2.12	Up-to-date	Baseline 17_1	None	09/10/2010 23:04:54	51580	TELM-06	1	Pass	0	0	0	1119
M_TCP_Security	m_Security024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security024\Hardware\m_Security024.exe	2.3	2010/03/26 13:06:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 18:19:58	30771	TELM-06	1	Pass	0	0	0	4
M_TCP_Security	m_Security025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security025\Hardware\m_Security025.exe	2.3	2010/06/10 14:39:09	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 17:42:06	48431	TELM-06	1	Pass	0	0	0	158
M_TCP_Security	m_Security026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security026\Hardware\m_Security026.exe	2.3	2010/02/18 15:15:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:07:50	4285	TELM-06	1	Pass	0	0	0	60
M_TCP_Security	m_Security027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security027\Hardware\m_Security027.exe	2.1	2010/06/29 19:03:00	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:03:19	42310	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security028\Hardware\m_Security028.exe	2.1	2010/06/29 19:26:29	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:00:51	42957	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security029\Hardware\m_Security029.exe	2.2	2010/03/11 18:29:14	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:56:16	20492	TELM-06	1	Pass	0	0	0	46
M_TCP_Security	m_Security030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security030\Hardware\m_Security030.exe	2.2	2010/02/08 13:02:02	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:54:34	11754	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security031\Hardware\m_Security031.exe	2.1	2010/06/03 18:44:12	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:52:06	22901	TELM-06	1	Pass	0	0	0	16
M_TCP_Security	m_Security032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security032\Hardware\m_Security032.exe	2.1	2010/02/03 12:51:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:49:18	8188	TELM-06	1	Pass	0	0	0	11
M_TCP_Security	m_Security033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security033\Hardware\m_Security033.exe	2.1	2010/01/27 21:32:15	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:46:55	13999	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security034\Hardware\m_Security034.exe	2.3	2010/03/26 13:19:51	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:39:40	118598	TELM-06	1	Pass	0	0	0	15
M_TCP_Security	m_Security035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security035\Hardware\m_Security035.exe	2.0	2010/01/14 21:18:30	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 00:37:09	17610	TELM-06	1	Pass	0	0	0	30
M_TCP_Security	m_Security036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security036\Hardware\m_Security036.exe	2.5	2010/05/07 14:41:43	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 00:32:50	86631	TELM-06	1	Pass	0	0	0	10
M_TCP_Security	m_Security037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security037\Hardware\m_Security037.exe	2.6	2010/06/03 18:54:41	2.6	Up-to-date	Baseline 17_1	None	09/11/2010 04:16:06	51328	TELM-06	1	Pass	0	0	0	119
M_TCP_Security	m_Security038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security038\Hardware\m_Security038.exe	2.9	2010/06/07 13:02:07	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 17:19:03	167458	TELM-06	1	Pass	0	0	0	170
M_TCP_Security	m_Security039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security039\Hardware\m_Security039.exe	2.2	2010/02/18 15:16:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:29:15	56574	TELM-06	1	Pass	0	0	0	55
M_TCP_Security	m_Security040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security040\Hardware\m_Security040.exe	2.3	2010/06/03 20:43:31	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:22:56	278514	TELM-06	1	Pass	0	0	0	33
M_TCP_Security	m_Security041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security041\Hardware\m_Security041.exe	2.6	2010/04/21 10:55:08	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 00:20:45	66745	TELM-06	1	Pass	0	0	0	77
M_TCP_Security	m_Security042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security042\Hardware\m_Security042.exe	2.1	2010/04/08 12:57:18	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:17:29	58300	TELM-06	1	Pass	0	0	0	48
M_TCP_Security	m_Security043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security043\Hardware\m_Security043.exe	2.0	2010/02/16 20:27:42	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 00:13:46	68894	TELM-06	1	Pass	0	0	0	56
M_TCP_Security	m_Security044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security044\Hardware\m_Security044.exe	2.2	2010/05/13 19:14:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:10:28	70750	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security045\Hardware\m_Security045.exe	2.3	2010/05/12 17:38:53	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:07:05	71039	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security046\Hardware\m_Security046.exe	2.2	2010/06/07 13:05:30	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:05:07	3341	TELM-06	1	Pass	0	0	0	30
M_TCP_WirelessWakeup	m_Wakeup001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup001\Hardware\m_Wakeup001.exe	2.3	2010/06/21 17:44:25	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 16:57:10	291431	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup002\Hardware\m_Wakeup002.exe	2.5	2010/05/12 16:16:45	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 00:01:34	35312	TELM-06	1	Pass	0	0	0	12
M_TCP_WirelessWakeup	m_Wakeup003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup003\Hardware\m_Wakeup003.exe	2.2	2010/03/10 17:59:29	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:57:56	14408	TELM-06	1	Pass	0	0	0	11
M_TCP_WirelessWakeup	m_Wakeup004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup004\Hardware\m_Wakeup004.exe	2.1	2009/12/08 16:08:20	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 04:14:36	19328	TELM-06	1	Pass	0	0	0	31
M_TCP_WirelessWakeup	m_Wakeup005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup005\Hardware\m_Wakeup005.exe	2.3	2010/06/08 21:32:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:56:30	20389	TELM-06	1	Pass	0	0	0	40
M_TCP_WirelessWakeup	m_Wakeup006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup006\Hardware\m_Wakeup006.exe	2.3	2010/01/27 00:35:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:51:49	10190	TELM-06	1	Pass	0	0	0	55
M_TCP_WirelessWakeup	m_Wakeup007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup007\Hardware\m_Wakeup007.exe	2.1	2009/12/04 19:28:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:49:29	12852	TELM-06	1	Pass	0	0	0	3
M_TCP_WirelessWakeup	m_Wakeup008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup008\Hardware\m_Wakeup008.exe	2.5	2010/05/07 15:41:21	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:47:10	54576	TELM-06	1	Pass	0	0	0	24
M_TCP_WirelessWakeup	m_Wakeup009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup009\Hardware\m_Wakeup009.exe	2.6	2010/05/12 16:17:49	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:44:24	51998	TELM-06	1	Pass	0	0	0	32
M_TCP_WirelessWakeup	m_Wakeup010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup010\Hardware\m_Wakeup010.exe	2.4	2010/05/12 16:20:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 23:42:31	45208	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup011\Hardware\m_Wakeup011.exe	2.3	2010/04/06 18:07:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:39:24	28013	TELM-06	1	Pass	0	0	0	25
M_TCP_WirelessWakeup	m_Wakeup012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup012\Hardware\m_Wakeup012.exe	2.6	2010/05/12 16:20:45	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:37:13	48813	TELM-06	1	Pass	0	0	0	45
M_TCP_WirelessWakeup	m_Wakeup013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup013\Hardware\m_Wakeup013.exe	2.2	2010/02/24 12:20:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:32:42	127541	TELM-06	1	Pass	0	0	0	2
M_TCP_WirelessWakeup	m_Wakeup014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode7_8\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup014\Hardware\m_Wakeup014.exe	2.3	2010/06/04 11:50:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:26:57	74712	TELM-06	1	Pass	0	0	0	30
M_TCP_AntennaSelection	m_Antenna001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna001\Hardware\m_Antenna001.exe	2.4	2010/06/16 12:31:48	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 12:12:09	232241	TELM-06	1	Pass	0	0	0	127
M_TCP_AntennaSelection	m_Antenna002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_AntennaSelection\m_Antenna002\Hardware\m_Antenna002.exe	2.0	2010/06/16 12:29:50	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 11:47:20	287043	TELM-06	1	Pass	0	0	0	99
M_TCP_Channel	m_ChannelRecovery001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery001\Hardware\m_ChannelRecovery001.exe	2.5	2010/04/08 14:25:18	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 08:17:49	15057	TELM-06	1	Pass	0	0	0	26
M_TCP_Channel	m_ChannelRecovery002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery002\Hardware\m_ChannelRecovery002.exe	2.2	2009/11/30 22:31:32	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:15:58	17609	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery003\Hardware\m_ChannelRecovery003.exe	2.2	2009/11/30 22:32:21	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:05:44	17517	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery004\Hardware\m_ChannelRecovery004.exe	2.2	2010/05/26 12:04:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 08:03:16	14543	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery005\Hardware\m_ChannelRecovery005.exe	2.5	2010/06/21 18:34:44	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 08:00:40	14243	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery006\Hardware\m_ChannelRecovery006.exe	2.4	2010/05/07 17:09:38	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 07:55:59	15829	TELM-06	1	Pass	0	0	0	35
M_TCP_Channel	m_ChannelRecovery007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery007\Hardware\m_ChannelRecovery007.exe	2.7	2010/05/11 11:49:24	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 07:48:21	27570	TELM-06	1	Pass	0	0	0	89
M_TCP_Channel	m_ChannelRecovery008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery008\Hardware\m_ChannelRecovery008.exe	2.6	2010/05/13 15:05:39	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 07:37:32	27203	TELM-06	1	Pass	0	0	0	576
M_TCP_Channel	m_ChannelRecovery009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery009\Hardware\m_ChannelRecovery009.exe	2.6	2010/06/22 15:16:15	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 07:29:01	57655	TELM-06	1	Pass	0	0	0	342
M_TCP_Channel	m_ChannelRecovery010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery010\Hardware\m_ChannelRecovery010.exe	2.8	2010/05/26 15:48:50	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 07:23:08	25068	TELM-06	1	Pass	0	0	0	144
M_TCP_Channel	m_ChannelRecovery011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery011\Hardware\m_ChannelRecovery011.exe	2.5	2010/05/13 12:31:10	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 07:16:10	16079	TELM-06	1	Pass	0	0	0	15
M_TCP_Channel	m_ChannelRecovery012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery012\Hardware\m_ChannelRecovery012.exe	2.6	2010/09/07 20:36:00	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 07:09:16	48170	TELM-06	1	Pass	0	0	0	199
M_TCP_Channel	m_ChannelRecovery013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery013\Hardware\m_ChannelRecovery013.exe	2.2	2010/01/05 19:17:54	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 07:02:40	24250	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery014\Hardware\m_ChannelRecovery014.exe	2.8	2010/05/13 13:19:47	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 06:55:22	51604	TELM-06	1	Pass	0	0	0	23
M_TCP_Channel	m_ChannelRecovery015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery015\Hardware\m_ChannelRecovery015.exe	2.6	2010/06/03 18:01:04	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 06:48:31	12589	TELM-06	1	Pass	0	0	0	88
M_TCP_Channel	m_ChannelRecovery016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery016\Hardware\m_ChannelRecovery016.exe	2.4	2010/02/11 20:10:03	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 06:45:38	38084	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery017\Hardware\m_ChannelRecovery017.exe	2.3	2010/02/11 17:38:13	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:41:03	10373	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery018\Hardware\m_ChannelRecovery018.exe	2.3	2010/02/01 19:41:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:35:26	18159	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery020\Hardware\m_ChannelRecovery020.exe	2.7	2010/06/23 17:42:40	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 06:32:48	37526	TELM-06	1	Pass	0	0	0	109
M_TCP_Channel	m_ChannelRecovery021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery021\Hardware\m_ChannelRecovery021.exe	2.3	2010/04/13 18:33:24	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:29:39	27269	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery022\Hardware\m_ChannelRecovery022.exe	2.3	2010/04/13 18:36:50	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 06:26:37	19975	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery023\Hardware\m_ChannelRecovery023.exe	2.5	2010/04/13 18:41:23	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 06:20:21	29948	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery024\Hardware\m_ChannelRecovery024.exe	2.4	2010/04/07 12:33:58	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 06:16:29	19832	TELM-06	1	Pass	0	0	0	9

M_TCP_Channel	m_ChannelRecovery025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery025\Hardware\m_ChannelRecovery025.exe	2.2	2010/02/01 19:43:28	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:12:34	11075	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery026\Hardware\m_ChannelRecovery026.exe	2.2	2010/02/11 21:01:06	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:09:31	11497	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery027\Hardware\m_ChannelRecovery027.exe	2.2	2010/02/03 17:18:54	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:06:21	13518	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery028\Hardware\m_ChannelRecovery028.exe	2.2	2010/02/03 17:19:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:03:08	11555	TELM-06	1	Pass	0	0	0	10
M_TCP_Channel	m_ChannelRecovery029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery029\Hardware\m_ChannelRecovery029.exe	2.2	2009/12/18 20:13:33	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 06:01:46	13536	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery030\Hardware\m_ChannelRecovery030.exe	2.10	2010/06/23 21:15:31	2.10	Up-to-date	Baseline 17_1	None	09/10/2010 05:58:44	18514	TELM-06	1	Pass	0	0	0	30
M_TCP_Channel	m_ChannelRecovery031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery031\Hardware\m_ChannelRecovery031.exe	2.6	2010/04/13 18:46:58	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:55:15	18861	TELM-06	1	Pass	0	0	0	9
M_TCP_Channel	m_ChannelRecovery032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery032\Hardware\m_ChannelRecovery032.exe	2.5	2010/06/23 21:50:13	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:50:56	18080	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery033\Hardware\m_ChannelRecovery033.exe	2.3	2010/04/07 17:51:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:48:54	19153	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery038\Hardware\m_ChannelRecovery038.exe	2.6	2010/05/26 15:55:12	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:47:36	20721	TELM-06	1	Pass	0	0	0	175
M_TCP_Channel	m_ChannelRecovery039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery039\Hardware\m_ChannelRecovery039.exe	2.3	2010/02/10 20:47:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:45:26	12345	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery040\Hardware\m_ChannelRecovery040.exe	2.5	2010/06/07 12:54:38	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:42:34	15484	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery041\Hardware\m_ChannelRecovery041.exe	2.1	2009/10/22 13:18:38	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:38:03	38430	TELM-06	1	Pass	0	0	0	1
M_TCP_Channel	m_ChannelRecovery042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery042\Hardware\m_ChannelRecovery042.exe	2.3	2009/12/03 15:57:39	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:36:09	21969	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery043\Hardware\m_ChannelRecovery043.exe	2.6	2010/06/24 18:37:58	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 05:34:13	55950	TELM-06	1	Pass	0	0	0	37
M_TCP_Channel	m_ChannelRecovery045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery045\Hardware\m_ChannelRecovery045.exe	2.2	2009/11/12 21:22:11	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:30:32	11488	TELM-06	1	Pass	0	0	0	42
M_TCP_Channel	m_ChannelRecovery046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery046\Hardware\m_ChannelRecovery046.exe	2.3	2009/11/16 18:40:10	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:29:05	12899	TELM-06	1	Pass	0	0	0	49
M_TCP_Channel	m_ChannelRecovery047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery047\Hardware\m_ChannelRecovery047.exe	2.1	2009/11/16 18:58:46	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 05:24:30	52547	TELM-06	1	Pass	0	0	0	95
M_TCP_Channel	m_ChannelRecovery048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery048\Hardware\m_ChannelRecovery048.exe	2.3	2010/06/24 18:46:14	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:21:48	62582	TELM-06	1	Pass	0	0	0	156
M_TCP_Channel	m_ChannelRecovery049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery049\Hardware\m_ChannelRecovery049.exe	2.5	2010/03/26 13:55:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 05:14:56	36436	TELM-06	1	Pass	0	0	0	207
M_TCP_Channel	m_ChannelRecovery050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery050\Hardware\m_ChannelRecovery050.exe	2.2	2009/11/20 13:38:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 05:09:00	18777	TELM-06	1	Pass	0	0	0	2
M_TCP_Channel	m_ChannelRecovery051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery051\Hardware\m_ChannelRecovery051.exe	2.3	2010/06/24 19:40:38	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:04:05	82726	TELM-06	1	Pass	0	0	0	196
M_TCP_Channel	m_ChannelRecovery053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery053\Hardware\m_ChannelRecovery053.exe	2.3	2010/06/24 21:57:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 05:00:51	31168	TELM-06	1	Pass	0	0	0	96
M_TCP_Channel	m_ChannelRecovery054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery054\Hardware\m_ChannelRecovery054.exe	2.4	2010/06/24 22:31:30	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:57:32	8879	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery055\Hardware\m_ChannelRecovery055.exe	2.4	2010/02/11 14:12:25	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 04:21:43	2022286	TELM-06	1	Pass	0	0	0	6
M_TCP_Channel	m_ChannelRecovery061	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery061\Hardware\m_ChannelRecovery061.exe	2.3	2010/04/06 11:52:17	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:53:53	20597	TELM-06	1	Pass	0	0	0	12
M_TCP_Channel	m_ChannelRecovery062	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery062\Hardware\m_ChannelRecovery062.exe	2.2	2010/06/25 15:47:28	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:52:06	12997	TELM-06	1	Pass	0	0	0	4
M_TCP_Channel	m_ChannelRecovery063	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery063\Hardware\m_ChannelRecovery063.exe	2.9	2010/05/13 17:43:56	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 04:50:10	24377	TELM-06	1	Pass	0	0	0	39
M_TCP_Channel	m_ChannelRecovery064	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery064\Hardware\m_ChannelRecovery064.exe	2.4	2010/05/06 14:14:13	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 04:47:29	40824	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery065\Hardware\m_ChannelRecovery065.exe	2.2	2010/05/06 15:43:30	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:45:15	37311	TELM-06	1	Pass	0	0	0	7
M_TCP_Channel	m_ChannelRecovery066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery066\Hardware\m_ChannelRecovery066.exe	2.2	2010/05/06 15:49:13	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:42:37	37536	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery067\Hardware\m_ChannelRecovery067.exe	2.2	2010/05/06 15:58:13	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:40:31	40789	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery068\Hardware\m_ChannelRecovery068.exe	2.3	2010/06/03 18:10:46	2.3	Up-to-date	Baseline 17_1	None	09/13/2010 09:12:25	34699	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery069	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery069\Hardware\m_ChannelRecovery069.exe	2.3	2010/05/10 15:40:47	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:18:25	1009306	TELM-06	1	Pass	0	0	0	5
M_TCP_Channel	m_ChannelRecovery070	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery070\Hardware\m_ChannelRecovery070.exe	2.1	2010/02/04 18:14:32	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:30:08	12802	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery071	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery071\Hardware\m_ChannelRecovery071.exe	2.1	2010/02/04 17:57:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 04:26:41	15727	TELM-06	1	Pass	0	0	0	3
M_TCP_Channel	m_ChannelRecovery072	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery072\Hardware\m_ChannelRecovery072.exe	2.3	2010/05/07 11:42:18	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:24:53	44181	TELM-06	1	Pass	0	0	0	8
M_TCP_Channel	m_ChannelRecovery073	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Channel\m_ChannelRecovery073\Hardware\m_ChannelRecovery073.exe	2.3	2010/05/07 11:43:00	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:18:10	44238	TELM-06	1	Pass	0	0	0	8
M_TCP_Close	m_Close001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close001\Hardware\m_Close001.exe	2.6	2010/03/29 17:53:05	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 12:09:36	7286	TELM-06	1	Pass	0	0	0	51
M_TCP_Close	m_Close002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close002\Hardware\m_Close002.exe	2.3	2010/01/26 20:03:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:07:19	8890	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close003\Hardware\m_Close003.exe	2.3	2010/06/24 15:01:11	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:05:19	7187	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close004\Hardware\m_Close004.exe	2.5	2010/03/16 12:03:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:03:22	14834	TELM-06	1	Pass	0	0	0	138
M_TCP_Close	m_Close005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close005\Hardware\m_Close005.exe	2.5	2010/05/24 17:00:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:01:25	19306	TELM-06	1	Pass	0	0	0	188
M_TCP_Close	m_Close006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close006\Hardware\m_Close006.exe	2.2	2009/11/19 20:38:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:56:27	11105	TELM-06	1	Pass	0	0	0	13
M_TCP_Close	m_Close007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close007\Hardware\m_Close007.exe	2.1	2010/02/04 18:46:09	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:52:10	16218	TELM-06	1	Pass	0	0	0	66
M_TCP_Close	m_Close008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close008\Hardware\m_Close008.exe	2.4	2010/05/12 15:22:15	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:48:06	7085	TELM-06	1	Pass	0	0	0	10
M_TCP_Close	m_Close009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close009\Hardware\m_Close009.exe	2.2	2010/02/04 15:29:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:46:17	7922	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close010\Hardware\m_Close010.exe	2.0	2009/09/21 12:45:39	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:43:49	15907	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close011\Hardware\m_Close011.exe	2.0	2009/09/21 12:46:09	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:42:27	8183	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close012\Hardware\m_Close012.exe	2.0	2009/09/21 12:46:36	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:40:41	11457	TELM-06	1	Pass	0	0	0	3
M_TCP_Close	m_Close013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close013\Hardware\m_Close013.exe	2.0	2009/09/21 12:18:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:38:07	7902	TELM-06	1	Pass	0	0	0	4
M_TCP_Close	m_Close014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close014\Hardware\m_Close014.exe	2.1	2010/01/29 16:45:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:34:49	8825	TELM-06	1	Pass	0	0	0	6
M_TCP_Close	m_Close016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close016\Hardware\m_Close016.exe	2.2	2010/08/24 12:05:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:32:15	8077	TELM-06	1	Pass	0	0	0	5
M_TCP_Close	m_Close017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close017\Hardware\m_Close017.exe	2.0	2009/09/21 12:47:07	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:30:09	9504	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close019\Hardware\m_Close019.exe	2.1	2009/10/01 14:47:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:28:58	5214	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close020\Hardware\m_Close020.exe	2.2	2009/10/30 14:47:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:27:07	9425	TELM-06	1	Pass	0	0	0	2
M_TCP_Close	m_Close021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close021\Hardware\m_Close021.exe	2.0	2009/09/21 12:48:28	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 11:25:58	2896	TELM-06	1	Pass	0	0	0	1
M_TCP_Close	m_Close022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Close\m_Close022\Hardware\m_Close022.exe	2.2	2009/10/30 14:53:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:24:34	12392	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_ARQ001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ001\Hardware\m_ARQ001.exe	2.5	2010/06/29 15:20:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:37:37	26897	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ002\Hardware\m_ARQ002.exe	2.0	2009/09/21 18:25:53	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:35:34	7872	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ003\Hardware\m_ARQ003.exe	2.3	2010/04/05 12:22:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:33:52	8422	TELM-06	1	Pass	0	0	0	148
M_TCP_Data	m_ARQ004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ004\Hardware\m_ARQ004.exe	2.1	2009/11/04 21:22:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:32:34	8341	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_ARQ005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ005\Hardware\m_ARQ005.exe	2.2	2009/12/23 13:26:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:30:39	7659	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ006\Hardware\m_ARQ006.exe	2.1	2009/09/24 16:41:52	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:28:39	8629	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_ARQ007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ007\Hardware\m_ARQ007.exe	2.0	2009/09/21 18:27:29	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:27:12	10508	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_ARQ008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ008\Hardware\m_ARQ008.exe	2.0	2009/09/21 18:28:06	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:25:52	10635	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ009\Hardware\m_ARQ009.exe	2.2	2010/02/04 13:08:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:23:42	8646	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ010\Hardware\m_ARQ010.exe	2.2	2010/02/04 13:19:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:22:17	8829	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_ARQ011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ011\Hardware\m_ARQ011.exe	2.3	2010/02/01 13:30:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:20:55	8275	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_ARQ012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ012\Hardware\m_ARQ012.exe	2.3	2010/02/18 15:17:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:19:26	8611	TELM-06	1	Pass	0	0	0	211
M_TCP_Data	m_ARQ013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ013\Hardware\m_ARQ013.exe	2.3	2010/02/03 16:38:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:18:14	10592	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_ARQ014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ014\Hardware\m_ARQ014.exe	2.4	2010/04/05 11:08:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:15:36	8414	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ015\Hardware\m_ARQ015.exe	2.6	2010/06/08 15:25:41	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:13:38	12838	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_ARQ016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ016\Hardware\m_ARQ016.exe	2.3	2009/11/19 20:22:28	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 19:10:50	9339	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_ARQ017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ017\Hardware\m_ARQ017.exe	2.2	2009/09/30 15:01:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:06:04	6809	TELM-06	1	Pass	0	0	0	66
M_TCP_Data	m_ARQ018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ018\Hardware\m_ARQ018.exe	2.7	2010/06/24 12:34:26	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 19:04:16	10873	TELM-06	1	Pass	0	0	0	634
M_TCP_Data	m_ARQ019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ019\Hardware\m_ARQ019.exe	2.4	2010/06/08 16:07:58	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:00:48	10235	TELM-06	1	Pass	0	0	0	102
M_TCP_Data	m_ARQ020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ020\Hardware\m_ARQ020.exe	2.2	2009/11/18 15:23:22	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:59:17	7522	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_ARQ021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ021\Hardware\m_ARQ021.exe	2.2	2009/11/19 18:43:52	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:57:55	7560	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ022\Hardware\m_ARQ022.exe	2.3	2010/02/03 16:30:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:54:09	8469	TELM-06	1	Pass	0	0	0	137
M_TCP_Data	m_ARQ023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ023\Hardware\m_ARQ023.exe	2.1	2009/11/10 19:12:36	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:50:44	9642	TELM-06	1	Pass	0	0	0	63
M_TCP_Data	m_ARQ024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ024\Hardware\m_ARQ024.exe	2.3	2010/02/08 18:44:31	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:46:42	10132	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ025\Hardware\m_ARQ025.exe	2.1	2009/09/25 14:20:39	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:43:59	8648	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ026\Hardware\m_ARQ026.exe	2.1	2009/09/25 14:21:05	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:38:23	8616	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_ARQ027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ027\Hardware\m_ARQ027.exe	2.1	2009/09/25 14:21:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:36:06	9154	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_ARQ028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ028\Hardware\m_ARQ028.exe	2.4	2010/04/19 14:36:56	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 18:34:50	18747	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_ARQ029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ029\Hardware\m_ARQ029.exe	2.1	2010/03/08 19:53:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 18:33:07	29550	TELM-06	1	Pass	0	0	0	555
M_TCP_Data	m_ARQ030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ030\Hardware\m_ARQ030.exe	2.5	2010/06/03 15:11:29	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 18:27:46	16840	TELM-06	1	Pass	0	0	0	356
M_TCP_Data	m_ARQ031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_ARQ031\Hardware\m_ARQ031.exe	2.3	2010/06/08 19:57:52	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:25:40	11215	TELM-06	1	Pass	0	0	0	596
M_TCP_Data	m_Data012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data012\Hardware\m_Data012.exe	2.4	2009/11/10 14:18:10	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 02:51:08	37620	TELM-06	1	Pass	0	0	0	18
M_TCP_Data	m_Data013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data013\Hardware\m_Data013.exe	2.3	2010/02/03 15:41:32	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:19:02	103177	TELM-06	1	Pass	0	0	0	335
M_TCP_Data	m_Data014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data014\Hardware\m_Data014.exe	2.3	2010/05/12 19:37:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:14:23	92582	TELM-06	1	Pass	0	0	0	36
M_TCP_Data	m_Data015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data015\Hardware\m_Data015.exe	2.2	2010/02/03 17:34:25	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:12:41	8191	TELM-06	1	Pass	0	0	0	14
M_TCP_Data	m_Data016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data016\Hardware\m_Data016.exe	2.2	2010/02/24 20:56:20	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:09:51	70052	TELM-06	1	Pass	0	0	0	26
M_TCP_Data	m_Data017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data017\Hardware\m_Data017.exe	2.2	2010/02/24 22:22:41	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:07:03	67961	TELM-06	1	Pass	0	0	0	30
M_TCP_Data	m_Data018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_Data018\Hardware\m_Data018.exe	2.2	2010/02/03 22:48:42	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:04:07	59067	TELM-06	1	Pass	0	0	0	19
M_TCP_Data	m_MemoryRead001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead001\Hardware\m_MemoryRead001.exe	2.2	2010/02/01 17:26:47	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 18:03:18	19011	TELM-06	1	Pass	0	0	0	439
M_TCP_Data	m_MemoryRead002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead002\Hardware\m_MemoryRead002.exe	2.3	2009/11/05 15:03:23	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 18:01:18	18775	TELM-06	1	Pass	0	0	0	31
M_TCP_Data	m_MemoryRead003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead003\Hardware\m_MemoryRead003.exe	2.4	2010/04/05 13:08:02	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:59:14	20089	TELM-06	1	Pass	0	0	0	320
M_TCP_Data	m_MemoryRead004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead004\Hardware\m_MemoryRead004.exe	2.3	2010/03/19 16:10:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:54:05	8311	TELM-06	1	Pass	0	0	0	1
M_TCP_Data	m_MemoryRead005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead005\Hardware\m_MemoryRead005.exe	2.3	2009/12/07 19:50:30	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:47:50	48545	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryRead006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead006\Hardware\m_MemoryRead006.exe	2.6	2010/06/29 21:05:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 17:44:18	12201	TELM-06	1	Pass	0	0	0	37
M_TCP_Data	m_MemoryRead007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead007\Hardware\m_MemoryRead007.exe	2.1	2009/11/03 12:42:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:41:52	10720	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_MemoryRead008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead008\Hardware\m_MemoryRead008.exe	2.1	2009/09/30 18:09:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:40:02	7808	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead009\Hardware\m_MemoryRead009.exe	2.1	2009/12/07 19:51:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:38:59	11324	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead010\Hardware\m_MemoryRead010.exe	2.2	2009/09/30 17:31:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:35:13	7626	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead011\Hardware\m_MemoryRead011.exe	2.2	2009/09/30 17:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:33:13	7684	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryRead012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead012\Hardware\m_MemoryRead012.exe	2.2	2009/09/30 17:33:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:31:01	7640	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead013\Hardware\m_MemoryRead013.exe	2.2	2009/09/30 17:33:27	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:29:15	7719	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead014\Hardware\m_MemoryRead014.exe	2.2	2009/09/30 17:33:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:27:16	7777	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryRead015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead015\Hardware\m_MemoryRead015.exe	2.2	2009/09/30 17:34:18	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:25:08	7694	TELM-06	1	Pass	0	0	0	40

M_TCP_Data	m_MemoryRead016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead016\Hardware\m_MemoryRead016.exe	2.1	2009/10/19 18:24:00	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 17:19:11	7646	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead017\Hardware\m_MemoryRead017.exe	2.2	2009/10/19 15:52:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:15:42	7645	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryRead019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead019\Hardware\m_MemoryRead019.exe	2.3	2009/12/03 13:57:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 17:08:47	9299	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead020\Hardware\m_MemoryRead020.exe	2.4	2010/02/09 20:49:06	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 17:04:20	6920	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryRead021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead021\Hardware\m_MemoryRead021.exe	2.2	2009/12/23 13:34:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 17:01:50	8444	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryRead030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead030\Hardware\m_MemoryRead030.exe	2.1	2009/10/16 13:54:03	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:55:57	6634	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead031\Hardware\m_MemoryRead031.exe	2.3	2009/12/28 18:23:34	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:53:21	9913	TELM-06	1	Pass	0	0	0	43
M_TCP_Data	m_MemoryRead033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead033\Hardware\m_MemoryRead033.exe	2.3	2009/11/23 18:55:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:49:26	11987	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead034\Hardware\m_MemoryRead034.exe	2.1	2009/10/16 14:15:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:43:27	8216	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_MemoryRead035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead035\Hardware\m_MemoryRead035.exe	2.4	2009/11/23 18:56:23	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:36:55	16817	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead037\Hardware\m_MemoryRead037.exe	2.2	2009/11/23 18:57:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:33:21	12320	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead038\Hardware\m_MemoryRead038.exe	2.3	2010/01/19 17:36:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:30:43	14622	TELM-06	1	Pass	0	0	0	982
M_TCP_Data	m_MemoryRead039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead039\Hardware\m_MemoryRead039.exe	2.1	2009/09/28 15:09:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:28:19	8578	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead040\Hardware\m_MemoryRead040.exe	2.6	2010/01/06 14:10:47	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 16:25:09	32251	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryRead041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead041\Hardware\m_MemoryRead041.exe	2.4	2009/11/16 12:52:24	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 16:22:18	18221	TELM-06	1	Pass	0	0	0	22
M_TCP_Data	m_MemoryRead042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead042\Hardware\m_MemoryRead042.exe	2.3	2010/06/29 22:24:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:20:32	26122	TELM-06	1	Pass	0	0	0	41
M_TCP_Data	m_MemoryRead043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead043\Hardware\m_MemoryRead043.exe	2.2	2010/02/08 16:55:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:18:42	17725	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryRead045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead045\Hardware\m_MemoryRead045.exe	2.0	2009/09/22 14:34:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:16:16	6898	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryRead049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead049\Hardware\m_MemoryRead049.exe	2.3	2009/11/16 12:52:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:13:46	8638	TELM-06	1	Pass	0	0	0	12
M_TCP_Data	m_MemoryRead051	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead051\Hardware\m_MemoryRead051.exe	2.1	2009/10/27 19:22:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:10:26	9632	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryRead052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryRead052\Hardware\m_MemoryRead052.exe	2.3	2010/06/29 23:18:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 16:08:13	11035	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite001\Hardware\m_MemoryWrite001.exe	2.2	2009/12/07 19:52:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 16:06:38	8634	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite002\Hardware\m_MemoryWrite002.exe	2.0	2009/09/22 14:48:30	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 16:04:46	10436	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite003\Hardware\m_MemoryWrite003.exe	2.1	2009/10/28 14:34:41	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 16:00:33	10072	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite005\Hardware\m_MemoryWrite005.exe	2.2	2009/12/15 19:05:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:53:51	7829	TELM-06	1	Pass	0	0	0	186
M_TCP_Data	m_MemoryWrite006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite006\Hardware\m_MemoryWrite006.exe	2.3	2009/11/24 17:33:37	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:50:37	11245	TELM-06	1	Pass	0	0	0	145
M_TCP_Data	m_MemoryWrite007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite007\Hardware\m_MemoryWrite007.exe	2.1	2009/10/28 15:09:57	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:48:47	6777	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite008\Hardware\m_MemoryWrite008.exe	2.1	2010/02/01 19:11:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:43:05	7350	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite009\Hardware\m_MemoryWrite009.exe	2.1	2009/10/28 15:17:22	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:40:32	7552	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite010\Hardware\m_MemoryWrite010.exe	2.1	2009/10/28 16:00:35	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:40:04	7521	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite011\Hardware\m_MemoryWrite011.exe	2.1	2009/10/28 16:07:34	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:37:12	7510	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite012\Hardware\m_MemoryWrite012.exe	2.2	2009/10/28 16:15:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:32:03	12274	TELM-06	1	Pass	0	0	0	120
M_TCP_Data	m_MemoryWrite013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite013\Hardware\m_MemoryWrite013.exe	2.1	2009/10/28 16:21:17	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:30:03	7482	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite014\Hardware\m_MemoryWrite014.exe	2.1	2009/10/28 16:29:59	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:29:00	7538	TELM-06	1	Pass	0	0	0	40
M_TCP_Data	m_MemoryWrite015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite015\Hardware\m_MemoryWrite015.exe	2.1	2009/10/28 16:36:30	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:27:20	7530	TELM-06	1	Pass	0	0	0	57
M_TCP_Data	m_MemoryWrite016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite016\Hardware\m_MemoryWrite016.exe	2.5	2010/02/01 21:11:41	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:26:09	9589	TELM-06	1	Pass	0	0	0	2270
M_TCP_Data	m_MemoryWrite017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite017\Hardware\m_MemoryWrite017.exe	2.3	2010/01/19 17:37:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:24:01	8540	TELM-06	1	Pass	0	0	0	1521
M_TCP_Data	m_MemoryWrite018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite018\Hardware\m_MemoryWrite018.exe	2.1	2009/10/29 20:06:15	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:22:27	6933	TELM-06	1	Pass	0	0	0	171
M_TCP_Data	m_MemoryWrite019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite019\Hardware\m_MemoryWrite019.exe	2.0	2009/09/22 15:13:38	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 15:21:08	9509	TELM-06	1	Pass	0	0	0	2
M_TCP_Data	m_MemoryWrite020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite020\Hardware\m_MemoryWrite020.exe	2.2	2009/10/29 19:44:00	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:20:00	7842	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite022\Hardware\m_MemoryWrite022.exe	2.3	2009/11/03 17:28:46	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 15:18:23	9924	TELM-06	1	Pass	0	0	0	4
M_TCP_Data	m_MemoryWrite023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite023\Hardware\m_MemoryWrite023.exe	2.4	2010/02/04 16:15:17	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 15:15:43	20581	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite024\Hardware\m_MemoryWrite024.exe	2.2	2009/11/24 17:35:21	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:14:16	7535	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_MemoryWrite025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite025\Hardware\m_MemoryWrite025.exe	2.1	2009/10/29 18:47:12	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:10:41	10965	TELM-06	1	Pass	0	0	0	7
M_TCP_Data	m_MemoryWrite029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite029\Hardware\m_MemoryWrite029.exe	2.2	2010/02/01 21:00:35	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 15:09:09	14014	TELM-06	1	Pass	0	0	0	15
M_TCP_Data	m_MemoryWrite032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite032\Hardware\m_MemoryWrite032.exe	2.5	2010/03/15 01:08:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 15:07:26	9629	TELM-06	1	Pass	0	0	0	509
M_TCP_Data	m_MemoryWrite038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite038\Hardware\m_MemoryWrite038.exe	2.1	2009/10/29 18:42:13	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:03:42	7244	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_MemoryWrite039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite039\Hardware\m_MemoryWrite039.exe	2.1	2010/02/10 18:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 15:01:50	9647	TELM-06	1	Pass	0	0	0	3
M_TCP_Data	m_MemoryWrite040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite040\Hardware\m_MemoryWrite040.exe	2.2	2010/02/10 17:56:56	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:58:48	9560	TELM-06	1	Pass	0	0	0	6
M_TCP_Data	m_MemoryWrite041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite041\Hardware\m_MemoryWrite041.exe	2.2	2010/02/01 20:45:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:57:28	13777	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite042\Hardware\m_MemoryWrite042.exe	2.2	2010/01/19 17:38:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:55:44	10275	TELM-06	1	Pass	0	0	0	981
M_TCP_Data	m_MemoryWrite046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite046\Hardware\m_MemoryWrite046.exe	2.3	2010/02/04 16:51:09	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:54:13	7273	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite047\Hardware\m_MemoryWrite047.exe	2.2	2009/11/11 19:12:51	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:52:27	8133	TELM-06	1	Pass	0	0	0	96
M_TCP_Data	m_MemoryWrite048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite048\Hardware\m_MemoryWrite048.exe	2.2	2009/11/11 18:32:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:51:22	9021	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite049\Hardware\m_MemoryWrite049.exe	2.5	2010/02/10 16:51:26	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:49:13	21256	TELM-06	1	Pass	0	0	0	13
M_TCP_Data	m_MemoryWrite052	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite052\Hardware\m_MemoryWrite052.exe	2.3	2009/12/23 13:35:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:48:34	11168	TELM-06	1	Pass	0	0	0	11
M_TCP_Data	m_MemoryWrite053	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite053\Hardware\m_MemoryWrite053.exe	2.2	2009/11/11 17:01:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:46:11	8459	TELM-06	1	Pass	0	0	0	9
M_TCP_Data	m_MemoryWrite054	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite054\Hardware\m_MemoryWrite054.exe	2.6	2010/06/30 17:58:00	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 14:43:54	10126	TELM-06	1	Pass	0	0	0	10
M_TCP_Data	m_MemoryWrite055	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite055\Hardware\m_MemoryWrite055.exe	2.2	2009/11/11 18:04:03	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:42:09	7327	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_MemoryWrite056	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_MemoryWrite056\Hardware\m_MemoryWrite056.exe	2.4	2010/02/10 16:32:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:40:41	8778	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode001\Hardware\m_TransferMode001.exe	2.8	2010/05/14 18:48:57	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 14:38:53	16926	TELM-06	1	Pass	0	0	0	4820
M_TCP_Data	m_TransferMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode002\Hardware\m_TransferMode002.exe	2.4	2010/07/02 13:00:52	2.4	Up-to-date	Baseline 17_1	None	09/13/2010 09:10:40	9933	TELM-06	1	Pass	0	0	0	166
M_TCP_Data	m_TransferMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode003\Hardware\m_TransferMode003.exe	2.0	2009/11/16 20:59:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:33:56	7078	TELM-06	1	Pass	0	0	0	129
M_TCP_Data	m_TransferMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode004\Hardware\m_TransferMode004.exe	2.3	2010/01/08 17:07:47	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:31:23	18134	TELM-06	1	Pass	0	0	0	314
M_TCP_Data	m_TransferMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode005\Hardware\m_TransferMode005.exe	2.7	2010/07/02 13:36:13	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 14:27:29	36416	TELM-06	1	Pass	0	0	0	460
M_TCP_Data	m_TransferMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode006\Hardware\m_TransferMode006.exe	2.4	2010/04/16 11:58:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 14:26:28	6929	TELM-06	1	Pass	0	0	0	8
M_TCP_Data	m_TransferMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode007\Hardware\m_TransferMode007.exe	2.2	2009/10/12 15:46:45	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 14:24:40	11767	TELM-06	1	Pass	0	0	0	112
M_TCP_Data	m_TransferMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode008\Hardware\m_TransferMode008.exe	2.0	2009/09/09 15:28:45	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 14:22:00	10813	TELM-06	1	Pass	0	0	0	76
M_TCP_Data	m_TransferMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode009\Hardware\m_TransferMode009.exe	2.1	2009/10/13 16:25:11	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:20:00	11416	TELM-06	1	Pass	0	0	0	39
M_TCP_Data	m_TransferMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode010\Hardware\m_TransferMode010.exe	2.1	2009/10/13 16:26:21	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 14:17:24	11293	TELM-06	1	Pass	0	0	0	46
M_TCP_Data	m_TransferMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode011\Hardware\m_TransferMode011.exe	2.3	2010/02/10 13:37:08	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:15:29	8152	TELM-06	1	Pass	0	0	0	72
M_TCP_Data	m_TransferMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode012\Hardware\m_TransferMode012.exe	2.3	2010/02/10 13:38:00	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:12:32	8220	TELM-06	1	Pass	0	0	0	23
M_TCP_Data	m_TransferMode013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode013\Hardware\m_TransferMode013.exe	2.3	2010/02/10 13:39:16	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:10:40	8180	TELM-06	1	Pass	0	0	0	44
M_TCP_Data	m_TransferMode014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode014\Hardware\m_TransferMode014.exe	2.3	2010/02/10 13:40:35	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 14:08:38	8218	TELM-06	1	Pass	0	0	0	16
M_TCP_Data	m_TransferMode015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode015\Hardware\m_TransferMode015.exe	2.5	2010/05/26 14:56:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 14:05:30	7458	TELM-06	1	Pass	0	0	0	5
M_TCP_Data	m_TransferMode016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode016\Hardware\m_TransferMode016.exe	2.8	2010/07/02 13:14:11	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 14:04:11	29949	TELM-06	1	Pass	0	0	0	47
M_TCP_Data	m_TransferMode017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode017\Hardware\m_TransferMode017.exe	2.7	2010/05/18 14:47:09	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 14:01:39	36097	TELM-06	1	Pass	0	0	0	257
M_TCP_Data	m_TransferMode018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode018\Hardware\m_TransferMode018.exe	2.6	2010/04/16 14:06:26	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 13:56:11	26477	TELM-06	1	Pass	0	0	0	509
M_TCP_Data	m_TransferMode019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Data\m_TransferMode019\Hardware\m_TransferMode019.exe	2.6	2010/03/19 18:58:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 13:52:29	16839	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics001\Hardware\m_Diagnostics001.exe	2.5	2009/12/30 17:45:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:49:03	9566	TELM-06	1	Pass	0	0	0	166
M_TCP_Diagnostic	m_Diagnostics002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics002\Hardware\m_Diagnostics002.exe	2.2	2009/11/23 20:15:02	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:47:19	7751	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics003\Hardware\m_Diagnostics003.exe	2.2	2010/03/14 20:01:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 13:43:52	6820	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics004\Hardware\m_Diagnostics004.exe	2.3	2009/12/03 13:22:44	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:41:22	15844	TELM-06	1	Pass	0	0	0	204
M_TCP_Diagnostic	m_Diagnostics005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics005\Hardware\m_Diagnostics005.exe	2.1	2009/10/12 14:14:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:40:11	7798	TELM-06	1	Pass	0	0	0	13
M_TCP_Diagnostic	m_Diagnostics006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics006\Hardware\m_Diagnostics006.exe	2.0	2009/09/22 17:02:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:37:27	14857	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics007\Hardware\m_Diagnostics007.exe	2.1	2009/09/28 13:31:55	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 13:33:14	15066	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics008\Hardware\m_Diagnostics008.exe	2.0	2009/09/22 17:03:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:28:31	21383	TELM-06	1	Pass	0	0	0	31
M_TCP_Diagnostic	m_Diagnostics009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics009\Hardware\m_Diagnostics009.exe	2.5	2010/06/25 17:09:14	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:24:43	15869	TELM-06	1	Pass	0	0	0	23
M_TCP_Diagnostic	m_Diagnostics010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics010\Hardware\m_Diagnostics010.exe	2.5	2010/07/20 19:18:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:22:33	10526	TELM-06	1	Pass	0	0	0	10
M_TCP_Diagnostic	m_Diagnostics011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics011\Hardware\m_Diagnostics011.exe	2.4	2009/12/23 13:29:38	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:18:19	29865	TELM-06	1	Pass	0	0	0	19
M_TCP_Diagnostic	m_Diagnostics012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics012\Hardware\m_Diagnostics012.exe	2.3	2010/04/16 14:20:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 13:14:29	17024	TELM-06	1	Pass	0	0	0	14
M_TCP_Diagnostic	m_Diagnostics013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics013\Hardware\m_Diagnostics013.exe	2.5	2010/06/08 17:29:50	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 13:11:45	20597	TELM-06	1	Pass	0	0	0	15
M_TCP_Diagnostic	m_Diagnostics014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics014\Hardware\m_Diagnostics014.exe	2.4	2010/03/14 17:27:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:08:14	17780	TELM-06	1	Pass	0	0	0	18
M_TCP_Diagnostic	m_Diagnostics015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics015\Hardware\m_Diagnostics015.exe	2.4	2010/01/19 17:35:47	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 13:04:32	25458	TELM-06	1	Pass	0	0	0	29
M_TCP_Diagnostic	m_Diagnostics016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics016\Hardware\m_Diagnostics016.exe	2.0	2009/09/22 17:09:46	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 13:00:23	5815	TELM-06	1	Pass	0	0	0	1
M_TCP_Diagnostic	m_Diagnostics017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics017\Hardware\m_Diagnostics017.exe	2.3	2009/11/23 20:15:57	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:56:59	24770	TELM-06	1	Pass	0	0	0	32
M_TCP_Diagnostic	m_Diagnostics018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics018\Hardware\m_Diagnostics018.exe	2.2	2009/12/23 13:30:36	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:54:37	11838	TELM-06	1	Pass	0	0	0	9
M_TCP_Diagnostic	m_Diagnostics019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics019\Hardware\m_Diagnostics019.exe	2.1	2009/10/13 12:56:53	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:52:39	7457	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics020\Hardware\m_Diagnostics020.exe	2.2	2009/11/23 20:16:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:49:31	9689	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics021\Hardware\m_Diagnostics021.exe	2.1	2009/10/13 13:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:46:47	6051	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics022\Hardware\m_Diagnostics022.exe	2.4	2010/02/03 17:09:03	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:44:47	12592	TELM-06	1	Pass	0	0	0	4
M_TCP_Diagnostic	m_Diagnostics023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics023\Hardware\m_Diagnostics023.exe	2.2	2010/02/03 14:24:26	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:42:28	39098	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics024\Hardware\m_Diagnostics024.exe	2.5	2010/02/03 14:04:47	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:40:30	15496	TELM-06	1	Pass	0	0	0	20
M_TCP_Diagnostic	m_Diagnostics026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics026\Hardware\m_Diagnostics026.exe	2.1	2009/10/15 19:32:14	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:38:06	6051	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics027\Hardware\m_Diagnostics027.exe	2.2	2009/11/02 13:03:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:33:07	3133	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics028\Hardware\m_Diagnostics028.exe	2.4	2009/11/16 20:37:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:30:39	11433	TELM-06	1	Pass	0	0	0	11
M_TCP_Diagnostic	m_Diagnostics029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics029\Hardware\m_Diagnostics029.exe	2.1	2009/10/15 19:46:18	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 12:27:43	2846	TELM-06	1	Pass	0	0	0	2
M_TCP_Diagnostic	m_Diagnostics030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics030\Hardware\m_Diagnostics030.exe	2.5	2010/07/20 18:09:08	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 12:26:25	7303	TELM-06	1	Pass	0	0	0	277
M_TCP_Diagnostic	m_Diagnostics031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics031\Hardware\m_Diagnostics031.exe	2.2	2009/11/11 14:33:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:24:48	10456	TELM-06	1	Pass	0	0	0	3
M_TCP_Diagnostic	m_Diagnostics032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics032\Hardware\m_Diagnostics032.exe	2.3	2009/12/23 13:31:13	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:21:24	9594	TELM-06	1	Pass	0	0	0	5
M_TCP_Diagnostic	m_Diagnostics034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics034\Hardware\m_Diagnostics034.exe	2.4	2009/11/16 20:30:31	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:19:46	9624	TELM-06	1	Pass	0	0	0	2

M_TCP_Diagnostic	m_Diagnostics035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics035\Hardware\m_Diagnostics035.exe	2.2	2009/12/23 13:31:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 12:17:23	7826	TELM-06	1	Pass	0	0	0	7
M_TCP_Diagnostic	m_Diagnostics036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics036\Hardware\m_Diagnostics036.exe	2.3	2010/03/13 01:44:14	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:15:44	8361	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics037\Hardware\m_Diagnostics037.exe	2.4	2010/03/13 01:23:35	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 12:13:44	7007	TELM-06	1	Pass	0	0	0	8
M_TCP_Diagnostic	m_Diagnostics038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Diagnostic\m_Diagnostics038\Hardware\m_Diagnostics038.exe	2.3	2010/06/07 20:16:48	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 12:11:59	48830	TELM-06	1	Pass	0	0	0	18
M_TCP_Discover	m_DiscoverFull001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull001\Hardware\m_DiscoverFull001.exe	2.8	2010/05/21 19:22:39	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 04:12:00	44414	TELM-06	1	Pass	0	0	0	150
M_TCP_Discover	m_DiscoverFull002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull002\Hardware\m_DiscoverFull002.exe	2.6	2010/03/19 18:02:47	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 04:08:11	144811	TELM-06	1	Pass	0	0	0	1815
M_TCP_Discover	m_DiscoverFull003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull003\Hardware\m_DiscoverFull003.exe	2.2	2010/02/25 20:00:01	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:05:36	12114	TELM-06	1	Pass	0	0	0	58
M_TCP_Discover	m_DiscoverFull004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull004\Hardware\m_DiscoverFull004.exe	2.4	2010/02/05 21:19:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 12:32:24	270981	TELM-06	1	Pass	0	0	0	3713
M_TCP_Discover	m_DiscoverFull005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull005\Hardware\m_DiscoverFull005.exe	2.2	2009/10/29 19:18:52	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 04:04:14	17528	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull006\Hardware\m_DiscoverFull006.exe	2.3	2010/02/05 21:19:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 09:49:17	130855	TELM-06	1	Pass	0	0	0	1727
M_TCP_Discover	m_DiscoverFull007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull007\Hardware\m_DiscoverFull007.exe	2.7	2009/12/14 21:43:07	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 04:02:21	13874	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull009\Hardware\m_DiscoverFull009.exe	2.3	2010/01/21 21:02:21	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 04:01:08	11016	TELM-06	1	Pass	0	0	0	65
M_TCP_Discover	m_DiscoverFull010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull010\Hardware\m_DiscoverFull010.exe	2.4	2010/05/12 17:07:31	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:59:52	8058	TELM-06	1	Pass	0	0	0	121
M_TCP_Discover	m_DiscoverFull011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull011\Hardware\m_DiscoverFull011.exe	2.3	2009/11/02 22:03:01	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:57:22	56691	TELM-06	1	Pass	0	0	0	9
M_TCP_Discover	m_DiscoverFull012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull012\Hardware\m_DiscoverFull012.exe	2.8	2010/06/29 12:07:18	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 18:05:12	366874	TELM-06	1	Pass	0	0	0	33
M_TCP_Discover	m_DiscoverFull013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull013\Hardware\m_DiscoverFull013.exe	2.2	2010/03/18 16:03:09	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:55:09	42748	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull014\Hardware\m_DiscoverFull014.exe	2.2	2010/03/18 16:03:45	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:52:42	24557	TELM-06	1	Pass	0	0	0	15
M_TCP_Discover	m_DiscoverFull015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull015\Hardware\m_DiscoverFull015.exe	2.1	2009/11/23 16:09:49	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:51:14	7757	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull016\Hardware\m_DiscoverFull016.exe	2.1	2009/11/03 20:58:07	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:49:06	8212	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull017\Hardware\m_DiscoverFull017.exe	2.1	2009/11/03 20:58:47	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:47:46	12098	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull022\Hardware\m_DiscoverFull022.exe	2.1	2009/11/23 16:11:12	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:46:25	8344	TELM-06	1	Pass	0	0	0	6
M_TCP_Discover	m_DiscoverFull023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull023\Hardware\m_DiscoverFull023.exe	2.1	2009/11/23 16:11:42	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:45:05	11433	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull024\Hardware\m_DiscoverFull024.exe	2.1	2009/11/23 16:12:04	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:44:14	12600	TELM-06	1	Pass	0	0	0	7
M_TCP_Discover	m_DiscoverFull025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull025\Hardware\m_DiscoverFull025.exe	2.1	2009/11/23 16:12:32	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:42:25	9182	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull029\Hardware\m_DiscoverFull029.exe	2.0	2009/11/13 21:27:59	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:37:34	12324	TELM-06	1	Pass	0	0	0	3
M_TCP_Discover	m_DiscoverFull030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull030\Hardware\m_DiscoverFull030.exe	2.3	2010/05/28 11:39:20	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:33:49	19166	TELM-06	1	Pass	0	0	0	10
M_TCP_Discover	m_DiscoverFull031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull031\Hardware\m_DiscoverFull031.exe	2.3	2010/05/28 12:38:20	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:32:06	35290	TELM-06	1	Pass	0	0	0	22
M_TCP_Discover	m_DiscoverFull032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull032\Hardware\m_DiscoverFull032.exe	2.0	2009/11/13 21:29:43	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:29:41	9334	TELM-06	1	Pass	0	0	0	12
M_TCP_Discover	m_DiscoverFull033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull033\Hardware\m_DiscoverFull033.exe	2.0	2009/11/13 21:30:18	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 03:27:15	9521	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull034\Hardware\m_DiscoverFull034.exe	2.2	2010/03/18 16:04:21	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:26:14	11228	TELM-06	1	Pass	0	0	0	2
M_TCP_Discover	m_DiscoverFull035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull035\Hardware\m_DiscoverFull035.exe	2.2	2010/02/09 20:53:53	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 01:05:58	43617	TELM-06	1	Pass	0	0	0	31
M_TCP_Discover	m_DiscoverFull036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull036\Hardware\m_DiscoverFull036.exe	2.2	2010/02/11 14:31:48	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 00:46:26	22037	TELM-06	1	Pass	0	0	0	38
M_TCP_Discover	m_DiscoverFull037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull037\Hardware\m_DiscoverFull037.exe	2.4	2010/02/03 21:18:36	2.4	Up-to-date	Baseline 17_1	None	09/11/2010 04:15:26	35253	TELM-06	1	Pass	0	0	0	4
M_TCP_Discover	m_DiscoverFull038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Discover\m_DiscoverFull038\Hardware\m_DiscoverFull038.exe	2.1	2009/12/01 20:49:48	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 03:22:08	141703	TELM-06	1	Pass	0	0	0	7
M_TCP_EEPROM	m_Eeprom001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom001\Hardware\m_Eeprom001.exe	2.3	2009/12/23 16:23:59	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 19:40:31	11123	TELM-06	1	Pass	0	0	0	22
M_TCP_EEPROM	m_Eeprom002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom002\Hardware\m_Eeprom002.exe	2.3	2009/12/18 21:07:43	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 13:10:08	10842	TELM-06	1	Pass	0	0	0	938
M_TCP_EEPROM	m_Eeprom003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom003\Hardware\m_Eeprom003.exe	2.2	2010/01/05 18:03:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:07:03	16695	TELM-06	1	Pass	0	0	0	702
M_TCP_EEPROM	m_Eeprom004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom004\Hardware\m_Eeprom004.exe	2.2	2010/01/25 19:26:31	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:02:12	109008	TELM-06	1	Pass	0	0	0	583
M_TCP_EEPROM	m_Eeprom005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom005\Hardware\m_Eeprom005.exe	2.2	2010/01/19 15:32:05	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:57:33	171734	TELM-06	1	Pass	0	0	0	704
M_TCP_EEPROM	m_Eeprom006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom006\Hardware\m_Eeprom006.exe	2.2	2010/05/28 12:55:48	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:55:09	46951	TELM-06	1	Pass	0	0	0	6
M_TCP_EEPROM	m_Eeprom007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_EEPROM\m_Eeprom007\Hardware\m_Eeprom007.exe	2.1	2010/02/18 18:38:33	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:53:16	5104	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency001\Hardware\m_Emergency001.exe	2.3	2009/11/03 21:54:33	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:13:39	9430	TELM-06	1	Pass	0	0	0	100
M_TCP_Emergency	m_Emergency002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency002\Hardware\m_Emergency002.exe	2.5	2010/06/25 20:19:36	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:11:31	12965	TELM-06	1	Pass	0	0	0	45
M_TCP_Emergency	m_Emergency003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency003\Hardware\m_Emergency003.exe	2.6	2010/06/29 14:19:35	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:08:54	28206	TELM-06	1	Pass	0	0	0	2564
M_TCP_Emergency	m_Emergency004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency004\Hardware\m_Emergency004.exe	2.3	2010/02/05 21:22:43	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:06:34	18963	TELM-06	1	Pass	0	0	0	4
M_TCP_Emergency	m_Emergency005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency005\Hardware\m_Emergency005.exe	2.2	2009/11/10 00:31:59	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:03:35	8204	TELM-06	1	Pass	0	0	0	5
M_TCP_Emergency	m_Emergency006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency006\Hardware\m_Emergency006.exe	2.1	2009/10/16 16:44:07	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:00:21	9255	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency007\Hardware\m_Emergency007.exe	2.1	2009/10/16 16:44:28	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:57:19	9354	TELM-06	1	Pass	0	0	0	33
M_TCP_Emergency	m_Emergency009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency009\Hardware\m_Emergency009.exe	2.1	2009/10/08 19:54:42	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:54:59	6017	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency010\Hardware\m_Emergency010.exe	2.2	2010/06/08 15:56:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:53:28	13309	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency011\Hardware\m_Emergency011.exe	2.0	2009/10/21 16:25:23	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:52:19	6741	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency012\Hardware\m_Emergency012.exe	2.1	2009/11/11 13:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:51:11	8170	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency013\Hardware\m_Emergency013.exe	2.1	2009/11/11 14:02:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:49:17	11509	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency014\Hardware\m_Emergency014.exe	2.3	2009/12/23 13:33:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:48:09	8239	TELM-06	1	Pass	0	0	0	6
M_TCP_Emergency	m_Emergency015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency015\Hardware\m_Emergency015.exe	2.3	2010/06/08 15:56:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:45:53	8969	TELM-06	1	Pass	0	0	0	8
M_TCP_Emergency	m_Emergency016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency016\Hardware\m_Emergency016.exe	2.2	2009/12/21 21:03:44	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:44:30	12430	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency017\Hardware\m_Emergency017.exe	2.2	2009/11/16 12:31:37	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:43:10	9781	TELM-06	1	Pass	0	0	0	21
M_TCP_Emergency	m_Emergency018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency018\Hardware\m_Emergency018.exe	2.3	2010/06/29 15:32:40	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:42:02	8066	TELM-06	1	Pass	0	0	0	22
M_TCP_Emergency	m_Emergency019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency019\Hardware\m_Emergency019.exe	2.2	2010/01/07 19:28:32	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:40:46	11097	TELM-06	1	Pass	0	0	0	17
M_TCP_Emergency	m_Emergency020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency020\Hardware\m_Emergency020.exe	2.6	2010/03/18 17:47:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:39:04	11415	TELM-06	1	Pass	0	0	0	26
M_TCP_Emergency	m_Emergency021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency021\Hardware\m_Emergency021.exe	2.3	2009/12/21 19:27:03	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:38:02	10471	TELM-06	1	Pass	0	0	0	83
M_TCP_Emergency	m_Emergency022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency022\Hardware\m_Emergency022.exe	2.2	2010/01/07 18:54:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:32:45	10198	TELM-06	1	Pass	0	0	0	82
M_TCP_Emergency	m_Emergency038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency038\Hardware\m_Emergency038.exe	2.2	2010/01/07 19:42:11	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:31:42	10439	TELM-06	1	Pass	0	0	0	14
M_TCP_Emergency	m_Emergency039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency039\Hardware\m_Emergency039.exe	2.2	2010/02/03 17:51:34	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:27:55	15424	TELM-06	1	Pass	0	0	0	2
M_TCP_Emergency	m_Emergency040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency040\Hardware\m_Emergency040.exe	2.0	2009/09/29 14:53:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:25:09	13843	TELM-06	1	Pass	0	0	0	7
M_TCP_Emergency	m_Emergency041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency041\Hardware\m_Emergency041.exe	2.0	2009/09/29 14:54:05	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:24:03	4943	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency042\Hardware\m_Emergency042.exe	2.0	2009/09/29 14:54:52	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 22:22:59	4920	TELM-06	1	Pass	0	0	0	3
M_TCP_Emergency	m_Emergency043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency043\Hardware\m_Emergency043.exe	2.4	2010/04/21 17:13:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:21:09	10403	TELM-06	1	Pass	0	0	0	9
M_TCP_Emergency	m_Emergency044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency044\Hardware\m_Emergency044.exe	2.5	2009/11/10 14:19:43	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 03:35:06	37644	TELM-06	1	Pass	0	0	0	18
M_TCP_Emergency	m_Emergency045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Emergency\m_Emergency045\Hardware\m_Emergency045.exe	2.3	2010/05/11 12:18:39	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:19:43	10842	TELM-06	1	Pass	0	0	0	16
M_TCP_General	m_General003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General003\Hardware\m_General003.exe	2.2	2009/10/13 14:50:43	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:38:44	67693	TELM-06	1	Pass	0	0	0	44646
M_TCP_General	m_General005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General005\Hardware\m_General005.exe	2.4	2009/12/01 19:02:30	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:35:33	67800	TELM-06	1	Pass	0	0	0	16772
M_TCP_General	m_General015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General015\Hardware\m_General015.exe	2.3	2009/12/01 19:02:59	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:34:00	36153	TELM-06	1	Pass	0	0	0	129327
M_TCP_General	m_General016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General016\Hardware\m_General016.exe	2.4	2010/03/29 10:43:22	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:31:42	8728	TELM-06	1	Pass	0	0	0	2
M_TCP_General	m_General017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General017\Hardware\m_General017.exe	2.3	2009/12/01 19:03:26	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:30:30	15763	TELM-06	1	Pass	0	0	0	10
M_TCP_General	m_General018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General018\Hardware\m_General018.exe	2.2	2010/02/03 18:07:07	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:27:40	29823	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General027\Hardware\m_General027.exe	2.7	2010/06/07 11:39:47	2.7	Up-to-date	Baseline 17_1	None	09/10/2010 19:51:55	802193	TELM-06	1	Pass	0	0	0	21
M_TCP_General	m_General028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General028\Hardware\m_General028.exe	2.5	2010/06/08 11:48:52	2.5	Up-to-date	Baseline 17_1	None	09/11/2010 02:02:24	635677	TELM-06	1	Pass	0	0	0	9
M_TCP_General	m_General029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General029\Hardware\m_General029.exe	2.3	2010/06/07 12:27:26	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 01:38:06	274149	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General030\Hardware\m_General030.exe	2.1	2010/06/07 12:30:50	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 10:26:16	17289	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General031\Hardware\m_General031.exe	2.5	2010/05/11 12:33:53	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 10:20:19	57916	TELM-06	1	Pass	0	0	0	12
M_TCP_General	m_General032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General032\Hardware\m_General032.exe	2.2	2010/02/05 18:56:06	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:16:31	29831	TELM-06	1	Pass	0	0	0	11
M_TCP_General	m_General033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General033\Hardware\m_General033.exe	2.0	2010/02/04 17:42:57	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 10:12:41	20714	TELM-06	1	Pass	0	0	0	4
M_TCP_General	m_General034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_General\m_General034\Hardware\m_General034.exe	2.0	2010/01/22 22:45:34	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 10:09:38	30370	TELM-06	1	Pass	0	0	0	2
M_TCP_Initialize	m_Initialize001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize001\Hardware\m_Initialize001.exe	2.8	2010/05/12 12:34:35	2.8	Up-to-date	Baseline 17_1	None	09/09/2010 21:58:58	10475	TELM-06	1	Pass	0	0	0	9
M_TCP_Initialize	m_Initialize002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize002\Hardware\m_Initialize002.exe	2.2	2009/10/20 12:16:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 21:57:20	19873	TELM-06	1	Pass	0	0	0	11
M_TCP_Initialize	m_Initialize003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize003\Hardware\m_Initialize003.exe	2.4	2010/06/08 15:57:16	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:53:35	24143	TELM-06	1	Pass	0	0	0	6
M_TCP_Initialize	m_Initialize004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Initialize\m_Initialize004\Hardware\m_Initialize004.exe	2.4	2010/06/08 15:57:44	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 21:50:31	40125	TELM-06	1	Pass	0	0	0	30
M_TCP_MedicalEvent	m_MedicalEvent001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent001\Hardware\m_MedicalEvent001.exe	2.4	2010/01/18 15:42:53	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 10:43:36	12482	TELM-06	1	Pass	0	0	0	36
M_TCP_MedicalEvent	m_MedicalEvent002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent002\Hardware\m_MedicalEvent002.exe	2.6	2010/02/09 19:03:39	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 10:31:42	73239	TELM-06	1	Pass	0	0	0	23
M_TCP_MedicalEvent	m_MedicalEvent003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent003\Hardware\m_MedicalEvent003.exe	2.4	2010/03/11 21:16:02	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 10:14:17	18974	TELM-06	1	Pass	0	0	0	24
M_TCP_MedicalEvent	m_MedicalEvent004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent004\Hardware\m_MedicalEvent004.exe	2.4	2010/01/18 15:44:34	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 10:00:25	12110	TELM-06	1	Pass	0	0	0	12
M_TCP_MedicalEvent	m_MedicalEvent008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent008\Hardware\m_MedicalEvent008.exe	2.4	2010/01/18 15:44:59	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 09:59:29	8461	TELM-06	1	Pass	0	0	0	6
M_TCP_MedicalEvent	m_MedicalEvent009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent009\Hardware\m_MedicalEvent009.exe	2.2	2010/01/18 15:46:20	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 09:39:57	7719	TELM-06	1	Pass	0	0	0	4
M_TCP_MedicalEvent	m_MedicalEvent010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent010\Hardware\m_MedicalEvent010.exe	2.5	2010/01/27 14:30:44	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 09:33:47	16634	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent011\Hardware\m_MedicalEvent011.exe	2.2	2010/01/18 15:47:08	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 09:16:04	15146	TELM-06	1	Pass	0	0	0	11
M_TCP_MedicalEvent	m_MedicalEvent012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent012\Hardware\m_MedicalEvent012.exe	2.3	2010/01/18 15:47:55	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 09:00:55	6177	TELM-06	1	Pass	0	0	0	3
M_TCP_MedicalEvent	m_MedicalEvent014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent014\Hardware\m_MedicalEvent014.exe	2.5	2010/09/08 13:20:31	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 08:50:14	15627	TELM-06	1	Pass	0	0	0	18
M_TCP_MedicalEvent	m_MedicalEvent015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent015\Hardware\m_MedicalEvent015.exe	2.9	2010/09/07 20:16:18	2.9	Up-to-date	Baseline 17_1	None	09/09/2010 09:47:10	56616	TELM-06	1	Pass	0	0	0	152
M_TCP_MedicalEvent	m_MedicalEvent016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent016\Hardware\m_MedicalEvent016.exe	2.9	2010/05/12 18:44:37	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 18:56:18	283901	TELM-06	1	Pass	0	0	0	375
M_TCP_MedicalEvent	m_MedicalEvent017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent017\Hardware\m_MedicalEvent017.exe	2.8	2010/09/07 20:17:30	2.8	Up-to-date	Baseline 17_1	None	09/10/2010 23:35:54	612345	TELM-06	1	Pass	0	0	0	13
M_TCP_MedicalEvent	m_MedicalEvent018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent018\Hardware\m_MedicalEvent018.exe	2.5	2010/01/18 15:50:21	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 18:25:47	272252	TELM-06	1	Pass	0	0	0	4627
M_TCP_MedicalEvent	m_MedicalEvent019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent019\Hardware\m_MedicalEvent019.exe	2.5	2010/01/18 15:50:59	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 08:40:01	7463	TELM-06	1	Pass	0	0	0	5
M_TCP_MedicalEvent	m_MedicalEvent020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MedicalEvent\m_MedicalEvent020\Hardware\m_MedicalEvent020.exe	2.3	2010/01/18 15:51:30	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 08:28:48	40607	TELM-06	1	Pass	0	0	0	9
M_TCP_MemoryTest	m_MemoryTest001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest001\Hardware\m_MemoryTest001.exe	2.7	2010/06/18 12:28:23	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 00:02:12	33003	TELM-06	1	Pass	0	0	0	33
M_TCP_MemoryTest	m_MemoryTest002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest002\Hardware\m_MemoryTest002.exe	2.4	2010/05/05 19:48:21	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 10:45:38	240621	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest003\Hardware\m_MemoryTest003.exe	2.3	2010/05/05 20:08:58	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 14:11:38	479589	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest004\Hardware\m_MemoryTest004.exe	2.3	2010/05/05 20:09:49	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:40:47	240203	TELM-06	1	Pass	0	0	0	75
M_TCP_MemoryTest	m_MemoryTest005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_MemoryTest\m_MemoryTest005\Hardware\m_MemoryTest005.exe	2.3	2010/05/05 20:10:26	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 13:38:15	478704	TELM-06	1	Pass	0	0	0	75
M_TCP_Network	m_Network001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network001\Hardware\m_Network001.exe	2.6	2010/05/21 15:45:43	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 21:49:09	24439	TELM-06	1	Pass	0	0	0	1461
M_TCP_Network	m_Network002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network002\Hardware\m_Network002.exe	2.6	2010/05/12 12:01:17	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 21:07:15	34753	TELM-06	1	Pass	0	0	0	82

M_TCP_Network	m_Network003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network003\Hardware\m_Network003.exe	2.1	2009/12/16 22:27:06	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:44:18	64790	TELM-06	1	Pass	0	0	0	86
M_TCP_Network	m_Network004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network004\Hardware\m_Network004.exe	2.1	2010/04/21 17:14:24	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 21:41:58	2961	TELM-06	1	Pass	0	0	0	6
M_TCP_Network	m_Network005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network005\Hardware\m_Network005.exe	2.3	2010/05/12 12:02:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:37:41	58135	TELM-06	1	Pass	0	0	0	50
M_TCP_Network	m_Network006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network006\Hardware\m_Network006.exe	2.5	2010/05/21 16:49:56	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:35:15	15254	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network007\Hardware\m_Network007.exe	2.5	2010/05/21 17:09:04	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:29:23	15426	TELM-06	1	Pass	0	0	0	1421
M_TCP_Network	m_Network008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network008\Hardware\m_Network008.exe	2.5	2010/05/21 17:13:03	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:25:31	10457	TELM-06	1	Pass	0	0	0	1380
M_TCP_Network	m_Network009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network009\Hardware\m_Network009.exe	2.5	2010/05/21 17:21:30	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:22:10	10415	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network010\Hardware\m_Network010.exe	2.5	2010/05/21 17:34:27	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:19:23	10276	TELM-06	1	Pass	0	0	0	1381
M_TCP_Network	m_Network011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network011\Hardware\m_Network011.exe	2.3	2010/05/12 12:03:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:12:49	209522	TELM-06	1	Pass	0	0	0	122
M_TCP_Network	m_Network013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network013\Hardware\m_Network013.exe	2.5	2010/07/01 13:00:48	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 21:07:31	141819	TELM-06	1	Pass	0	0	0	392
M_TCP_Network	m_Network014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network014\Hardware\m_Network014.exe	2.3	2010/05/12 12:06:29	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:04:16	26650	TELM-06	1	Pass	0	0	0	12
M_TCP_Network	m_Network015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network015\Hardware\m_Network015.exe	2.3	2010/05/24 13:53:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 21:02:10	35725	TELM-06	1	Pass	0	0	0	25
M_TCP_Network	m_Network016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network016\Hardware\m_Network016.exe	2.1	2010/05/12 12:09:43	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:59:26	55337	TELM-06	1	Pass	0	0	0	262
M_TCP_Network	m_Network017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network017\Hardware\m_Network017.exe	2.4	2010/06/07 13:19:37	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 20:58:10	12750	TELM-06	1	Pass	0	0	0	41
M_TCP_Network	m_Network018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network018\Hardware\m_Network018.exe	2.5	2010/06/07 13:14:52	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 20:54:31	21581	TELM-06	1	Pass	0	0	0	11
M_TCP_Network	m_Network019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network019\Hardware\m_Network019.exe	2.2	2010/05/12 12:12:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:49:45	56605	TELM-06	1	Pass	0	0	0	51
M_TCP_Network	m_Network020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network020\Hardware\m_Network020.exe	2.2	2010/05/12 12:14:53	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:46:49	60302	TELM-06	1	Pass	0	0	0	67
M_TCP_Network	m_Network021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network021\Hardware\m_Network021.exe	2.2	2010/05/12 12:16:17	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:43:26	60362	TELM-06	1	Pass	0	0	0	133
M_TCP_Network	m_Network022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network022\Hardware\m_Network022.exe	2.2	2010/05/12 12:17:24	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:38:40	60371	TELM-06	1	Pass	0	0	0	175
M_TCP_Network	m_Network023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network023\Hardware\m_Network023.exe	2.1	2010/05/12 12:18:23	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:36:32	19623	TELM-06	1	Pass	0	0	0	454
M_TCP_Network	m_Network024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network024\Hardware\m_Network024.exe	2.2	2010/05/24 14:41:01	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:33:56	47546	TELM-06	1	Pass	0	0	0	46
M_TCP_Network	m_Network025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network025\Hardware\m_Network025.exe	2.1	2010/05/12 12:20:01	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:31:33	19600	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network026\Hardware\m_Network026.exe	2.3	2010/05/12 12:20:56	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:26:32	20481	TELM-06	1	Pass	0	0	0	178
M_TCP_Network	m_Network027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network027\Hardware\m_Network027.exe	2.1	2010/05/12 12:22:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:24:16	20988	TELM-06	1	Pass	0	0	0	7
M_TCP_Network	m_Network028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network028\Hardware\m_Network028.exe	2.1	2010/06/07 13:22:27	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:21:46	14278	TELM-06	1	Pass	0	0	0	22
M_TCP_Network	m_Network029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Network\m_Network029\Hardware\m_Network029.exe	2.4	2010/02/09 21:21:17	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 15:31:52	271665	TELM-06	1	Pass	0	0	0	48
M_TCP_Open	m_Open001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open001\Hardware\m_Open001.exe	2.11	2010/05/20 13:50:26	2.11	Up-to-date	Baseline 17_1	None	09/09/2010 20:19:10	31634	TELM-06	1	Pass	0	0	0	335
M_TCP_Open	m_Open002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open002\Hardware\m_Open002.exe	2.3	2010/01/18 15:52:02	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:15:58	18560	TELM-06	1	Pass	0	0	0	31
M_TCP_Open	m_Open003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open003\Hardware\m_Open003.exe	2.0	2009/10/15 19:09:20	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 20:14:17	6273	TELM-06	1	Pass	0	0	0	3
M_TCP_Open	m_Open004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open004\Hardware\m_Open004.exe	2.6	2010/06/22 20:01:21	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:12:47	13119	TELM-06	1	Pass	0	0	0	21
M_TCP_Open	m_Open005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open005\Hardware\m_Open005.exe	2.3	2010/06/22 17:40:45	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 20:10:54	17053	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open006\Hardware\m_Open006.exe	2.2	2010/03/15 14:26:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 20:08:54	21301	TELM-06	1	Pass	0	0	0	1173
M_TCP_Open	m_Open007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open007\Hardware\m_Open007.exe	2.7	2010/03/15 14:46:20	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 20:06:48	17486	TELM-06	1	Pass	0	0	0	1181
M_TCP_Open	m_Open008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open008\Hardware\m_Open008.exe	2.1	2009/10/30 14:23:08	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 20:04:58	7731	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open009\Hardware\m_Open009.exe	2.7	2010/06/16 00:36:52	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 20:03:18	12412	TELM-06	1	Pass	0	0	0	1268
M_TCP_Open	m_Open010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open010\Hardware\m_Open010.exe	2.6	2010/03/15 15:05:50	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 20:01:23	17360	TELM-06	1	Pass	0	0	0	3965
M_TCP_Open	m_Open011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open011\Hardware\m_Open011.exe	2.2	2010/02/10 13:31:30	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:59:13	6291	TELM-06	1	Pass	0	0	0	4
M_TCP_Open	m_Open012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open012\Hardware\m_Open012.exe	2.4	2010/02/26 17:01:14	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:57:41	15407	TELM-06	1	Pass	0	0	0	10
M_TCP_Open	m_Open013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open013\Hardware\m_Open013.exe	2.5	2010/02/10 13:34:06	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 19:56:22	9410	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open014\Hardware\m_Open014.exe	2.6	2010/02/10 13:35:39	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 19:53:56	8622	TELM-06	1	Pass	0	0	0	101
M_TCP_Open	m_Open015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open015\Hardware\m_Open015.exe	2.0	2010/01/28 19:00:26	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:52:17	8925	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open030\Hardware\m_Open030.exe	2.1	2010/05/10 13:57:56	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 03:32:09	58182	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open031\Hardware\m_Open031.exe	2.7	2009/11/19 21:48:20	2.7	Up-to-date	Baseline 17_1	None	09/11/2010 03:29:37	63777	TELM-06	1	Pass	0	0	0	13
M_TCP_Open	m_Open032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open032\Hardware\m_Open032.exe	2.3	2010/02/03 22:17:27	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 00:33:43	32344	TELM-06	1	Pass	0	0	0	18
M_TCP_Open	m_Open033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open033\Hardware\m_Open033.exe	2.3	2010/02/03 22:32:54	2.3	Up-to-date	Baseline 17_1	None	09/11/2010 00:31:47	9989	TELM-06	1	Pass	0	0	0	5
M_TCP_Open	m_Open034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open034\Hardware\m_Open034.exe	2.2	2010/02/17 14:19:09	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:50:07	57031	TELM-06	1	Pass	0	0	0	42
M_TCP_Open	m_Open035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open035\Hardware\m_Open035.exe	2.2	2010/02/17 14:10:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 19:47:45	9933	TELM-06	1	Pass	0	0	0	32
M_TCP_Open	m_Open036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open036\Hardware\m_Open036.exe	2.1	2009/10/21 15:52:46	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 03:26:51	36257	TELM-06	1	Pass	0	0	0	16
M_TCP_Open	m_Open037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open037\Hardware\m_Open037.exe	2.2	2009/10/21 15:51:07	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 03:14:24	31234	TELM-06	1	Pass	0	0	0	9
M_TCP_Open	m_Open038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open038\Hardware\m_Open038.exe	2.7	2010/02/17 14:00:40	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 19:45:23	78354	TELM-06	1	Pass	0	0	0	46
M_TCP_Open	m_Open039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open039\Hardware\m_Open039.exe	2.0	2009/11/09 14:20:44	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 19:43:33	9120	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open040\Hardware\m_Open040.exe	2.1	2010/02/01 19:45:37	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 19:42:09	17308	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open041\Hardware\m_Open041.exe	2.0	2009/11/09 14:21:53	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 15:07:52	513732	TELM-06	1	Pass	0	0	0	1
M_TCP_Open	m_Open042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Open\m_Open042\Hardware\m_Open042.exe	2.4	2010/07/02 12:38:07	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 19:39:31	16075	TELM-06	1	Pass	0	0	0	68
M_TCP_Polling	m_Polling016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling016\Hardware\m_Polling016.exe	2.0	2009/09/23 19:25:05	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 00:37:43	49144	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling017\Hardware\m_Polling017.exe	2.1	2010/01/26 20:04:46	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 09:46:15	6243	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling018\Hardware\m_Polling018.exe	2.1	2010/01/26 20:05:23	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:34:29	81418	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling024\Hardware\m_Polling024.exe	2.5	2010/01/26 20:05:45	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 00:30:26	78557	TELM-06	1	Pass	0	0	0	10
M_TCP_Polling	m_Polling025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling025\Hardware\m_Polling025.exe	2.1	2009/11/24 20:51:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:27:50	46108	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling026\Hardware\m_Polling026.exe	2.2	2009/12/03 14:59:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:21:56	19475	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling040\Hardware\m_Polling040.exe	2.1	2010/01/26 20:06:14	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:18:30	6364	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling045\Hardware\m_Polling045.exe	2.1	2010/01/26 20:06:42	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:14:57	36048	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling046\Hardware\m_Polling046.exe	2.6	2010/04/16 14:21:47	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 00:11:41	47316	TELM-06	1	Pass	0	0	0	38
M_TCP_Polling	m_Polling047	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling047\Hardware\m_Polling047.exe	2.1	2010/01/26 20:07:28	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:08:28	36162	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling048	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling048\Hardware\m_Polling048.exe	2.3	2009/11/24 20:53:00	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 00:05:24	10042	TELM-06	1	Pass	0	0	0	3
M_TCP_Polling	m_Polling049	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling049\Hardware\m_Polling049.exe	2.4	2010/03/10 19:44:21	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:02:21	46205	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling050	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling050\Hardware\m_Polling050.exe	2.6	2010/06/29 17:44:57	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:58:12	22806	TELM-06	1	Pass	0	0	0	5
M_TCP_Polling	m_Polling065	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling065\Hardware\m_Polling065.exe	2.6	2010/06/09 18:52:38	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:54:25	42154	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling066	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling066\Hardware\m_Polling066.exe	2.3	2009/11/09 22:57:18	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:52:11	52677	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling067	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling067\Hardware\m_Polling067.exe	2.1	2009/10/22 14:28:02	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:50:04	84845	TELM-06	1	Pass	0	0	0	11
M_TCP_Polling	m_Polling068	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling068\Hardware\m_Polling068.exe	2.3	2009/11/05 18:23:27	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:48:53	7137	TELM-06	1	Pass	0	0	0	8
M_TCP_Polling	m_Polling076	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling076\Hardware\m_Polling076.exe	2.1	2009/11/03 15:55:29	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 23:44:12	10040	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling077	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling077\Hardware\m_Polling077.exe	2.1	2009/11/05 18:34:39	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 22:26:15	527433	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling080	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling080\Hardware\m_Polling080.exe	2.2	2009/11/05 18:24:04	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 21:57:47	277401	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling081	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling081\Hardware\m_Polling081.exe	2.3	2010/05/27 14:19:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:41:17	24095	TELM-06	1	Pass	0	0	0	4
M_TCP_Polling	m_Polling082	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling082\Hardware\m_Polling082.exe	2.2	2009/11/02 18:25:15	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 23:38:59	9431	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling083	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling083\Hardware\m_Polling083.exe	2.3	2009/11/02 18:45:07	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 23:36:03	19395	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling085	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling085\Hardware\m_Polling085.exe	2.1	2009/10/22 13:55:52	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 21:42:11	529316	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling088	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling088\Hardware\m_Polling088.exe	2.5	2010/05/12 17:59:19	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 23:32:16	23450	TELM-06	1	Pass	0	0	0	27
M_TCP_Polling	m_Polling089	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling089\Hardware\m_Polling089.exe	2.7	2010/05/11 19:28:17	2.7	Up-to-date	Baseline 17_1	None	09/09/2010 23:25:18	97044	TELM-06	1	Pass	0	0	0	26
M_TCP_Polling	m_Polling090	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling090\Hardware\m_Polling090.exe	2.6	2010/02/15 20:02:24	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:21:08	77453	TELM-06	1	Pass	0	0	0	32
M_TCP_Polling	m_Polling091	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling091\Hardware\m_Polling091.exe	2.6	2010/02/15 20:16:12	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 23:17:55	13112	TELM-06	1	Pass	0	0	0	13
M_TCP_Polling	m_Polling092	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling092\Hardware\m_Polling092.exe	2.1	2009/11/05 18:41:35	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 21:27:53	277281	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling093	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling093\Hardware\m_Polling093.exe	2.0	2009/11/09 18:30:12	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:16:21	16845	TELM-06	1	Pass	0	0	0	1
M_TCP_Polling	m_Polling094	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling094\Hardware\m_Polling094.exe	2.0	2009/11/09 18:30:49	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 15:46:15	307357	TELM-06	1	Pass	0	0	0	2
M_TCP_Polling	m_Polling095	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Polling\m_Polling095\Hardware\m_Polling095.exe	2.0	2009/11/09 18:31:27	2.0	Up-to-date	Baseline 17_1	None	09/09/2010 23:15:09	8279	TELM-06	1	Pass	0	0	0	2
M_TCP_PowerInhibit	m_Power001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power001\Hardware\m_Power001.exe	2.2	2010/06/07 13:10:06	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:12:54	58877	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power002\Hardware\m_Power002.exe	2.2	2010/01/05 19:15:56	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:07:56	10588	TELM-06	1	Pass	0	0	0	614
M_TCP_PowerInhibit	m_Power003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power003\Hardware\m_Power003.exe	2.4	2010/06/22 14:04:45	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 01:05:08	53982	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power004\Hardware\m_Power004.exe	2.3	2010/03/22 14:26:21	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:02:12	17805	TELM-06	1	Pass	0	0	0	46
M_TCP_PowerInhibit	m_Power005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power005\Hardware\m_Power005.exe	2.6	2010/07/02 12:34:05	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 10:57:38	118219	TELM-06	1	Pass	0	0	0	310
M_TCP_PowerInhibit	m_Power006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power006\Hardware\m_Power006.exe	2.6	2010/06/22 12:27:46	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 00:57:13	193282	TELM-06	1	Pass	0	0	0	2884
M_TCP_PowerInhibit	m_Power007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power007\Hardware\m_Power007.exe	2.1	2009/12/14 16:39:57	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:55:13	19227	TELM-06	1	Pass	0	0	0	13
M_TCP_PowerInhibit	m_Power008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power008\Hardware\m_Power008.exe	2.2	2009/12/14 16:44:18	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 00:52:51	34768	TELM-06	1	Pass	0	0	0	15
M_TCP_PowerInhibit	m_Power009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power009\Hardware\m_Power009.exe	2.4	2010/07/01 15:00:10	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 00:49:28	12164	TELM-06	1	Pass	0	0	0	9
M_TCP_PowerInhibit	m_Power010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power010\Hardware\m_Power010.exe	2.1	2010/02/16 22:18:45	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:47:12	11853	TELM-06	1	Pass	0	0	0	26
M_TCP_PowerInhibit	m_Power011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_PowerInhibit\m_Power011\Hardware\m_Power011.exe	2.1	2010/03/11 19:04:17	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 00:41:54	90189	TELM-06	1	Pass	0	0	0	116
M_TCP_RawMode	m_RawMode001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode001\Hardware\m_RawMode001.exe	2.4	2010/01/26 20:08:13	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:18:44	7421	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode002\Hardware\m_RawMode002.exe	2.6	2010/01/27 13:00:04	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:17:00	9629	TELM-06	1	Pass	0	0	0	87
M_TCP_RawMode	m_RawMode003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode003\Hardware\m_RawMode003.exe	2.5	2010/01/27 13:03:49	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 22:15:08	11286	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode004\Hardware\m_RawMode004.exe	2.6	2010/01/27 13:09:25	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 22:13:26	7165	TELM-06	1	Pass	0	0	0	2
M_TCP_RawMode	m_RawMode005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode005\Hardware\m_RawMode005.exe	2.3	2010/01/26 20:10:41	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:11:37	13495	TELM-06	1	Pass	0	0	0	9
M_TCP_RawMode	m_RawMode006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode006\Hardware\m_RawMode006.exe	2.2	2010/01/26 20:11:57	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 09:44:52	9083	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode007\Hardware\m_RawMode007.exe	2.3	2010/06/09 21:03:53	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:08:50	11319	TELM-06	1	Pass	0	0	0	11
M_TCP_RawMode	m_RawMode008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode008\Hardware\m_RawMode008.exe	2.4	2010/01/27 13:14:20	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 22:07:34	16149	TELM-06	1	Pass	0	0	0	4
M_TCP_RawMode	m_RawMode009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode009\Hardware\m_RawMode009.exe	2.2	2009/12/14 16:51:58	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:05:12	15700	TELM-06	1	Pass	0	0	0	7
M_TCP_RawMode	m_RawMode010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode010\Hardware\m_RawMode010.exe	2.1	2009/10/27 17:32:20	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 22:04:08	3157	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode011\Hardware\m_RawMode011.exe	2.2	2010/02/01 16:15:38	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 22:02:22	8478	TELM-06	1	Pass	0	0	0	3
M_TCP_RawMode	m_RawMode012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_RawMode\m_RawMode012\Hardware\m_RawMode012.exe	2.3	2010/01/26 20:13:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 22:00:24	11734	TELM-06	1	Pass	0	0	0	3
M_TCP_Reset	m_Reset001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset001\Hardware\m_Reset001.exe	2.3	2010/01/06 21:25:12	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 03:20:19	9078	TELM-06	1	Pass	0	0	0	23
M_TCP_Reset	m_Reset002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Reset\m_Reset002\Hardware\m_Reset002.exe	2.4	2010/01/28 21:52:07	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:17:49	20362	TELM-06	1	Pass	0	0	0	39

M_TCP_Security	m_Security002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security002\Hardware\m_Security002.exe	2.9	2010/06/03 15:18:25	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 17:34:11	457359	TELM-06	1	Pass	0	0	0	154
M_TCP_Security	m_Security003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security003\Hardware\m_Security003.exe	2.4	2010/02/17 16:20:41	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:14:06	6849	TELM-06	1	Pass	0	0	0	73
M_TCP_Security	m_Security004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security004\Hardware\m_Security004.exe	2.4	2010/02/15 15:01:00	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:07:34	253980	TELM-06	1	Pass	0	0	0	112
M_TCP_Security	m_Security005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security005\Hardware\m_Security005.exe	2.2	2010/02/17 16:23:51	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 03:06:51	4078	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security006\Hardware\m_Security006.exe	2.4	2010/02/17 14:44:38	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 03:00:58	145599	TELM-06	1	Pass	0	0	0	108
M_TCP_Security	m_Security007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security007\Hardware\m_Security007.exe	2.6	2010/02/18 15:13:38	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 02:56:56	97975	TELM-06	1	Pass	0	0	0	126
M_TCP_Security	m_Security008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security008\Hardware\m_Security008.exe	2.3	2010/02/17 16:28:05	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:54:46	6828	TELM-06	1	Pass	0	0	0	72
M_TCP_Security	m_Security009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security009\Hardware\m_Security009.exe	2.3	2010/02/17 16:32:40	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:52:52	6318	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security010\Hardware\m_Security010.exe	2.3	2010/02/18 17:45:03	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:48:07	38593	TELM-06	1	Pass	0	0	0	113
M_TCP_Security	m_Security011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security011\Hardware\m_Security011.exe	2.5	2010/06/03 18:16:10	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:43:38	20145	TELM-06	1	Pass	0	0	0	58
M_TCP_Security	m_Security012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security012\Hardware\m_Security012.exe	2.9	2010/02/18 15:14:18	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 02:41:24	4894	TELM-06	1	Pass	0	0	0	132
M_TCP_Security	m_Security013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security013\Hardware\m_Security013.exe	2.5	2010/02/18 15:15:19	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:35:46	11578	TELM-06	1	Pass	0	0	0	140
M_TCP_Security	m_Security014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security014\Hardware\m_Security014.exe	2.2	2010/06/03 18:21:27	2.2	Up-to-date	Baseline 17_1	None	09/11/2010 04:14:18	14808	TELM-06	1	Pass	0	0	0	36
M_TCP_Security	m_Security015	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security015\Hardware\m_Security015.exe	2.4	2010/02/17 17:58:26	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 23:14:54	596197	TELM-06	1	Pass	0	0	0	34
M_TCP_Security	m_Security016	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security016\Hardware\m_Security016.exe	2.2	2010/06/08 19:56:40	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:32:59	11128	TELM-06	1	Pass	0	0	0	146
M_TCP_Security	m_Security017	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security017\Hardware\m_Security017.exe	2.2	2009/12/08 17:29:14	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:30:41	14646	TELM-06	1	Pass	0	0	0	39
M_TCP_Security	m_Security018	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security018\Hardware\m_Security018.exe	2.4	2010/03/18 16:05:37	2.4	Up-to-date	Baseline 17_1	None	09/10/2010 02:28:01	40549	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security019	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security019\Hardware\m_Security019.exe	2.3	2010/03/26 13:43:17	2.3	Up-to-date	Baseline 17_1	None	09/13/2010 09:10:52	90532	TELM-06	1	Pass	0	0	0	1222
M_TCP_Security	m_Security020	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security020\Hardware\m_Security020.exe	2.3	2010/06/03 18:22:30	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:21:41	22027	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security021	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security021\Hardware\m_Security021.exe	2.5	2010/06/03 18:37:52	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 02:18:26	40870	TELM-06	1	Pass	0	0	0	81
M_TCP_Security	m_Security022	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security022\Hardware\m_Security022.exe	2.9	2010/06/18 19:05:16	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 11:13:09	290910	TELM-06	1	Pass	0	0	0	55505
M_TCP_Security	m_Security023	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security023\Hardware\m_Security023.exe	2.12	2010/06/10 17:30:47	2.12	Up-to-date	Baseline 17_1	None	09/10/2010 23:03:25	51607	TELM-06	1	Pass	0	0	0	1119
M_TCP_Security	m_Security024	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security024\Hardware\m_Security024.exe	2.3	2010/03/26 13:06:23	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 17:12:52	30658	TELM-06	1	Pass	0	0	0	4
M_TCP_Security	m_Security025	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security025\Hardware\m_Security025.exe	2.3	2010/06/10 14:39:09	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 16:55:59	48458	TELM-06	1	Pass	0	0	0	158
M_TCP_Security	m_Security026	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security026\Hardware\m_Security026.exe	2.3	2010/02/18 15:15:54	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 02:17:05	4160	TELM-06	1	Pass	0	0	0	60
M_TCP_Security	m_Security027	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security027\Hardware\m_Security027.exe	2.1	2010/06/29 19:03:00	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:14:33	42290	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security028	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security028\Hardware\m_Security028.exe	2.1	2010/06/29 19:26:29	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 02:12:16	47163	TELM-06	1	Pass	0	0	0	35
M_TCP_Security	m_Security029	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security029\Hardware\m_Security029.exe	2.2	2010/03/11 18:29:14	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:08:57	20501	TELM-06	1	Pass	0	0	0	46
M_TCP_Security	m_Security030	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security030\Hardware\m_Security030.exe	2.2	2010/02/08 13:02:02	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 02:07:01	12462	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security031	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security031\Hardware\m_Security031.exe	2.1	2010/06/03 18:44:12	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:59:47	23053	TELM-06	1	Pass	0	0	0	16
M_TCP_Security	m_Security032	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security032\Hardware\m_Security032.exe	2.1	2010/02/03 12:51:08	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:58:39	8150	TELM-06	1	Pass	0	0	0	11
M_TCP_Security	m_Security033	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security033\Hardware\m_Security033.exe	2.1	2010/01/27 21:32:15	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:55:04	13934	TELM-06	1	Pass	0	0	0	12
M_TCP_Security	m_Security034	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security034\Hardware\m_Security034.exe	2.3	2010/03/26 13:19:51	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:49:47	118709	TELM-06	1	Pass	0	0	0	15
M_TCP_Security	m_Security035	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security035\Hardware\m_Security035.exe	2.0	2010/01/14 21:18:30	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 01:48:11	17724	TELM-06	1	Pass	0	0	0	30
M_TCP_Security	m_Security036	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security036\Hardware\m_Security036.exe	2.5	2010/05/07 14:41:43	2.5	Up-to-date	Baseline 17_1	None	09/10/2010 01:45:27	86768	TELM-06	1	Pass	0	0	0	10
M_TCP_Security	m_Security037	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security037\Hardware\m_Security037.exe	2.6	2010/06/03 18:54:41	2.6	Up-to-date	Baseline 17_1	None	09/11/2010 04:11:14	51213	TELM-06	1	Pass	0	0	0	119
M_TCP_Security	m_Security038	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security038\Hardware\m_Security038.exe	2.9	2010/06/07 13:02:07	2.9	Up-to-date	Baseline 17_1	None	09/10/2010 16:24:42	167462	TELM-06	1	Pass	0	0	0	170
M_TCP_Security	m_Security039	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security039\Hardware\m_Security039.exe	2.2	2010/02/18 15:16:25	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:41:28	56729	TELM-06	1	Pass	0	0	0	55
M_TCP_Security	m_Security040	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security040\Hardware\m_Security040.exe	2.3	2010/06/03 20:43:31	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:34:02	278558	TELM-06	1	Pass	0	0	0	33
M_TCP_Security	m_Security041	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security041\Hardware\m_Security041.exe	2.6	2010/04/21 10:55:08	2.6	Up-to-date	Baseline 17_1	None	09/10/2010 01:32:04	65792	TELM-06	1	Pass	0	0	0	77
M_TCP_Security	m_Security042	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security042\Hardware\m_Security042.exe	2.1	2010/04/08 12:57:18	2.1	Up-to-date	Baseline 17_1	None	09/10/2010 01:29:26	58063	TELM-06	1	Pass	0	0	0	48
M_TCP_Security	m_Security043	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security043\Hardware\m_Security043.exe	2.0	2010/02/16 20:27:42	2.0	Up-to-date	Baseline 17_1	None	09/10/2010 01:26:37	68761	TELM-06	1	Pass	0	0	0	56
M_TCP_Security	m_Security044	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security044\Hardware\m_Security044.exe	2.2	2010/05/13 19:14:37	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:23:16	70753	TELM-06	1	Pass	0	0	0	44
M_TCP_Security	m_Security045	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security045\Hardware\m_Security045.exe	2.3	2010/05/12 17:38:53	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 01:19:54	71201	TELM-06	1	Pass	0	0	0	62
M_TCP_Security	m_Security046	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_Security\m_Security046\Hardware\m_Security046.exe	2.2	2010/06/07 13:05:30	2.2	Up-to-date	Baseline 17_1	None	09/10/2010 01:15:43	3341	TELM-06	1	Pass	0	0	0	30
M_TCP_WirelessWakeup	m_Wakeup001	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup001\Hardware\m_Wakeup001.exe	2.3	2010/06/21 17:44:25	2.3	Up-to-date	Baseline 17_1	None	09/10/2010 14:37:38	291373	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup002	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup002\Hardware\m_Wakeup002.exe	2.5	2010/05/12 16:16:45	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:22:46	35491	TELM-06	1	Pass	0	0	0	12
M_TCP_WirelessWakeup	m_Wakeup003	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup003\Hardware\m_Wakeup003.exe	2.2	2010/03/10 17:59:29	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 11:21:21	14545	TELM-06	1	Pass	0	0	0	11
M_TCP_WirelessWakeup	m_Wakeup004	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup004\Hardware\m_Wakeup004.exe	2.1	2009/12/08 16:08:20	2.1	Up-to-date	Baseline 17_1	None	09/11/2010 02:48:30	19480	TELM-06	1	Pass	0	0	0	31
M_TCP_WirelessWakeup	m_Wakeup005	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup005\Hardware\m_Wakeup005.exe	2.3	2010/06/08 21:32:21	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:19:43	20372	TELM-06	1	Pass	0	0	0	40
M_TCP_WirelessWakeup	m_Wakeup006	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup006\Hardware\m_Wakeup006.exe	2.3	2010/01/27 00:35:22	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:18:26	10378	TELM-06	1	Pass	0	0	0	55
M_TCP_WirelessWakeup	m_Wakeup007	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup007\Hardware\m_Wakeup007.exe	2.1	2009/12/04 19:28:48	2.1	Up-to-date	Baseline 17_1	None	09/09/2010 11:17:14	12843	TELM-06	1	Pass	0	0	0	3
M_TCP_WirelessWakeup	m_Wakeup008	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup008\Hardware\m_Wakeup008.exe	2.5	2010/05/07 15:41:21	2.5	Up-to-date	Baseline 17_1	None	09/09/2010 11:14:18	54526	TELM-06	1	Pass	0	0	0	24
M_TCP_WirelessWakeup	m_Wakeup009	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup009\Hardware\m_Wakeup009.exe	2.6	2010/05/12 16:17:49	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 11:11:42	52041	TELM-06	1	Pass	0	0	0	32
M_TCP_WirelessWakeup	m_Wakeup010	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup010\Hardware\m_Wakeup010.exe	2.4	2010/05/12 16:20:12	2.4	Up-to-date	Baseline 17_1	None	09/09/2010 11:09:26	45200	TELM-06	1	Pass	0	0	0	16
M_TCP_WirelessWakeup	m_Wakeup011	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup011\Hardware\m_Wakeup011.exe	2.3	2010/04/06 18:07:38	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 11:05:52	27968	TELM-06	1	Pass	0	0	0	25
M_TCP_WirelessWakeup	m_Wakeup012	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup012\Hardware\m_Wakeup012.exe	2.6	2010/05/12 16:20:45	2.6	Up-to-date	Baseline 17_1	None	09/09/2010 10:59:08	48823	TELM-06	1	Pass	0	0	0	45
M_TCP_WirelessWakeup	m_Wakeup013	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup013\Hardware\m_Wakeup013.exe	2.2	2010/02/24 12:20:40	2.2	Up-to-date	Baseline 17_1	None	09/09/2010 10:55:16	127480	TELM-06	1	Pass	0	0	0	2
M_TCP_WirelessWakeup	m_Wakeup014	\\data\pace\fwvt\mozart\FWTest\FWVT_FORMAL\TestMode9_10\Tests\am_fwvt\mozart\vts\M_TCP_WirelessWakeup\m_Wakeup014\Hardware\m_Wakeup014.exe	2.3	2010/06/04 11:50:36	2.3	Up-to-date	Baseline 17_1	None	09/09/2010 10:51:38	74648	TELM-06	1	Pass	0	0	0	30

This is a description of the heading columns that are found under the "Tests" tab.

The intent is to provide more detailed information on what will be found.

	Column Name	**Description**
1	Feature	Common feature group, or sub-group, that test being run is related to.
2	Test	Name of the test (.cpp) that was run.
3	Test Path	Location of the executable (.exe) that was run.
4	Revision #	CVS Revision number extracted from test (.cpp) file.
5	Revision Date	Date of last check in to CVS of test (.cpp).
6	CVS Rev	Revision number of test (.cpp) file.
7	CVS State	Current CVS State of test (.cpp) being run.
8	TelM Firmware Revision	Version of Telemetry M Firmware under test.
9	RAMware	Version of RAMware for TelM
10	Sys Init Time	Time test (.exe) was started for Formal Run.
11	Test Time (mS)	Total time that test (.exe) was running.
12	Tester	What test station was used for the test execution.
13	Tester Rev	Revision number of the test station used for the test execution.
14	Status	Test Script execution - Pass or Fail
15	Warnings	Number of Warnings the executed test script had.
16	Errors	Number of Errors the executed test script had.
17	Fails	Number of Fail events the executed test script had.
18	Passes	Number of Pass events the executed test script had.

Medtronic	CONFIDENTIAL	DOCUMENT/RECORD

This document/record is electronically controlled; printed copies are considered uncontrolled.
System of Record: QADoc

Name	Version	Author
DSN001257	2.0	Kahle, Ken

Title: Mozart (Phase 1) Summary Specification to Test Requirements Traceability Report

APPROVALS

Signed By	Meaning of Signature	Date/Time (UTC)
Kahle, Ken	FW VT Manager/Leader	15-Sep-2010 16:07:15
Kahle, Ken	FW Verification Manager/Leader	15-Sep-2010 16:10:33
Mastro, Marcy	CRDM Config Assur	16-Sep-2010 11:49:47

MEDTRONIC CONFIDENTIAL

THIS DOCUMENT IS THE PROPERTY OF MEDTRONIC, INC. AND MUST BE ACCOUNTED FOR. INFORMATION HEREIN IS CONFIDENTIAL. DO NOT REPRODUCE IT, REVEAL IT TO UNAUTHORIZED PERSONS OR SEND IT OUTSIDE MEDTRONIC WITHOUT PROPER AUTHORIZATION.

PAGE
Cover

PROJECT NAME
Telemetry M

TITLE

Mozart (Phase 1) Summary Specification to Test Requirements Traceability Report

DHF Reference # 11.02 (5.11.10)

EFFECTIVE
UPON APPROVAL

DHF Deliverable: Traceability Report

VERSION	DESCRIPTION
2.0	Initial Release

Author(s)	Automatically Generated		
Reviewed and Approved By:	**Name**	**Signatures & Date**	
	Ken Kahle	<Electronic Signature>	

Tracing	Classification	Requirement Title					
				Test Case		Test	
	Class	Decomp Title		ID	Class	Class	Name
SPEC	DOC/REV	**TCP-C/H+**					

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_IE_001 - DOWNLINK PACKET CONTROL BYTES**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite006\memorywrite006
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch002\discovermatch002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency002\emergency002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open015\open015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead031\memoryread031
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead033\memoryread033
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite046\memorywrite046
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics032\diagnostics032
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch001\discovermatch001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Open\Open001\open001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite005\memorywrite005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_General\General015\general015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close002\close002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ006\arq006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead003\memoryread003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\TransferMode001\transfermode001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName001\discovername001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency038\emergency038
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_General\General004\general004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite010\memorywrite010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\TransferMode002\transfermode002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency001\emergency001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_General\General016\general016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite013\memorywrite013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\TransferMode003\transfermode003

Tracing	Classification	Requirement Title				
Decomp Class	Title		Test Case ID	Class	Test Class	Name

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull002\discoverfull002

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName024\discovername024

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite047\memorywrite047

Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001

' FILE_NAME: 'datalink.fm' FILE_REV: '1.14' SECTION: '5.1.1 Downlink Packet Control Bytes ' Page No: ' 38'

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class** **Name**

TRACING	FWVT Tested	**TLMC_IE_002 - UPLINK PACKET CONTROL BYTES**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead020\memoryread020
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead031\memoryread031
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_General\General003\general003
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Open\Open001\open001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead001\memoryread001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead034\memoryread034
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\TransferMode002\transfermode002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics003\diagnostics003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close005\close005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead003\memoryread003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead032\memoryread032
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite005\memorywrite005
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead015\memoryread015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead033\memoryread033
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\TransferMode003\transfermode003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency008\emergency008
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead014\memoryread014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite006\memorywrite006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite046\memorywrite046
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\TransferMode001\transfermode001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_General\General015\general015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead010\memoryread010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency001\emergency001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency002\emergency002

Tracing	Classification	Requirement Title	
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite045\memorywrite045
			Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName001\discovername001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch001\discovermatch001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General005\general005
		' FILE_NAME: 'datalink.fm' FILE_REV: '1.14' SECTION: '5.1.2 Uplink Packet Control Bytes ' Page No: ' 39'	
TRACING	FWVT Tested	**TLMC_IE_003 - PACKET TYPES**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite014\memorywrite014
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName001\discovername001
			Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead020\memoryread020
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite006\memorywrite006
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General015\general015
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\ARQ006\arq006
			Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General003\general003
			Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open001\open001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead003\memoryread003
			Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close001\close001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General005\general005
		' FILE_NAME: 'datalink.fm' FILE_REV: '1.14' SECTION: '5.1.3.4 Packet Type ' Page No: ' 41'	

Tracing **Classification** **Requirement Title**

Decomp Class Title Test Case ID Class Test Class Name

TRACING	FWVT Tested	**TLMC_IE_004 - INSTRUMENT ID**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ006\arq006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_General\General015\general015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite005\memorywrite005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite009\memorywrite009
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Open\Open001\open001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName001\discovername001

' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.4 General Requirements ' Page No: ' 46'

TRACING	FWVT Tested	**TLMC_IE_006 - EXTERNAL SERVICE REQUEST PROCESSING**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite005\memorywrite005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode002\rawmode002
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode001\rawmode001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling069\polling069
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics027\diagnostics027
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency001\emergency001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Open\Open001\open001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics014\diagnostics014
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency012\emergency012

' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.4 General Requirements ' Page No: ' 47'

TRACING	FWVT Tested	**TLMC_I_008 - IMPLANT INITIALIZE SERVICE REQUEST FROM C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Initialize\Initialize001\initialize001
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Initialize\Initialize002\initialize002

' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.5.1 Requirements ' Page No: ' 53'

Tracing	Classification	Requirement Title				
Class	**Decomp**	**Title**	**Test Case ID**	**Test Case Class**	**Test Class**	**Test Name**
TRACING	FWVT Tested	**TLMC_I_009 - INITIALIZE SERVICE RESPONSE TO C INTERFACE LAYER.**				
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.5.1 Requirements ' Page No: ' 53'				

Tracing	Classification	Requirement Title						
Class	Decomp	Title			ID	Test Case Class		Test Class Name
TRACING	FWVT Tested	**TLMC_E_010 - DISCOVER SERVICE REQUEST FROM C INTERFACE LAYER**	Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch044\discovermatch044
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName006\discovername006
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch010\discovermatch010
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch005\discovermatch005
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch007\discovermatch007
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName001\discovername001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName007\discovername007
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull005\discoverfull005
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch002\discovermatch002
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch045\discovermatch045
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName008\discovername008
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Discover\DiscoverFull027\discoverfull027
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch001\discovermatch001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch004\discovermatch004
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch009\discovermatch009
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch042\discovermatch042
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName004\discovername004
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull007\discoverfull007
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Discover\DiscoverFull028\discoverfull028
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName002\discovername002
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull002\discoverfull002
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull004\discoverfull004
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull008\discoverfull008
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch003\discovermatch003
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverMatch043\discovermatch043
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName003\discovername003

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName005\discovername005
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull003\discoverfull003
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.1 Requirements ' Page No: ' 65'	
TRACING	FWVT Tested	**TLMC_E_011 - VALID IMPLANT ID WITH STOP ON MATCH**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch011\discovermatch011
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch001\discovermatch001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.1 Requirements ' Page No: ' 65'	
TRACING	FWVT Tested	**TLMC_IE_012 - DISCOVER FULL - DISCOVER ID PACKET FORMAT**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 66'	
TRACING	FWVT Tested	**TLMC_E_013 - DISCOVER FULL - DOWNLINK DISCOVER ID REQUEST PACKET**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull006\discoverfull006
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 66'	
TRACING	FWVT Tested	**TLMC_E_014 - DISCOVER FULL - RETRY DISCOVER ID REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull010\discoverfull010
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull011\discoverfull011
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull002\discoverfull002
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull009\discoverfull009
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 66'	
TRACING	FWVT Tested	**TLMC_I_015 - DISCOVER FULL - DISCOVER ID RESPONSE**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull011\discoverfull011
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull029\discoverfull029
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 66'	
TRACING	FWVT Tested	**TLMC_E_016 - DISCOVER FULL - LISTEN BEFORE TALK TIMER DURING ID**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull002\discoverfull002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 67'	
TRACING	FWVT Tested	**TLMC_IE_017 - DISCOVER FULL - DISCOVER ID RESPONSE PACKET FORMAT**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 67'	

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_IE_018 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 67'			
TRACING	FWVT Tested	**TLMC_E_019 - DISCOVER FULL - DOWNLINK DISCOVER NAMEREQUEST PACKET**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.2 Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001 Test: .\C_TCP_Discover\DiscoverFull002\discoverfull002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 67'			
TRACING	FWVT Tested	**TLMC_E_020 - DISCOVER FULL - RETRY DISCOVER NAME REQUEST**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.1 Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull012\discoverfull012 Test: .\C_TCP_Discover\DiscoverFull013\discoverfull013
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 68'			
TRACING	FWVT Tested	**TLMC_I_021 - DISCOVER FULL - DISCOVER NAME RESPONSE**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.2 Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001 Test: .\C_TCP_Discover\DiscoverFull013\discoverfull013
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 68'			
TRACING	FWVT Tested	**TLMC_IE_022 - DISCOVER FULL - DISCOVER NAME RESPONSE PACKET FORMATS**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 68'			
TRACING	FWVT Tested	**TLMC_I_023 - DISCOVER FULL - UPLINK RESPONSE PARTITIONING**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 70'			

Tracing	Classification	Requirement Title			
	Decomp Class	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_024 - DISCOVER FULL - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull015\discoverfull015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull024\discoverfull024
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull005\discoverfull005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull025\discoverfull025
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull014\discoverfull014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull022\discoverfull022
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull023\discoverfull023
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName015\discovername015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull016\discoverfull016
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull017\discoverfull017
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.2 Discover Full ' Page No: ' 70'			
TRACING	FWVT Tested	**TLMC_E_025 - DISCOVER NAME - DOWNLINK REQUEST PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName002\discovername002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.3 Discover Name ' Page No: ' 71'			
TRACING	FWVT Tested	**TLMC_E_026 - DISCOVER NAME - RETRY DOWNLINK PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName009\discovername009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName003\discovername003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName010\discovername010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName002\discovername002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.3 Discover Name ' Page No: ' 71'			
TRACING	FWVT Tested	**TLMC_E_027 - DISCOVER NAME -LISTEN BEFORE TALK TIMER DURING ID**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName002\discovername002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.3 Discover Name ' Page No: ' 71'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_028 - DISCOVER NAME - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName002\discovername002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName011\discovername011
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName001\discovername001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName014\discovername014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName013\discovername013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName015\discovername015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName027\discovername027
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName028\discovername028
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName005\discovername005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName012\discovername012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName025\discovername025
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName026\discovername026
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.3 Discover Name ' Page No: ' 72'			
TRACING	FWVT Tested	**TLMC_E_029 - DISCOVER STOP ON MATCH - DISCOVER ID RESPONSE MATCH INSTR ID**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch013\discovermatch013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch014\discovermatch014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch015\discovermatch015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch012\discovermatch012
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.4 Discover Stop On Match ' Page No: ' 73'			
TRACING	FWVT Tested	**TLMC_E_030 - DISCOVER STOP ON MATCH - DISCOVER NAME REQUEST**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch004\discovermatch004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch016\discovermatch016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch013\discovermatch013
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.4 Discover Stop On Match ' Page No: ' 73'			
TRACING	FWVT Tested	**TLMC_E_031 - DISCOVER STOP ON MATCH - MATCHING IMPLANT ID**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch001\discovermatch001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch002\discovermatch002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch017\discovermatch017
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.4 Discover Stop On Match ' Page No: ' 73'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Test Case Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_034 - DISCOVER STOP ON MATCH - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch017\discovermatch017
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch005\discovermatch005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch018\discovermatch018
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch002\discovermatch002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch001\discovermatch001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch020\discovermatch020
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch040\discovermatch040
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch046\discovermatch046
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.7.4 Discover Stop On Match ' Page No: ' 74'			
TRACING	FWVT Tested	**TLMC_E_040 - OPEN - C INTERFACE LAYER PARAMETERS**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open003\open003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open005\open005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open002\open002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open004\open004
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Open\Open001\open001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 78'			
TRACING	FWVT Tested	**TLMC_IE_041 - OPEN - DOWNLINK REQUEST PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open014\open014
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Open\Open001\open001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open020\open020
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 78'			
TRACING	FWVT Tested	**TLMC_IE_042 - OPEN - UPLINK OPEN RESPONSE PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open014\open014
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Open\Open001\open001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 79'			
TRACING	FWVT Tested	**TLMC_E_043 - OPEN - DOWNLINK REQUEST**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open002\open002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open003\open003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open004\open004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open005\open005
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 79'			

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_044 - OPEN - NO VALID UPLINK RESPONSE RETRY DOWNLINK REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open002\open002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 79'	
TRACING	FWVT Tested	**TLMC_E_045 - OPEN - LISTEN BEFORE TALK TIMER OPERATIONS**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open022\open022 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open002\open002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 79'	
TRACING	FWVT Tested	**TLMC_I_046 - OPEN - UPLINK RESPONSE TO A VALID OPEN REQUEST**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling073\polling073 Status: **Up-to-date** Rev: 2.3 Test: .\C_TCP_Polling\Polling076\polling076 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling077\polling077 Status: **Up-to-date** Rev: 2.3 Test: .\C_TCP_Polling\Polling074\polling074 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open001\open001 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling078\polling078 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling079\polling079 Status: **Up-to-date** Rev: 2.3 Test: .\C_TCP_Polling\Polling072\polling072 Status: **Up-to-date** Rev: 2.3 Test: .\C_TCP_Polling\Polling075\polling075 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling080\polling080 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling081\polling081
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 80'	
TRACING	FWVT Tested	**TLMC_E_047 - OPEN - A VALID UPLINK RESPONSE**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open001\open001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 80'	

Tracing	Classification	Requirement Title	
Class	Decomp Title		Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_048 - OPEN - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open002\open002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open007\open007 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open019\open019 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open005\open005 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open008\open008 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open001\open001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open018\open018
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.1 Open - General Requirements ' Page No: ' 80'	
TRACING	FWVT Tested	**TLMC_E_049 - DATA - C INTERFACE LAYER TO DATA LINK LAYER PARAMETERS**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite003\memorywrite003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite004\memorywrite004 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite007\memorywrite007 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite005\memorywrite005 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead035\memoryread035 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite006\memorywrite006 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite049\memorywrite049 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite002\memorywrite002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite001\memorywrite001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 87'	
TRACING	FWVT Tested	**TLMC_IE_050 - DATA - DOWNLINK REQUEST PACKET FORMAT(S)**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite005\memorywrite005
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 88'	
TRACING	FWVT Tested	**TLMC_E_051 - DATA - DOWNLINK OF DATA PACKET**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite001\memorywrite001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite005\memorywrite005
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 89'	
TRACING	FWVT Tested	**TLMC_I_052 - DATA - ACKNOWLEDGEMENT OF DOWNLINK DATA PACKET**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite002\memorywrite002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite006\memorywrite006
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 89'	

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_053 - DATA - PACKETIZATION OF THE DOWNLINK DATA**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite005\memorywrite005 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite013\memorywrite013 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite011\memorywrite011 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite014\memorywrite014 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite012\memorywrite012 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite015\memorywrite015 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite009\memorywrite009 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite010\memorywrite010
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 89'	
TRACING	FWVT Tested	**TLMC_E_054 - DATA - RETRANSMISSION OF DOWNLINK PACKET - NO ACK RECEIVED**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite018\memorywrite018 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite016\memorywrite016 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite017\memorywrite017 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite019\memorywrite019
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 90'	
TRACING	FWVT Tested	**TLMC_I_055 - DATA - DOWNLINK PACKETS WITH DUPLICATE SEQUENCE NUMBERS**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead002\memoryread002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead001\memoryread001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 90'	
TRACING	FWVT Tested	**TLMC_I_056 - DATA - VALIDATION OF DOWNLINK DATA REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead003\memoryread003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite008\memorywrite008 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite010\memorywrite010 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite018\memorywrite018 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite021\memorywrite021 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite050\memorywrite050 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite005\memorywrite005 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite016\memorywrite016 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite020\memorywrite020 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead004\memoryread004

Tracing	Classification	Requirement Title	
Decomp Class		Title	Test Case ID Class Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 90'	
TRACING	FWVT Tested	**TLMC_I_057 - DATA - DOWNLINK PACKETS INTO TRANSACTION**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite006\memorywrite006
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 90'	
TRACING	FWVT Tested	**TLMC_E_058 - DATA - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead005\memoryread005 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite024\memorywrite024 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite025\memorywrite025 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Data\MemoryRead044\memoryread044 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead045\memoryread045 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite022\memorywrite022 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite026\memorywrite026 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite049\memorywrite049 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Data\MemoryRead043\memoryread043 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite023\memorywrite023 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite027\memorywrite027
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.2 Downlink Data ' Page No: ' 91'	
TRACING	FWVT Tested	**TLMC_I_059 - DATA - UPLINK C INTERFACE LAYER PARAMETERS**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead006\memoryread006 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead001\memoryread001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead008\memoryread008 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead007\memoryread007
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.3 Uplink Data ' Page No: ' 91'	
TRACING	FWVT Tested	**TLMC_I_060 - DATA - UPLINK OF DATA PACKETS**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead009\memoryread009 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead006\memoryread006
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.3 Uplink Data ' Page No: ' 92'	
TRACING	FWVT Tested	**TLMC_I_061 - DATA - UPLINK DATA PACKET FORMAT(S)**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead006\memoryread006
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.3 Uplink Data ' Page No: ' 93'	

Tracing	Classification	Requirement Title			
Class	Decomp	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_I_062 - DATA - PACKETIZATION OF UPLINK DATA**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead010\memoryread010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead016\memoryread016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead006\memoryread006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead014\memoryread014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead017\memoryread017
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead012\memoryread012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead015\memoryread015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead018\memoryread018
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead011\memoryread011
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead013\memoryread013
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.3 Uplink Data ' Page No: ' 94'			
TRACING	FWVT Tested	**TLMC_E_063 - DATA - RECEIPT OF AN INVALID UPLINK PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead009\memoryread009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead003\memoryread003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead020\memoryread020
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead019\memoryread019
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead021\memoryread021
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.3 Uplink Data ' Page No: ' 94'			
TRACING	FWVT Tested	**TLMC_E_064 - DATA - ASSEMBLING DATA PACKET INTO TRANSACTIONS**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead024\memoryread024
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead022\memoryread022
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead023\memoryread023
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.3 Uplink Data ' Page No: ' 94'			
TRACING	FWVT Tested	**TLMC_E_065 - DATA - RESPONSE BEFORE DATA INDICATION**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite058\memorywrite058
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.9.3 Uplink Data ' Page No: ' 95'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_066 - DIAGNOSTIC SERVICE REQUEST FROM C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics013\diagnostics013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics004\diagnostics004
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics008\diagnostics008
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics010\diagnostics010
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics011\diagnostics011
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics015\diagnostics015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics016\diagnostics016
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics007\diagnostics007
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics012\diagnostics012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics005\diagnostics005
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics006\diagnostics006
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics009\diagnostics009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics027\diagnostics027
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics002\diagnostics002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics003\diagnostics003
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics014\diagnostics014
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.1 Requirements ' Page No: ' 97'			
TRACING	FWVT Tested	**TLMC_E_067 - REMOTE DIAGNOSTIC - DOWNLINK PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics002\diagnostics002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics003\diagnostics003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics004\diagnostics004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics005\diagnostics005
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.2 Remote Diagnostic ' Page No: ' 97'			
TRACING	FWVT Tested	**TLMC_I_068 - REMOTE DIAGNOSTIC - UPLINK PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.2 Remote Diagnostic ' Page No: ' 98'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID Class		Test Class Name
TRACING	FWVT Tested	**TLMC_I_069 - REMOTE DIAGNOSTIC - STATUS BYTE FORMAT**	Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics030\diagnostics030
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics017\diagnostics017
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.2 Remote Diagnostic ' Page No: ' 98'			
TRACING	FWVT Tested	**TLMC_I_070 - REMOTE DIAGNOSTIC - UPLINK RESPONSE PACKET**	Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics009\diagnostics009
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics007\diagnostics007
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics006\diagnostics006
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics018\diagnostics018
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics008\diagnostics008
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.2 Remote Diagnostic ' Page No: ' 99'			
TRACING	FWVT Tested	**TLMC_E_071 - REMOTE DIAGNOSTIC - DOWNLINK REQUEST RETRANSMISSION**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics018\diagnostics018
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics020\diagnostics020
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics033\diagnostics033
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics004\diagnostics004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics035\diagnostics035
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics036\diagnostics036
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics019\diagnostics019
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics037\diagnostics037
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.2 Remote Diagnostic ' Page No: ' 99'			

Tracing Class	Classification Decomp Title	Requirement Title	Test Case ID	Test Case Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_072 - REMOTE DIAGNOSTIC - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics008\diagnostics008
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics009\diagnostics009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics020\diagnostics020
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics021\diagnostics021
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics022\diagnostics022
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics024\diagnostics024
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics005\diagnostics005
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics007\diagnostics007
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics010\diagnostics010
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics006\diagnostics006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics017\diagnostics017
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics018\diagnostics018
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics025\diagnostics025
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics026\diagnostics026
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics003\diagnostics003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics034\diagnostics034
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.2 Remote Diagnostic ' Page No: ' 99'			
TRACING	FWVT Tested	**TLMC_E_073 - LOCAL DIAGNOSTIC - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics014\diagnostics014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics015\diagnostics015
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics012\diagnostics012
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Diagnostic\Diagnostics011\diagnostics011
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics013\diagnostics013
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.3 Local Diagnostic ' Page No: ' 101'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_E_074 - MAPPING DIAGNOSTIC - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics025\diagnostics025
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics028\diagnostics028
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics016\diagnostics016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics022\diagnostics022
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics023\diagnostics023
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics024\diagnostics024
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics029\diagnostics029
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics005\diagnostics005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics026\diagnostics026
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics027\diagnostics027
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.4 Mapping Diagnostic ' Page No: ' 103'			
TRACING	FWVT Tested	**TLMC_E_075 - DIAGNOSTIC FILLER - DOWNLINK FILLER PACKETS**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ020\arq020
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ012\arq012
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Diagnostic\Diagnostics030\diagnostics030
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.5 Diagnostic Filler Packet ' Page No: ' 104'			
TRACING	FWVT Tested	**TLMC_E_077 - EMERGENCY - C INTERFACE LAYER PARAMETERS**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency001\emergency001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency006\emergency006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency043\emergency043
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency003\emergency003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency005\emergency005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency002\emergency002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency004\emergency004
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.11.2 Instrument ' Page No: ' 106'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_E_078 - EMERGENCY - DOWNLINK OF EMERGENCY REQUEST**	Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverMatch020\discovermatch020
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency003\emergency003
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite024\memorywrite024
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode003\rawmode003
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead030\memoryread030
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Diagnostic\Diagnostics031\diagnostics031
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency001\emergency001
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverName014\discovername014
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency007\emergency007
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open008\open008
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Discover\DiscoverFull017\discoverfull017
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open017\open017
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.11.2 Instrument ' Page No: ' 106'				
TRACING	FWVT Tested	**TLMC_E_079 - EMERGENCY - DOWNLINK REQUEST NOT ACKNOWLEDGED**	Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency003\emergency003
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.11.2 Instrument ' Page No: ' 107'				
TRACING	FWVT Tested	**TLMC_E_080 - EMERGENCY - DOWNLINK REQUEST ACKNOWLEDGED**	Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency008\emergency008
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.11.2 Instrument ' Page No: ' 107'				

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_081 - EMERGENCY - STATUS RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency006\emergency006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency040\emergency040
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency042\emergency042
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency010\emergency010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency041\emergency041
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency039\emergency039
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency004\emergency004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency009\emergency009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency011\emergency011
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency002\emergency002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.11.2 Instrument ' Page No: ' 107'			
TRACING	FWVT Tested	**TLMC_E_082 - RAW MODE - C INTERFACE LAYER DOWNLINK PARAMETERS**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode001\rawmode001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode003\rawmode003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode002\rawmode002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.12.1 Requirements ' Page No: ' 109'			
TRACING	FWVT Tested	**TLMC_E_083 - RAW MODE - DOWNLINK ATTACHED OPERATION**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode002\rawmode002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode004\rawmode004
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.12.2 Raw Mode (Downlink Attached) ' Page No: ' 109'			

Decomp Class Title Test Case ID Class Test Class Name

Tracing	Classification	Requirement Title			
TRACING	FWVT Tested	**TLMC_E_084 - RAW MODE - DOWNLINK ATTACHED UPLINK RESPONSE**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode010\rawmode010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode008\rawmode008
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode009\rawmode009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode006\rawmode006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode012\rawmode012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode002\rawmode002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode007\rawmode007
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode003\rawmode003
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode005\rawmode005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode004\rawmode004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode011\rawmode011
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.12.2 Raw Mode (Downlink Attached) ' Page No: ' 110'			
TRACING	FWVT Tested	**TLMC_E_085 - RAW MODE - ECHO ON STATUS TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode001\rawmode001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.12.3 Raw Mode (Echo On) ' Page No: ' 111'			
TRACING	FWVT Tested	**TLMC_E_086 - RAW MODE - ECHO ON - DOWNLINK PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode013\rawmode013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode014\rawmode014
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.12.3 Raw Mode (Echo On) ' Page No: ' 111'			
TRACING	FWVT Tested	**TLMC_E_087 - RAW MODE - ECHO ON - UPLINK PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode013\rawmode013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode014\rawmode014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode015\rawmode015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode017\rawmode017
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode016\rawmode016
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.12.3 Raw Mode (Echo On) ' Page No: ' 112'			
TRACING	FWVT Tested	**TLMC_E_088 - RAW MODE - ECHO OFF STATUS TO C INTERFACE LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_RawMode\RawMode001\rawmode001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.12.4 Raw Mode (Echo Off) ' Page No: ' 112'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_E_089 - CLOSE - C INTERFACE LAYER DOWNLINK PARAMETERS**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close004\close004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close006\close006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close002\close002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close005\close005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close018\close018
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close003\close003
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 113'			
TRACING	FWVT Tested	**TLMC_IE_090 - CLOSE - DOWNLINK PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 113'			
TRACING	FWVT Tested	**TLMC_IE_091 - CLOSE - UPLINK PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 114'			
TRACING	FWVT Tested	**TLMC_E_092 - CLOSE - DOWNLINK REQUEST**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close002\close002
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 114'			
TRACING	FWVT Tested	**TLMC_E_093 - CLOSE - RETRANSMISSION OF DOWNLINK REQUEST**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close001\close001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close007\close007
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close005\close005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close008\close008
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 114'			
TRACING	FWVT Tested	**TLMC_I_094 - CLOSE - RECEIPT OF A VALID CLOSE REQUEST**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close009\close009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Close\Close010\close010
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 114'			
TRACING	FWVT Tested	**TLMC_I_095 - CLOSE - INDICATION TO IMPLANT C INTERFACE LAYER**			
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 114'			

Tracing **Classification** **Requirement Title**

Decomp Class Title **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_E_096 - CLOSE - RECEIPT OF A VALID RESPONSE**	Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close001\close001
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close002\close002
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 115'					
TRACING	FWVT Tested	**TLMC_E_097 - CLOSE - IMPLICITLY SUCCESSFUL**	Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close012\close012
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close013\close013
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close017\close017
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close014\close014
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close011\close011
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 115'					
TRACING	FWVT Tested	**TLMC_E_098 - CLOSE - RESPONSE TO C INTERFACE LAYER**	Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close008\close008
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close012\close012
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close013\close013
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close011\close011
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close001\close001
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close014\close014
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close007\close007
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Close\Close005\close005
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 115'					
TRACING	FWVT Tested	**TLMC_E_100 - ARQ - STOP AND WAIT DOWNLINK ACKNOWLEDGMENT**	Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\ARQ003\arq003
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryWrite008\memorywrite008
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\ARQ005\arq005
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\ARQ001\arq001
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\ARQ004\arq004
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\ARQ002\arq002
			Status:	**Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryWrite039\memorywrite039
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.2 Instrument ' Page No: ' 125'					

Tracing	Classification	Requirement Title			
	Class	Decomp Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_IE_101 - ARQ - DOWNLINK ARQ PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ006\arq006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ009\arq009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ007\arq007
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ008\arq008
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ010\arq010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ011\arq011
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.2 Instrument ' Page No: ' 128'			
TRACING	FWVT Tested	**TLMC_E_102 - ARQ - UPDATING OF ARQ INFORMATION**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ012\arq012
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.2 Instrument ' Page No: ' 129'			
TRACING	FWVT Tested	**TLMC_E_103 - ARQ - MEMORY DATA UPLINK PACKET ACKNOWLEDGMENT**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ013\arq013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ016\arq016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ015\arq015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ012\arq012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ014\arq014
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.2 Instrument ' Page No: ' 129'			
TRACING	FWVT Tested	**TLMC_E_104 - ARQ - UPLINK ACK NOT RECEIVED**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ018\arq018
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ016\arq016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ017\arq017
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.2 Instrument ' Page No: ' 129'			
TRACING	FWVT Tested	**TLMC_E_105 - ARQ - INITIATE DOUBLE-UP WAVEFORM**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ028\arq028
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ029\arq029
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.2 Instrument ' Page No: ' 129'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class** **Name**

TRACING	FWVT Tested	**TLMC_I_106 - ARQ - STOP AND WAIT DOWNLINK PACKETS**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ017\arq017
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ020\arq020
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ019\arq019
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ012\arq012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ021\arq021
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.3 Implant ' Page No: ' 129'			
TRACING	FWVT Tested	**TLMC_I_107 - ARQ - RECEIPT OF AN ARQ PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ024\arq024
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ022\arq022
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ023\arq023
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.3 Implant ' Page No: ' 130'			
TRACING	FWVT Tested	**TLMC_I_108 - ARQ - ARQ PACKET DOWNLINK NOT RECEIVED**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ017\arq017
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.3 Implant ' Page No: ' 130'			
TRACING	FWVT Tested	**TLMC_I_109 - ARQ - IMPLICIT ACK OF A VALID MEMORY DATA UPLINK**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ026\arq026
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ023\arq023
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ025\arq025
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.3 Implant ' Page No: ' 130'			
TRACING	FWVT Tested	**TLMC_I_111 - ARQ - RETRANSMIT WAVEFORM TRIGGER**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ030\arq030
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.3 Implant ' Page No: ' 130'			
TRACING	FWVT Tested	**TLMC_E_112 - CH MAPPING - COLLECT MAPPING DATA**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.3 Requirements ' Page No: ' 133'			
TRACING	FWVT Tested	**TLMC_E_113 - CH MAPPING - PROCESS MAPPING DATA**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.3 Requirements ' Page No: ' 133'			
TRACING	FWVT Tested	**TLMC_E_114 - CH MAPPING - CALCULATE AVERAGES**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.3 Requirements ' Page No: ' 133'			

Tracing	Classification	Requirement Title	
	Decomp Class	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_115 - CH MAPPING - COLLECT MAPPING DATA**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.3 Requirements ' Page No: ' 135'	
TRACING	FWVT Tested	**TLMC_E_116 - CH MAPPING - EVALUATE COLLECTED DATA**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.3 Requirements ' Page No: ' 135'	
TRACING	FWVT Tested	**TLMC_E_117 - CH MAPPING - EVALUATION CONTROL**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.3 Requirements ' Page No: ' 135'	
TRACING	FWVT Tested	**TLMC_E_118 - CH MAPPING - DIAGNOSTIC DATA COLLECTION**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.4 Diagnostic Mapping ' Page No: ' 136'	
TRACING	FWVT Tested	**TLMC_E_119 - CH MAPPING - CALCULATE DIAGNOSTIC DATA AVERAGES**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.2.4 Diagnostic Mapping ' Page No: ' 136'	
TRACING	FWVT Tested	**TLMC_E_120 - CH RECOVERY - TRIGGER CHANNEL RECOVERY**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery004\channelrecovery004 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery033\channelrecovery033 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery060\channelrecovery060 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery002\channelrecovery002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery001\channelrecovery001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery003\channelrecovery003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery005\channelrecovery005 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery010\channelrecovery010
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 139'	

Tracing	Classification	Requirement Title			
Class	Decomp Title		Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_121 - CH RECOVERY - START SAME CHANNEL RECOVERY**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery008\channelrecovery008
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery001\channelrecovery001
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery006\channelrecovery006
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery007\channelrecovery007
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery032\channelrecovery032
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 139'			
TRACING	FWVT Tested	**TLMC_E_122 - CH RECOVERY - SAME CHANNEL RECOVERY OPEN DOWNLINK**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery010\channelrecovery010
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery013\channelrecovery013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery009\channelrecovery009
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery011\channelrecovery011
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery012\channelrecovery012
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 140'			
TRACING	FWVT Tested	**TLMC_E_123 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery014\channelrecovery014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery012\channelrecovery012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery001\channelrecovery001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 140'			
TRACING	FWVT Tested	**TLMC_E_124 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery012\channelrecovery012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery014\channelrecovery014
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 140'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID Class		Test Class Name
TRACING	FWVT Tested	**TLMC_E_125 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery012\channelrecovery012
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery016\channelrecovery016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery017\channelrecovery017
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery038\channelrecovery038
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery015\channelrecovery015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery043\channelrecovery043
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery020\channelrecovery020
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery018\channelrecovery018
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery042\channelrecovery042
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 141'			
TRACING	FWVT Tested	**TLMC_E_126 - REMAP CHANNEL - UNSPECIFIed CHANNEL RECOVERY TIMER EXPIRATION**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery020\channelrecovery020
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 141'			
TRACING	FWVT Tested	**TLMC_E_127 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery020\channelrecovery020
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 142'			
TRACING	FWVT Tested	**TLMC_E_128 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 142'			
TRACING	FWVT Tested	**TLMC_I_129 - CH RECOVERY - TRIGGER CHANNEL RECOVERY**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery023\channelrecovery023
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery014\channelrecovery014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery015\channelrecovery015
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery024\channelrecovery024
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery022\channelrecovery022
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Channel\ChannelRecovery021\channelrecovery021
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 142'			

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_I_130 - CH RECOVERY - START SAME CHANNEL RECOVERY**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery028\channelrecovery028 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery029\channelrecovery029 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery026\channelrecovery026 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery025\channelrecovery025 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery027\channelrecovery027
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 142'	
TRACING	FWVT Tested	**TLMC_I_131 - CH RECOVERY - SAME CHANNEL RECOVERY INVOKE SCAN**	Status: **Up-to-date** Rev: 2.3 Test: .\C_TCP_Channel\ChannelRecovery040\channelrecovery040 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery001\channelrecovery001 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Channel\ChannelRecovery031\channelrecovery031
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 143'	
TRACING	FWVT Tested	**TLMC_I_132 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery001\channelrecovery001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 143'	
TRACING	FWVT Tested	**TLMC_I_133 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 143'	
TRACING	FWVT Tested	**TLMC_I_134 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 143'	
TRACING	FWVT Tested	**TLMC_I_135 - CH RECOVERY - UCR TIMER ACTIVE NON OPEN DOWNLINK REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery030\channelrecovery030 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery041\channelrecovery041
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 144'	
TRACING	FWVT Tested	**TLMC_I_136 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery030\channelrecovery030 Status: **Up-to-date** Rev: 2.3 Test: .\C_TCP_Channel\ChannelRecovery040\channelrecovery040
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 144'	

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID Class** **Test Class Name**

TRACING FWVT Tested **TLMC_I_137 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL**

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery039\channelrecovery039
Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Channel\ChannelRecovery031\channelrecovery031

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.3 Implant ' Page No: ' 144'

TRACING FWVT Tested **TLMC_E_139 - DATA TRANSFER MODE - MODE TRANSITIONS**

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode003\transfermode003
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode002\transfermode002
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode001\transfermode001

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.4.2 Instrument ' Page No: ' 147'

TRACING FWVT Tested **TLMC_E_140 - DATA TRANSFER MODE - NOMINAL TO LOW ENERGY MODE TRANSITIONS.**

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode010\transfermode010
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode001\transfermode001

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.4.2 Instrument ' Page No: ' 147'

TRACING FWVT Tested **TLMC_E_141 - DATA TRANSFER MODE - NEGATIVE PREAMBLE RATIO IN LOW ENERGY MODE**

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode001\transfermode001
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode004\transfermode004

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.4.2 Instrument ' Page No: ' 148'

TRACING FWVT Tested **TLMC_E_142 - DATA TRANSFER MODE - LOW ENERGY TO NOMINAL TRANSITION**

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode001\transfermode001

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.4.2 Instrument ' Page No: ' 148'

TRACING FWVT Tested **TLMC_E_143 - DATA TRANSFER MODE - TRANSITION TO NEW MODE ON DOWNLINK**

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode009\transfermode009
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode008\transfermode008
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode001\transfermode001
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode006\transfermode006
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode005\transfermode005
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode007\transfermode007

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.4.2 Instrument ' Page No: ' 148'

Tracing	Classification	Requirement Title						
	Decomp				Test Case		Test	
Class		Title			ID	Class	Class	Name
TRACING	FWVT Tested	**TLMC_E_158 - CIL - REQUEST-RESPONSE-DEFINITION**	Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_General\General003\general003
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Close\Close001\close001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Data\MemoryWrite006\memorywrite006
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Open\Open001\open001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Diagnostic\Diagnostics027\diagnostics027
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Emergency\Emergency001\emergency001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_General\General017\general017
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Diagnostic\Diagnostics001\diagnostics001
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Diagnostic\Diagnostics011\diagnostics011
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverName001\discovername001
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Open\Open014\open014
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_RawMode\RawMode002\rawmode002
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull014\discoverfull014
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_General\General001\general001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_RawMode\RawMode001\rawmode001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2 ICCC Commands ' Page No: ' 158'						
TRACING	FWVT Tested	**TLMC_E_159 - CIL - ID ALL REQUEST AND RESPONSE VIA DISCOVER FULL**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Discover\DiscoverFull001\discoverfull001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull018\discoverfull018
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull021\discoverfull021
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull009\discoverfull009
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull020\discoverfull020
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull002\discoverfull002
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull014\discoverfull014
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Discover\DiscoverFull019\discoverfull019
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.1 ID All ' Page No: ' 159'						

Tracing	Classification	Requirement Title	
	Decomp Class	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_161 - CIL - REQUEST ID WHEN DEVICE NAME NOT KNOWN**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName015\discovername015 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName022\discovername022 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName016\discovername016 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName017\discovername017 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName019\discovername019 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName018\discovername018
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.3 Request ID String ' Page No: ' 159'	
TRACING	FWVT Tested	**TLMC_E_162 - CIL - REQUEST ID STRING DEVICE NAME KNOWN**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName023\discovername023
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.3 Request ID String ' Page No: ' 160'	
TRACING	FWVT Tested	**TLMC_E_164 - CIL - OPEN IMMEDIATE REQUEST**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open001\open001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open004\open004
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.5 Open Immediate ' Page No: ' 160'	
TRACING	FWVT Tested	**TLMC_E_165 - CIL - INVALID ICCC OPEN IMMEDIATE REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open009\open009
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.5 Open Immediate ' Page No: ' 160'	
TRACING	FWVT Tested	**TLMC_E_166 - CIL - OPEN WITH DEVICE FOUND REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch009\discovermatch009 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch022\discovermatch022 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch001\discovermatch001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch023\discovermatch023 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch024\discovermatch024 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch002\discovermatch002
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.6 Open With Device Found ' Page No: ' 161'	
TRACING	FWVT Tested	**TLMC_E_167 - CIL - INVALID OPEN WITH DEVICE FOUND REQUEST**	
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.6 Open With Device Found ' Page No: ' 161'	
TRACING	FWVT Tested	**TLMC_E_217 - CIL - OPEN HIBERNATE REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open014\open014
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.7 Waveform Disable ' Page No: ' 161'	

Tracing Class	Classification Decomp	Requirement Title Title	Test Case ID Class / Test Class Name
TRACING	FWVT Tested	**TLMC_E_168 - CIL - TARGET DEVICE ID**	
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.9 Open And Target Device ID ' Page No: ' 162'	
TRACING	FWVT Tested	**TLMC_E_169 - CIL -VALIDATION OF TARGET DEVICE-ID**	
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.9 Open And Target Device ID ' Page No: ' 162'	
TRACING	FWVT Tested	**TLMC_E_170 - CIL - CLOSE REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close001\close001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close002\close002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close005\close005
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.10 Close ' Page No: ' 163'	
TRACING	FWVT Tested	**TLMC_E_171 - CIL - DOWNLINK IN CLOSE REQUEST**	
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.10 Close ' Page No: ' 163'	
TRACING	FWVT Tested	**TLMC_E_172 - CIL - RESET SERIAL NUMBER AFTER CLOSE**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close015\close015
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.10 Close ' Page No: ' 163'	
TRACING	FWVT Tested	**TLMC_E_173 - CIL - EMERGENCY DURING CLOSE REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close016\close016
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.10 Close ' Page No: ' 163'	
TRACING	FWVT Tested	**TLMC_E_174 - CIL - RAW MODE DOWNLINK (ATTACHED)**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode003\rawmode003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode002\rawmode002
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.11.1 Downlink Attached ' Page No: ' 163'	
TRACING	FWVT Tested	**TLMC_E_175 - CIL - REJECT RAW MODE DOWNLINK ATTACHED**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode002\rawmode002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode018\rawmode018
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.11.1 Downlink Attached ' Page No: ' 164'	
TRACING	FWVT Tested	**TLMC_E_176 - CIL - RAW MODE DOWNLINK (ECHO ON)**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode001\rawmode001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.11.2 Raw Mode - Echo On ' Page No: ' 165'	

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID** **Class** **Test Class** **Name**

TRACING FWVT Tested **TLMC_E_177 - CIL - RAW MODE DOWNLINK TO APPLICATION LAYER (ECHO ON)**
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode001\rawmode001

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.11.2 Raw Mode - Echo On ' Page No: ' 165'

TRACING FWVT Tested **TLMC_E_178 - CIL - RAW MODE UPLINK DATA**
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode001\rawmode001

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.11.2 Raw Mode - Echo On ' Page No: ' 165'

TRACING FWVT Tested **TLMC_E_179 - CIL - RAW MODE DOWNLINK (ECHO OFF)**
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode001\rawmode001

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.11.2 Raw Mode - Echo On ' Page No: ' 165'

TRACING FWVT Tested **TLMC_E_180 - CIL - UPLINK DATA INDICATION TIMERS**
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead003\memoryread003
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead027\memoryread027
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead025\memoryread025
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead026\memoryread026
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead028\memoryread028
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead029\memoryread029

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.3.1 Data Indication ' Page No: ' 167'

TRACING FWVT Tested **TLMC_E_181 - CIL - RESPONSE-APP-DATA-UPLINK**
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General003\general003

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.3.2 Unrequested Data Indication ' Page No: ' 167'

TRACING FWVT Tested **TLMC_E_182 - CIL - FINAL ICCC DOWNLINK REQUEST**
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite059\memorywrite059
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite030\memorywrite030
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite006\memorywrite006
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite060\memorywrite060
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite005\memorywrite005
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite029\memorywrite029
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite007\memorywrite007
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite033\memorywrite033

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.4.2 One ICCC Downlink Req, Multiple Uplink Data Indications ' Page No: ' 168'

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_183 - CIL - MULTIPLE TRANSACTION DOWNLINK - NOT FINAL**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite007\memorywrite007 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite036\memorywrite036 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite032\memorywrite032 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite061\memorywrite061 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite031\memorywrite031 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite062\memorywrite062
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.4.3 Multiple ICCC Downlink Requests For One Message ' Page No: ' 169'	
TRACING	FWVT Tested	**TLMC_E_184 - CIL - SERVICE DURATION TIMER**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite004\memorywrite004 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite034\memorywrite034 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite035\memorywrite035 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite036\memorywrite036 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite037\memorywrite037 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite033\memorywrite033 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite049\memorywrite049
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.4.3 Multiple ICCC Downlink Requests For One Message ' Page No: ' 170'	
TRACING	FWVT Tested	**TLMC_E_185 - CIL - EMERGENCY WHILE WAITING FOR DATA**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead030\memoryread030
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.5 Emergency Request While Waiting For Data Indication ' Page No: ' 170'	
TRACING	FWVT Tested	**TLMC_E_186 - CIL - MESSAGE TAG IN THE DOWNLINK MESSAGE**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite038\memorywrite038 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead047\memoryread047
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.6 Downlink and uplink Message Tag ' Page No: ' 171'	
TRACING	FWVT Tested	**TLMC_E_187 - CIL - UPLINK WITH NON-MATCHING TAG**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite038\memorywrite038
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.6 Downlink and uplink Message Tag ' Page No: ' 171'	

Tracing	Classification	Requirement Title	
	Decomp Class	Decomp Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_188 - CIL - EMERGENCY ICCC REQUEST DOWNLINK**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close016\close016 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open017\open017 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Diagnostic\Diagnostics020\diagnostics020 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead030\memoryread030 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName014\discovername014 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull017\discoverfull017 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch020\discovermatch020 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open008\open008 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_RawMode\RawMode003\rawmode003
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.13 Emergency ' Page No: ' 171'	
TRACING	FWVT Tested	**TLMC_E_189 - CIL - ICCC EMERGENCY REQUEST DOWNLINK**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency001\emergency001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency003\emergency003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency002\emergency002
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.13 Emergency ' Page No: ' 172'	
TRACING	FWVT Tested	**TLMC_E_192 - CIL - SECOND ICCC EMERGENCY REQUEST WHILE PROCESSING FIRST**	
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.13 Emergency ' Page No: ' 172'	
TRACING	FWVT Tested	**TLMC_E_193 - CIL - UPLINK EMERGENCY DATA INDICATION TIMEOUT**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency025\emergency025 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency001\emergency001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency027\emergency027 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency026\emergency026
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.13.1 Emergency Timeout Indication ' Page No: ' 172'	
TRACING	FWVT Tested	**TLMC_E_194 - CIL - RECOVERY-WHILE-WAITING-FOR-EMERGENCY**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency028\emergency028
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.13.2 Channel Recovery During Emergency ' Page No: ' 173'	

Tracing	Classification	Requirement Title	
	Decomp Class	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_E_195 - CIL - EMERGENCY COMPLETE REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency031\emergency031 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency035\emergency035 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency036\emergency036 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency029\emergency029 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency032\emergency032 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency037\emergency037 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency030\emergency030 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency033\emergency033 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency034\emergency034
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.13.3 Emergency Complete ' Page No: ' 173'	
TRACING	FWVT Tested	**TLMC_E_196 - CIL - EMERGENCY COMPLETE RESPONSE**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency002\emergency002
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.13.3 Emergency Complete ' Page No: ' 173'	
TRACING	FWVT Tested	**TLMC_E_197 - CIL - REMOTE STATISTICS ICCC REQUEST**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Diagnostic\Diagnostics002\diagnostics002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Diagnostic\Diagnostics005\diagnostics005 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Diagnostic\Diagnostics003\diagnostics003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Diagnostic\Diagnostics004\diagnostics004
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.14.1 Remote Statistics ' Page No: ' 174'	
TRACING	FWVT Tested	**TLMC_E_198 - CIL - LOCAL STATISTICS ICCC REQUEST**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Diagnostic\Diagnostics011\diagnostics011
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.14.2 Local Statistics ' Page No: ' 174'	
TRACING	FWVT Tested	**TLMC_E_199 - CIL - MAPPING STATISTICS ICCC REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Diagnostic\Diagnostics027\diagnostics027
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.15 Mapping Statistics ' Page No: ' 174'	
TRACING	FWVT Tested	**TLMC_E_200 - CIL - TELEMETRY ON STATUS TO APPLICATION**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open014\open014 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open016\open016 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open001\open001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.1 Telemetry On Status To Application ' Page No: ' 176'	

Tracing	Classification	Requirement Title	Test Case ID / Class / Test Class Name
	Decomp Class	Title	
TRACING	FWVT Tested	**TLMC_E_201 - CIL - TELEMETRY ON STATUS TO APPLICATION - CHANNEL RECOVERY**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery001\channelrecovery001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.1 Telemetry On Status To Application ' Page No: ' 176'	
TRACING	FWVT Tested	**TLMC_E_202 - CIL - TELEMETRY OFF MESSAGE APPLICATION**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close001\close001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open014\open014
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.2 Telemetry Off Message To Application ' Page No: ' 176'	
TRACING	FWVT Tested	**TLMC_E_203 - CIL -TELEMETRY OFF MESSAGE DURING CHANNEL RECOVERY**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery012\channelrecovery012
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.2 Telemetry Off Message To Application ' Page No: ' 176'	
TRACING	FWVT Tested	**TLMC_E_204 - CIL - CHART RECORDER DOWN ARROW - DOWNLINK STARTED**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite051\memorywrite051
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.3 Chart Recorder - Downlink And Uplink Status ' Page No: ' 177'	
TRACING	FWVT Tested	**TLMC_E_205 - CIL - CHART RECORDER UP ARROW - UPLINK CONFIRMED**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull026\discoverfull026 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close012\close012 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch041\discovermatch041 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close001\close001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close013\close013 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close014\close014 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull014\discoverfull014 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open025\open025 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Diagnostic\Diagnostics001\diagnostics001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName029\discovername029
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.4 Confirm Detected ' Page No: ' 177'	
TRACING	FWVT Tested	**TLMC_E_206 - CIL - CONFIRM-DETECTED-STATE-CHANGE-2**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead046\memoryread046
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.4 Confirm Detected ' Page No: ' 177'	

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class / Test Class Name
TRACING	FWVT Tested	**TLMC_E_207 - CIL - CONFIRM-DETECTED-STATE-CHANGE-3**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency001\emergency001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.4 Confirm Detected ' Page No: ' 177'	
TRACING	FWVT Tested	**TLMC_E_237 - SERVICE DURATION TOLERANCE**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite003\memorywrite003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Diagnostic\Diagnostics002\diagnostics002 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull006\discoverfull006 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency003\emergency003 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close002\close002 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Discover\DiscoverFull001\discoverfull001 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch006\discovermatch006 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open016\open016 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName006\discovername006
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.4 General Requirements ' Page No: ' 48'	
TRACING	FWVT Tested	**TLMC_E_243 - UPLINK PACKET RESPONSE TIMEOUT**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch008\discovermatch008 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverMatch041\discovermatch041 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverName030\discovername030 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open020\open020 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery018\channelrecovery018 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Discover\DiscoverFull010\discoverfull010 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open021\open021
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.4 General Requirements ' Page No: ' 48'	

Tracing	Classification	Requirement Title			
	Decomp Class	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_I_240 - POLLING OPERATION SCAN AND SLEEP TIMERS**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling016\polling016
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling050\polling050
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling040\polling040
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling017\polling017
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling025\polling025
			Status: **Up-to-date**	Rev: 2.4	Test: .\C_TCP_Polling\Polling047\polling047
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling049\polling049
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling051\polling051
			Status: **Up-to-date**	Rev: 2.4	Test: .\C_TCP_Polling\Polling046\polling046
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling024\polling024
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling039\polling039
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Polling\Polling048\polling048
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Polling\Polling018\polling018
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling026\polling026
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling045\polling045
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling052\polling052
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.1 Requirements ' Page No: ' 55'			
TRACING	FWVT Tested	**TLMC_I_241 - POLLING ENABLING CRITERIA**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling034\polling034
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling043\polling043
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling054\polling054
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling016\polling016
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling044\polling044
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Polling\Polling004\polling004
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling005\polling005
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.1 Requirements ' Page No: ' 55'			

Tracing **Classification** **Requirement Title**

Decomp **Class** **Title** **Test Case ID** **Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_I_242 - POLLING TERMINATION**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Polling\Polling041\polling041
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Polling\Polling042\polling042
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Polling\Polling058\polling058
			Status:	**Up-to-date**	Rev:	2.3	Test:	.\C_TCP_Polling\Polling006\polling006
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_Polling\Polling057\polling057
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_Polling\Polling007\polling007
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.1 Requirements ' Page No: ' 55'						
TRACING	FWVT Tested	**TLMC_E_218 - MEDICAL EVENT - C INTERFACE PARAMETERS**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_MedicalEvent\MedicalEvent002\medicalevent002
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_MedicalEvent\MedicalEvent003\medicalevent003
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 117'						
TRACING	FWVT Tested	**TLMC_IE_219 - MEDICAL EVENT - UPLINK PACKET FORMAT**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 117'						
TRACING	FWVT Tested	**TLMC_E_220 - MEDICAL EVENT - MEDICAL EVENT REQUEST**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 117'						
TRACING	FWVT Tested	**TLMC_E_221 - MEDICAL EVENT - MEDICAL EVENT UPLINK**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent004\medicalevent004
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 117'						
TRACING	FWVT Tested	**TLMC_E_222 - MEDICAL EVENT - INVALID MEDICAL EVENT UPLINK**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent004\medicalevent004
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 117'						
TRACING	FWVT Tested	**TLMC_E_223 - MEDICAL EVENT - ANTENNA TIMEOUT**						
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 117'						
TRACING	FWVT Tested	**TLMC_E_224 - MEDICAL EVENT - LISTEN BEFORE TALK TIMEOUT**	Status:	**Up-to-date**	Rev:	2.2	Test:	.\C_TCP_MedicalEvent\MedicalEvent004\medicalevent004
			Status:	**Up-to-date**	Rev:	2.1	Test:	.\C_TCP_MedicalEvent\MedicalEvent005\medicalevent005

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID Class	Test Class	Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 118'			
TRACING	FWVT Tested	**TLMC_E_228 - MEDICAL EVENT - RESPONSE TO C INTERFACE LAYER**	Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.2 Rev: 2.1 Rev: 2.1 Rev: 2.1 Rev: 2.1 Rev: 2.1 Rev: 2.1 Rev: 2.1	Test: .\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001 Test: .\C_TCP_MedicalEvent\MedicalEvent014\medicalevent014 Test: .\C_TCP_MedicalEvent\MedicalEvent010\medicalevent010 Test: .\C_TCP_MedicalEvent\MedicalEvent013\medicalevent013 Test: .\C_TCP_MedicalEvent\MedicalEvent008\medicalevent008 Test: .\C_TCP_MedicalEvent\MedicalEvent009\medicalevent009 Test: .\C_TCP_MedicalEvent\MedicalEvent011\medicalevent011 Test: .\C_TCP_MedicalEvent\MedicalEvent012\medicalevent012
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 118'			
TRACING	FWVT Tested	**TLMC_I_231 - MEDICAL EVENT - INVALID OPEN DOWNLINK**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_MedicalEvent\MedicalEvent016\medicalevent016
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 118'			
TRACING	FWVT Tested	**TLMC_I_232 - MEDICAL EVENT - MEDICAL EVENT DURATION**	Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.2 Rev: 2.2 Rev: 2.2 Rev: 2.2	Test: .\C_TCP_MedicalEvent\MedicalEvent015\medicalevent015 Test: .\C_TCP_MedicalEvent\MedicalEvent017\medicalevent017 Test: .\C_TCP_MedicalEvent\MedicalEvent018\medicalevent018 Test: .\C_TCP_MedicalEvent\MedicalEvent016\medicalevent016
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 119'			
TRACING	FWVT Tested	**TLMC_E_236 - CIL - OPEN STOP HIBERNATE REQUEST**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Open\Open014\open014
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.8 Waveform Enable ' Page No: ' 162'			
TRACING	FWVT Tested	**TLMC_E_235 - CIL - MEDICAL EVENT REQUEST AND RESPONSE**	Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.1 Rev: 2.1 Rev: 2.2	Test: .\C_TCP_MedicalEvent\MedicalEvent003\medicalevent003 Test: .\C_TCP_MedicalEvent\MedicalEvent002\medicalevent002 Test: .\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.6 Medical Event ' Page No: ' 179'			
TRACING	FWVT Tested	**TLMC_E_244 - RF HEAD LEDS**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.4 RF Head LEDS For Telemetry C ' Page No: ' 152'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_IE_246 - CIL - PROGRAMMING HEAD PRESENCE OR ABSENCE STATUS**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open028\open028 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open029\open029
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.5 Implant Programming Head Status ' Page No: ' 178'	
TRACING	FWVT Tested	**TLMC_E_247 - CIL - PROGRAMMING HEAD STATUS TO APPLICATION LAYER**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open028\open028 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Open\Open001\open001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.5 Implant Programming Head Status ' Page No: ' 178'	
TRACING	FWVT Tested	**TLMC_E_215 - OPEN - HIBERNATE DOWNLINK REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open023\open023 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open014\open014 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open017\open017 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open024\open024 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open027\open027
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.2 Open - Hibernate Requirements ' Page No: ' 81'	
TRACING	FWVT Tested	**TLMC_I_216 - OPEN - HIBERNATE UPLINK RESPONSE**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open023\open023 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Open\Open014\open014
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.8.2 Open - Hibernate Requirements ' Page No: ' 82'	
TRACING	FWVT Tested	**TLMC_I_076 - DIAGNOSTIC FILLER - UPLINK FILLER PACKETS**	Status: **Up-to-date** Rev: 2.3 Test: .\C_TCP_Diagnostic\Diagnostics030\diagnostics030
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.10.5 Diagnostic Filler Packet ' Page No: ' 104'	
TRACING	FWVT Tested	**TLMC_E_099 - CLOSE - RECOVERY DURING CLOSE**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Close\Close008\close008
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.13.1 Requirements ' Page No: ' 115'	
TRACING	FWVT Tested	**TLMC_I_144 - DATA TRANSFER MODE - VALIDATE TRANSFER MODE AND RESPONSE**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\TransferMode001\transfermode001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.4.3 Implant ' Page No: ' 150'	

Tracing	Classification	Requirement Title				
Class	Decomp	Title		Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_145 - RESET SERVICE - DOWNLINK PACKET WITH RESET BIT**	Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryRead035\memoryread035
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryRead036\memoryread036
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryRead049\memoryread049
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryRead037\memoryread037
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryRead048\memoryread048
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryWrite038\memorywrite038
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryWrite041\memorywrite041
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Emergency\Emergency014\emergency014
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Emergency\Emergency015\emergency015
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryWrite048\memorywrite048
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Emergency\Emergency012\emergency012
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryWrite029\memorywrite029
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryRead050\memoryread050
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryRead051\memoryread051
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Data\MemoryWrite040\memorywrite040
			Status: **Up-to-date**	Rev: 2.1	Test:	.\C_TCP_Emergency\Emergency013\emergency013

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.5.1 Requirements ' Page No: ' 151'

Tracing	Classification	Requirement Title			
Decomp Class		Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_E_146 - RESET SERVICE - DOWNLINK RESET REQUEST RETRANSMISSION**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead038\memoryread038
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency014\emergency014
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead037\memoryread037
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite042\memorywrite042
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead039\memoryread039
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead040\memoryread040
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency011\emergency011
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryRead036\memoryread036
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite043\memorywrite043
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite044\memorywrite044
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite040\memorywrite040
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\MemoryWrite041\memorywrite041
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency005\emergency005
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Emergency\Emergency015\emergency015

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.5.1 Requirements ' Page No: ' 151'

Tracing	Classification	Requirement Title	
Decomp Class		Title	Test Case ID Class / Test Class Name
TRACING	FWVT Tested	**TLMC_I_147 - RESET SERVICE - DOWNLINK RESET REQUEST VALIDATION**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite055\memorywrite055
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead041\memoryread041
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite032\memorywrite032
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite038\memorywrite038
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite054\memorywrite054
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency016\emergency016
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency018\emergency018
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead035\memoryread035
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite029\memorywrite029
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency017\emergency017
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency019\emergency019
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead049\memoryread049
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite053\memorywrite053
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite052\memorywrite052
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryWrite056\memorywrite056
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.5.1 Requirements ' Page No: ' 151'	
TRACING	FWVT Tested	**TLMC_E_148 - RESET SERVICE - OPERATION SUCCESSFUL**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead035\memoryread035
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency022\emergency022
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Data\MemoryRead042\memoryread042
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency013\emergency013
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency020\emergency020
			Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Emergency\Emergency021\emergency021
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.5.1 Requirements ' Page No: ' 152'	
TRACING	FWVT Tested	**TLMC_E_156 - SIGNAL - STRENGTH INDICATOR**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.4.1 Signal Strength Indicator ' Page No: ' 153'	
TRACING	FWVT Tested	**TLMC_E_210 - CIL - DISABLE DOWNLINK ICCC REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General001\general001

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.16 Enable/disable Downlinks ' Page No: ' 174'	
TRACING	FWVT Tested	**TLMC_E_211 - CIL - ENABLE DOWNLINK ICCC REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General001\general001
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.16 Enable/disable Downlinks ' Page No: ' 175'	
TRACING	FWVT Tested	**TLMC_I_035 - POLLING - SERVICE REQUEST FROM C INTERFACE LAYER**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling057\polling057 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling058\polling058
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.3 Requirements ' Page No: ' 56'	
TRACING	FWVT Tested	**TLMC_I_036 - POLLING - SCANNING CHANNEL AND SERVICE DURATION TIME**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Polling\Polling045\polling045 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Polling\Polling062\polling062 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Polling\Polling034\polling034 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Polling\Polling059\polling059 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling060\polling060 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Polling\Polling061\polling061 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_Polling\Polling016\polling016 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Polling\Polling043\polling043 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Polling\Polling054\polling054
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.3 Requirements ' Page No: ' 57'	
TRACING	FWVT Tested	**TLMC_E_248 - CIL - REACTION TO APPLICATION SYSTEM ERROR**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General023\general023 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General022\general022 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General024\general024 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General025\general025
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.4 Application System Error ' Page No: ' 179'	
TRACING	FWVT Tested	**TLMC_I_037 - POLLING - INVALID DOWNLINK PACKET RECEIVED**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.3 Requirements ' Page No: ' 57'	

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_I_038 - POLLING - SCAN TERMINATION**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling051\polling051
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling058\polling058
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling025\polling025
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling040\polling040
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling057\polling057
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Polling\Polling066\polling066
			Status: **Up-to-date**	Rev: 2.4	Test: .\C_TCP_Polling\Polling046\polling046
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Polling\Polling067\polling067
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling050\polling050
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling065\polling065
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Polling\Polling068\polling068
			Status: **Up-to-date**	Rev: 2.4	Test: .\C_TCP_Polling\Polling047\polling047
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Polling\Polling063\polling063
			Status: **Up-to-date**	Rev: 2.3	Test: .\C_TCP_Polling\Polling064\polling064

' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.3 Requirements ' Page No: ' 58'

TRACING	FWVT Tested	**TLMC_I_110 - ARQ - MEMORY DATA ACKNOWLEDGEMENT NOT RECEIVED**	Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ013\arq013
			Status: **Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\ARQ027\arq027

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.1.3 Implant ' Page No: ' 130'

TRACING	FWVT Tested	**TLMC_I_252 - POLLING - RESTART**	Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling084\polling084
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling086\polling086
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling082\polling082
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling085\polling085
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling083\polling083
			Status: **Up-to-date**	Rev: 2.2	Test: .\C_TCP_Polling\Polling087\polling087

' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.1 Requirements ' Page No: ' 55'

TRACING	FWVT Tested	**TLMC_I_256 - POLLING - ACKNOWLEDGEMENT RESPONSE TO C INTERFACE LAYER**

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class / Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.3 Requirements ' Page No: ' 57'	
TRACING	FWVT Tested	**TLMC_I_039 - POLLING - SERVICE RESPONSE TO C INTERFACE LAYER**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.6.3 Requirements ' Page No: ' 58'	
TRACING	FWVT Tested	**TLMC_I_229 - MEDICAL EVENT - MEDICAL EVENT REQUEST RESPONSE**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 118'	
TRACING	FWVT Tested	**TLMC_I_230 - MEDICAL EVENT - UPLINK TIMEOUT**	Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_MedicalEvent\MedicalEvent004\medicalevent004 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_MedicalEvent\MedicalEvent024\medicalevent024 Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_MedicalEvent\MedicalEvent015\medicalevent015
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 118'	
TRACING	FWVT Tested	**TLMC_I_257 - MEDICAL EVENT - RESPONSE TO C INTERFACE LAYER**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.14.1 Requirements ' Page No: ' 119'	
TRACING	FWVT Tested	**TLMC_I_258 - PATIENT NAME REQUEST**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.15.1 Requirements ' Page No: ' 119'	
TRACING	FWVT Tested	**TLMC_I_259 - PATIENT NAME ACKNOWLEDGEMENT RESPONSE TO C INTERFACE LAYER**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.41' SECTION: '5.2.15.1 Requirements ' Page No: ' 120'	
TRACING	FWVT Tested	**TLMC_E_245 - CIL - UPDATE DEVICE STATUS ICCC REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General020\general020 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General017\general017 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General021\general021 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General018\general018 Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_General\General019\general019
		' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.17 Update Device Status ' Page No: ' 175'	

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class Name**

TRACING FWVT Tested **TLMC_E_260 - CIL - MEDICAL EVENT UPLINK TO APPLICATION**

Status: **Up-to-date** Rev: 2.2 Test: .\C_TCP_MedicalEvent\MedicalEvent001\medicalevent001

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.3.5 Implant Programming Head Status ' Page No: ' 178'

TRACING FWVT Tested **TLMC_E_261 - CH RECOVERY - TOO MANY SAME CHANNEL RECOVERIES**

Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery053\channelrecovery053
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery058\channelrecovery058
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery048\channelrecovery048
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery050\channelrecovery050
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery057\channelrecovery057
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery045\channelrecovery045
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery052\channelrecovery052
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery047\channelrecovery047
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery055\channelrecovery055
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery046\channelrecovery046
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery049\channelrecovery049
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery051\channelrecovery051
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery054\channelrecovery054
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery056\channelrecovery056
Status: **Up-to-date** Rev: 2.1 Test: .\C_TCP_Channel\ChannelRecovery059\channelrecovery059

' FILE_NAME: 'internal.fm' FILE_REV: '1.42' SECTION: '5.3.3.2 Instrument ' Page No: ' 141'

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_E_262 - CIL - RETRY DOWNLINK**	Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data004\data004
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data011\data011
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data006\data006
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data009\data009
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data001\data001
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data002\data002
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data007\data007
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data003\data003
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data005\data005
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data008\data008
			Status:	**Up-to-date**	Rev: 2.1	Test: .\C_TCP_Data\Data010\data010

' FILE_NAME: 'c_intfc.fm' FILE_REV: '1.35' SECTION: '6.2.12.4.2 One ICCC Downlink Req, Multiple Uplink Data Indications '
Page No: ' 169'

SPEC DOC/REV **TELM/C**

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_MS_003 - PACKET TYPES**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_General\m_General005\m_general005
			Status: **Up-to-date**	Rev: 2.11	Test: .\M_TCP_Open\m_Open001\m_open001
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Close\m_Close001\m_close001
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Security\m_Security004\m_security004
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_ARQ006\m_arq006
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_General\m_General015\m_general015
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_MedicalEvent\m_MedicalEvent001\m_medicalevent001
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Security\m_Security011\m_security011
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryWrite006\m_memorywrite006
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Discover\m_DiscoverFull032\m_discoverfull032
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_General\m_General003\m_general003
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Security\m_Security021\m_security021
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Security\m_Security020\m_security020
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead003\m_memoryread003
			Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Network\m_Network001\m_network001

' FILE_NAME: 'datalink.fm' FILE_REV: '1.34' SECTION: '5.1.2.4 Packet Type ' Page No: ' 98'

TRACING	FWVT Tested	**TLMC_M_100 - ARQ - STOP AND WAIT MASTER ACKNOWLEDGMENT**	Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryWrite039\m_memorywrite039
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryWrite008\m_memorywrite008
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_ARQ001\m_arq001
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_ARQ004\m_arq004
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_ARQ003\m_arq003
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Data\m_ARQ002\m_arq002
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_ARQ005\m_arq005

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.2 MASTER ' Page No: ' 277'

Tracing Class	Classification Decomp	Requirement Title Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_MS_101 - ARQ - MASTER ARQ PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Data\m_ARQ008\m_arq008
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Data\m_ARQ015\m_arq015
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Data\m_ARQ007\m_arq007
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_ARQ011\m_arq011
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_ARQ010\m_arq010
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_ARQ009\m_arq009
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_ARQ006\m_arq006
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.2 MASTER ' Page No: ' 281'			
TRACING	FWVT Tested	**TLMC_M_102 - ARQ - UPDATING OF ARQ INFORMATION**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_ARQ012\m_arq012
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.2 MASTER ' Page No: ' 281'			
TRACING	FWVT Tested	**TLMC_M_103 - ARQ - SLAVE MEMORY DATA PACKET ACKNOWLEDGMENT**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_ARQ014\m_arq014
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_ARQ012\m_arq012
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_ARQ013\m_arq013
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Data\m_ARQ015\m_arq015
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_ARQ016\m_arq016
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.2 MASTER ' Page No: ' 282'			
TRACING	FWVT Tested	**TLMC_M_104 - ARQ - SLAVE ACK NOT RECEIVED**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_ARQ016\m_arq016
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_ARQ017\m_arq017
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Data\m_ARQ018\m_arq018
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.2 MASTER ' Page No: ' 282'			
TRACING	FWVT Tested	**TLMM_M_357 - CH MAPPING - SELECT CHANNEL**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.2.2 MAPPING SYSTEM PARTITIONING ' Page No: ' 288'			
TRACING	FWVT Tested	**TLMM_M_358 - CH MAPPING - DIAGNOSTIC COLLECTION**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.2.3 DIAGNOSTIC MAPPING ' Page No: ' 288'			

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_M_120 - CH RECOVERY - TRIGGER CHANNEL RECOVERY**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery001\m_channelrecovery001
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery011\m_channelrecovery011
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery002\m_channelrecovery002
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery005\m_channelrecovery005
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery009\m_channelrecovery009
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Channel\m_ChannelRecovery004\m_channelrecovery004
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery003\m_channelrecovery003
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery033\m_channelrecovery033
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 291'			
TRACING	FWVT Tested	**TLMC_M_121 - CH RECOVERY - START SAME CHANNEL RECOVERY**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery006\m_channelrecovery006
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery001\m_channelrecovery001
			Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Channel\m_ChannelRecovery007\m_channelrecovery007
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery032\m_channelrecovery032
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery008\m_channelrecovery008
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 291'			
TRACING	FWVT Tested	**TLMC_M_122 - CH RECOVERY - SAME CHANNEL RECOVERY OPEN MASTER PACKET**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery009\m_channelrecovery009
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery011\m_channelrecovery011
			Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Channel\m_ChannelRecovery010\m_channelrecovery010
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery012\m_channelrecovery012
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery013\m_channelrecovery013
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 293'			
TRACING	FWVT Tested	**TLMC_M_123 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL**	Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Channel\m_ChannelRecovery014\m_channelrecovery014
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery001\m_channelrecovery001
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery012\m_channelrecovery012
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery015\m_channelrecovery015
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 293'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID Class		Test Class Name
TRACING	FWVT Tested	**TLMC_M_124 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery012\m_channelrecovery012
			Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Channel\m_ChannelRecovery014\m_channelrecovery014
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery015\m_channelrecovery015
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 295'			
TRACING	FWVT Tested	**TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**	Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Channel\m_ChannelRecovery014\m_channelrecovery014
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery064\m_channelrecovery064
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery068\m_channelrecovery068
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery045\m_channelrecovery045
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery020\m_channelrecovery020
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery069\m_channelrecovery069
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery065\m_channelrecovery065
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery066\m_channelrecovery066
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery067\m_channelrecovery067
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 296'			
TRACING	FWVT Tested	**TLMC_M_261 - CH RECOVERY - TOO MANY SAME CHANNEL RECOVERIES**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery048\m_channelrecovery048
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery053\m_channelrecovery053
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery055\m_channelrecovery055
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery045\m_channelrecovery045
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Channel\m_ChannelRecovery047\m_channelrecovery047
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery049\m_channelrecovery049
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery050\m_channelrecovery050
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Channel\m_ChannelRecovery051\m_channelrecovery051
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery046\m_channelrecovery046
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery054\m_channelrecovery054
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 297'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_M_125 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery012\m_channelrecovery012
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery015\m_channelrecovery015
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery017\m_channelrecovery017
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery020\m_channelrecovery020
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery016\m_channelrecovery016
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery018\m_channelrecovery018
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 298'			
TRACING	FWVT Tested	**TLMC_M_126 - REMAP CHANNEL - UNSPECIFIED CHANNEL RECOVERY TIMER EXPIRATION**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery038\m_channelrecovery038
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery020\m_channelrecovery020
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 298'			
TRACING	FWVT Tested	**TLMC_M_127 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery020\m_channelrecovery020
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 299'			
TRACING	FWVT Tested	**TLMM_M_360 - CH RECOVERY - NATIVE TIMEOUT**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery073\m_channelrecovery073
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery064\m_channelrecovery064
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery072\m_channelrecovery072
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 299'			
TRACING	FWVT Tested	**TLMC_M_128 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery012\m_channelrecovery012
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery042\m_channelrecovery042
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.2 MASTER ' Page No: ' 299'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_S_129 - CH RECOVERY - TRIGGER CHANNEL RECOVERY**	Status:	**Up-to-date**	Rev:	2.3	Test:	.\M_TCP_Channel\m_ChannelRecovery021\m_channelrecovery021
			Status:	**Up-to-date**	Rev:	2.3	Test:	.\M_TCP_Channel\m_ChannelRecovery022\m_channelrecovery022
			Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_Channel\m_ChannelRecovery023\m_channelrecovery023
			Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_Channel\m_ChannelRecovery043\m_channelrecovery043
			Status:	**Up-to-date**	Rev:	2.8	Test:	.\M_TCP_Channel\m_ChannelRecovery014\m_channelrecovery014
			Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_PowerInhibit\m_Power005\m_power005
			Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_Channel\m_ChannelRecovery009\m_channelrecovery009
			Status:	**Up-to-date**	Rev:	2.4	Test:	.\M_TCP_Channel\m_ChannelRecovery024\m_channelrecovery024

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 301'

TRACING	FWVT Tested	**TLMC_S_130 - CH RECOVERY - START SAME CHANNEL RECOVERY**	Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_Channel\m_ChannelRecovery009\m_channelrecovery009
			Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_Channel\m_ChannelRecovery043\m_channelrecovery043
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\M_TCP_Channel\m_ChannelRecovery025\m_channelrecovery025
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\M_TCP_Channel\m_ChannelRecovery026\m_channelrecovery026
			Status:	**Up-to-date**	Rev:	2.3	Test:	.\M_TCP_Channel\m_ChannelRecovery017\m_channelrecovery017
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\M_TCP_Channel\m_ChannelRecovery027\m_channelrecovery027
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\M_TCP_Channel\m_ChannelRecovery029\m_channelrecovery029
			Status:	**Up-to-date**	Rev:	2.2	Test:	.\M_TCP_Channel\m_ChannelRecovery028\m_channelrecovery028

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 301'

TRACING	FWVT Tested	**TLMC_S_131 - CH RECOVERY - SAME CHANNEL RECOVERY INVOKE SCAN**	Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_Channel\m_ChannelRecovery040\m_channelrecovery040
			Status:	**Up-to-date**	Rev:	2.6	Test:	.\M_TCP_Channel\m_ChannelRecovery031\m_channelrecovery031
			Status:	**Up-to-date**	Rev:	2.8	Test:	.\M_TCP_Channel\m_ChannelRecovery010\m_channelrecovery010

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 302'

TRACING	FWVT Tested	**TLMC_S_132 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL**	Status:	**Up-to-date**	Rev:	2.5	Test:	.\M_TCP_Channel\m_ChannelRecovery001\m_channelrecovery001
			Status:	**Up-to-date**	Rev:	2.4	Test:	.\M_TCP_Channel\m_ChannelRecovery032\m_channelrecovery032
			Status:	**Up-to-date**	Rev:	2.8	Test:	.\M_TCP_Channel\m_ChannelRecovery010\m_channelrecovery010

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 302'

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_S_133 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Channel\m_ChannelRecovery016\m_channelrecovery016
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery012\m_channelrecovery012
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 303'			
TRACING	FWVT Tested	**TLMC_S_134 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY**	Status: **Up-to-date**	Rev: 2.9	Test: .\M_TCP_Channel\m_ChannelRecovery063\m_channelrecovery063
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery061\m_channelrecovery061
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Channel\m_ChannelRecovery062\m_channelrecovery062
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 303'			
TRACING	FWVT Tested	**TLMC_S_135 - CH RECOVERY - UCR TIMER ACTIVE NON OPEN MASTER PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Channel\m_ChannelRecovery041\m_channelrecovery041
			Status: **Up-to-date**	Rev: 2.10	Test: .\M_TCP_Channel\m_ChannelRecovery030\m_channelrecovery030
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 303'			
TRACING	FWVT Tested	**TLMC_S_136 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Channel\m_ChannelRecovery040\m_channelrecovery040
			Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Channel\m_ChannelRecovery010\m_channelrecovery010
			Status: **Up-to-date**	Rev: 2.10	Test: .\M_TCP_Channel\m_ChannelRecovery030\m_channelrecovery030
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery038\m_channelrecovery038
			Status: **Up-to-date**	Rev: 2.9	Test: .\M_TCP_Channel\m_ChannelRecovery063\m_channelrecovery063
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 304'			
TRACING	FWVT Tested	**TLMM_S_361 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Channel\m_ChannelRecovery031\m_channelrecovery031
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery018\m_channelrecovery018
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Channel\m_ChannelRecovery071\m_channelrecovery071
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Channel\m_ChannelRecovery070\m_channelrecovery070
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Channel\m_ChannelRecovery039\m_channelrecovery039
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.3.3 SLAVE ' Page No: ' 304'			

Tracing **Classification** **Requirement Title**

Decomp
Class **Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_M_139 - DATA TRANSFER MODE - MODE TRANSITIONS**	Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Data\m_TransferMode001\m_transfermode001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_TransferMode004\m_transfermode004
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Data\m_TransferMode003\m_transfermode003
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_TransferMode007\m_transfermode007
			Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Data\m_TransferMode017\m_transfermode017
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_TransferMode002\m_transfermode002
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.4.2 MASTER ' Page No: ' 307'			
TRACING	FWVT Tested	**TLMC_M_140 - DATA TRANSFER MODE - NOMINAL TO LOW ENERGY MODE TRANSITIONS.**	Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Data\m_TransferMode001\m_transfermode001
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_TransferMode010\m_transfermode010
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.4.2 MASTER ' Page No: ' 307'			
TRACING	FWVT Tested	**TLMC_M_141 - DATA TRANSFER MODE - NEGATIVE PREAMBLE RATIO IN LOW ENERGY MODE**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Data\m_TransferMode018\m_transfermode018
			Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Data\m_TransferMode017\m_transfermode017
			Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Data\m_TransferMode001\m_transfermode001
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_TransferMode015\m_transfermode015
			Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Data\m_TransferMode016\m_transfermode016
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_TransferMode004\m_transfermode004
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.4.2 MASTER ' Page No: ' 308'			
TRACING	FWVT Tested	**TLMC_M_142 - DATA TRANSFER MODE - LOW ENERGY TO NOMINAL TRANSITION**	Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Data\m_TransferMode001\m_transfermode001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.4.2 MASTER ' Page No: ' 308'			

Decomp Class | Title | Test Case ID | Class | Test Class | Name

Tracing	Classification	Requirement Title			
TRACING	FWVT Tested	**TLMC_M_143 - DATA TRANSFER MODE - TRANSITION TO NEW MODE ON MASTER PACKET**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_TransferMode006\m_transfermode006
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Data\m_TransferMode008\m_transfermode008
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_TransferMode009\m_transfermode009
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Data\m_TransferMode019\m_transfermode019
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_TransferMode007\m_transfermode007
			Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Data\m_TransferMode001\m_transfermode001
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_TransferMode005\m_transfermode005
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.4.2 MASTER ' Page No: ' 308'			
TRACING	FWVT Tested	**TLMC_S_144 - DATA TRANSFER MODE - VALIDATE TRANSFER MODE AND RESPONSE**	Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Data\m_TransferMode001\m_transfermode001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_TransferMode011\m_transfermode011
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_TransferMode014\m_transfermode014
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_TransferMode005\m_transfermode005
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_TransferMode013\m_transfermode013
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_TransferMode012\m_transfermode012
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.4.3 SLAVE ' Page No: ' 310'			
TRACING	FWVT Tested	**TLMM_M_362 - RESET SERVICE - INVOKE SERVICE 1**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Reset\m_Reset001\m_reset001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.5.1 Requirements ' Page No: ' 311'			
TRACING	FWVT Tested	**TLMM_M_363 - RESET SERVICE - INVOKE SERVICE 2**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Reset\m_Reset002\m_reset002
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.5.1 Requirements ' Page No: ' 311'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_M_145 - RESET SERVICE - MASTER PACKET WITH RESET BIT**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryRead049\m_memoryread049
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency012\m_emergency012
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency014\m_emergency014
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite029\m_memorywrite029
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite041\m_memorywrite041
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryWrite038\m_memorywrite038
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency013\m_emergency013
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency015\m_emergency015
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead035\m_memoryread035
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead037\m_memoryread037
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryRead051\m_memoryread051
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite040\m_memorywrite040
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite048\m_memorywrite048
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.5.1 Requirements ' Page No: ' 312'			
TRACING	FWVT Tested	**TLMC_M_146 - RESET SERVICE - MASTER RESET REQUEST RETRANSMISSION**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency014\m_emergency014
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead037\m_memoryread037
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency015\m_emergency015
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite042\m_memorywrite042
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite040\m_memorywrite040
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency005\m_emergency005
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite041\m_memorywrite041
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryRead038\m_memoryread038
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.5.1 Requirements ' Page No: ' 312'			

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class** **Name**

TRACING	FWVT Tested	**TLMC_S_147 - RESET SERVICE - RESET REQUEST VALIDATION**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_MemoryWrite032\m_memorywrite032
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryWrite052\m_memorywrite052
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite055\m_memorywrite055
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryWrite056\m_memorywrite056
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency017\m_emergency017
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead035\m_memoryread035
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryWrite038\m_memorywrite038
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_MemoryWrite054\m_memorywrite054
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency018\m_emergency018
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead041\m_memoryread041
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite053\m_memorywrite053
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite029\m_memorywrite029
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency016\m_emergency016
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryRead049\m_memoryread049
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency019\m_emergency019

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.5.1 Requirements ' Page No: ' 312'

TRACING	FWVT Tested	**TLMC_M_148 - RESET SERVICE - OPERATION SUCCESSFUL**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead035\m_memoryread035
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead042\m_memoryread042
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency022\m_emergency022
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency013\m_emergency013
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency021\m_emergency021
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Emergency\m_Emergency020\m_emergency020

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.5.1 Requirements ' Page No: ' 312'

TRACING	FWVT Tested	**TLMM_M_364 - WIRELESS WAKEUP - INITIATE INSTIGATOR WAKEUP**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_WirelessWakeup\m_Wakeup008\m_wakeup008

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.2 Requirements ' Page No: ' 317'

Tracing	Classification	Requirement Title			
	Class	Decomp Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMM_MS_365 - WIRELESS WAKEUP - WAKEUP PACKET**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_WirelessWakeup\m_Wakeup011\m_wakeup011
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_WirelessWakeup\m_Wakeup008\m_wakeup008
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_WirelessWakeup\m_Wakeup002\m_wakeup002
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_WirelessWakeup\m_Wakeup012\m_wakeup012
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_WirelessWakeup\m_Wakeup009\m_wakeup009
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_WirelessWakeup\m_Wakeup010\m_wakeup010
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 318'			
TRACING	FWVT Tested	**TLMM_MS_366 - WIRELESS WAKEUP -WAKEUP HEADER BYTE**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_WirelessWakeup\m_Wakeup008\m_wakeup008
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_WirelessWakeup\m_Wakeup011\m_wakeup011
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_WirelessWakeup\m_Wakeup009\m_wakeup009
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 318'			
TRACING	FWVT Tested	**TLMM_MS_367 - WIRELESS WAKEUP -WAKEUP SEQUENCE NUMBER BYTES**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 320'			
TRACING	FWVT Tested	**TLMM_MS_368 - WIRELESS WAKEUP -WAKEUP STATION ID BYTES**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_WirelessWakeup\m_Wakeup002\m_wakeup002
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 320'			
TRACING	FWVT Tested	**TLMM_MS_369 - WIRELESS WAKEUP -WAKEUP RECEPTOR BIT MAP BYTE**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_WirelessWakeup\m_Wakeup010\m_wakeup010
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 321'			
TRACING	FWVT Tested	**TLMM_MS_370 - WIRELESS WAKEUP -WAKEUP CONFIGURATION BYTES**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_WirelessWakeup\m_Wakeup012\m_wakeup012
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 321'			
TRACING	FWVT Tested	**TLMM_MS_371 - WIRELESS WAKEUP -WAKEUP USER DATA BYTES**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_WirelessWakeup\m_Wakeup011\m_wakeup011

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID Class** **Test Class Name**

		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 322'	
TRACING	FWVT Tested	**TLMM_M_373 - WIRELESS WAKEUP -WAKEUP INTERVAL SYNC START TIME**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_WirelessWakeup\m_Wakeup014\m_wakeup014
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.6 Requirements ' Page No: ' 324'	
TRACING	FWVT Tested	**TLMM_M_374 - WIRELESS WAKEUP -WAKEUP INITIAL CCA**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.6 Requirements ' Page No: ' 324'	
TRACING	FWVT Tested	**TLMM_M_375 - WIRELESS WAKEUP -WAKEUP WITH CCA BREAKS**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.6 Requirements ' Page No: ' 325'	
TRACING	FWVT Tested	**TLMM_S_377 - WIRELESS WAKEUP - WAKEUP SNIFF FREQUENCY RANGE**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 326'	
TRACING	FWVT Tested	**TLMM_S_380 - WIRELESS WAKEUP - WAKEUP COMPLETE INDICATION**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_WirelessWakeup\m_Wakeup003\m_wakeup003
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 327'	
TRACING	FWVT Tested	**TLMM_S_381 - WIRELESS WAKEUP - WAKEUP IN PROGRESS USER DATA AVAILABLE INDICATION**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_WirelessWakeup\m_Wakeup006\m_wakeup006 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_WirelessWakeup\m_Wakeup005\m_wakeup005 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_WirelessWakeup\m_Wakeup007\m_wakeup007
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 327'	
TRACING	FWVT Tested	**TLMM_S_382 - WIRELESS WAKEUP - WAKEUP COMPLETE INDICATION ONLY**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_WirelessWakeup\m_Wakeup003\m_wakeup003 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_WirelessWakeup\m_Wakeup005\m_wakeup005
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 328'	
TRACING	FWVT Tested	**TLMM_S_383 - WIRELESS WAKEUP - INDICATIONS**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_WirelessWakeup\m_Wakeup006\m_wakeup006 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_WirelessWakeup\m_Wakeup005\m_wakeup005 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_WirelessWakeup\m_Wakeup004\m_wakeup004

Tracing	Classification	Requirement Title			
	Decomp Class	Title	Test Case ID	Class	Test Class Name
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 328'			
TRACING	FWVT Tested	**TLMM_M_386 - ANTENNA SELECTION - TRANSMIT**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.0 Rev: 2.3	Test: .\M_TCP_AntennaSelection\m_Antenna002\m_antenna002 Test: .\M_TCP_AntennaSelection\m_Antenna001\m_antenna001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.7 ANTENNA SELECTION ' Page No: ' 329'			
TRACING	FWVT Tested	**TLMC_M_156 - SIGNAL - STRENGTH INDICATOR**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.8 Signal Strength Indicator ' Page No: ' 330'			
TRACING	FWVT Tested	**TLMM_M_387 - SIGNAL STRENGTH INDICATOR SETUP**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.8 Signal Strength Indicator ' Page No: ' 331'			
TRACING	FWVT Tested	**TLMM_MS_391 - PLL LOCK ERROR**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.9.1 REQUIREMENTS: ' Page No: ' 332'			
TRACING	FWVT Tested	**TLMM_M_392 - MASTER TRIM SUCCESS**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.9.1 REQUIREMENTS: ' Page No: ' 332'			
TRACING	FWVT Tested	**TLMM_S_393 - SLAVE TRIM SUCCESS**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.9.1 REQUIREMENTS: ' Page No: ' 333'			
TRACING	FWVT Tested	**TLMM_M_394 - MASTER TRIM FAIL**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.9.1 REQUIREMENTS: ' Page No: ' 333'			
TRACING	FWVT Tested	**TLMM_S_395 - SLAVE TRIM FAIL**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.9.1 REQUIREMENTS: ' Page No: ' 333'			
TRACING	FWVT Tested	**TLMM_MS_396 - TELEMETRY CEASED INDICATION**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.9.1 REQUIREMENTS: ' Page No: ' 334'			
TRACING	FWVT Tested	**TLMM_MS_397 - TRIM FAILURE INDICATION**			
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.9.1 REQUIREMENTS: ' Page No: ' 334'			

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMM_MS_415 - MEMORY TEST - RAM STRUCTURES**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_MemoryTest\m_MemoryTest004\m_memorytest004 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_MemoryTest\m_MemoryTest003\m_memorytest003 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_MemoryTest\m_MemoryTest005\m_memorytest005 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MemoryTest\m_MemoryTest002\m_memorytest002
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.10.4 Requirements ' Page No: ' 338'	
TRACING	FWVT Tested	**TLMM_MS_406 - EEPROM - NON-VOLATILE REGISTERS**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.2.4 READY INDICATION ' Page No: ' 104'	
TRACING	FWVT Tested	**TLMM_MS_407 - EEPROM - LOAD RAMWARE APPLICATION**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.2.4 READY INDICATION ' Page No: ' 104'	
TRACING	FWVT Tested	**TLMM_MS_262- READY INDICATION - INDICATION TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_EEPROM\m_Eeprom001\m_eeprom001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.2.4 READY INDICATION ' Page No: ' 104'	
TRACING	FWVT Tested	**TLMM_MS_409 - EEPROM - FAILURE INDICATION**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.2.5 EEPROM FAILED indicationS ' Page No: ' 105'	
TRACING	FWVT Tested	**TLMC_MS_004 - STATION ID**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.6 General Requirements ' Page No: ' 110'	

Tracing	Classification	Requirement Title			
	Decomp Class	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_MS_006 - EXTERNAL SERVICE REQUEST PROCESSING**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Close\m_Close020\m_close020
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Network\m_Network001\m_network001
			Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics014\m_diagnostics014
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency012\m_emergency012
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_RawMode\m_RawMode002\m_rawmode002
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Security\m_Security013\m_security013
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_MedicalEvent\m_MedicalEvent001\m_medicalevent001
			Status: **Up-to-date**	Rev: 2.11	Test: .\M_TCP_Open\m_Open001\m_open001
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite005\m_memorywrite005
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics027\m_diagnostics027
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull015\m_discoverfull015
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Security\m_Security010\m_security010
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Close\m_Close001\m_close001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryRead005\m_memoryread005
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics028\m_diagnostics028
			Status: **Up-to-date**	Rev: 2.8	Test: .\M_TCP_Initialize\m_Initialize001\m_initialize001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_RawMode\m_RawMode005\m_rawmode005
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency001\m_emergency001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Initialize\m_Initialize003\m_initialize003
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_MedicalEvent\m_MedicalEvent014\m_medicalevent014
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Network\m_Network002\m_network002
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Open\m_Open008\m_open008

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.6 General Requirements ' Page No: ' 111'

Tracing	Classification	Requirement Title			
	Class	Decomp Title	Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_M_237 - SERVICE DURATION TOLERANCE**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Security\m_Security015\m_security015
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Emergency\m_Emergency003\m_emergency003
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Close\m_Close002\m_close002
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Discover\m_DiscoverFull006\m_discoverfull006
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_MedicalEvent\m_MedicalEvent002\m_medicalevent002
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_RawMode\m_RawMode002\m_rawmode002
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics002\m_diagnostics002
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryWrite003\m_memorywrite003
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Network\m_Network005\m_network005
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Open\m_Open005\m_open005
			Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.6 General Requirements ' Page No: ' 112'			
TRACING	FWVT Tested	**TLMM_MS_404 - FAULT INDICATION - INDICATION TO UPPER LAYER**			
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.6.1 Fault indication ' Page No: ' 118'			
TRACING	FWVT Tested	**TLMM_S_263 - SHUTDOWN INDICATION - INDICATION TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_General\m_General027\m_general027
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_General\m_General028\m_general028
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_General\m_General029\m_general029
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.6.2 SHUTDOWN INDICATION ' Page No: ' 119'			
TRACING	FWVT Tested	**TLMM_MS_265 - INITIALIZE ADDRESS WITH VALUES**			
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.7 Initialize ' Page No: ' 122'			
TRACING	FWVT Tested	**TLMM_M_266 - MASTER INITIALIZE SERVICE REQUEST FROM UPPER LAYER**			
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.7.1 MASTER ' Page No: ' 122'			
TRACING	FWVT Tested	**TLMM_M_267 - MASTER INITIALIZE SERVICE RESPONSE TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Security\m_Security025\m_security025
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Initialize\m_Initialize004\m_initialize004

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class / Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.7.1 MASTER ' Page No: ' 122'	
TRACING	FWVT Tested	**TLMC_S_008 - SLAVE INITIALIZE SERVICE REQUEST FROM UPPER LAYER**	Status: **Up-to-date** Rev: 2.8 Test: .\M_TCP_Initialize\m_Initialize001\m_initialize001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.7.2 SLAVE ' Page No: ' 124'	
TRACING	FWVT Tested	**TLMC_S_009 - INITIALIZE SERVICE RESPONSE TO UPPER LAYER.**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Initialize\m_Initialize002\m_initialize002 Status: **Up-to-date** Rev: 2.8 Test: .\M_TCP_Initialize\m_Initialize001\m_initialize001 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Initialize\m_Initialize003\m_initialize003 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security025\m_security025
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.7.2 SLAVE ' Page No: ' 124'	
TRACING	FWVT Tested	**TLMM_S_241 - POLLING ENABLING CRITERIA**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Polling\m_Polling089\m_polling089
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 127'	
TRACING	FWVT Tested	**TLMM_S_035 - POLLING - SERVICE REQUEST FROM UPPER LAYER**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 128'	
TRACING	FWVT Tested	**TLMM_S_256 - POLLING - SERVICE RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Polling\m_Polling090\m_polling090 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Polling\m_Polling091\m_polling091
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 128'	
TRACING	FWVT Tested	**TLMM_S_268 - POLLING - VALID WAKEUP PACKET**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Polling\m_Polling088\m_polling088
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 131'	

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_S_038 - POLLING - TERMINATION**	Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Polling\m_Polling094\m_polling094 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Polling\m_Polling024\m_polling024 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Polling\m_Polling095\m_polling095 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Polling\m_Polling066\m_polling066 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Polling\m_Polling093\m_polling093 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Polling\m_Polling068\m_polling068 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Polling\m_Polling065\m_polling065 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Polling\m_Polling067\m_polling067
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 131'	
TRACING	FWVT Tested	**TLMC_S_039 - POLLING INDICATION TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Polling\m_Polling046\m_polling046
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 131'	
TRACING	FWVT Tested	**TLMC_M_010 - DISCOVER SERVICE REQUEST FROM UPPER LAYER**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Discover\m_DiscoverFull005\m_discoverfull005
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.1 REQUIREMENTS ' Page No: ' 136'	
TRACING	FWVT Tested	**TLMC_M_011 - VALID SLAVE ID**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Discover\m_DiscoverFull031\m_discoverfull031 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Discover\m_DiscoverFull030\m_discoverfull030
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.1 REQUIREMENTS ' Page No: ' 136'	

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Test Case Class	Test Class Name
TRACING	FWVT Tested	**TLMC_M_024 - DISCOVER FULL - RESPONSE TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull025\m_discoverfull025
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Discover\m_DiscoverFull005\m_discoverfull005
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull016\m_discoverfull016
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull023\m_discoverfull023
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull017\m_discoverfull017
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Discover\m_DiscoverFull033\m_discoverfull033
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Discover\m_DiscoverFull014\m_discoverfull014
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull022\m_discoverfull022
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull024\m_discoverfull024
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull015\m_discoverfull015
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 137'			
TRACING	FWVT Tested	**TLMC_MS_012 - DISCOVER FULL - DISCOVER ID PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 138'			
TRACING	FWVT Tested	**TLMC_M_013 - DISCOVER FULL - DISCOVER ID REQUEST PACKET**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Discover\m_DiscoverFull006\m_discoverfull006
			Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 138'			
TRACING	FWVT Tested	**TLMM_M_269 - DISCOVER FULL - DISCOVER ID RESPONSE WINDOW**			
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 139'			
TRACING	FWVT Tested	**TLMM_M_270 - DISCOVER FULL - RESTART NATIVE MODE TIMER**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Discover\m_DiscoverFull037\m_discoverfull037
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 139'			
TRACING	FWVT Tested	**TLMC_M_014 - DISCOVER FULL - RETRY DISCOVER ID REQUEST**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Discover\m_DiscoverFull002\m_discoverfull002
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Discover\m_DiscoverFull003\m_discoverfull003
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Discover\m_DiscoverFull009\m_discoverfull009
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Discover\m_DiscoverFull004\m_discoverfull004
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Discover\m_DiscoverFull010\m_discoverfull010

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID Class		Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 139'			
TRACING	FWVT Tested	**TLMC_S_015 - DISCOVER FULL - DISCOVER ID RESPONSE**	Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.7 Rev: 2.0 Rev: 2.3	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001 Test: .\M_TCP_Discover\m_DiscoverFull029\m_discoverfull029 Test: .\M_TCP_Discover\m_DiscoverFull011\m_discoverfull011
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 140'			
TRACING	FWVT Tested	**TLMC_M_016 - DISCOVER FULL - LBT OR NATIVE MODE TIMEOUT**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.6 Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull002\m_discoverfull002 Test: .\M_TCP_Discover\m_DiscoverFull007\m_discoverfull007
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 140'			
TRACING	FWVT Tested	**TLMC_MS_017 - DISCOVER FULL - DISCOVER ID RESPONSE PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 140'			
TRACING	FWVT Tested	**TLMC_MS_018 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 140'			
TRACING	FWVT Tested	**TLMC_M_019 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.7 Rev: 2.6	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001 Test: .\M_TCP_Discover\m_DiscoverFull002\m_discoverfull002
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 141'			
TRACING	FWVT Tested	**TLMC_M_020 - DISCOVER FULL - RETRY DISCOVER NAME REQUEST**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.2 Rev: 2.7	Test: .\M_TCP_Discover\m_DiscoverFull013\m_discoverfull013 Test: .\M_TCP_Discover\m_DiscoverFull012\m_discoverfull012
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 141'			
TRACING	FWVT Tested	**TLMC_S_021 - DISCOVER FULL - DISCOVER NAME RESPONSE**	Status: **Up-to-date** Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.7 Rev: 2.7 Rev: 2.2	Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001 Test: .\M_TCP_Discover\m_DiscoverFull012\m_discoverfull012 Test: .\M_TCP_Discover\m_DiscoverFull013\m_discoverfull013
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 142'			

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class / Test Class Name
TRACING	FWVT Tested	**TLMC_MS_022 - DISCOVER FULL - DISCOVER NAME RESPONSE PACKET FORMATS**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 142'	
TRACING	FWVT Tested	**TLMM_M_272 - DISCOVER FULL - SUCCESSFUL NAME**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Discover\m_DiscoverFull017\m_discoverfull017 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Discover\m_DiscoverFull014\m_discoverfull014 Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Discover\m_DiscoverFull003\m_discoverfull003
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 143'	
TRACING	FWVT Tested	**TLMC_S_023 - DISCOVER FULL - RESPONSE PARTITIONING**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 145'	
TRACING	FWVT Tested	**TLMM_M_273 - DISCOVER FULL - TIMEOUT INDICATION**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Discover\m_DiscoverFull035\m_discoverfull035
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 145'	
TRACING	FWVT Tested	**TLMM_M_274 - DISCOVER FULL - STOP**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Discover\m_DiscoverFull036\m_discoverfull036
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 145'	
TRACING	FWVT Tested	**TLMC_M_040 - OPEN - UPPER LAYER PARAMETERS**	Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Open\m_Open041\m_open041 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Open\m_Open039\m_open039 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Open\m_Open002\m_open002 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Open\m_Open004\m_open004 Status: **Up-to-date** Rev: 2.11 Test: .\M_TCP_Open\m_Open001\m_open001 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Open\m_Open040\m_open040
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 150'	

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMM_MS_281 - OPEN - REQUEST PACKET FORMAT**	Status: **Up-to-date** Rev: 2.11 Test: .\M_TCP_Open\m_Open001\m_open001 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Open\m_Open003\m_open003 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open011\m_open011 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Open\m_Open012\m_open012 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open006\m_open006
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 150'	
TRACING	FWVT Tested	**TLMM_MS_282 - OPEN - RESPONSE PACKET FORMAT**	Status: **Up-to-date** Rev: 2.11 Test: .\M_TCP_Open\m_Open001\m_open001 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Open\m_Open002\m_open002
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 151'	
TRACING	FWVT Tested	**TLMM_M_283 - OPEN - TRANSMIT MASTER PACKET**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Open\m_Open032\m_open032
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 153'	
TRACING	FWVT Tested	**TLMM_M_284 - OPEN - NO VALID SLAVE RESPONSE RETRY REQUEST**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Open\m_Open032\m_open032
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 153'	
TRACING	FWVT Tested	**TLMM_M_285 - OPEN - LISTEN BEFORE TALK TIMER OPERATIONS**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Open\m_Open033\m_open033
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 153'	
TRACING	FWVT Tested	**TLMM_S_287 - OPEN - NO RESPONSE TO A VALID OPEN REQUEST**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Open\m_Open031\m_open031
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 154'	
TRACING	FWVT Tested	**TLMM_S_288 - OPEN - RESPONSE TO A VALID OPEN UNSECURE REQUEST**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Open\m_Open036\m_open036
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 154'	
TRACING	FWVT Tested	**TLMM_S_289 - OPEN - SESSION KEY COMPUTATION UPON VALID OPEN REQUEST**	

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 155'	
TRACING	FWVT Tested	**TLMM_S_290 - OPEN - RESPONSE SUCCESSFUL**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open037\m_open037
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 155'	
TRACING	FWVT Tested	**TLMM_S_292 -OPEN - ENABLE SECURITY BLOCK**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 155'	
TRACING	FWVT Tested	**TLMM_M_294 - OPEN - A VALID SLAVE RESPONSE**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open042\m_open042
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 157'	
TRACING	FWVT Tested	**TLMM_M_295 - OPEN - RESPONSE WITHOUT KBAN OR KEBAN**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Open\m_Open038\m_open038
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 157'	
TRACING	FWVT Tested	**TLMM_M_296 - OPEN - RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open035\m_open035 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open034\m_open034 Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Open\m_Open038\m_open038
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 158'	
TRACING	FWVT Tested	**TLMC_MS_049 - DATA - UPPER LAYER TO DATA LINK LAYER PARAMETERS**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryWrite006\m_memorywrite006 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryRead001\m_memoryread001 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite012\m_memorywrite012 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite005\m_memorywrite005 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite007\m_memorywrite007
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 165'	
TRACING	FWVT Tested	**TLMC_MS_050 - DATA - MASTER REQUEST PACKET FORMAT(S)**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite005\m_memorywrite005
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 165'	

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class** **Name**

TRACING FWVT Tested **TLMC_M_051 - DATA - TRANSMISSION OF MASTER DATA PACKET**

Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite005\m_memorywrite005
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite025\m_memorywrite025
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite001\m_memorywrite001
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Emergency\m_Emergency002\m_emergency002
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryRead001\m_memoryread001

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 166'

TRACING FWVT Tested **TLMC_S_052 - DATA - ACKNOWLEDGEMENT OF MASTER DATA PACKET**

Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Data\m_MemoryWrite002\m_memorywrite002
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryWrite006\m_memorywrite006

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 167'

TRACING FWVT Tested **TLMC_M_053 - DATA - PACKETIZATION OF THE MASTER DATA**

Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite014\m_memorywrite014
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite012\m_memorywrite012
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite013\m_memorywrite013
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite005\m_memorywrite005
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite009\m_memorywrite009
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite011\m_memorywrite011
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite015\m_memorywrite015

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 167'

TRACING FWVT Tested **TLMC_M_054 - DATA - RETRANSMISSION OF MASTER PACKET - NO ACK RECEIVED**

Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Data\m_MemoryWrite019\m_memorywrite019
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite018\m_memorywrite018
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Data\m_MemoryWrite016\m_memorywrite016
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryWrite017\m_memorywrite017

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 167'

TRACING FWVT Tested **TLMC_S_055 - DATA - MASTER PACKETS WITH DUPLICATE SEQUENCE NUMBERS**

Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryRead001\m_memoryread001
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryRead002\m_memoryread002

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 167'

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID Class** **Test Class Name**

TRACING	FWVT Tested	**TLMC_S_056 - DATA - VALIDATION OF MASTER DATA REQUEST**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_MemoryWrite016\m_memorywrite016
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite020\m_memorywrite020
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryWrite018\m_memorywrite018
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead003\m_memoryread003
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 168'			
TRACING	FWVT Tested	**TLMC_S_057 - DATA - MASTER PACKETS INTO TRANSACTION**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite005\m_memorywrite005
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite001\m_memorywrite001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryWrite006\m_memorywrite006
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 168'			
TRACING	FWVT Tested	**TLMC_M_058 - DATA - RESPONSE TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_MemoryWrite049\m_memorywrite049
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryWrite022\m_memorywrite022
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryRead039\m_memoryread039
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead043\m_memoryread043
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Data\m_MemoryRead045\m_memoryread045
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryWrite025\m_memorywrite025
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryWrite006\m_memorywrite006
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryWrite023\m_memorywrite023
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryRead005\m_memoryread005
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite012\m_memorywrite012
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite024\m_memorywrite024
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 169'			
TRACING	FWVT Tested	**TLMM_M_297 - DATA - WAIT FOR INDICATION**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_Data013\m_data013
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 171'			
TRACING	FWVT Tested	**TLMM_M_298 - DATA - RECOVERY**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_Data014\m_data014
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 171'			
TRACING	FWVT Tested	**TLMM_M_299 - DATA - REPLY TIMEOUT**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_Data012\m_data012

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class** **Name**

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 171'

TRACING FWVT Tested **TLMC_S_059 - DATA - SLAVE UPPER LAYER PARAMETERS**
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryRead007\m_memoryread007
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Data\m_MemoryRead006\m_memoryread006
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryRead008\m_memoryread008
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryRead001\m_memoryread001

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.5 SLAVE DATA ' Page No: ' 172'

TRACING FWVT Tested **TLMC_S_058 - DATA - RESPONSE TO UPPER LAYER**
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Data\m_MemoryWrite049\m_memorywrite049
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryRead005\m_memoryread005
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Data\m_MemoryWrite023\m_memorywrite023
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryRead007\m_memoryread007
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Data\m_MemoryRead006\m_memoryread006
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryRead004\m_memoryread004
Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Data\m_MemoryRead040\m_memoryread040
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryWrite022\m_memorywrite022

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.5 SLAVE DATA ' Page No: ' 173'

TRACING FWVT Tested **TLMC_S_060 - DATA - SLAVE TRANSMISSION OF DATA PACKETS**
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryRead052\m_memoryread052
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryRead009\m_memoryread009
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Data\m_MemoryRead006\m_memoryread006

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.5 SLAVE DATA ' Page No: ' 173'

TRACING FWVT Tested **TLMC_S_061 - DATA - SLAVE DATA PACKET FORMAT(S)**
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Data\m_MemoryRead006\m_memoryread006

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.5 SLAVE DATA ' Page No: ' 174'

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Test Case Class	Test Class Name
TRACING	FWVT Tested	**TLMC_S_062 - DATA - PACKETIZATION OF SLAVE DATA**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead014\m_memoryread014
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead011\m_memoryread011
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead013\m_memoryread013
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Data\m_MemoryRead006\m_memoryread006
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead012\m_memoryread012
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead010\m_memoryread010
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryRead016\m_memoryread016
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead017\m_memoryread017
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead015\m_memoryread015
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.5 SLAVE DATA ' Page No: ' 175'			
TRACING	FWVT Tested	**TLMC_M_063 - DATA - RECEIPT OF AN INVALID SLAVE PACKET**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead003\m_memoryread003
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryRead009\m_memoryread009
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryRead021\m_memoryread021
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Data\m_MemoryRead020\m_memoryread020
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Data\m_MemoryRead019\m_memoryread019
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.5 SLAVE DATA ' Page No: ' 176'			
TRACING	FWVT Tested	**TLMC_M_066 - DIAGNOSTIC SERVICE REQUEST FROM UPPER LAYER**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics011\m_diagnostics011
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Diagnostic\m_Diagnostics006\m_diagnostics006
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics027\m_diagnostics027
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.1 REQUIREMENTS ' Page No: ' 178'			
TRACING	FWVT Tested	**TLMC_M_067 - REMOTE DIAGNOSTIC - MASTER PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics002\m_diagnostics002
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics004\m_diagnostics004
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics003\m_diagnostics003
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.2 REMOTE DIAGNOSTIC ' Page No: ' 179'			

Tracing	Classification	Requirement Title			
Class	Decomp	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_S_068 - REMOTE DIAGNOSTIC - SLAVE PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.2 REMOTE DIAGNOSTIC ' Page No: ' 179'			
TRACING	FWVT Tested	**TLMC_S_069 - REMOTE DIAGNOSTIC - STATUS BYTE FORMAT**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics030\m_diagnostics030
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics017\m_diagnostics017
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.2 REMOTE DIAGNOSTIC ' Page No: ' 179'			
TRACING	FWVT Tested	**TLMC_S_070 - REMOTE DIAGNOSTIC - SLAVE RESPONSE PACKET**	Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Diagnostic\m_Diagnostics007\m_diagnostics007
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics018\m_diagnostics018
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics002\m_diagnostics002
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics038\m_diagnostics038
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Diagnostic\m_Diagnostics006\m_diagnostics006
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Diagnostic\m_Diagnostics008\m_diagnostics008
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics009\m_diagnostics009
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.2 REMOTE DIAGNOSTIC ' Page No: ' 180'			
TRACING	FWVT Tested	**TLMC_M_071 - REMOTE DIAGNOSTIC - MASTER REQUEST RETRANSMISSION**	Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Diagnostic\m_Diagnostics019\m_diagnostics019
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics036\m_diagnostics036
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics020\m_diagnostics020
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics037\m_diagnostics037
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics018\m_diagnostics018
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics035\m_diagnostics035
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics004\m_diagnostics004
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.2 REMOTE DIAGNOSTIC ' Page No: ' 180'			

Tracing **Classification** **Requirement Title**

Class	Decomp	Title	Test Case ID	Test Case Class	Test Class Name
TRACING	FWVT Tested	**TLMC_M_072 - REMOTE DIAGNOSTIC - RESPONSE TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Diagnostic\m_Diagnostics006\m_diagnostics006
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Diagnostic\m_Diagnostics007\m_diagnostics007
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics017\m_diagnostics017
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Diagnostic\m_Diagnostics021\m_diagnostics021
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Diagnostic\m_Diagnostics026\m_diagnostics026
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics028\m_diagnostics028
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Diagnostic\m_Diagnostics008\m_diagnostics008
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics010\m_diagnostics010
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics022\m_diagnostics022
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics034\m_diagnostics034
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics018\m_diagnostics018
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Diagnostic\m_Diagnostics005\m_diagnostics005
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics020\m_diagnostics020
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics003\m_diagnostics003
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics009\m_diagnostics009
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics024\m_diagnostics024
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.2 REMOTE DIAGNOSTIC ' Page No: ' 181'			
TRACING	FWVT Tested	**TLMC_M_073 - LOCAL DIAGNOSTIC - RESPONSE TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics011\m_diagnostics011
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics012\m_diagnostics012
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics015\m_diagnostics015
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Diagnostic\m_Diagnostics024\m_diagnostics024
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics014\m_diagnostics014
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Diagnostic\m_Diagnostics013\m_diagnostics013
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Diagnostic\m_Diagnostics005\m_diagnostics005
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Diagnostic\m_Diagnostics017\m_diagnostics017
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.3 LOCAL DIAGNOSTIC ' Page No: ' 183'			

Tracing	Classification	Requirement Title	
Decomp Class		Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMC_M_074 - MAPPING DIAGNOSTIC - RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Diagnostic\m_Diagnostics029\m_diagnostics029 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Diagnostic\m_Diagnostics016\m_diagnostics016 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Diagnostic\m_Diagnostics023\m_diagnostics023 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Diagnostic\m_Diagnostics024\m_diagnostics024 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Diagnostic\m_Diagnostics028\m_diagnostics028 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Diagnostic\m_Diagnostics005\m_diagnostics005 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Diagnostic\m_Diagnostics017\m_diagnostics017 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Diagnostic\m_Diagnostics022\m_diagnostics022 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Diagnostic\m_Diagnostics026\m_diagnostics026 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Diagnostic\m_Diagnostics027\m_diagnostics027
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.4 MAPPING DIAGNOSTIC ' Page No: ' 185'	
TRACING	FWVT Tested	**TLMC_M_075 - DIAGNOSTIC FILLER - MASTER FILLER PACKETS**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_ARQ020\m_arq020 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_ARQ012\m_arq012 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Diagnostic\m_Diagnostics030\m_diagnostics030
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.5 DIAGNOSTIC FILLER PACKET ' Page No: ' 186'	
TRACING	FWVT Tested	**TLMC_S_076 - DIAGNOSTIC FILLER - SLAVE FILLER PACKETS**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Diagnostic\m_Diagnostics030\m_diagnostics030
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.12.5 DIAGNOSTIC FILLER PACKET ' Page No: ' 186'	
TRACING	FWVT Tested	**TLMC_M_077 - EMERGENCY - UPPER LAYER PARAMETERS**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Emergency\m_Emergency002\m_emergency002 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Emergency\m_Emergency001\m_emergency001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 188'	

Tracing	Classification	Requirement Title			
Class	Decomp	Title	Test Case ID	Class	Test Class Name
TRACING	FWVT Tested	**TLMC_M_078 - EMERGENCY - TRANSMISSION OF EMERGENCY REQUEST**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Data\m_MemoryWrite024\m_memorywrite024
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Diagnostic\m_Diagnostics031\m_diagnostics031
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_RawMode\m_RawMode003\m_rawmode003
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency007\m_emergency007
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Open\m_Open008\m_open008
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency001\m_emergency001
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Data\m_MemoryRead030\m_memoryread030
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Discover\m_DiscoverFull017\m_discoverfull017
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Emergency\m_Emergency003\m_emergency003
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency004\m_emergency004
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Security\m_Security029\m_security029
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 189'			
TRACING	FWVT Tested	**TLMC_M_079 - EMERGENCY - MASTER REQUEST NOT ACKNOWLEDGED**	Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Emergency\m_Emergency003\m_emergency003
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 189'			
TRACING	FWVT Tested	**TLMC_M_081 - EMERGENCY - STATUS RESPONSE TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency004\m_emergency004
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Emergency\m_Emergency011\m_emergency011
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Emergency\m_Emergency039\m_emergency039
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency006\m_emergency006
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Emergency\m_Emergency040\m_emergency040
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Emergency\m_Emergency043\m_emergency043
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Emergency\m_Emergency041\m_emergency041
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Emergency\m_Emergency042\m_emergency042
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Emergency\m_Emergency001\m_emergency001
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency009\m_emergency009
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Emergency\m_Emergency010\m_emergency010
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 189'			

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID Class** **Test Class Name**

TRACING FWVT Tested **TLMM_M_301 - EMERGENCY - WAIT FOR INDICATION**
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Emergency\m_Emergency045\m_emergency045
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 191'

TRACING FWVT Tested **TLMM_M_302 - EMERGENCY - RECOVERY**
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Emergency\m_Emergency045\m_emergency045
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 191'

TRACING FWVT Tested **TLMM_M_303 - EMERGENCY - REPLY TIMEOUT**
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Emergency\m_Emergency044\m_emergency044
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 191'

TRACING FWVT Tested **TLMC_M_082 - RAW MODE - UPPER LAYER RAW MODE PARAMETERS**
Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_RawMode\m_RawMode002\m_rawmode002
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_RawMode\m_RawMode001\m_rawmode001
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.14.1 REQUIREMENTS ' Page No: ' 194'

TRACING FWVT Tested **TLMC_M_083 - RAW MODE - MASTER PACKET ATTACHED OPERATION**
Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_RawMode\m_RawMode002\m_rawmode002
Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_RawMode\m_RawMode004\m_rawmode004
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.14.2 RAW MODE (MASTER PACKET ATTACHED) ' Page No: ' 194'

TRACING FWVT Tested **TLMC_M_089 - CLOSE - UPPER LAYER CLOSE PARAMETERS**
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Close\m_Close002\m_close002
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 197'

TRACING FWVT Tested **TLMC_MS_090 - CLOSE - MASTER PACKET FORMAT**
Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Close\m_Close001\m_close001
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 197'

TRACING FWVT Tested **TLMC_MS_091 - CLOSE - SLAVE PACKET FORMAT**
Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Close\m_Close001\m_close001
' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 197'

TRACING FWVT Tested **TLMC_M_092 - CLOSE - MASTER REQUEST**
Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Close\m_Close001\m_close001
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Close\m_Close003\m_close003
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Close\m_Close004\m_close004
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Close\m_Close002\m_close002

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 198'	
TRACING	FWVT Tested	**TLMC_M_093 - CLOSE - RETRANSMISSION OF MASTER REQUEST**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Close\m_Close005\m_close005 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Close\m_Close001\m_close001 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Close\m_Close007\m_close007 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Close\m_Close008\m_close008
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 198'	
TRACING	FWVT Tested	**TLMC_S_094 - CLOSE - RECEIPT OF A VALID CLOSE REQUEST**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Close\m_Close009\m_close009 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Close\m_Close010\m_close010
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 198'	
TRACING	FWVT Tested	**TLMC_S_095 - CLOSE - INDICATION TO SLAVE UPPER LAYER**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Close\m_Close006\m_close006 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Close\m_Close001\m_close001 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Close\m_Close003\m_close003
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 198'	
TRACING	FWVT Tested	**TLMC_M_096 - CLOSE - RECEIPT OF A VALID RESPONSE**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Close\m_Close002\m_close002 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Close\m_Close001\m_close001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 198'	
TRACING	FWVT Tested	**TLMC_M_097 - CLOSE - IMPLICITLY SUCCESSFUL**	Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Close\m_Close017\m_close017 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Close\m_Close012\m_close012 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Close\m_Close014\m_close014 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Close\m_Close011\m_close011 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Close\m_Close013\m_close013
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 199'	

Tracing	Classification	Requirement Title			
Class	Decomp	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMC_M_098 - CLOSE - RESPONSE TO UPPER LAYER**	Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Close\m_Close014\m_close014
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Close\m_Close022\m_close022
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Close\m_Close001\m_close001
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Close\m_Close007\m_close007
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Close\m_Close008\m_close008
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Close\m_Close011\m_close011
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Close\m_Close013\m_close013
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Close\m_Close020\m_close020
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Close\m_Close005\m_close005
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Close\m_Close021\m_close021
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Close\m_Close012\m_close012
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Close\m_Close019\m_close019
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 199'			
TRACING	FWVT Tested	**TLMC_M_099 - CLOSE - RECOVERY DURING CLOSE**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Close\m_Close008\m_close008
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.15.1 REQUIREMENTS ' Page No: ' 199'			
TRACING	FWVT Tested	**TLMC_S_414 - MEDICAL EVENT - SLAVE PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_MedicalEvent\m_MedicalEvent001\m_medicalevent001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 202'			
TRACING	FWVT Tested	**TLMC_S_229 - MEDICAL EVENT - REQUEST**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_MedicalEvent\m_MedicalEvent002\m_medicalevent002
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 202'			
TRACING	FWVT Tested	**TLMC_S_418 - MEDICAL EVENT - RESPONSE**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_MedicalEvent\m_MedicalEvent014\m_medicalevent014
			Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_MedicalEvent\m_MedicalEvent002\m_medicalevent002
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_MedicalEvent\m_MedicalEvent012\m_medicalevent012
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_MedicalEvent\m_MedicalEvent011\m_medicalevent011
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_MedicalEvent\m_MedicalEvent008\m_medicalevent008
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 203'			

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class / Test Class Name
TRACING	FWVT Tested	**TLMC_S_257 - MEDICAL EVENT - INDICATION TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent003\m_medicalevent003 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_MedicalEvent\m_MedicalEvent002\m_medicalevent002
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 204'	
TRACING	FWVT Tested	**TLMC_M_230 - MEDICAL EVENT - HOST PARAMETERS**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent001\m_medicalevent001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 205'	
TRACING	FWVT Tested	**TLMC_M_231 - MEDICAL EVENT - RESPONSE TO HOST**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent001\m_medicalevent001 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_MedicalEvent\m_MedicalEvent011\m_medicalevent011 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_MedicalEvent\m_MedicalEvent009\m_medicalevent009 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_MedicalEvent\m_MedicalEvent010\m_medicalevent010 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent008\m_medicalevent008 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_MedicalEvent\m_MedicalEvent012\m_medicalevent012 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent014\m_medicalevent014
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 206'	
TRACING	FWVT Tested	**TLMC_M_232 - MEDICAL EVENT - MEDICAL EVENT PACKET**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent001\m_medicalevent001 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent004\m_medicalevent004
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 207'	
TRACING	FWVT Tested	**TLMC_S_258 - PATIENT NAME REQUEST**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_General\m_General017\m_general017
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.17.1 REQUIREMENTS ' Page No: ' 209'	
TRACING	FWVT Tested	**TLMC_S_259 - PATIENT NAME SERVICE RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_General\m_General018\m_general018 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_General\m_General017\m_general017
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.17.1 REQUIREMENTS ' Page No: ' 209'	
TRACING	FWVT Tested	**TLMM_MS_309 - CONFIGURE NETWORK - UPPER LAYER PARAMETERS**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.3.1 CONFIGURE NETWORK service requirements ' Page No: ' 213'	

Decomp Class Title Test Case ID Class Test Class Name

Tracing	Classification	Requirement Title	
TRACING	FWVT Tested	**TLMM_MS_310 - CONFIGURE NETWORK - RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network003\m_network003 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network004\m_network004
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.3.1 CONFIGURE NETWORK service requirements ' Page No: ' 214'	
TRACING	FWVT Tested	**TLMM_M_311 - NETWORK DATA - UPPER LAYER PARAMETERS**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4 NETWORK DATA ' Page No: ' 215'	
TRACING	FWVT Tested	**TLMM_M_312 - NETWORK DATA - REQUEST AND RESPONSE PACKET FORMAT**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Network\m_Network001\m_network001 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network010\m_network010 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network011\m_network011 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network005\m_network005 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network006\m_network006 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network008\m_network008 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network009\m_network009 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network007\m_network007
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4 NETWORK DATA ' Page No: ' 220'	
TRACING	FWVT Tested	**TLMM_S_313 - NETWORK DATA - FILLER PACKET FORMAT**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network010\m_network010 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network005\m_network005 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network011\m_network011 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network006\m_network006 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network008\m_network008 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network009\m_network009 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Network\m_Network001\m_network001 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network007\m_network007
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4 NETWORK DATA ' Page No: ' 225'	
TRACING	FWVT Tested	**TLMM_M_314 - NETWORK DATA - TRANSMITTING REQUEST PACKETS**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Network\m_Network029\m_network029 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network026\m_network026

Tracing **Classification** **Requirement Title**

Class **Decomp** **Title** **Test Case ID** **Class** **Test Class** **Name**

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 225'

TRACING FWVT Tested **TLMM_M_315 - NETWORK DATA - UPPER LAYER RESPONSE (MESSAGE LEVEL RESPONSE)** Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network026\m_network026

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 226'

TRACING FWVT Tested **TLMM_M_316 - NETWORK DATA - DATA LINK LAYER RESPONSE (PACKET LEVEL RESPONSE)** Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network025\m_network025

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 226'

TRACING FWVT Tested **TLMM_M_317 - NETWORK DATA - NO ACK** Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network023\m_network023

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 227'

TRACING FWVT Tested **TLMM_M_318 - NETWORK DATA - SUCCESSFUL RESPONSE TO UPPER LAYER** Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Network\m_Network002\m_network002

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 227'

TRACING FWVT Tested **TLMM_M_319 - NETWORK DATA - UNSUCCESSFUL RESPONSE TO UPPER LAYER** Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Network\m_Network002\m_network002

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 228'

TRACING FWVT Tested **TLMM_M_320 - NETWORK DATA - INDICATION TO UPPER LAYER** Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network013\m_network013

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 229'

TRACING FWVT Tested **TLMM_M_321 - NETWORK DATA - RESPONSE COMPLETE AT MASTER** Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network013\m_network013

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 231'

TRACING FWVT Tested **TLMM_S_322 - NETWORK DATA - NO ACK OF INVALID REQUEST PACKET** Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Network\m_Network015\m_network015

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 232'

Tracing	Classification	Requirement Title	
Class	Decomp Title		Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMM_S_323 - NETWORK DATA - NO ACK OF REQUEST PACKET**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Network\m_Network015\m_network015
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 232'	
TRACING	FWVT Tested	**TLMM_S_324 - NETWORK DATA - ACK OF REQUEST PACKET**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network016\m_network016
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 232'	
TRACING	FWVT Tested	**TLMM_S_325 - NETWORK DATA - SERVICE INDICATION TO THE UPPER LAYER**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network016\m_network016 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network024\m_network024
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 233'	
TRACING	FWVT Tested	**TLMM_S_327 - NETWORK DATA - NETWORK DATA RESPONSE**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Network\m_Network014\m_network014
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 234'	
TRACING	FWVT Tested	**TLMM_S_328 - NETWORK DATA - RESPONSE TIME SLOT**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Network\m_Network020\m_network020 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Network\m_Network019\m_network019 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Network\m_Network021\m_network021 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Network\m_Network022\m_network022
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 234'	
TRACING	FWVT Tested	**TLMM_S_329 - NETWORK DATA - SEQUENCE NUMBER**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network016\m_network016
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 235'	
TRACING	FWVT Tested	**TLMM_S_330 - NETWORK DATA - INVALID RESPONSE TO INDICATION**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Network\m_Network027\m_network027
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 235'	
TRACING	FWVT Tested	**TLMM_MS_403 - SECURITY - UPPER LAYER PARAMETERS**	Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Security\m_Security035\m_security035
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.1 SECURITY SERVICE - General Requirements ' Page No: ' 238'	

Tracing	Classification	Requirement Title			
	Decomp Class	Title		Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMM_MS_334 - SECURITY - CREATE BAN KEY/EPHEMERAL BAN KEY LOCAL**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Security\m_Security007\m_security007
			Status: **Up-to-date**	Rev: 2.9	Test: .\M_TCP_Security\m_Security012\m_security012
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Security\m_Security006\m_security006
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Security\m_Security008\m_security008
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Security\m_Security009\m_security009
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Security\m_Security010\m_security010
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Security\m_Security003\m_security003
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Security\m_Security013\m_security013
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Security\m_Security024\m_security024
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Security\m_Security026\m_security026
			Status: **Up-to-date**	Rev: 2.9	Test: .\M_TCP_Security\m_Security002\m_security002
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Security\m_Security005\m_security005
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.2 CREATE BAN KEY / Ephemeral ban key Local ' Page No: ' 241'			
TRACING	FWVT Tested	**TLMM_MS_401 - SECURITY - MASTER/SLAVE REJECTS KEY REQUEST IF IT HAS NO BAN/EBAN KEY**	Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Security\m_Security014\m_security014
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.4 REQUEST BAN KEY/EPHEMERAL BAN KEY LOCAL ' Page No: ' 243'			
TRACING	FWVT Tested	**TLMM_M_337 - SECURITY - PROXIMITY SECURITY REQUEST PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Security\m_Security004\m_security004
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5 REQUEST BAN KEY/EBAN key VIA proximity SWITCH ' Page No: ' 248'			
TRACING	FWVT Tested	**TLMM_S_338 - SECURITY - PROXIMITY RESPONSE PACKET FORMAT**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Security\m_Security020\m_security020
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5 REQUEST BAN KEY/EBAN key VIA proximity SWITCH ' Page No: ' 249'			

Class | Decomp Title | Test Case ID | Class | Test Class Name

Tracing	Classification	Requirement Title	
TRACING	FWVT Tested	**TLMM_M_339 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH REQUEST PACKET FORMAT**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security011\m_security011
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5 REQUEST BAN KEY/EBAN key VIA proximity SWITCH ' Page No: ' 249'	
TRACING	FWVT Tested	**TLMM_S_340 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH RESPONSE PACKET FORMAT**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Security\m_Security033\m_security033 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security021\m_security021
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5 REQUEST BAN KEY/EBAN key VIA proximity SWITCH ' Page No: ' 250'	
TRACING	FWVT Tested	**TLMM_M_341 - SECURITY - TRANSMIT PROXIMITY SECURITY REQUEST PACKET**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Security\m_Security016\m_security016
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.1 MASTER ' Page No: ' 251'	
TRACING	FWVT Tested	**TLMM_M_342 - SECURITY - NO RESPONSE TO PROXIMITY SECURITY REQUEST**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security015\m_security015
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.1 MASTER ' Page No: ' 251'	
TRACING	FWVT Tested	**TLMM_M_343 - SECURITY - VALID RESPONSE TO PROXIMITY SECURITY REQUEST**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security017\m_security017
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.1 MASTER ' Page No: ' 251'	
TRACING	FWVT Tested	**TLMM_M_344 - SECURITY - NO RESPONSE OR REJECTION TO REQUEST BAN/EBAN KEY VIA PROXIMITY SWITCH**	Status: **Up-to-date** Rev: 2.12 Test: .\M_TCP_Security\m_Security023\m_security023 Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Security\m_Security022\m_security022
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.1 MASTER ' Page No: ' 251'	

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMM_M_346 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH - RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Security\m_Security028\m_security028 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Security\m_Security031\m_security031 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security029\m_security029 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Security\m_Security027\m_security027 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security030\m_security030 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Security\m_Security032\m_security032
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.1 MASTER ' Page No: ' 252'	
TRACING	FWVT Tested	**TLMM_M_347 - SECURITY - LBT TIMEOUT DURING REQUEST BAN/EBAN KEY EXCHANGE VIA PROXIMITY SWITCH SERVICE**	Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Security\m_Security043\m_security043 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security044\m_security044 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Security\m_Security042\m_security042
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.1 MASTER ' Page No: ' 252'	
TRACING	FWVT Tested	**TLMM_S_348 - SECURITY - SLAVE RESPONSE TO A VALID PROXIMITY SWITCH SECURITY REQUEST PACKET**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security017\m_security017 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security018\m_security018
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.2 SLAVE ' Page No: ' 252'	
TRACING	FWVT Tested	**TLMM_S_349 - SECURITY - SLAVE REJECTS A VALID REQUEST BAN/EBAN KEY VIA PROXIMITY SWITCH REQUEST**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security019\m_security019
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.2 SLAVE ' Page No: ' 253'	
TRACING	FWVT Tested	**TLMM_S_350 - SECURITY - SLAVE ACCEPTS A VALID REQUEST BAN/EBAN key VIA PROXIMITY SWITCH REQUEST**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security034\m_security034
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.2 SLAVE ' Page No: ' 253'	
TRACING	FWVT Tested	**TLMM_S_351 - SECURITY - AUTHORIZATION TO GIVE BAN/EBAN KEY SERVICE RESPONSE**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security045\m_security045 Status: **Up-to-date** Rev: 2.9 Test: .\M_TCP_Security\m_Security002\m_security002 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security039\m_security039 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Security\m_Security041\m_security041

Tracing	Classification	Requirement Title	
	Decomp Class	Title	Test Case ID Class Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.2 SLAVE ' Page No: ' 254'	
TRACING	FWVT Tested	**TLMM_S_352 - SECURITY - USAGE OF EPHEMERAL BAN KEY**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Security\m_Security036\m_security036
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.5.2 SLAVE ' Page No: ' 254'	
TRACING	FWVT Tested	**TLMM_S_419 - SECURITY - UPDATE PROXIMITY STATUS**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Security\m_Security037\m_security037
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.7 Security - update proximity status ' Page No: ' 258'	
TRACING	FWVT Tested	**TLMM_S_420 - SECURITY - SERVICE DURATION TIMEOUT**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security040\m_security040
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.7 Security - update proximity status ' Page No: ' 258'	
TRACING	FWVT Tested	**TLMM_M_422 - POWER INHIBIT - CHANNEL RECOVERY THRESHOLD**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_PowerInhibit\m_Power002\m_power002
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.2 MASTER ' Page No: ' 263'	
TRACING	FWVT Tested	**TLMM_S_425 - POWER INHIBIT - RESPONSE**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_PowerInhibit\m_Power006\m_power006
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 265'	
TRACING	FWVT Tested	**TLMM_MS_353 - SHUTDOWN - SERVICE REQUEST PARAMETERS**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.22.1 REQUIREMENTS ' Page No: ' 274'	
TRACING	FWVT Tested	**TLMM_MS_354 - SHUTDOWN - RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_General\m_General030\m_general030 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_General\m_General031\m_general031 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_General\m_General032\m_general032 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_General\m_General033\m_general033 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_General\m_General034\m_general034
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.22.1 REQUIREMENTS ' Page No: ' 274'	
TRACING	FWVT Tested	**TLMM_MS_413 - SHUTDOWN INDICATION - DISABLE TIMER EXPIRES**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_General\m_General027\m_general027
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.6.2 SHUTDOWN INDICATION ' Page No: ' 120'	

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMM_M_300 - DATA - FIRST INDICATION**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data015\m_data015
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.11.4 MASTER DATA ' Page No: ' 172'	
TRACING	FWVT Tested	**TLMM_S_326 -NETWORK DATA-ENABLE SECURITY BLOCK**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 234'	
TRACING	FWVT Tested	**TLMC_S_106 - ARQ - STOP AND WAIT MASTER PACKETS**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_ARQ012\m_arq012 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_ARQ017\m_arq017 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_ARQ019\m_arq019 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_ARQ020\m_arq020 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_ARQ021\m_arq021
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.3 Slave ' Page No: ' 285'	
TRACING	FWVT Tested	**TLMC_S_107 - ARQ - RECEIPT OF AN ARQ PACKET**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_ARQ024\m_arq024 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_ARQ022\m_arq022 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_ARQ023\m_arq023
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.3 Slave ' Page No: ' 285'	
TRACING	FWVT Tested	**TLMC_S_108 - ARQ - MASTER ARQ PACKET NOT RECEIVED**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_ARQ017\m_arq017
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.3 Slave ' Page No: ' 285'	
TRACING	FWVT Tested	**TLMC_S_109 - ARQ - IMPLICIT ACK OF VALID SLAVE MEMORY DATA**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_ARQ026\m_arq026 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_ARQ023\m_arq023 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_ARQ025\m_arq025
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.3 Slave ' Page No: ' 285'	
TRACING	FWVT Tested	**TLMC_S_110 - ARQ - MEMORY DATA ACKNOWLEDGEMENT NOT RECEIVED**	Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Data\m_ARQ018\m_arq018 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_ARQ027\m_arq027 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_ARQ031\m_arq031 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_ARQ013\m_arq013
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.3 Slave ' Page No: ' 285'	

Tracing	Classification	Requirement Title				
Class	Decomp	Title		Test Case ID	Test Case Class	Test Class Name
TRACING	FWVT Tested	**TLMM_S_356 - ARQ - RETRANSMIT WAVEFORM PACKET**	Status: **Up-to-date**	Rev: 2.2		Test: .\M_TCP_EEPROM\m_Eeprom006\m_eeprom006
			Status: **Up-to-date**	Rev: 2.5		Test: .\M_TCP_Data\m_ARQ030\m_arq030

' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.3 Slave ' Page No: ' 286'

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID** **Class** **Test Class Name**

Tracing	Classification	Requirement Title	
TRACING	FWVT Tested	**TLMC_MS_001 - PACKET CONTROL BYTES**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryRead001\m_memoryread001
			Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Data\m_MemoryRead006\m_memoryread006
			Status: **Up-to-date** Rev: 2.8 Test: .\M_TCP_Data\m_TransferMode001\m_transfermode001
			Status: **Up-to-date** Rev: 2.11 Test: .\M_TCP_Open\m_Open001\m_open001
			Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Close\m_Close001\m_close001
			Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryRead014\m_memoryread014
			Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite005\m_memorywrite005
			Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_TransferMode002\m_transfermode002
			Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Discover\m_DiscoverFull002\m_discoverfull002
			Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Emergency\m_Emergency002\m_emergency002
			Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_General\m_General015\m_general015
			Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security015\m_security015
			Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security020\m_security020
			Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_ARQ006\m_arq006
			Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryRead031\m_memoryread031
			Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryRead033\m_memoryread033
			Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Data\m_TransferMode003\m_transfermode003
			Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Discover\m_DiscoverFull001\m_discoverfull001
			Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Discover\m_DiscoverFull032\m_discoverfull032
			Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_PowerInhibit\m_Power006\m_power006
			Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryRead012\m_memoryread012
			Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Diagnostic\m_Diagnostics001\m_diagnostics001
			Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Emergency\m_Emergency038\m_emergency038
			Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open005\m_open005
			Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Close\m_Close002\m_close002
			Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryWrite046\m_memorywrite046
			Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Diagnostic\m_Diagnostics032\m_diagnostics032

Tracing **Classification** **Requirement Title**

Class **Decomp Title** **Test Case ID Class** **Test Class Name**

Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_General\m_General003\m_general003
Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Network\m_Network001\m_network001
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite010\m_memorywrite010
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_MedicalEvent\m_MedicalEvent001\m_medicalevent001
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security021\m_security021
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Data\m_MemoryRead003\m_memoryread003
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Data\m_MemoryWrite006\m_memorywrite006
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryWrite011\m_memorywrite011
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_MemoryWrite047\m_memorywrite047
Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Emergency\m_Emergency001\m_emergency001
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_General\m_General005\m_general005
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security011\m_security011
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_ARQ017\m_arq017
Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_MemoryRead034\m_memoryread034
Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Discover\m_DiscoverFull034\m_discoverfull034
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_General\m_General016\m_general016
Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security004\m_security004
Status: **Up-to-date** Rev: 2.12 Test: .\M_TCP_Security\m_Security023\m_security023

' FILE_NAME: 'datalink.fm' FILE_REV: '1.34' SECTION: '5.1.1 Packet Control Bytes ' Page No: ' 95'

TRACING FWVT Tested **TLMM_S_408 - EEPROM - WAKEUP DURING EEPROM READ** Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_EEPROM\m_Eeprom001\m_eeprom001

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.2.4 READY INDICATION ' Page No: ' 104'

TRACING FWVT Tested **TLMM_MS_426 - DEFAULT NATIVE MODE RATE**

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.6 General Requirements ' Page No: ' 110'

TRACING FWVT Tested **TLMM_MS_428 - INITIALIZE - UPPER LAYER TO DATA LINK LAYER PARAMETERS**

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.7 Initialize ' Page No: ' 121'

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class** **Name**

TRACING	FWVT Tested	**TLMM_S_432 - POLLING - HOST REQUIRES SECURE BUT LOCAL WAKEUP PACKET DOES NOT INCLUDE SECURITY**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Polling\m_Polling091\m_polling091
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 129'			
TRACING	FWVT Tested	**TLMM_S_240 - POLLING INITIATION**	Status: **Up-to-date**	Rev: 2.6	Test: .\M_TCP_Polling\m_Polling046\m_polling046
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling047\m_polling047
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Polling\m_Polling050\m_polling050
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling025\m_polling025
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling040\m_polling040
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling017\m_polling017
			Status: **Up-to-date**	Rev: 2.4	Test: .\M_TCP_Polling\m_Polling049\m_polling049
			Status: **Up-to-date**	Rev: 2.0	Test: .\M_TCP_Polling\m_Polling016\m_polling016
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Polling\m_Polling026\m_polling026
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling045\m_polling045
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Polling\m_Polling048\m_polling048
			Status: **Up-to-date**	Rev: 2.5	Test: .\M_TCP_Polling\m_Polling024\m_polling024
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 130'			
TRACING	FWVT Tested	**TLMC_S_252 - POLLING - RESTART NATIVE MODE TIMER OR CHANNEL LISTEN TIMER**	Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Polling\m_Polling080\m_polling080
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling092\m_polling092
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling076\m_polling076
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Polling\m_Polling081\m_polling081
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling085\m_polling085
			Status: **Up-to-date**	Rev: 2.1	Test: .\M_TCP_Polling\m_Polling077\m_polling077
			Status: **Up-to-date**	Rev: 2.2	Test: .\M_TCP_Polling\m_Polling082\m_polling082
			Status: **Up-to-date**	Rev: 2.3	Test: .\M_TCP_Polling\m_Polling083\m_polling083
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 130'			

Tracing	Classification	Requirement Title	
Class	Decomp Title		Test Case ID Class / Test Class Name
TRACING	FWVT Tested	**TLMC_S_037 - POLLING - NATIVE MODE TIMEOUT/ CHANNEL LISTEN TIMEOUT**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Polling\m_Polling018\m_polling018 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Polling\m_Polling016\m_polling016
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.8.1 REQUIREMENTS ' Page No: ' 131'	
TRACING	FWVT Tested	**TLMM_S_271 - DISCOVER FULL - CANCEL COLLISION AVOIDANCE TIMER**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Discover\m_DiscoverFull038\m_discoverfull038
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.9.2 DISCOVER FULL ' Page No: ' 142'	
TRACING	FWVT Tested	**TLMC_M_084 - RAW MODE - MASTER PACKET ATTACHED RESPONSE**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_RawMode\m_RawMode003\m_rawmode003 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_RawMode\m_RawMode008\m_rawmode008 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_RawMode\m_RawMode002\m_rawmode002 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_RawMode\m_RawMode004\m_rawmode004 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_RawMode\m_RawMode006\m_rawmode006 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_RawMode\m_RawMode011\m_rawmode011 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_RawMode\m_RawMode012\m_rawmode012 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_RawMode\m_RawMode009\m_rawmode009 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_RawMode\m_RawMode010\m_rawmode010 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_RawMode\m_RawMode007\m_rawmode007 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_RawMode\m_RawMode005\m_rawmode005
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.14.2 RAW MODE (MASTER PACKET ATTACHED) ' Page No: ' 195'	
TRACING	FWVT Tested	**TLMM_S_440 - MEDICAL EVENT - POLL UPON TELM REQUEST**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_MedicalEvent\m_MedicalEvent020\m_medicalevent020
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 203'	
TRACING	FWVT Tested	**TLMM_M_431 - NETWORK DATA - TRANSMIT SECURED NETWORK DATA PACKET**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.2 MASTER ' Page No: ' 231'	
TRACING	FWVT Tested	**TLMM_S_434 - NETWORK DATA - PROCESSING POLLING REQUEST**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Network\m_Network018\m_network018

Tracing **Classification** **Requirement Title**

Decomp Class **Title** **Test Case ID** **Class** **Test Class** **Name**

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 232'

Tracing	Classification	Requirement Title	
TRACING	FWVT Tested	**TLMM_MS_335 - SECURITY - ASSERT BAN KEY/EPHEMERAL BAN KEY LOCAL**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security005\m_security005 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security003\m_security003 Status: **Up-to-date** Rev: 2.9 Test: .\M_TCP_Security\m_Security012\m_security012 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security026\m_security026 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security006\m_security006 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Security\m_Security013\m_security013 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security009\m_security009 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security045\m_security045 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Security\m_Security007\m_security007 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security008\m_security008 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security010\m_security010 Status: **Up-to-date** Rev: 2.9 Test: .\M_TCP_Security\m_Security038\m_security038

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.3 aSSert ban key/ephemeral ban key Local ' Page No: ' 242'

Tracing	Classification	Requirement Title	
TRACING	FWVT Tested	**TLMM_MS_336 - SECURITY - REQUEST BAN KEY OR EPHEMERAL BAN KEY LOCAL**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Security\m_Security005\m_security005 Status: **Up-to-date** Rev: 2.9 Test: .\M_TCP_Security\m_Security002\m_security002 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Security\m_Security007\m_security007 Status: **Up-to-date** Rev: 2.9 Test: .\M_TCP_Security\m_Security012\m_security012 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security006\m_security006 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security045\m_security045 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security008\m_security008 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security010\m_security010 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Security\m_Security003\m_security003 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security009\m_security009 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Security\m_Security026\m_security026 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Security\m_Security013\m_security013

' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.4 REQUEST BAN KEY/EPHEMERAL BAN KEY LOCAL ' Page No: ' 243'

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID / Class / Test Class Name
TRACING	FWVT Tested	**TLMM_S_264 - POWER INHIBIT - UPPER LAYER TO DATA LINK LAYER PARAMETERS**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 264'	
TRACING	FWVT Tested	**TLMM_S_421 - POWER INHIBIT - TRANSMIT PACKET**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_PowerInhibit\m_Power002\m_power002
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 264'	
TRACING	FWVT Tested	**TLMM_S_427 - POWER INHIBIT SERVICE RESPONSE TO UPPER LAYER**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_PowerInhibit\m_Power004\m_power004
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 264'	
TRACING	FWVT Tested	**TLMM_S_423 - POWER INHIBIT IDLE IMMEDIATE - ASSERTED**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_PowerInhibit\m_Power003\m_power003
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 265'	
TRACING	FWVT Tested	**TLMM_S_424 - POWER INHIBIT IDLE IMMEDIATE - DEASSERTED**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_PowerInhibit\m_Power003\m_power003 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_PowerInhibit\m_Power008\m_power008 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_PowerInhibit\m_Power007\m_power007
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 265'	
TRACING	FWVT Tested	**TLMM_M_355 - ARQ - UPDATING THE EXPECTED WAVEFORM SEQUENCE NUMBER 1**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Data\m_ARQ028\m_arq028
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.2 MASTER ' Page No: ' 283'	
TRACING	FWVT Tested	**TLMM_M_429 - ARQ - UPDATING THE EXPECTED WAVEFORM SEQUENCE NUMBER 2**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Data\m_ARQ029\m_arq029
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.1.2 MASTER ' Page No: ' 284'	
TRACING	FWVT Tested	**TLMM_M_441 - WIRELESS WAKEUP - CCA AFTER WAKEUP**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.6 Requirements ' Page No: ' 325'	
TRACING	FWVT Tested	**TLMM_S_412 - RAMWARE RESTRICTIONS**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_EEPROM\m_Eeprom006\m_eeprom006 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_EEPROM\m_Eeprom007\m_eeprom007

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class Test Class Name
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6 Ramware ' Page No: ' 364'	
TRACING	FWVT Tested	**TLMM_MS_388 - MOZART MODULE MEMORY READ**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data017\m_data017 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data016\m_data016 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data018\m_data018
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6.3.1 Mozart module Memory Read ' Page No: ' 367'	
TRACING	FWVT Tested	**TLMM_MS_389 - MOZART MODULE MEMORY WRITE**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data018\m_data018 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data016\m_data016 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data017\m_data017
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6.3.2 Mozart module Memory Write ' Page No: ' 368'	
TRACING	FWVT Tested	**TLMM_MS_436 - MOZART MODULE WRITE TO EEPROM**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_EEPROM\m_Eeprom002\m_eeprom002 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_EEPROM\m_Eeprom003\m_eeprom003
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6.3.3 Special function commands ' Page No: ' 369'	
TRACING	FWVT Tested	**TLMM_S_437 - MOZART MODULE WRITE TO EEPROM AFTER CLOSE WITHOUT POLLING INDICATION**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_EEPROM\m_Eeprom005\m_eeprom005 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_EEPROM\m_Eeprom004\m_eeprom004
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6.3.3 Special function commands ' Page No: ' 369'	
TRACING	FWVT Tested	**TLMM_MS_438 - MOZART MODULE ENABLE RAMWARE**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_EEPROM\m_Eeprom002\m_eeprom002
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6.3.3 Special function commands ' Page No: ' 370'	
TRACING	FWVT Tested	**TLMM_MS_439 - MOZART MODULE CLEAR RAMWARE**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_EEPROM\m_Eeprom002\m_eeprom002
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6.3.3 Special function commands ' Page No: ' 370'	

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class / Test Class Name
TRACING	FWVT Tested	**TLMC_S_286 - OPEN - RESPONSE TO A VALID OPEN REQUEST**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Open\m_Open006\m_open006 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Open\m_Open013\m_open013 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Open\m_Open014\m_open014 Status: **Up-to-date** Rev: 2.0 Test: .\M_TCP_Open\m_Open015\m_open015 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Open\m_Open009\m_open009 Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Open\m_Open012\m_open012 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_Open\m_Open010\m_open010 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Polling\m_Polling024\m_polling024 Status: **Up-to-date** Rev: 2.11 Test: .\M_TCP_Open\m_Open001\m_open001 Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_Open\m_Open002\m_open002 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Open\m_Open004\m_open004 Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_Open\m_Open007\m_open007
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 154'	
TRACING	FWVT Tested	**TLMM_S_293 - OPEN - UNSUCCESSFUL - MISSING KEY**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_Open\m_Open030\m_open030
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.10.1 OPEN - GENERAL REQUIREMENTS ' Page No: ' 156'	
TRACING	FWVT Tested	**TLMC_M_304 - EMERGENCY DURING CLOSE**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Close\m_Close016\m_close016
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 192'	
TRACING	FWVT Tested	**TLMC_M_305 - EMERGENCY - SECOND EMERGENCY REQUEST WHILE PROCESSING FIRST**	Status: **Up-to-date** Rev: 2.4 Test: .\M_TCP_Emergency\m_Emergency043\m_emergency043
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.13.2 MASTER ' Page No: ' 192'	
TRACING	FWVT Tested	**TLMC_S_307 - MEDICAL EVENT - CHANNEL TRANSMIT TIMOUT**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_MedicalEvent\m_MedicalEvent015\m_medicalevent015 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_MedicalEvent\m_MedicalEvent018\m_medicalevent018 Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_MedicalEvent\m_MedicalEvent017\m_medicalevent017
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 202'	
TRACING	FWVT Tested	**TLMC_S_416 - MEDICAL EVENT - INVALID OPEN REQUEST**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_MedicalEvent\m_MedicalEvent019\m_medicalevent019 Status: **Up-to-date** Rev: 2.9 Test: .\M_TCP_MedicalEvent\m_MedicalEvent016\m_medicalevent016

Tracing	Classification	Requirement Title	
Class	Decomp	Title	Test Case ID Class / Test Class Name
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 203'	
TRACING	FWVT Tested	**TLMC_S_417 - MEDICAL EVENT - SERVICE DURATION**	Status: **Up-to-date** Rev: 2.6 Test: .\M_TCP_MedicalEvent\m_MedicalEvent017\m_medicalevent017 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_MedicalEvent\m_MedicalEvent018\m_medicalevent018 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_MedicalEvent\m_MedicalEvent015\m_medicalevent015
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.16.1 REQUIREMENTS ' Page No: ' 203'	
TRACING	FWVT Tested	**TLMM_S_430 - MOZART DATA REQUEST INDICATION**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data018\m_data018 Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_Data\m_Data017\m_data017
		' FILE_NAME: 'appendix.fm' FILE_REV: '1.111' SECTION: '7.6 Ramware ' Page No: ' 364'	
TRACING	FWVT Tested	**TLMM_M_372 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA INSTIGATOR**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.4 Requirements ' Page No: ' 323'	
TRACING	FWVT Tested	**TLMM_S_378 - WIRELESS WAKEUP - SNIFF INTERVAL**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_WirelessWakeup\m_Wakeup001\m_wakeup001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 326'	
TRACING	FWVT Tested	**TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_WirelessWakeup\m_Wakeup013\m_wakeup013
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 326'	
TRACING	FWVT Tested	**TLMM_S_379 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA RECEPTOR**	
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.6.8 Requirements ' Page No: ' 327'	
TRACING	FWVT Tested	**TLMM_S_442 - POWER INHIBIT - WAKEUP COMPLETE INDICATION DURING SYNC SNIFF**	Status: **Up-to-date** Rev: 2.3 Test: .\M_TCP_PowerInhibit\m_Power009\m_power009
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 266'	
TRACING	FWVT Tested	**TLMM_S_447 - POWER INHIBIT - WAKEUP COMPLETE INDICATION DURING ASYNC SNIFF**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_PowerInhibit\m_Power010\m_power010
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 266'	

Tracing	Classification	Requirement Title	
	Decomp Class	Title	Test Case ID Class Test Class Name
TRACING	FWVT Tested	**TLMM_S_448 - POWER INHIBIT - CHANNEL LISTEN TIMER FOR SCAN DURING POLLING**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 266'	
TRACING	FWVT Tested	**TLMM_S_449 - POWER INHIBIT - MEDICAL EVENT TRANSMIT TIME**	
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 266'	
TRACING	FWVT Tested	**TLMM_S_399 - PARAMETER CRC CHECK**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_MemoryTest\m_MemoryTest001\m_memorytest001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.10.2 Requirements ' Page No: ' 337'	
TRACING	FWVT Tested	**TLMM_S_400- PARAMETER CORRUPTION INDICATION**	Status: **Up-to-date** Rev: 2.7 Test: .\M_TCP_MemoryTest\m_MemoryTest001\m_memorytest001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.10.2 Requirements ' Page No: ' 337'	
TRACING	FWVT Tested	**TLMM_S_451- CHARGE NOISE - USE NEW CHANNEL RECOVERY THRESHOLD**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_PowerInhibit\m_Power005\m_power005
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.21 Charging NOise ' Page No: ' 270'	
TRACING	FWVT Tested	**TLMM_S_453- CHARGE NOISE - SAME CHANNEL RECOVERY IMMINENT**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_PowerInhibit\m_Power005\m_power005 Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_PowerInhibit\m_Power011\m_power011
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.21 Charging NOise ' Page No: ' 271'	
TRACING	FWVT Tested	**TLMM_S_454- CHARGE NOISE - SAME CHANNEL RECOVERY IMMINENT RESUMED**	Status: **Up-to-date** Rev: 2.1 Test: .\M_TCP_PowerInhibit\m_Power011\m_power011 Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_PowerInhibit\m_Power005\m_power005
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.21 Charging NOise ' Page No: ' 271'	
TRACING	FWVT Tested	**TLMM_MS_410 - POWER INHIBIT DURING EEPROM OPERATION**	Status: **Up-to-date** Rev: 2.2 Test: .\M_TCP_PowerInhibit\m_Power001\m_power001
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.2.6 EEPROM and power inhibit ' Page No: ' 106'	
TRACING	FWVT Tested	**TLMM_S_331 - NETWORK DATA - SERVICE SUCCESSFUL AT SLAVE**	Status: **Up-to-date** Rev: 2.5 Test: .\M_TCP_Network\m_Network018\m_network018
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 235'	

Tracing	Classification	Requirement Title				
Class	Decomp	Title			Test Case ID Class	Test Class Name
TRACING	FWVT Tested	**TLMM_S_332 - NETWORK DATA - SERVICE UNSUCCESSFUL AT SLAVE**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.4 Rev: 2.1		Test: .\M_TCP_Network\m_Network017\m_network017 Test: .\M_TCP_Network\m_Network028\m_network028
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.18.4.3 SLAVE ' Page No: ' 235'				
TRACING	FWVT Tested	**TLMM_S_405 - SECURITY - ASSERT EBAN KEY DURATION**	Status: **Up-to-date** Status: **Up-to-date**	Rev: 2.9 Rev: 2.2		Test: .\M_TCP_Security\m_Security038\m_security038 Test: .\M_TCP_Security\m_Security046\m_security046
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.19.6 SECURITY - Assert eban key duration ' Page No: ' 255'				
TRACING	FWVT Tested	**TLMM_S_450 - POWER INHIBIT - OPEN RESPONSE PACKETS**				
		' FILE_NAME: 'external.fm' FILE_REV: '1.267' SECTION: '5.2.20.3 SLAVE ' Page No: ' 267'				
TRACING	FWVT Tested	**TLMM_S_398 - PARAMETER CRC UPDATE**	Status: **Up-to-date**	Rev: 2.7		Test: .\M_TCP_MemoryTest\m_MemoryTest001\m_memorytest001
		' FILE_NAME: 'internal.fm' FILE_REV: '1.131' SECTION: '5.3.10.2 Requirements ' Page No: ' 336'				

Medtronic

TYPE
SOFTWARE SPECIFICATION

NUMBER
A36244

REV
C

PAGE
1 OF 396

TITLE
TELEMETRY M
COMMUNICATION PROTOCOL SPECIFICATION

MEDTRONIC CONFIDENTIAL

DOCUMENTATION
QIN ZHANG

DATE
AUGUST 2, 2010

DESIGN
JAVAID MASOUD

DATE
AUGUST 2, 2010

FIRST USED ON
MOZART

REVISIONS

REV	DESCRIPTION	ECO NO.
A	Initial Release into eMatrix. This version reflects firmware Baseline 11.1.	10-00156
B	This release includes updates since Rev. A. The updates are documented under SCRs 186183, 186189, 186623, 186931, 187162, 187540, 187611, 187968, 188074, 188115, 188184, 188393, 188870, 189448, 189885, 190273, 190369, 190661, 190923, 191014, 191062., 192070, 192228, 192305, 192393, 193057 and 193715. It reflects Mozart Firmware Baseline 17.1.	10-01936
C	This release includes updates since Rev. B. The updates are documented 195259. It reflects Mozart Firmware Baseline 17.1.	10-02089

Table of Contents

1.0 INTRODUCTION

1.1 PURPOSE

The System M — Telemetry Communication Protocol defines the protocol for all data communication between a system M slave and master. In this document, the terms instrument and programmer are used interchangeably.

The purpose of this document is to ensure that all Medtronic devices making use of telemetry system M are designed to conform to a single specification to guarantee that a common headless telemetry system can be used regardless of the application.

The use of the common protocol is also beneficial in that it creates the possibility that different organizations within Medtronic can leverage a common telemetry system and shorten time-to-market while being guaranteed that a quality telemetry system is being delivered.

1.2 SCOPE

This specification defines the communication protocol for Telemetry M. All Medtronic products using Telemetry M shall conform to this specification.

This specification defines only the communication protocol for the data exchange between an slave device and the master. Each device product specification or system interface specification specifies aspects of communication which are unique to a particular application.

1.3 APPLICABLE DOCUMENTS

DOC. NO.	DOCUMENT TITLE
A40675	Telemetry M RF Module System Interface Specification
A17245	Telemetry M RF Module Requirements Specification
A34639	Telemetry M Design Input Source Document
123970	Mozart Programmer’s Manual
A44088	Telemetry M RF Module EEPROM Specification
217420	System B Telemetry Communication Protocol Specification

1.4 DEFINITIONS AND ACRONYMS

ITEM	DEFINITION
ACK	Acknowledgment.
ad hoc network	A network comprised solely of stations within mutual communication range of each other via wireless medium. An ad hoc network is typically created in a spontaneous manner. The principal characteristic of an ad hoc network is limited extent in time and space. These limitations allow creation and dissolving of ad hoc networks to be sufficiently straightforward and convenient so as to be achievable by non-technical users of the network facilities.
AES	The “advanced encryption standard”; a block cipher (used by the security feature) which is the de facto industrial/governmental standard. AES is defined in 128, 192 and 256 bit versions. A 128 bit AES block cipher is used by telemetry M.
ARQ	Automatic Request Repeat. A method of error control involving the retransmission of packets.
BAN	Body area network (for the security feature).
baseband	the band of frequencies occupied by the signal before it modulates the carrier (or subcarrier) frequency to form the transmitted line or radio signal
block cipher	A deterministic function which takes as input a data block and key (each of the same length) and outputs a data block (again of the same length). A block cipher also has the property that given the input data block and the output data block, it is exceedingly difficult to determine the key. The block cipher is used by the security feature.
bps	bits per second
broadband	refers to multiple channels
broadband interference	a disturbance that has a spectral energy distribution sufficiently broad so that the response of the measuring receiver in use does not vary significantly when tuned over a specified number of receiver bandwidths
byte	a group of eight bits
CCA	Clear channel assessment. Used interchangeably with channel mapping.
channel	a band of frequencies of a sufficient width to permit its use for radio communications
channel mapping	The process of selecting a clear communication channel. Used interchangeably with mapping.
channel recovery	The process of recovering communication between a master and slave.
Comm	one of the Data Link states; for the enumeration of all Data Link states, see **5.2.3 (States and Transitions)**.
Command code	A byte which identifies a slave message Contains a type (receive ready, reject, etc.) and a tag. The slave repeats this byte on every slave transaction (indication). The master will remove the redundant command codes (which occur on the 2nd, 3rd,..., nth indication).
communication hardware	a conglomeration of RF and baseband hardware; the term is used by Telemetry M when no distinction is necessary.
CRC	Cyclic Redundancy Check
CRDM	Cardiac Rhythm Disease Management. A business unit of Medtronic.

ITEM	DEFINITION
Data	the name of one of the External Services; see **5.2 (External Services)** for an enumeration of all the services
Data indication	A message from the master's Data Link Layer to the Upper Layer. This message contains up to 240 bytes of Memory Data (one to 8 fragments (packets)). Used interchangeably with data transaction.
Data Link Layer	The layer responsible for establishing and maintaining communication. This layer resides between the upper layer and the physical layer.
Data link task	The task responsible for establishing and maintaining communication. This task resides between the upper layer task and the telemetry digital hardware.
data transaction	see **TLMC_M_120 - CH RECOVERY - TRIGGER CHANNEL RECOVERY** in section **5.3.3.2 (MASTER)**.
Destination station	The destination station for a packet. For master packets the slave is the destination station. For slave packets the master is the destination station. Also see source station.
device	Implantable Pulse Generator (IPG) or Implantable Cardioverter Defibrillator (ICD). See also 'implant'.
Device name	see station name
Discover Full	a type of Discover service; see **5.2.9 (Discover)**.
Emergency indication	A message from the master's Data Link Layer to the Upper Layer. This message contains the application receive ready or reject response (~4 bytes). Used interchangeably with Emergency transaction.
Event indication	A message from the a station's Data Link Layer to the Upper Layer. Examples include: channel recovery initiated, channel recovery successful, first request packet sent, and system error.
External Services	refer to **5.2 (External Services)**
External service duration timeout	A service duration timeout occurs when the data link task is not able to complete an external service in time. The time duration is a parameter of the external service request. The data link task notifies the upper layer that a timeout occurred. Used interchangeably with service duration timeout.
External service request	A message from a station's Upper Layer to its Data Link Layer. Used to identify stations, establish communications, terminate communication, exchange data between stations, and reset communication parameters.
External service response	A message from a station's Data Link Layer to its Upper Layer. Contains the status of the external service request.
fade	the variation of radio field strength caused by changes in the transmission path with time
FDMA	Frequency Division Multiple Access - a frequency based access technique which divides the RF spectrum into individual channels with separate users on each channel
FEC	Forward Error Correction. A method of error control allowing the receiver to correct errors.
field	a region or space in which a given effect exists
Finite State Machine (FSM)	a computational model consisting of a finite number of states and transitions between these states
FOP	First Outstanding Packet. The leading or oldest outstanding packet expected by the destination station.

ITEM	DEFINITION
frame	A time-based entity encapsulating one or more master packets. A time-based entity encapsulating one or more slave packets.
identifier	unique manufacturer assigned identifier (ID) for each transmitting station: master ID, slave ID
implant	A synonym for the variety of implant devices produced by Medtronic. Examples include IPG, ICD, etc. Occurs in this specification as ‘***implant device’*** or simply ‘***the implant***’. See also ‘device’.
Implant ID/ Implant station ID	The implant station ID is a 6 byte identification which uniquely identifies a Telemetry M device. It is a composite, consisting of Model ID, SubModel ID, and Serial #. For details, see **5.2.6 (General Requirements)**. Used interchangeably with device ID.
Indication	A message from the master’s Data Link Layer to the Upper Layer. An indication may be for Memory Data, or unrequested data. Used interchangeably with transaction.
instigator	See wakeup-instigator
instrument	A synonym for the external Telemetry M station. Examples include the programmer, bedside monitor, patient activator, etc.
Internal Services	refer to **5.3 (Internal Services)**
IOP	Input/Output Processor
Joule	Absolute unit of energy. One Joule equals one Watt Second.
K_{BAN}	The body area network key. All telemetry M nodes in a body area network share K_{BAN}. K_{BAN} is used by the security feature.
K_{Sess}	The session key. A new session key is calculated at the start of each new secure telemetry session, and is used to secure packets within the telemetry session. K_{Sess} is used by the security feature.
layer	A structuring technique to reduce the complexity of a network architecture. A layer is created where a different level of abstraction is needed. Each layer performs a well defined function. Layer boundaries are chosen to minimize information flow across the interfaces. Layering increases the portability of applications.
LEM	Link Electronic Module
link	The term used to refer to the channel over a short period of time. A link is present when both the master and the slave can communicate with each other in a closed loop.
LSB	Least Significant Bit
M_*nnn*	see the entry below for (master) ‘requirement’.
MAC	Message authentication code. Replaces the CRC used in telemetry C.
Mapping	The process of selecting a clear communication channel. Used interchangeably with channel mapping.
Master	The station which sends the first (wakeup or protocol) packet. The master sends request packets. The slave sends response packets. For example, a master sends an 8K memory read request (using about 20 bytes). A slave responds with 8K bytes using multiple response packets.
master message	One or more master transactions. The master’s application fragments the message into one or more transactions.
master packet	Telemetry communication where the master is transmitting and the slave is receiving. The master is the station which initiated the communication (i.e., open request, etc.).

ITEM	DEFINITION
Medical Event	A patient event which is communicated to an master. In this scenario the slave is allowed to talk first.
message	Upper layer data unit for transferring user data between two stations. A message is composed of a finite number of one or more Telemetry M packets.
mobile	A communication device that can be moved during operation. See the definition for portable.
ms	millisecond
MSB	Most Significant Bit
network	A collection of up to 8 stations which share a common network ID. A network facilitates efficient communication (i.e., broadcast).
network coordinator	A network node whose ID used as the local network’s ID.
network ID	The network coordinator’s ID. Shared among the network nodes.
Octet	Synonym for byte
OSI	Open Systems Interconnection
OSI Reference Model	A standard communication model consisting of Application, Presentation, Session, Transport, Network, Data Link and Physical layers.
packet	A group of binary digits including data and control elements which is switched and transmitted as a composite whole. The data and control elements and possibly error control information are arranged in a specified format. Within Telemetry M, the Data Link layer protocol data unit for transferring information between two stations.
Packet Error Rate	a ratio expressing the number of error packets detected over the total number of packets
Polling off	Polling request with a duration of zero
Polling on	Polling request with a non-zero duration
portable	A communication device that can be moved from location to location, but is only used while at a fixed location.
programmer	External Medtronic product capable being a master or slave using Telemetry M. It can communicate with an implanted device, sensor or another instrument.
protocol	A set of rules governing the format and meaning of the frames, packets or messages that are exchanged by the peer entities within a layer. Entities use protocols to implement their service definitions.
Receive ready	The application version of acknowledgment. Used in the slave to notify the master that processing of the message has completed and that another message can be sent.
receptor	See wakeup-receptor
requirement	A singularly stated, traceable condition stated in terms that are unambiguous, measurable and testable. In this document, our convention for requirement tags takes this form: **TLMC_M_nnn**, **TLMC_S_nnn** or **TLMC_MS_nnn**, where **M** = Master; **S** = Slave; and **MS** = pertains to both.
RF	Radio Frequency
RH	Only radiation hardened (RH) always on (3V) registers are immune to ionizing radiation at the energy levels of concern.
RSSI	Received Signal Strength Indication

ITEM	DEFINITION
Rx	Receive
S_*nnn*	see entry below for (slave) ‘requirement’.
same channel	see **5.3.3 (Channel Recovery)**.
scan	listen to different channels in succession. This term pertains only to the slave, not the master; for details, see **2. (A master provides the Total Polling Duration Time as part of open master packet.)**.
SCR	Same Channel Recovery; see **5.3.3.1 (Requirements)**.
security	The security feature provides for privacy, integrity, freshness, and authenticity.
service	A set of primitives (operations) that Telemetry M provides to the upper layer. The service defines what operations Telemetry M is prepared to perform on behalf of its users, but it says nothing at all on how these operations are implemented.
Service duration timeout	A service duration timeout occurs when the data link task is not able to complete an external service in time. The time duration is a parameter of the external service request. The data link task notifies the upper layer task that a timeout occurred. Used interchangeably with external service duration timeout.
service region	the multi-dimensional space time extent within which Telemetry M stations may communicate
Service request	A service request may be an external service request (discover, open, etc.). The Upper Layer sends external service request messages to the Data Link Layer. A service request may be an internal service request (ARQ, mapping, etc.).
Service response	A service response may be an external service request (discover, open, etc.). The Data Link Layer sends external service response messages to the upper layer. A service response may be an internal service request (mapping, etc.).
session	a state where point to point communication between a specific master and slave has been established and is being maintained by closed loop communication between the two stations
Slave	The station which receives the first (wakeup or protocol) packet. The master sends request packets. The slave receives the request packet and sends response packet(s) in return. For example, a master sends an 8K memory read request (using about 20 bytes). A slave responds with 8K bytes using multiple response packets.
slave packet	Telemetry communication where the slave is transmitting and the master is receiving
slave message	One or more slave data indications. The master’s application assembles the slave data indications into an slave message.
sniff interval	The period that a station sniffs for wakeup transmissions. A trade-off exists: a short sniff interval consumes more current but is very responsive; a long sniff interval consumes less current but is less responsive.
SNR	Signal to Noise Ratio
Source station	The station where a packet originated. For master packets the master is the source station. For slave packets the slave is the source station. Also see destination station.
SR-ARQ	Selective Repeat ARQ algorithm
Standby	A data link state. Also, an application term for waveform disable (see waveform disable).

ITEM	DEFINITION
station	A generic reference to any Medtronic product that is Telemetry M compliant. When no distinction is necessary, ‘station’ is used to refer to either the master or the slave. (Accordingly, a ‘local station/remote station’ pair might refer *either* to an ‘master/slave’ configuration *or* to an ‘master/slave’ configuration; see the text that accompanies **Fig. 20 (Service Request Response Primitives)**.)
station (FCC definition)	One or more transmitters or receivers or a combination of transmitters and receivers, including the accessory equipment, necessary at one location for carrying on a radio communication service or radio astronomy service.
Station ID	The station ID is a 6 byte identification which uniquely identifies a Telemetry M device.
Station name	The station name consists of the following five fields: 1. Station serial number (null terminated ASCII text) 2. Station name (null terminated ASCII text) 3. Station Medtronic/Vitatron (16 bit) 4. Station status (optional) 5. Application specific information (optional)
station services	the set of services which support Telemetry M communications
Supplemental marker (slave transaction)	Communicates the slave’s therapy status on a cycle by cycle basis. Also referred to as unrequested (slave) data.
TDD	**TDD** - Time Division Duplex - Communication system designation which uses time to separate master and slave communications.
Telemetry System M	formal name for ‘Telemetry M’ or ‘Telemetry M system’ or ‘System M’
TLMC	A requirement ported from telemetry C. A singularly stated, traceable condition stated in terms that are unambiguous, measurable and testable. In this document, our convention for requirement tags takes this form: **TLMC_M_nnn**, **TLMC_S_nnn** or **TLMC_MS_nnn**, where **M** = Master; **S** = Slave; and **MS** = pertains to both.
TLMM	A requirement for telemetry M. A singularly stated, traceable condition stated in terms that are unambiguous, measurable and testable. In this document, our convention for requirement tags takes this form: **TLMM_M_nnn**, **TLMM_S_nnn** or **TLMM_MS_nnn**, where **M** = Master; **S** = Slave; and **MS** = pertains to both.
transaction	roughly synonymous with ‘a Data service request’; for details, see **5.2.11.4 (MASTER DATA)**.
transaction (master)	One to 16 fragments (master packets) comprise a master transaction. The maximum transaction size is 480 bytes (since the master content size is fixed at 30 bytes). Transactions are used for Memory Data (read device memory or write device memory).
transaction (slave)	One to 16 fragments (slave packets) comprise an slave transaction. The maximum transaction size is 480 bytes (since the slave content size is fixed at 30 bytes). Transactions are used for slave (memory) data, slave unrequested data, and slave discover name. Used interchangeably with indication.
Tx	Transmit
UCR	Unspecified Channel Recovery

ITEM	DEFINITION
ULH byte/ upper layer header byte	A byte added to each transaction (by the source station). A byte checked (and removed) from each transaction (by the destination station). The byte contains the type (Data or Emergency). It also contains the reset bit and final bit (final transaction of a multi-transaction message, or transaction other than the final one). The terms ‘ULH byte’ and ‘Upper layer header byte’ are used interchangeably.
Unrequested indication	A message from the Data Link Layer to the Upper Layer. This message contains up to 480 bytes of unrequested data (one to 16 fragments (packets)). Used interchangeably with unrequested transaction.
Unrequested (slave) data	Communicates the slave’s therapy status on a cycle by cycle basis. Also referred to as a supplemental marker.
upper layer	generic reference to the protocol stack layer(s) above the Data Link Layer; in this document, the term is used interchangeably with ‘application’.
us	microsecond (in this document, we do not use the abbreviation ‘µs’)
user data	Information generated and used by upper layer or layers outside of the Telemetry M protocol. Telemetry M simply transfers user data between stations without comprehension of user data contents.
wakeup	A method of causing a station to go from a very low current state to a higher current state so native mode telemetry operations can occur.
wakeup-instigator	The station transmitting the wakeup pattern.
wakeup-receptor	The station receiving the wakeup pattern.
waveform data	Real-time data generated on the slave device and transmitted to the master. Real-time data and waveform data are synonymous in this document.
waveform interval	fixed time interval at which slave waveform packets are transmitted
wireless medium	the medium used to convey Telemetry M communications

2.0 GENERAL DESCRIPTION

Telemetry M is a headless RF telemetry system that provides high speed, long range, bidirectional communications between any two nodes (i.e., implanted device and external instrument). Telemetry M differentiates itself from earlier Medtronic telemetry systems with the following features:

1. Wireless wakeup
2. Networking
3. Increased data rate
4. Forward error correction
5. Security

2.1 OVERVIEW

There are multiple ways to view Telemetry M. These different views provide understanding and insight into the system and its features.

2.1.1 MEDTRONIC VIEW

Telemetry M is designed to operate with Medtronic products: implanted devices, external instruments and sensors. The implant device is designed for implant within a human body and provides medical diagnostic or therapeutic functions. The instrument resides outside the human body. The instrument typically provides a user interface, allowing the user to communicate and interact with an implant device. A sensor may be internal or external. It may provide data to the implanted device for therapy.

Earlier Medtronic telemetry systems such as A/B/V require a programming head connected to an instrument to communicate at low data rates with one implant device. Telemetry M eliminates the programming head, increasing the range and data rate. **Figure 1** illustrates this view.

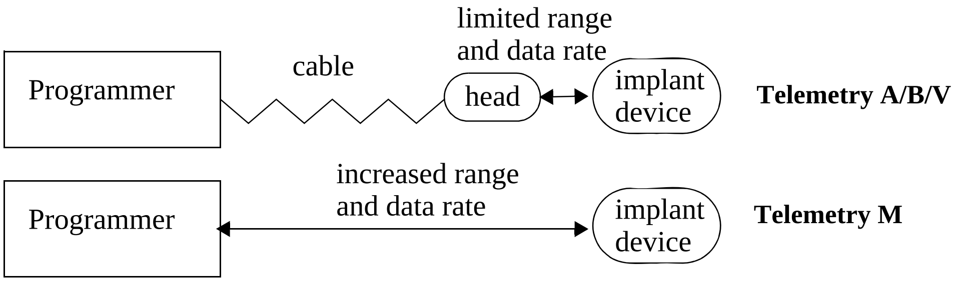


Figure 1: Medtronic Telemetry

An operator using a programming head with a short range telemetry system easily concludes that a specific programming head is communicating with a specific implant device. The programming head provides a tangible mechanism helping the operator control and verify the communication between a specific instrument and a specific implant device. Telemetry M uses other techniques to initiate and maintain communications between a specific instrument and implant device.

Medtronic has different product combinations and use models requiring telemetry. These use models include:

- External instrument and implanted device
- Implanted device and sensor
- External instrument and sensor
- External instrument and external instrument
- Implanted device and implanted device
- Sensor and sensor

From a user's perspective, each model appears to have more differences than similarities in usage, operation and data exchange. Telemetry M is designed such that the telemetry operations of each model are very similar. Where differences exist, Telemetry M provides additional features to support the requirements of the particular use model.

2.1.2 WIRELESS VIEW

A significant industry and body of knowledge exists for wireless RF communication systems. The wireless industry uses its own standard terminology to describe system components, concepts and behaviors. This section uses the terminology of the wireless industry to provide a wireless view of Telemetry M.

Telemetry M is an ad hoc, FDMA/TDD telemetry system operating as a secondary user in the regulated frequency spectrum between 402 and 405MHz (MICS). Telemetry M also has a MEDS band (401-402 MHz and 405-406 MHz).

Telemetry M communicates in the frequency range of 401.000MHz to 406.000MHz. This range of frequencies is a regulated spectrum which Telemetry M shares with other users of that spectrum. Competent authorities and regulatory agencies have extensively analyzed and reviewed this spectrum, the primary users, and Medtronic's proposals to share this spectrum for medical implant communications. The outcome of this extensive effort is that Telemetry M has received nearly world-wide approval to operate as a secondary user in this spectrum.

As a secondary user, Telemetry M must not interfere with the primary users of the spectrum. Other regulatory requirements such as transmit power, receiver sensitivity, channel width, Listen Before Talk, etc. impact the design, operation and behavior of Telemetry M. Refer to the specifications listed in **1.3 Applicable Documents**.

Figure 2 illustrates the relationships of Telemetry M frequency spectrum, channel bandwidth, and channel granularity.

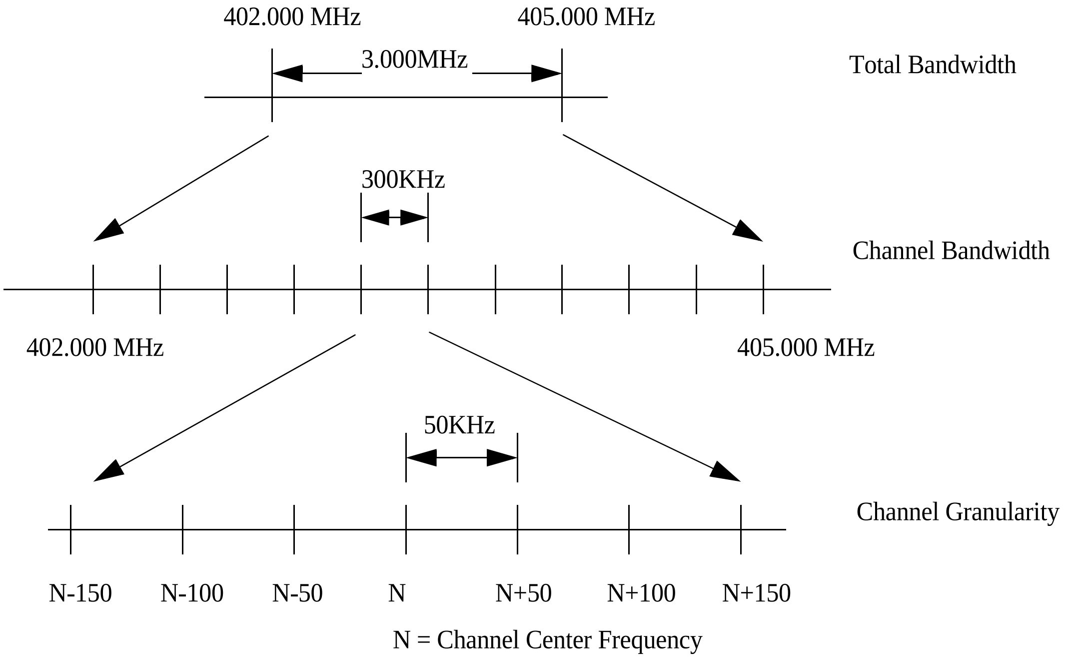


Figure 2: Telemetry M Frequency Relationships (MICS)

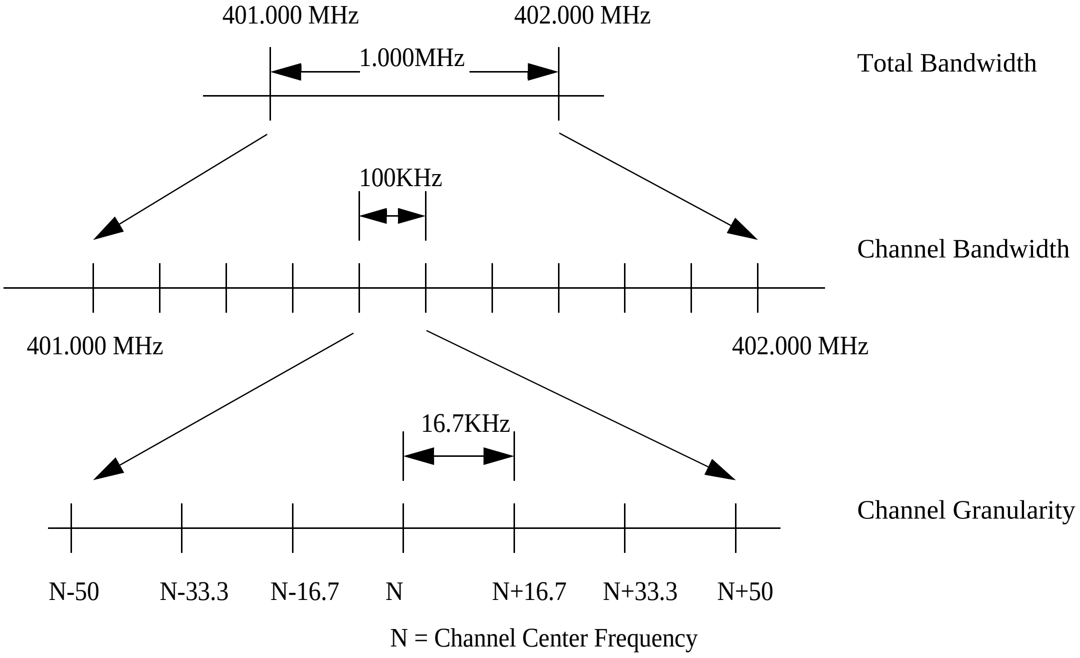


Figure 3: Telemetry M Frequency Relationships (MEDS lower)

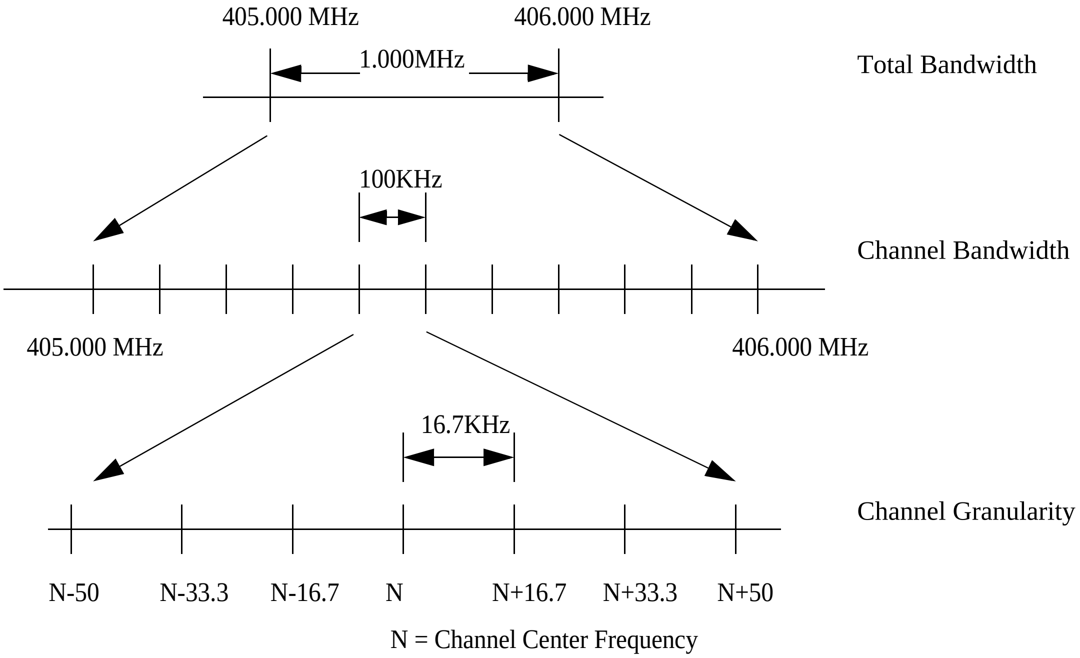


Figure 4: Telemetry M Frequency Relationships (MEDS upper)

A Telemetry M channel consists of a predetermined center frequency with a specified bandwidth on either side of the center frequency. Multiple channels coexist within the overall total bandwidth of Telemetry M. There are many factors associated with the determination and specification of Telemetry M channels. This information resides in **Medtronic Document A34639** (**Telemetry M Design Input Source Document**) and **Medtronic Document 123970** (**Mozart Programmer's Manual**). From the perspective of the Telemetry M communication protocol, it is sufficient to state that Telemetry M has a finite set of communication channels.

Telemetry M channels have dynamic signal levels. A channel that is available at one location may not be available at another location. In a similar manner, channel availability can change over time. A channel that was available today for a particular location may not be available tomorrow at the same location. Telemetry M provides the algorithms for a node to measure the channel signal levels and determine channel availability.

Frequency Division Multiple Access (FDMA) provides separate RF communication channels. Both instrument and implant device must be on the same channel to communicate with each other. Instrument and implant devices on different channels cannot communicate with each other. Channels are occupied on demand by two nodes. A channel is freed when the two nodes terminate their communications. Time Division Duplexing (TDD) allows bidirectional communication over a single radio channel through time sharing. Time is separately allocated for master communications and slave communications.

Ad hoc networking allows Telemetry M communications to be established in a spontaneous manner that is limited in time and space. A pre-existing or permanent communication infrastructure is not required. By definition, setup, use and termination of an ad hoc network must be sufficiently straightforward and convenient enough to be accomplished by non-technical users.

Figure 5 illustrates FDMA and TDD. (For a channel scan example, see **Figure 5**.)

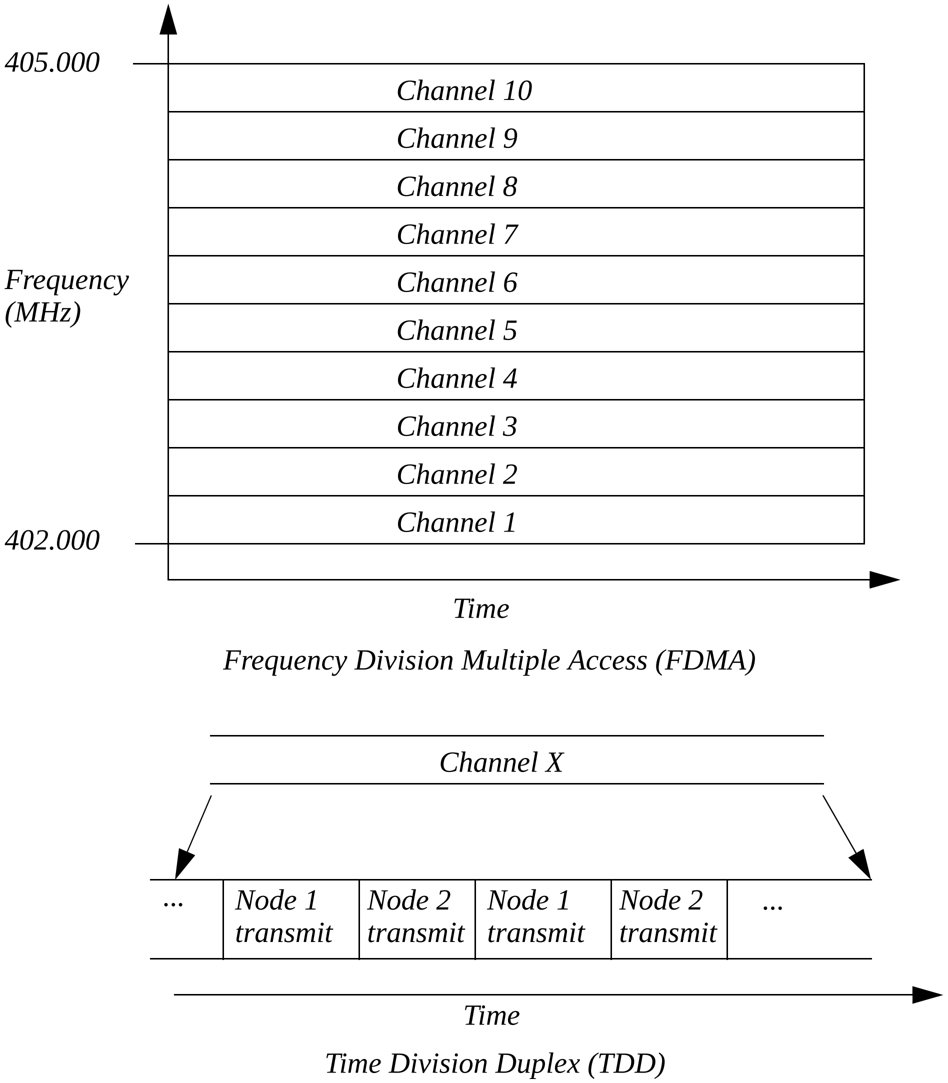


Figure 5: Depictions of FDMA and TDD in Telemetry M (MICS)

Telemetry M channels are not preallocated. Wireless systems such as pagers or cellular telephones preallocate one or more channels for specific purposes. These systems have the advantages of established locations and infrastructure which allow resources like channels, antenna towers, electronic equipment, etc. to be preallocated or dedicated. In addition cellular telephone systems are more sophisticated and have higher authorities such as telephone companies which can manage and control channel and equipment usage and coverage. Telemetry M does not have these advantages.

By definition, each instrument, implant device or sensor containing Telemetry M is a ‘station’. A Telemetry M station transmits and receives to exchange information with other Telemetry M stations. When no distinction is needed, ‘station’ refers to either instrument, implant device or sensor.

Telemetry M stations use a finite set of services to provide Telemetry M communication operations. Services help provide a well-known interface by which Telemetry M users control certain Telemetry M operations. Users invoke Telemetry M services to establish communications, transfer data and terminate communications between Telemetry M stations.

Telemetry M stations communicate over a finite distance, also known as range. Two stations within a common communication range can communicate with each other. If both stations are outside of a common communication range, they cannot communicate with each other. Range is multi-dimensional with three spatial dimensions and one time dimension. Two stations must share a common extent of space and time before communications can be established. This common extent of space and time is called a service region.

Well defined service regions rarely exist for wireless communication systems. Propagation characteristics of RF signals are dynamic and difficult to predict. Small changes in position or direction may result in significant changes in signal strength. Similar effects occur whether a station is mobile or stationary. Moving objects near a station can impact RF signal propagation characteristics. Figure 6 illustrates a simple two dimensional view of service regions. Though the illustration has sharp boundaries for service regions, this is an artifact of the pictorial representation, not a physical reality.

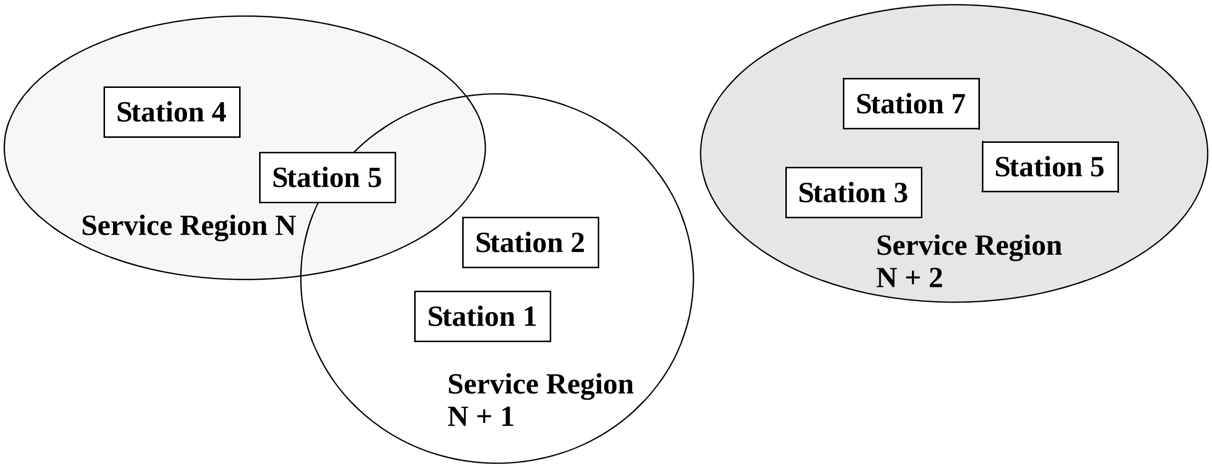


Figure 6: Telemetry M Service Region Example

In this illustration, several stations and their associated service regions are depicted. Service regions can overlap or be disjoint. Service region overlap can occur such that a station may reside in two or more service regions. This overlap can lead to a hidden station problem. For example, assume Station 5 is communicating with Station 4. Stations 1 and 2 may be able to detect Station 5 communications; however, they may not be able to detect Station 4 communications.

Interference sources are another important challenge in wireless RF communication systems. These are electromagnetic signal sources which can interfere with wireless communications. Examples include: other communication systems, electric or electronic equipment, light switches, natural phenomenon, etc. In the time domain, interference sources can be of short duration or long duration affecting single bits or spanning multiple packets. In the frequency domain, interference sources can affect one channel, i.e. narrow band or all channels, i.e. broadband. Interference sources are not readily apparent to the casual observer. Special equipment and techniques are required to detect interference sources. Interference sources are not always present, nor may they always interfere with wireless communications. However, when they do exist, interference sources can impact wireless communications.

Figure 7 illustrates how established Telemetry M communication links may radiate RF energy towards other Telemetry M programmers and implants. In this example, Programmer A communicates with implant device 1 and Programmer X communicates with implant device N. Implant device 1 may receive incidental RF energy from Programmer X or implant device N, depending on the range, channel selection, signal levels, timing, etc. Other stations may receive similar examples of incidental RF energy. Telemetry M uses a number of known and proprietary wireless communication design techniques in the Physical and Data Link layers to reduce the effects of interference and incidental radiation techniques. Channelization, station identification, packet retransmission, etc. are some of the techniques used by the Data Link layer of the communication protocol and discussed later in this document. Reference the corresponding Telemetry M documentation (in 1.3) to learn the design details of the physical layer components of the antenna, RF hardware and baseband hardware.

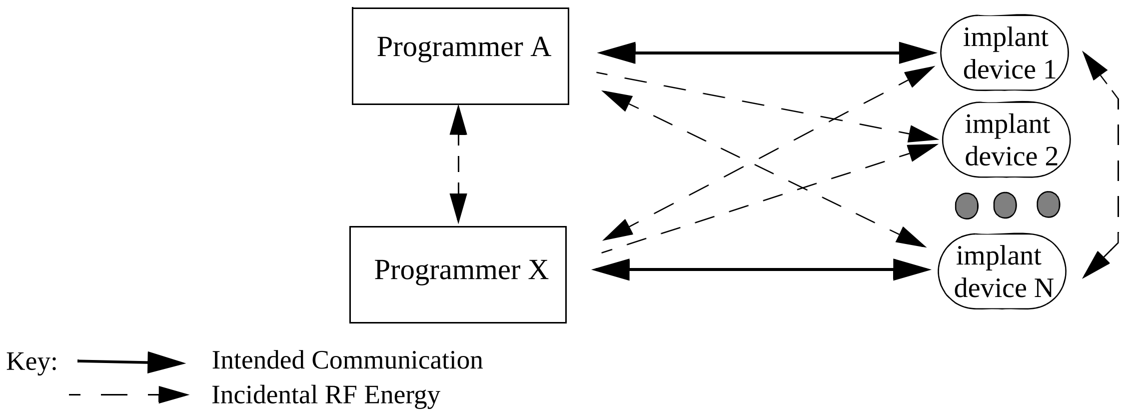


Figure 7: Incidental RF Energy Example

Fades are an important challenge for Telemetry M. By definition, a fade is the variation of radio field strength caused by changes in the transmission path. The effect of a fade is the loss of communication signals and data. Fades can impact both hardware and software operations. Fades have a number of important dynamic characteristics such as strength, duration, rate, etc. Telemetry M uses several proprietary techniques to reduce the effect of fades.

3.0 TELEMETRY SYSTEM M OVERVIEW

3.1 TELEMETRY SYSTEM M GOALS

There are number of performance goals for the Telemetry M system. In general the goals are:

- Provide a standard communication module for all Medtronic products,
- Increased communication range,
- Ease-of-use,
- Higher data rate with respect to Telemetry A, B, and C
- Develop a modular system which will minimize changes for the Application layer.

The following rules are used to achieve the goals stated above:

A standard telemetry module is produced. This has the analog, digital and firmware on one chip. The module may be instantiated as an implanted device, external instrument, or sensor.

3.2 LAYERED MODEL

To simplify definition and implementation, Telemetry System M has been broken down into layers which are modeled on the international Open System Interconnect (OSI) reference model.

The partitioning has been done to follow as closely as possible the OSI definitions for the lowest layers of the standard model. However, the model has been adapted to our specific data needs.

- Each layer performs a well-defined function.
- The layer boundaries are chosen to minimize information flow across the interfaces.

The System M telemetry communication protocol is divided into four distinct communication layers:

- **The Physical Layer:** specifies the characteristics of the physical link. This includes: *physical signal transmission and reception, wakeup, antenna selection and control, communication channel selection and control, and communication timing.*
- **The Data Link Layer:** deals primarily with data encoding and decoding. This includes: *translation between packets and messages, bit-level encoding, channel management and security, error detection and correction, and reliable data transmission/retransmission.*
- **The Application Layer:** defines the content of the transmitted message. This includes: *communication of application messages (request and response), Memory read processing, Memory write processing, and Application specific special commands processing.*

3.3 DATA FLOW BETWEEN LAYERS

Information flow between layers is an important concept to understand. Simply stated, information physically flows between adjacent vertical layers and logically flows between horizontal peer layers. Taking a closer look, data bridges the physical layer in a U-shape (dark arrows in **Figure 8**), and information is mirrored accordingly, in the three upper layers (pale dashed arrows in **Figure 8**). The vertical flow of information between layers represents services between adjacent layers of the protocol stack. The horizontal layers represent a notional flow of information between peer layers on separate products. Only the peer physical layers directly communicate with each other via RF communication signals. All other peer to peer communications represents a logical flow of information.

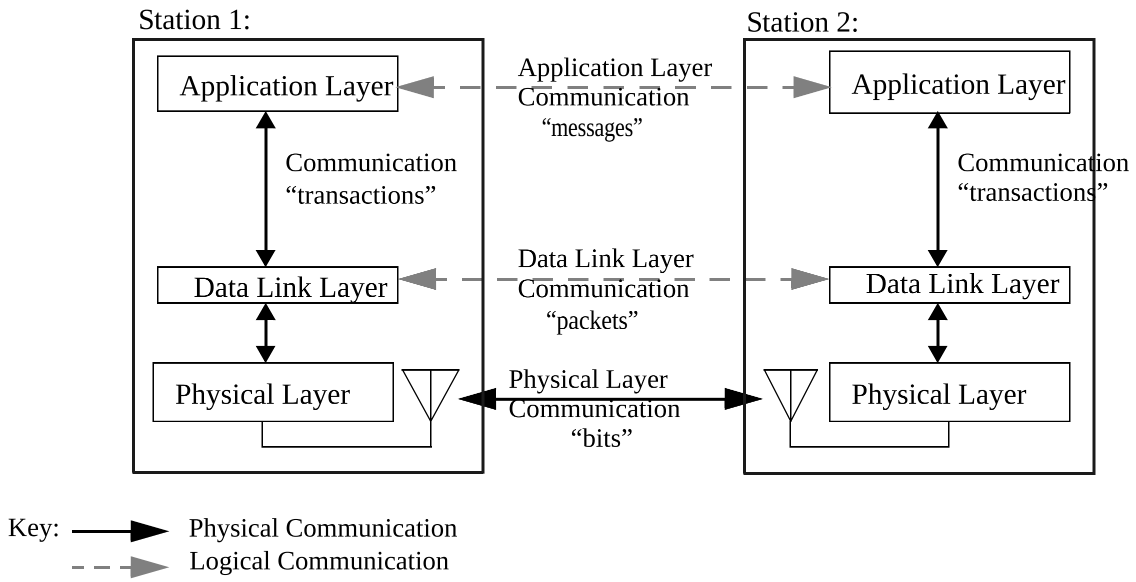


Figure 8: Layer Communications

Information originates in a particular layer or arrives from a higher layer. As information flows through lower layers of the system it is encapsulated as required by each lower layer to ensure proper recovery at the destination peer layer. Data originating in a particular layer normally transfers to the peer layer at the destination.

Figure 9 shows the data flow between two stations.

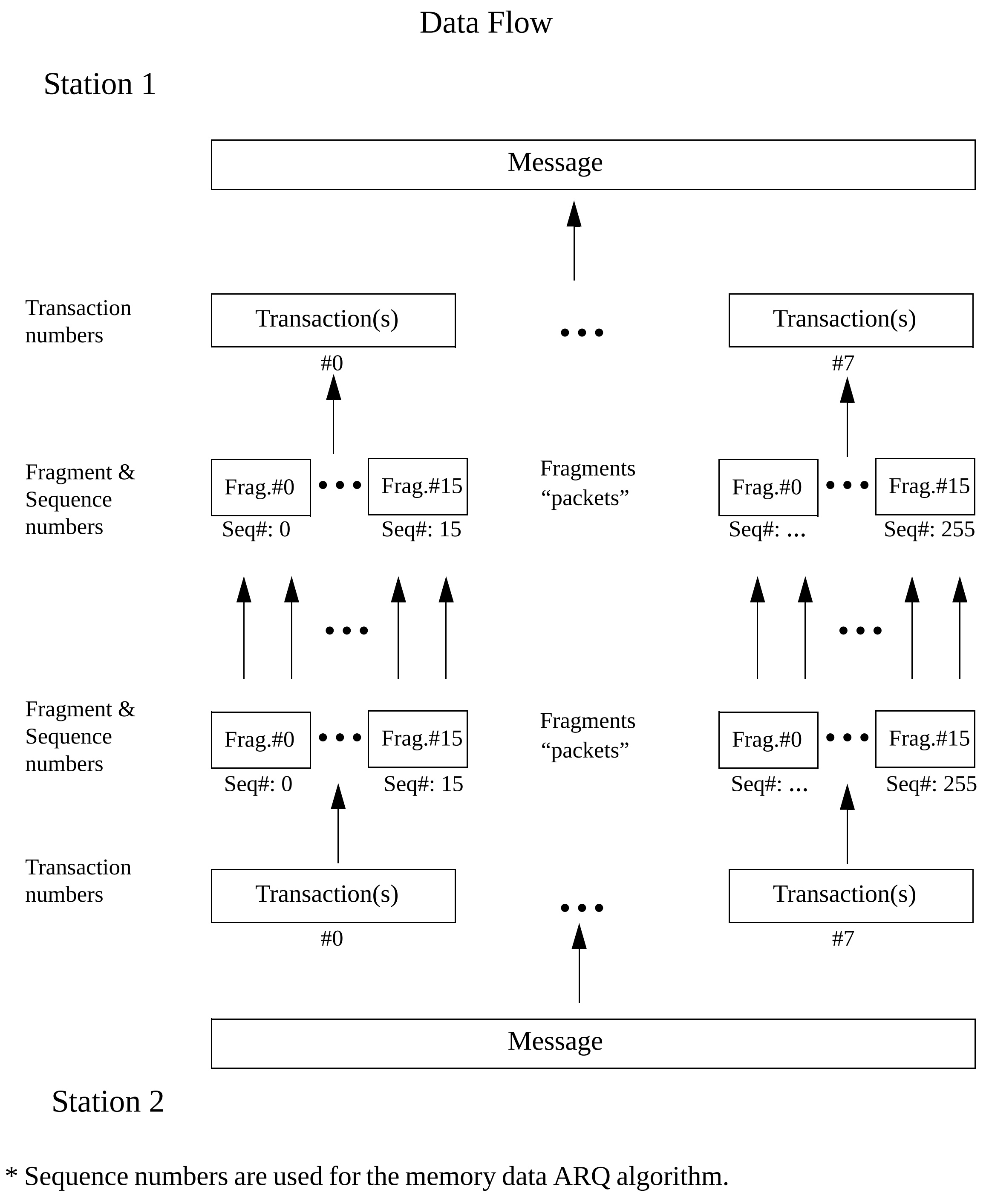
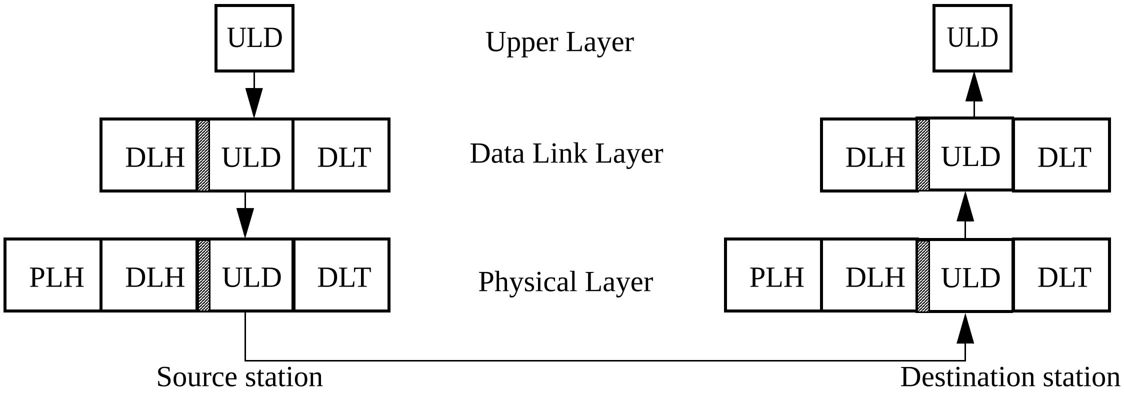


Figure 9: Data Flow Between Two Stations

Figure 10 illustrates application layer data encapsulation as it flows through the data link and physical layers of the source station. On the source station each layer adds their respective header and trailer information. At the destination station, the peer layers remove their respective headers and trailers as the information flows to the application layer. (See also **Figure 20**.)



Key:

ULD Upper Layer Data
DLH Data Link Layer Header
PLH Physical Layer Header

DLT Data Link Layer Trailer
▨ Upper Layer Header (ULH) byte

Figure 10: Upper Layer Data Encapsulation

3.4 TELEMETRY M OVERVIEW

Telemetry M provides a finite set of services which define specific operations related to Telemetry M communications. These services are partitioned into external and internal services. External services are invoked by the application layer, causing Telemetry M to perform specific operations on behalf of the application layer. Internal services are invoked within Telemetry M to support Telemetry M operations.

External services: allow the application layer to perform the following operations:

- Enable and disable Telemetry M using the **Initialize** and **Shutdown** services.
- Establish and terminate communications using the **Discover**, **Polling**, **Open** and **Close** services.
- Transfer user memory data using the **Data** or **Emergency** service. Real-time data transfer is setup via the **Open** service or the **Medical Event** service.
- Transfer a small amount of data using the **Network** service. This service does not require a two way session (open packet exchange).
- Link information using the **Diagnostic** service.
- Provide for privacy, integrity, freshness and authenticity using the **Security** service.
- Manage peak current constraints (or charging interference) using the **Power Inhibit** service.

Internal services: allow Telemetry M to perform the following operations:

- ARQ Algorithms
- Channel Mapping
- Channel Recovery
- Data Transfer Modes
- Memory Test
- Reset
- Tune and Trim
- Wakeup

These external and internal services work together to:

- support request and response communication between two stations (nodes)
- provide high level control of Telemetry M communication operations
- simplify user data transfer operations
- provide transparent communication maintenance and error handling operations

3.4.1 OPERATIONAL EXAMPLE

The following example provides a high level description of Telemetry M operation using an instrument and implant device.

On the instrument, Telemetry M is activated via the application layer. The associated event may originate from various user interface sources. For example, when the instrument is powered up or via a specific user interface action by an operator. Eventually, Telemetry M is commanded to begin communications. Telemetry M selects the best available channel and transmits wakeup packets followed by discover transmissions to determine what implants are available for communications.

On the implant device, Telemetry M is activated via a wakeup transmission from the instrument. After initialization, Telemetry M listens on the channel specified in the wakeup packet. If Telemetry M detects the correct request, it responds with its station identification information.

On the instrument, Telemetry M collects information from the responding implant device(s) and presents this information to the application layer. The M interface layer passes this information up the protocol stack to the appropriate user interface for display to the operator. The operator selects the desired implant device and passes this information back to the application layer which commands Telemetry M to establish communications with the specific implant device. Telemetry M then transmits specific requests containing both the instrument station ID and the implant device station ID to establish communications with the specific implant device.

On the implant device, Telemetry M continues scanning its channels listening for instrument requests. When the implant device receives the specific request containing its station ID, the implant verifies the contents and responds to the specific instrument with both station ID’s. The use of station ID’s allows other programmers or implants detecting these communications to reject them as errors. The correct instrument verifies the implant device response and establishes communications with the specific implant device. At this point, one-to-one communications are established between the specific instrument and the specific implant device.

After communications are established between the instrument and implant device, user data can be transferred between the two stations. Link management, error handling, and channel management are automatically provided by Telemetry M without intervention from the application layer. When no significant data transfer is needed, Telemetry M optimizes the communications to save implant device energy. As needed, the instrument and implant change channels to avoid interference sources and maintain communications.

When communication between instrument and implant device is no longer desired, the instrument operator closes the communications. This command is propagated down to the implant's application layer which commands Telemetry M to close communications. The instrument transmits this command to the implant device. Both Telemetry M stations close their communications and terminate their respective telemetry operations. On the implant device, the application layer may power down Telemetry M to save energy. On the instrument, Telemetry M enters a standby state from which the application layer can command Telemetry M to start another communications or shutdown.

In this example, simple operations allow the application layers of the instrument and implant to use Telemetry M to establish communications, transfer data and terminate communications. Telemetry M handles all the communication specific details between two Telemetry M stations. Telemetry M frees the application layer from the burden of communication specific details.

3.4.2 OPERATIONAL EXAMPLE (MEDICAL EVENT - HOME MONITOR)

The following example provides a high level description of Telemetry M operation using a home monitor and implant device. A home monitor which supports Telemetry M may receive data from the implant via a Medical Event transmission.

When a medical event condition is detected by the implant (e.g., lead impedance out of range, therapies exhausted, etc.) then the implant initiates a session by transmitting its ID. In this model the home monitor continuously listens for an implant. When the home monitor hears from a desired implant a communication session is opened, once home monitor receives the appropriate data then session is closed.

4.0 PHYSICAL LAYER

4.1 INITIALIZATION

The physical layer requires initialization when it is powered up or enabled. This initialization consists of writing parameters to various control registers. After initialization is complete, the physical layer is available for communication operations via the Data Link Layer.

4.2 CHANNEL FREQUENCIES

A finite number of communication channels are provided by the physical layer. These channels are spaced 300KHz apart, each around a center frequency. **Table 1** summarizes the different communication frequencies provided by the physical layer. The Data Link Layer associates a channel number which each frequency.

TABLE 1: MICS COMMUNICATION CHANNEL FREQUENCIES

CHANNEL NUMBER	CENTER FREQUENCY (MHZ)	CHANNEL NUMBER	CENTER FREQUENCY (MHZ)
1	402.15	11	402.300
2	402.45	12	402.600
3	402.75	13	402.900
4	403.05	14	403.200
5	403.35	15	403.500
6	403.65	16	403.800
7	403.95	17	404.100
8	404.25	18	404.400
9	404.55	19	404.700
10	404.85		

TABLE 2: LOWER MEDS COMMUNICATION CHANNEL FREQUENCIES

CHANNEL NUMBER	CENTER FREQUENCY (MHZ)
1	401.05
2	401.15
3	401.25
4	401.35
5	401.45
6	401.55
7	401.65

TABLE 2: LOWER MEDS COMMUNICATION CHANNEL FREQUENCIES

CHANNEL NUMBER	CENTER FREQUENCY (MHZ)
8	401.75
9	401.85
10	401.95

TABLE 3: UPPER MEDS COMMUNICATION CHANNEL FREQUENCIES

CHANNEL NUMBER	CENTER FREQUENCY (MHZ)
1	405.05
2	405.15
3	405.25
4	405.35
5	405.45
6	405.55
7	405.65
8	405.75
9	405.85
10	405.95

4.3 CHANNEL SELECTION

Channels provided by the physical layer are selected by the Data Link Layer. The Data Link Layer determines the desired channel and writes the corresponding physical layer parameters to the appropriate physical layer control registers.

After the Data Link Layer has written the channel selection, a finite amount of time is needed for the physical layer to change to the desired frequency before communication can begin on the selected frequency.

4.4 COMMUNICATION

4.4.1 FRAME

Time division duplexing is used to alternate between station 1 (master) and station 2 (slave) transmission. Each transmission contains information and timing entities that are encapsulated by a larger entity called a frame. **Figure 11** illustrates the major components of a frame.

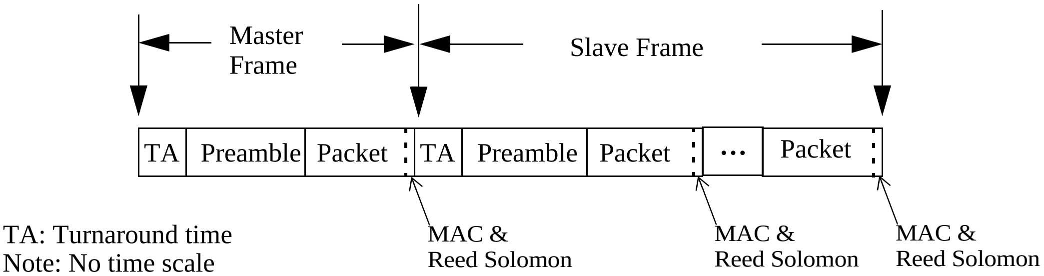


Figure 11: Telemetry M Frame

All frames include the following components:

- **Turnaround Time**: A physical layer time for separating station 1 and station 2 transmissions. The Data Link Layer uses this time for communication operations setup, and for transitions between transmit and receive.
- **Preamble**: A physical layer sequence of bits providing bit and frame synchronization information. The physical layer uses the preamble for master and slave synchronization.
- **Packet, MAC and Reed Solomon**: A fixed size protocol data unit containing control, data and error protection information. One or more packet may be transmitted for either master or slave communications. A fixed number of bytes representing packet error detection information. The physical layer calculates and appends the MAC to each transmitted packet. On the receiving station, the physical layer calculates the MAC of the incoming packet, compares the calculated MAC with the received MAC value, removes the received MAC value while storing the packet in a receive buffer and provides a MAC good/bad indication to the Data Link Layer. The physical layer (at the transmitting station) performs Reed Solomon encoding on a packet before it is transmitted. Errors may occur during transmission (noise, interference, etc.). The Reed Solomon decoder (at the receiving station) processes the packet and attempts to recover the original data. If security is enabled, the data content of the packet will be encrypted or decrypted by the transmitter or receiver.

Master and slave frame sizes are determined by several factors including baud rate, turnaround time, preamble size, packet count, packet size, etc. The baud rate, turnaround time and preamble size are determined by the physical layer. The packet count and packet size are determined by the Data Link Layer. Together, these factors combine to determine the overall size of a master frame or slave frame. For examples, see **Figure 12**, **Figure 13**.

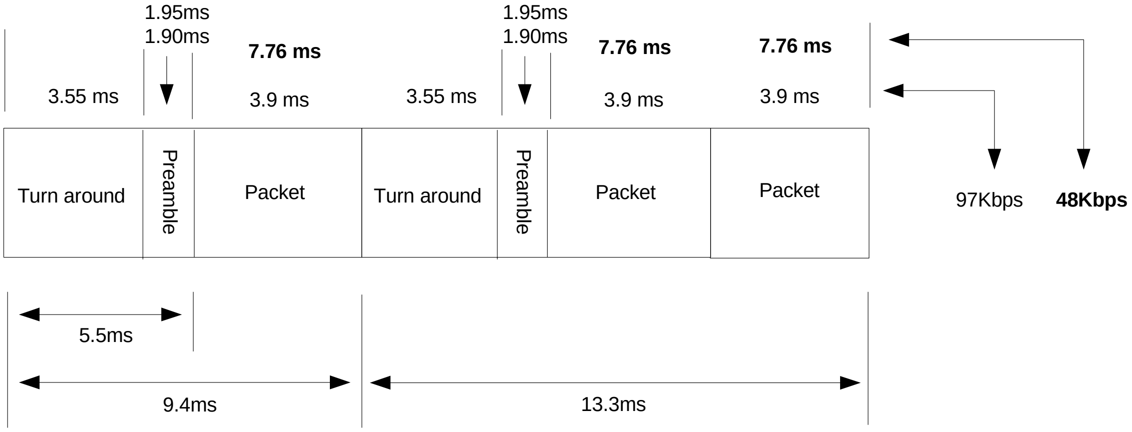


Figure 12: **Frame Timing (#Master Packet, #Slave Packets) = (1,2) Data Rate = 48Kbits/s OR 97Kbits/s**

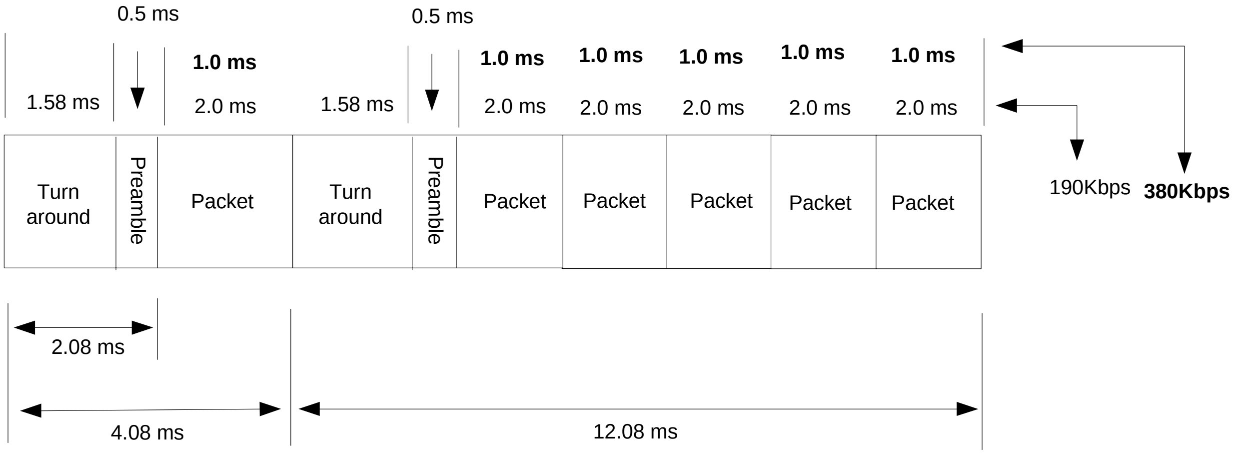


Figure 13: **Frame Timing (#Master Packet, #Slave Packets) = (1, 5) Data Rate = 190Kbits/s OR 380Kbps**

To properly achieve data transfer between local and remote stations, the hardware from both stations needs to be synchronized for transmission and reception. The receiving station has to be listening for the transmitting station to have successful transmission. The Telemetry M hardware has different modes of operations to facilitate this synchronization necessary for successful data transmission and reception.

Three communication modes are provided by the system.

Asynchronous Communication: Managed by the Data Link Layer. This mode provides one master frame and one slave frame. The mode terminates after completion of the slave frame. The time for the master frame and slave frame transmission is fixed.

One-to-one Synchronous Communication: Managed by the Data Link Layer. This mode provides multiple pairs of master and slave frames. The first cycle of the one-to-one synchronous mode provides the capability to synchronize the master and slave so that their frame cycles are aligned.

One-to-N Synchronous Communication: Managed by the Data Link Layer. This mode provides repeated cycles of one master frame and "N" slave frames. The first (transmit) cycle of the one-to-N synchronous mode provides the capability to synchronize the master and slaves so that their frame cycles are aligned.

Figure 14 illustrates the asynchronous and synchronous communication modes.

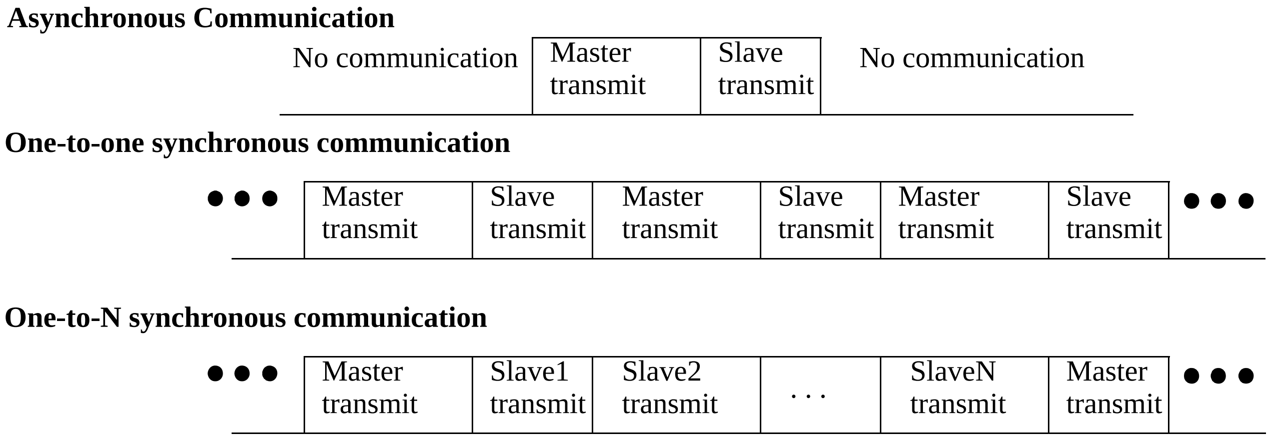


Figure 14: Async and Sync Example

These modes are designed for different profiles of data transfer. The Asynchronous Mode or first cycle sync is used when establishing and reestablishing communications. Second cycle and beyond is used once the communication is established.

The external services create different profiles of data transfer. They need to use these hardware states correctly in order to successfully transfer data. The external services must be invoked by the upper layer in a correct progression in order to properly use these hardware states. The Data Link Layer provides this progression of the various states.

Figure 15 shows the major components of a frame. In this example:

1. A packet is generated by the Data Link Layer and passed to the Physical Layer.
2. The transmit frame begins with the turnaround time and preamble.
3. As the packet is transmitted, the MAC is calculated and the Reed Solomon code is attached and appended to the packet.If security is enabled, the data content of the packet will be encrypted by the transmitter.
4. The frame components are received by the Physical Layer of the destination station.
5. The preamble is removed, the packet received, Reed Solomon error detection/ correction is performed and the MAC result calculated by the Physical Layer.If security is enabled, the data content of the packet will be decrypted by the receiver.
6. The Physical Layer notifies the Data Link Layer for each received packet along with the MAC results.

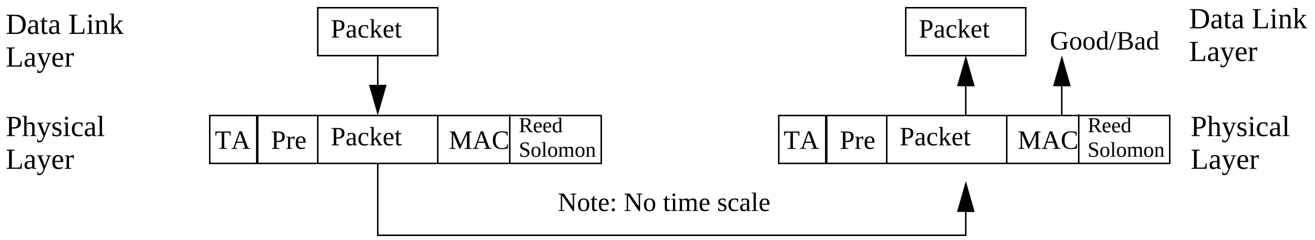


Figure 15: Frame Example

Key: TA = Turnaround time
Pre = Preamble

If multiple packets are transmitted in a frame, the Data Link Layer builds each packet, and the physical layer calculates, transmits and receives MAC information for each packet. The physical layer notifies the Data Link Layer when each packet is received along with the MAC results of that packet.

4.5 REED SOLOMON ERROR DETECTION AND CORRECTION

The physical layer (at the transmitting station) performs Reed Solomon encoding on a packet before it is transmitted. Errors may occur during transmission (noise, interference, etc.). The Reed Solomon decoder (at the receiving station) processes the packet and attempts to recover the original data.The physical layer notifies the Data Link Layer when each packet is received along with the Reed Solomon result (Good or Bad) of that packet. If the error is corrected or there is no error, it means “Good”, otherwise it is “Bad”.

A Reed-Solomon codes RS(63,51) has been implemented in Telemetry M.

4.6 MAC GENERATION AND DETECTION

The physical layer provides packet MAC (message authentication code) generation and detection. The Data Link Layer receives a good/bad MAC indication from the physical layer for each packet that is received.

4.7 RSSI (RECEIVED SIGNAL STRENGTH INDICATION)

The physical layer on an instrument provides the capability to measure the received signal strength on a channel. The Data Link Layer uses this capability during channel mapping operations to determine channel availability.

5.0 DATA LINK LAYER

The Data Link Layer section is divided into three sub-sections:

- Packet Formats (Master and Slave) [5.1]
- External Services [5.2]
- Internal Services [5.3].

5.1 PACKET FORMATS

All the master and slave packets contain the following fields:

- **Control Bytes field**: Control information for all packets. This field is generated and processed by the Data Link Layer.
- **Packet Content field**: This contains either the User Data or the Telemetry M specific data. The User data is generated by application layer and transferred to the appropriate layer for processing. The Telemetry M specific data is generated and processed by the Data Link Layer.
- **MAC (message authentication code) field**: Error detection information. This field is generated and detected by the physical layer.
- **Reed Solomon field**: This contains information to detect/ correct errors.

The packet size is the number of bytes from the start of the control bytes field to the end of the Reed Solomon field.

A graphical representation of master and slave packet format is given in **Fig. 16 (Master or Slave Packet Layout)**.

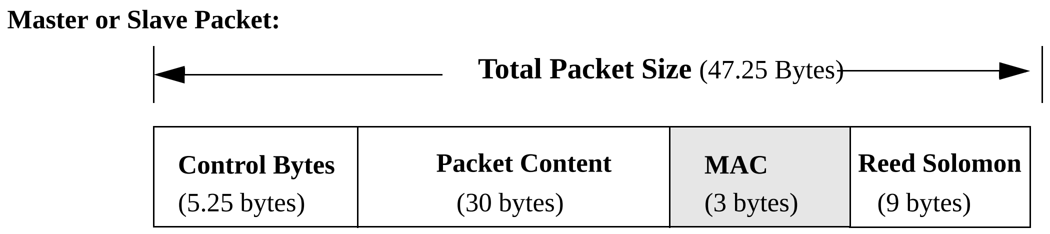


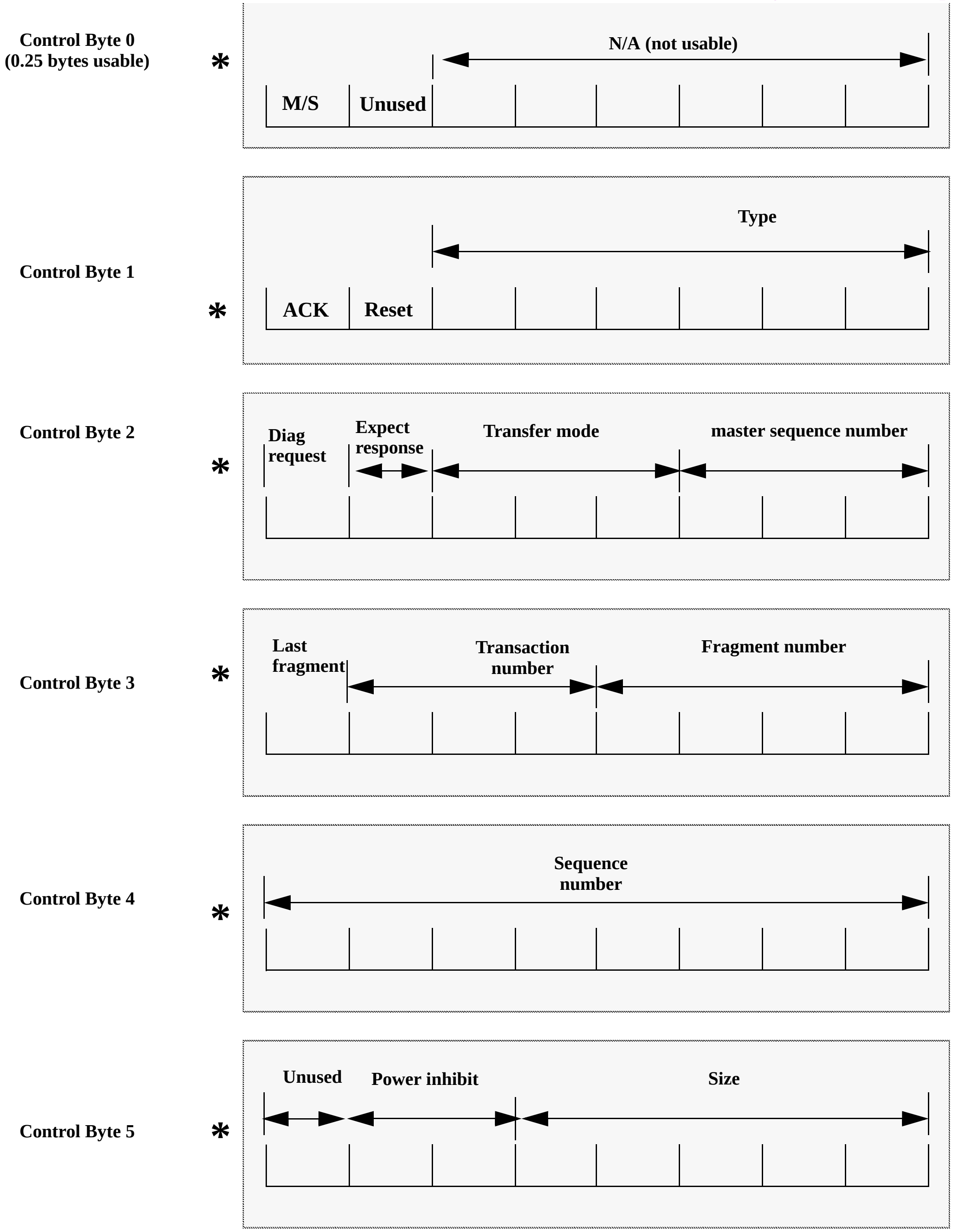
Figure 16: Master or Slave Packet Layout

5.1.1 PACKET CONTROL BYTES

There are five control bytes (and two bits) allocated for packets.

TLMC_MS_001 - PACKET CONTROL BYTES

Fig. 17 (Packet Control Bytes) defines the fields within the control bytes of a master packet.



* The terms ACK, Reset, M/S (master/slave), etc. are defined in **5.1.2 Content Of Control Bytes**.

Figure 17: Packet Control Bytes

Service	PKT Type	M/S	Ack	Reset	Pkt Type	Diag Req	Exp resp.	Xfer Mode	Mstr Seq #	LF	Trans #	Frag #	Seq Num	Power inh	Size
Discover Full	Discover ID	Y	Y	N	Y	N	N	N	N	N	N	N	N	Y	N
Discover Full	Discover Name	Y	Y	N	Y	N	N	N	N	N	Y	Y	N	Y	Y
Open	Open	Y	Y	N	Y	N	N	N	N	N	N	N	N	Y	N
Close	Close	Y	Y	N	Y	N	N	N	N	N	N	N	N	Y	N
ARQ	ARQ	Y	Y	N	Y	N	N	Y	N	N	N	N	N	Y	N
Diag	Diagnostic	Y	Y	N	Y	Y	N	Y	N	N	Y	Y	N	Y	N
Data	Data	Y	Y	Y	Y	N	N	Y	Y	Y	Y	Y	Y	Y	Y
Data	Unrequested	Y	Y	Y	Y	N	N	Y	N	Y	Y	Y	Y	Y	Y
Data	Waveform	Y	Y	Y	Y	N	N	Y	N	Y	Y	Y	Y	Y	Y
Network	Network filler	Y	N	N	Y	N	Y	N	N	N	N	N	Y	Y	N
Network	Network data	Y	N	N	Y	N	Y	N	N	N	N	N	Y	Y	Y
Security	Security	Y	Y	N	Y	N	N	N	N	Y	Y	N	N	Y	N
Medical Evt	Medical Evt	Y	N	N	Y	N	N	N	N	N	N	N	N	Y	N

Key
Y - Entity used
N - Entity *not* used - a don't care

Figure 18: Summary of Control Bytes

5.1.2 CONTENT OF CONTROL BYTES

5.1.2.1 ACKNOWLEDGE (ACK)

The ACK field provides acknowledgment information for the packet(s) of the previous frame. An ACK field with a value of 1 indicates that all the packets received from the previous frame have good MAC check. The value of 0 indicates at least one packet was received with bad MAC.

5.1.2.2 RESET/ RESET ACK

The RESET field is used by the master to instruct the slave to reset its communication parameters. A RESET field with a value of 1 in the master requests the slave to reset its communication parameters. A value of 0 indicates that this command is not active. The RESET ACK is used by the slave to acknowledge the master’s reset request.

5.1.2.3 TRANSFER MODE (DATA XFER MODE)

The Transfer Mode field identifies the current transfer mode between communicating stations. This field is also used when a station initiates a change in transfer mode. Refer to Internal Services section for additional details on transfer mode.

5.1.2.4 PACKET TYPE

The Type field identifies the type of Telemetry M packet.

TLMC_MS_003 - PACKET TYPES

Table 4 (PACKET TYPES) defines the permissible types of Telemetry M packet.

TABLE 4: PACKET TYPES

PACKET TYPE	DIRECTION[1]	DESCRIPTION	VALUE[2]
Discover ID	M/S	Discover service	0x00
Discover name	M/S	Discover service	0x01
Open	M/S	Open service	0x02
Close	M/S	Close service	0x03
Diagnostic	M/S	Diagnostic operations	0x05
ARQ	M	ARQ operations	0x06
Medical event	S	Medical event service	0x07
Memory data	M/S	Data service	0x0A
Unrequested data	S	Data service	0x0C
Waveform data	S	Data service	0x0E
Spare	M/S	Unused (Reserved for Tel M)	0x10
Network data	M/S	Network service	0x11
Network filler	S	Network service	0x12
Security - proximity security	M/S	Security service	0x13
Security - request key	M/S	Security service	0x14

1. M = Master, and S = Slave. Also see the control field “source of packet is master/slave”.
2. Most of the “unused” values are used by telemetry C. For example, a value of 0x0B is a “memory data with size” packet for telemetry C.

5.1.2.5 LAST FRAGMENT (LF)

Zero (0) indicates there are more packets for this transaction. One (1) indicates this packet is the last packet of this transaction.

5.1.2.6 SIZE (COMMON)

The Size field is used to indicate the number of bytes of user data in the packet content field. It is present for the packet types memory data, unrequested data, waveform data, discover name, and network data. The size ranges from one (1) byte up to the (maximum Packet Content size bytes). Specifically, the size of a packet ranges from 1 to 30 bytes.

5.1.2.7 TRANSACTION NUMBER

The Transaction field identifies the transaction number of an (fragment) packet. All the (fragment) packets resulting from one external service request are of the same transaction and have the same transaction number.

5.1.2.8 FRAGMENT NUMBER

The Fragment field identifies the fragment number of a particular packet within an transaction. The fragment number is used to assemble the packets back into one transaction.

5.1.2.9 SEQUENCE NUMBER

The Sequence Number field contains the sequence number for the packet. The eight (8) bit sequence number has a range between zero to 255.

For slave transmitted packet, there are 3 sets of sequence numbers. They are used by the following packet types during communication session:

- Memory Data
- Unrequested Data
- Waveform Data

Each packet is uniquely identified by the sequence number within its packet type.

For master transmitted packet, it is used to indicate the expected waveform sequence number during communication session.

For network data request packet, sequence number is "0" all the time. The slave will ignore network data request packets with any other sequence number.

For network data and network filler response packet, sequence number is "0" all the time.

5.1.2.10 MASTER/SLAVE (M/S) - SOURCE OF PACKET

The source field is set to one if the master is the sender (source) of the packet. This type of packet is also known as a request packet.

The source field is set to zero if the slave is the sender (source) of the packet. This type of packet is also known as a response packet.

5.1.2.11 DIAGNOSTIC REQUEST (MASTER ONLY)

This field is used to distinguish between diagnostic request packets and diagnostic filler packets.

5.1.2.12 EXPECT RESPONSE (MASTER ONLY)

The expect response field is set to true if the request packet desires a data link acknowledgement. In general this is always true, except for a network data request that desires no data link acknowledgement.

[A transmit only node may use this feature.]

5.1.2.13 MASTER SEQUENCE NUMBER

The master uses this master sequence number for Data packet transmission.

5.1.2.14 POWER INHIBIT

The slave controls the power inhibit mode which indirectly controls the channel recovery threshold. The control bits mirror the power inhibit modes (see **Table 71 (POWER INHIBIT MODES)**).

5.1.3 CONTENTS OF PACKET(S)

The packet contents for various packet types are defined in **5.2 External Services** and **5.3 Internal Services**.

5.2 EXTERNAL SERVICES

The Data Link provides user services to identify stations, establish and terminate a communication path between master and slave, exchange application data between stations, reset communication parameters and obtain communication statistics.

The External Services available to the Data Link Users are:

- Initialize - initializes Data Link operation [5.2.7]
- Discover - obtain station IDs and Name from scanning slaves [5.2.9]
- Polling - provide station ID and Name to masters using Discover [5.2.8]
- Open - establish communication between master and slave [5.2.10]
- Data - application data exchange between two stations [5.2.11]
- Diagnostic - communication diagnostic information [5.2.12]
- Emergency - Application data between two stations (this service can interrupt the service in progress) [5.2.13] (Master)
- Raw Mode - transmit the user specified packet [5.2.14]
- Close - terminate communication between master and slave [5.2.15]
- Medical event - establish communication between master and slave [5.2.16]
- Network - small application data exchange between two or more stations (without open packet exchange). Also facilitates network configuration [5.2.18]
- Security - provides for privacy, integrity, freshness and authenticity [5.2.19]
- Power inhibit - provides ability to pause telemetry to reduce the peak current demands [5.2.20]
- Charging Noise - provides ability to manage charging interference [5.2.21]
- Shutdown - disable Data Link operations

Initialize service allows the Upper Layer to enable or disable Telemetry M Data Link operations on the local station.

Discover and **Polling** services provide automatic communication identification between master and slave device.

Open and **Close** services allow the Upper Layer to establish and terminate Telemetry M communications between a specific master and a specific slave device.

Data service provides user the data transfer capability between two communicating stations.

Diagnostic service can provide diagnostic information on both the local and the remote stations. (Development use ONLY — for debugging purposes.)

Emergency service provides user the data transfer capability between two communicating stations. Note: This is the only service which can interrupt a service in progress.

Raw Mode service provides a mechanism for sending and receiving data directly, without mediation by the lower layers. (Development use ONLY — for debugging purposes.)

Medical Event service allows the Upper Layer to establish communications between a specific master and a specific slave. This differs from the Open service in that the slave talks first. In the Open service, the slave listens first.

Network service provides the capability to transfer a small amount of user data (no open packet exchange needed). Also facilitates network configuration.

Security service provides the provides a mechanism to exchange secret keys. Once the keys have been exchanged the service provides a means of communicating with privacy, integrity, freshness and authenticity.

Power inhibit service provide a mechanism to the host to control the transceiver. When Power Inhibit is asserted the Tel M transceiver is turned off and when power inhibit is deasserted the transceiver is turned ON.

Charging Noise provides a mechanism to slave host to control charging interference on downlinks (not on uplinks). Charging noise feature is implemented by using the power inhibit service request.

Shutdown service provides a graceful mechanism to cease the data link layer operations prior to turning off the 2V supply.

5.2.1 POWER UP SEQUENCE

The power up sequence for this module consists of three states:

Disable (CPU OFF) — This is a power-off state. Transition is made out of this state to read EEPROM on the following:

Reg_bp_boot_stat:

- bp_cold_nreset_boot - a cold reset from NRESET
- bp_warm_nreset_boot - a warm reset from NRESET
- bp_irqc_rst_boot - a warm reset from interrupt controller block (i.e. fault reset)
- bp_hst_rst_boot - a warm reset from host interface.
- bp_hdw_boot - a host dummy write.

Reg_wu_irq0:

- syn_pll_lock_err - Synthesizer PLL lock error interrupt/status
- wu_it_wrap_up - wakeup interval timer wrap-up threshold
- valid wakeup (wu_snf_first_pkt, wu_snf_last_pkt)
- wu_snf_mics_high current
- wu_snf_meds_high_current

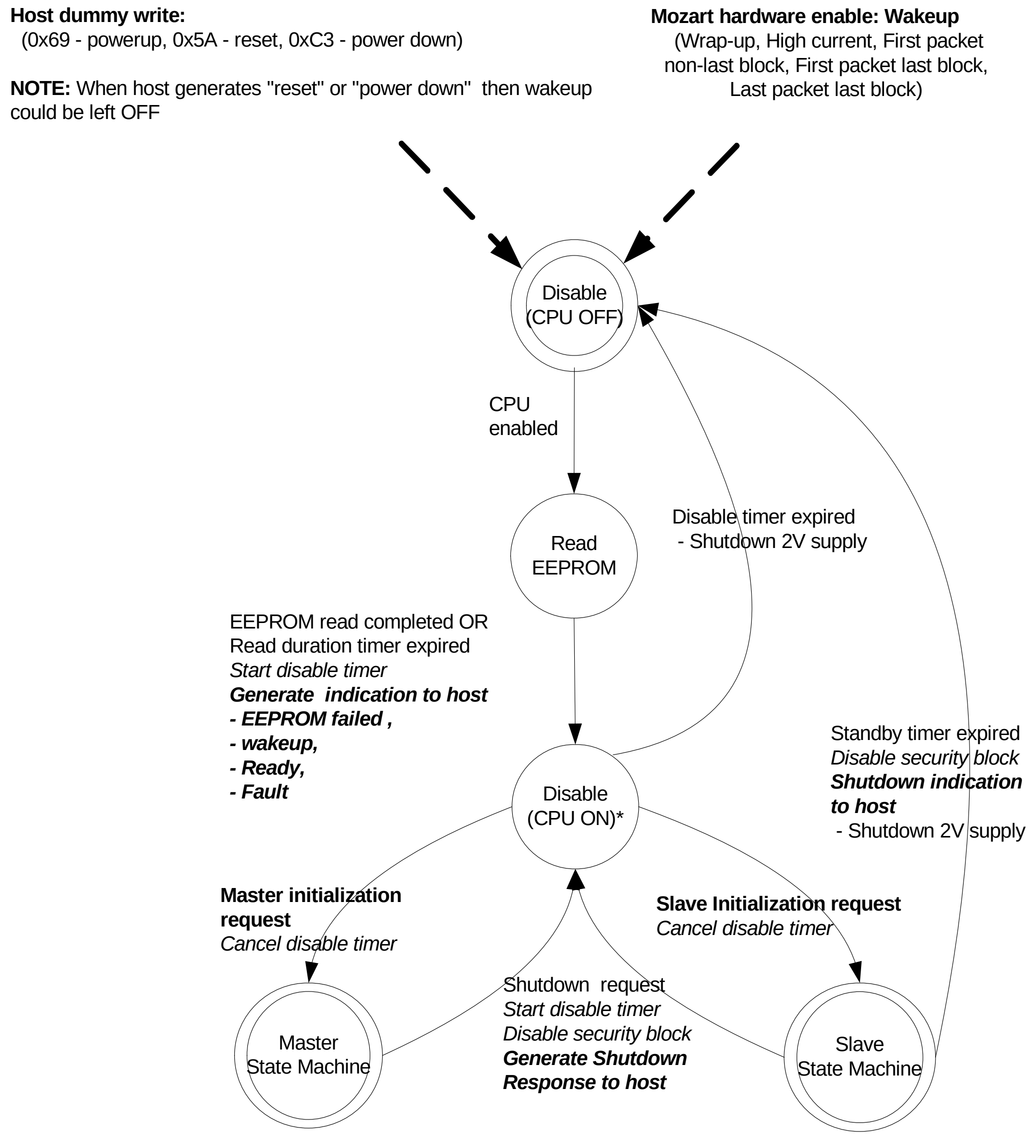
For further detail see programmer’s manual: 123970

Read EEPROM — In this state the CPU is enabled and reads the pertinent structures from the EEPROM. Transition is made out of this state to disable (CPU ON) state when:

- Read EEPROM is complete (successful, failed), or
- Read EEPROM duration timer expires

For pertinent structures see section (7.8 EEPROM Content) and for ramware see sections (7.6 Ramware) in the appendix of this document.

Disable (CPU ON) — In this state CPU is ready to process the initialization request from the host. When disable timer expires and shutdown indication is generated to the host 2V supply is also turned off. Note: sniffing will function as configured by the host.



Note: Disable timer will need to accommodate wakeup last packet last block

Figure 19: Powerup Sequence State Machine

5.2.2 EEPROM

The Telemetry Module shall be capable of reading and writing to EEPROM throughout its the operating life. When telemetry module is powered ON or it is reset there is a group of parameters (See **TABLE 122: STRUCTURES IN EEPROM**) shall be copied from EEPROM to RAM memory.

5.2.2.1 EEPROM DATA ARRANGEMENT

The data in the EEPROM is arranged in structures where each structure consists of 128 bytes. Each structure contains various 16 byte blocks. These 16 bytes block contain 14 bytes of data and 2 CRC bytes. During the read function if the 14 bytes data block has no CRC error then the EEPROM read continues with the next block and so on till all data is moved to the RAM. The total number of data content written to the RAM from the EEPROM for each of the data structure is 112 bytes where 110 bytes are the data followed by 2 bytes of total CRC in reverse order for 110 data bytes. The total CRC bytes of each structure will always result in 0x0000.

The capability exist to support up to 43 structures of RAMWARE code patches in the EEPROM, these are indicated by the RAMWARE control parameter. Prior to reading RAMWARE code from the EEPROM the ramware control parameter is read to know how many RAMWARE code patches are present. The total CRC for each RAMWARE patch is also stored in the ramware control parameter. Each time RAMWARE patch is read the total CRC will be checked to insure the ramware validity.

5.2.2.2 EEPROM READ AND WRITE

When the Mozart CPU is enabled, the firmware will restart operating in a well defined manner. It will first set all critical RAM locations to a known value. It will then recover operating parameters from the EEPROM. The EEPROM is used to preserve non-volatile always on registers to mitigate alpha particle corruption. If the EEPROM is intact, it will overwrite the non-volatile always on registers which they mirror. The EEPROM is also used to preserve some volatile always on registers, some firmware operating parameters and RAMware (if it exists).

These steps will initialize all RAM to a known value as follows:

1. Clear all RAM

   Fixed RAM (0x00 - 0xff), 0x600 - 0x3EFF)
   Master RAM: (0x14000 - 0x17FFF)
   Slave RAM: (0x24000 - 0x27EFF)
   Dual Port: (0x75000 - 0x75FFF)
2. Set all RAM which requires non-zero initial values
3. Update RAM from the EEPROM if it has survived intact:

   Mirror of non-volatile always on registers. This is p_hdw1, p_hdw2 and part of p_wakeup.
   Mirror of volatile always on registers. This is a portion of p_wakeup. Note that volatile registers are only overwritten upon a reset.
   Firmware operating parameters. This is p_config and a portion of p_wakeup
   Ramware control and RAMware

The ability to modify the EEPROM of the telemetry module is to be allowed under at least three conditions:

- telemetry module needs to be reconfigured with a different set of hardware parameters
- telemetry module has had a reset, and EEPROM can be restored to manufactured original condition, and Download RAMWARE to be saved in EEPROM. The primary use of the EEPROM is data recovery upon a power up or a reset

5.2.2.3 REQUIREMENTS

5.2.2.4 READY INDICATION

TLMM_MS_406 - EEPROM - NON-VOLATILE REGISTERS

When the CPU is enabled and the EEPROM is read successfully, the Mozart module shall update the non-volatile always on registers. Upon a reset, the Mozart module shall update the volatile registers.

[The data read from the EEPROM is copied to a specific location into the RAM **TABLE 122: STRUCTURES IN EEPROM**]

TLMM_MS_407 - EEPROM - LOAD RAMWARE APPLICATION

The Mozart module will load the RAMWARE application (if one exists) into RAM if there are no unrecoverable error reading the RAMWARE section of the EEPROM. Additionally it will begin executing the RAMWARE application.

TLMM_S_408 - EEPROM - WAKEUP DURING EEPROM READ

When CPU is enabled via wakeup then wakeup indication will be generated to the host upon the successful read of the EEPROM. For wakeup indication see **TABLE 93: WAKEUP INDICATION (SLAVE)**

TLMM_MS_262- READY INDICATION - INDICATION TO UPPER LAYER

The Ready indication will be generated to the Upper Layer when Data link layer is enabled by Host, or via Reset and the EEPROM has been successfully read. See **Table 5 (READY INDICATION (MASTER OR SLAVE))**

TABLE 5: READY INDICATION (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x20 - Ready indication
Sub-type	N/A	1	The indication sub-type	0x01 - Host dummy write(1) 0x02 - Host reset(2) 0x03 - Host dummy write followed by Host reset 0x08 - Warm Nreset(3) 0x10 - Cold NReset(4)
Size	N/A	2	The size of the info field	1
Info	Ready indication status	1	Status of the indication	0x01 - Successful 0x0A - Already ready
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

1. This value occurs when 0x69 is written to the host enable reset register while the RFM is not powered up. Note, if the RFM is not powered up, any write to the RFM will also cause this indication sub-type.
2. This value occurs when 0x5A is written to the host enable reset register and the CPU is on.

3. This value occurs when the Nreset line is toggled when the CPU and the TAP controller are both on.
4. This value occurs when 0x69 is written to the host enable reset register while the RFM is powered up.

5.2.2.5 EEPROM FAILED INDICATIONS

During the read operation of the EEPROM if a CRC error is encountered data link layer will make three attempts to read the EEPROM before generating a failure indication to the host. Upon a successful read of the EEPROM a ready indication will be generated to the host.

The Data Link Layer will read the EEPROM each time it is enabled (by the wakeup feature or by the host). If the EEPROM read fails after three attempts an indication will be sent to the host see **Table 6 (EEPROM READ FAILED INDICATION (MASTER OR SLAVE))**

TLMM_MS_409 - EEPROM - FAILURE INDICATION

When the CPU is enabled (host dummy write or wakeup) data will be read from the EEPROM. The Mozart module will send an EEPROM read failure indication to the host if there are unrecoverable errors encountered while reading any of the following:

a. p_hdw1 & p_hdw2 - mirror of non-volatile always on registers
b. p_wakeup - mirror of non-volatile always on registers, and mirror of volatile always on registers, and firmware operating parameters
c. p_config - firmware operating parameters
d. p_ramware_control and RAMWARE code

See **Table 6 (EEPROM READ FAILED INDICATION (MASTER OR SLAVE))**.

[ROM defaults will be used if (a) - (c) do not survive intact.

The content read from the EEPROM is listed in appendix section 7.7]

TABLE 6: EEPROM READ FAILED INDICATION (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x30 - EEPROM read failed indication
Sub-type	N/A	1	The indication sub-type	0x01 - EEPROM Parameter read failed 0x02 - EEPROM RAMware read failed
Size	N/A	1	The size of the info field	1
Info	EEPROM Read failed indication status	1	Status of the indication	0x0B - block level CRC failed during EEPROM read 0x0C - Max EEPROM read duration expired 0x11 - Max EEPROM read duration expired because Power Inhibit takes too long(slave only)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

5.2.2.6 EEPROM AND POWER INHIBIT

TLMM_MS_410 - POWER INHIBIT DURING EEPROM OPERATION

During the EEPROM operation if power inhibit is invoked then EEPROM operation will be suspended.

[During an EEPROM operation if power inhibit is invoked by the host data link layer will complete the on going EEPROM structure operation. The operation on the remaining structure will resume upon the termination of power inhibit.

A timeout indication will be generated to the Upper Layer if the total duration of the EEPROM read operation lasts longer than 10 seconds. Upon a successful read of the EEPROM transition will be made to DISABLE CPU ON state where disable ON timer will be started, power inhibit interrupts will be unmasked. Wakeup indication “Wakeup Complete”, containing the wakeup packet contents, shall be sent to the upper layer. (see **Table 93 (WAKEUP INDICATION (SLAVE))** if slave is waiting for the “last packet last block” wakeup packet (after the “first packet last block” wakeup packet has enabled the CPU).

EEPROM write operation will occur in Standby state. For detail about the EEPROM write operation, see **7.6.2 Mozart module Memory Read and Write**. A timeout indication (if EEPROM write command from Master) or response (if EEPROM write command from slave host) will be generated to the Upper Layer if the total duration of the EEPROM write operation lasts longer than Standby Timer because Power Inhibit takes too long (For “standby timer” see **Table 129 (PARAMETERS FROM EEPROM)**).

Power monitor bit shall only be set during EEPROM operation.]

Slave host can use the Power monitor bit to know whether Slave Mozart is in the middle of EEPROM operation. Slave host should take the power inhibit timing into consideration while picking the standby timer duration if EEPROM write is expected to occur.

Also on the other hand, care should be given in extending the standby timeout since the slave does not sniff nor listen in native mode during standby. The worst case write of a 4K byte RAMware should take about 400msec in the lack of power inhibit.

5.2.3 STATES AND TRANSITIONS

The Data Link States are:

Disable —	This is a power-up state. If initialization is successful, then transition is made to Standby state; otherwise, no transition is made to any other state.
Standby —	In this state, telemetry is enabled, waiting for external request.
Acquire —	The system is transmitting or receiving for identification and for establishing communications.
Comm —	Telemetry communication is established between two stations.
Channel Recovery —	This is a Data Link Layer internal state, and is used to recover the communication between the slave and the master.
Inactive —	This state applies to master only. In this state the telemetry is temporarily inactive because Channel Recovery has failed. While in this state if a data request, emergency request, open request, or remote diagnostic request is received then telemetry is activated by triggering Channel Recovery.

5.2.4 REQUESTS AND SERVICES

The various External Services have the same general behavior which can be summarized as follows:

- The Upper Layer requests a particular External Service.
- Data Link Layer performs the specific actions for the particular service.
- Data Link Layer returns status information and/or responses associated with the service.

Most of the External Service requests issued by the Upper Layer communicate to the remote station, and involve operations from the remote station. This is achieved by sending telemetry request packets to the remote station. Completion of these External Service requests may involve response packets from the remote station. **Fig. 20 (Service Request Response Primitives)** shows the terminology used to describe the External Service request operations. In this figure, the local station could be the instrument, implant, or sensor.

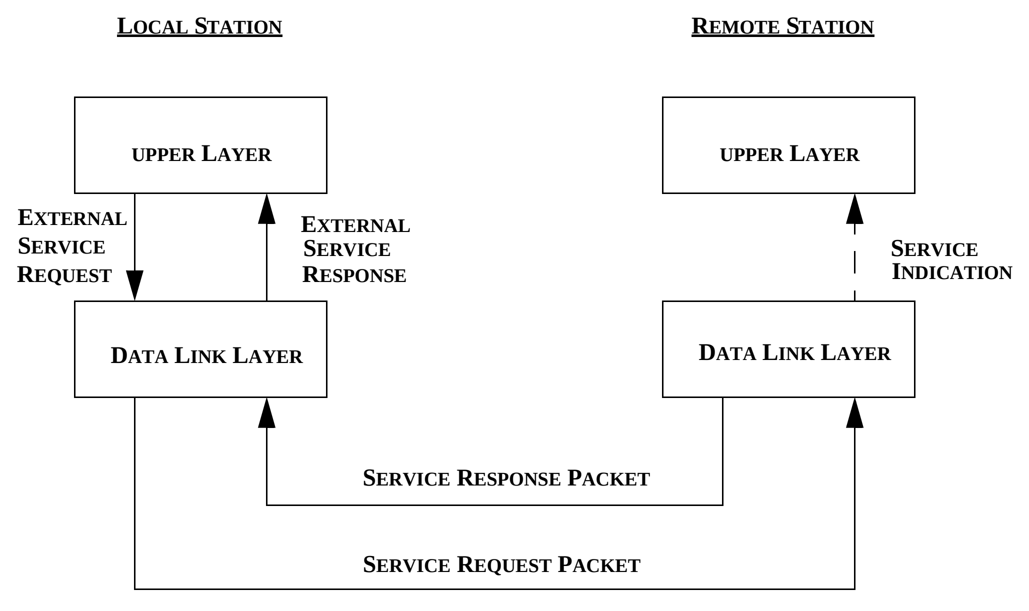


Figure 20: Service Request Response Primitives

The local station is the source station of the External Service. External services such as Initialize operate only on the local station. The other External Service requests require the generation and transmission of a service request packet to the remote station. The remote station is the destination station that receives and responds to the service request packet. The response for the External Services generally does not involve the Upper Layer of the remote station.

The Service Duration Time is the time duration for the Data Link Layer to perform the service as requested by the External Service request. The Data Link Layer returns an External Service response to the Upper Layer upon completion of the service. The completion of the service is when a response packet is successfully received from the remote station, or upon expiration of the Service Duration Time.

Multi-bye parameters contained in a service request should use “big-endian” convention. Additionally, multi-byte counters (in the diagnostic service response) will use the “big-endian” convention.

Fig. 21 (Station Layers) describes the boundary of the Upper Layer and the Data Link Layer in the implementation of the station (master or slave).

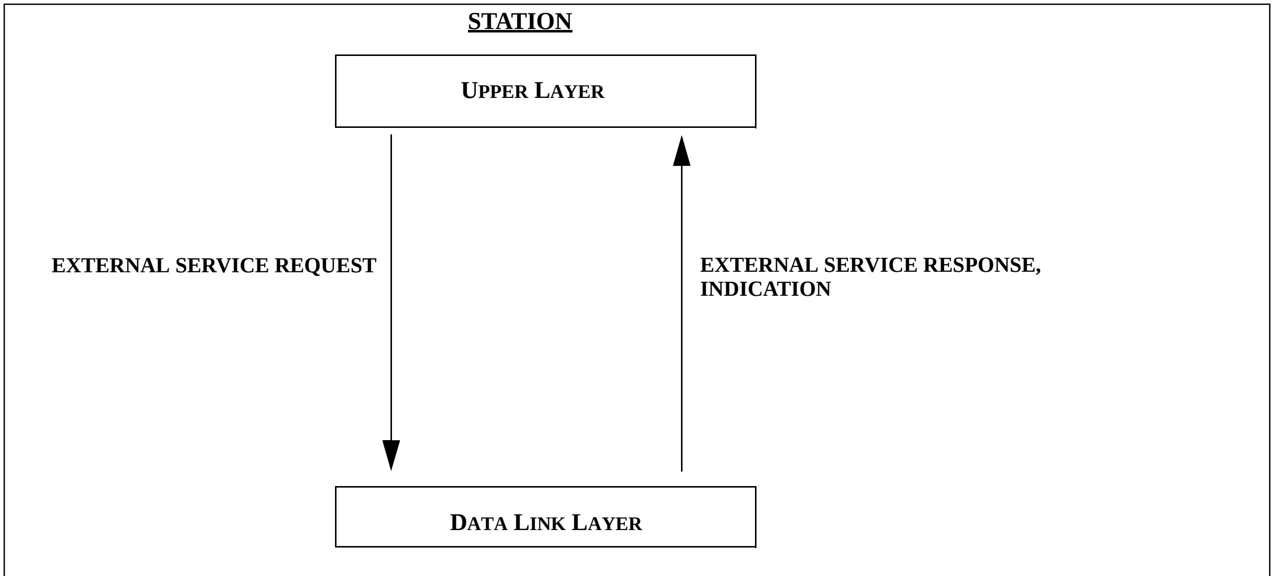


Figure 21: Station Layers

5.2.5 CHANNEL MAPPING TESTING

The channel mapping algorithm is implemented in the hardware, and the testing of the algorithm is done as part of the hardware verification. This document refers to the channel mapping algorithm in certain requirements for context or implementation continuity.

5.2.6 GENERAL REQUIREMENTS

The Station ID is the manufacturer-assigned station ID to uniquely identify each station. A zero valued ID is not valid.

TLMC_MS_004 - STATION ID

The ID is a 6-byte identifier which uniquely identifies each station. A valid ID shall have non-zero value.

TLMM_MS_426 - DEFAULT NATIVE MODE RATE

The master and slave shall transmit and receive native mode packets using a default rate before two way communication is established.

[See “default downlink native mode rate” and “default uplink native mode rate” in the p_config block of the EEPROM. Note that the open downlink packet will communicate the downlink and uplink rate for two way communication

Note that the slave’s default data rates come from EEPROM until a wakeup packet is received.]

Each external service request will be verified per **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)**. If the external service request is N/A, it will not be processed, otherwise service will be processed. In either case, a service response with status will be returned to the Upper Layer except a (waveform) data service request. That is, slave will process the waveform data request in all the slave states but will not send any waveform data service response (successful or not successful) to the upper layer.(Note the waveform service response interrupt is generated, but the response buffer is not filled in.).

TABLE 7: EXTERNAL SERVICES REQUEST VERIFICATION

EXTERNAL SERVICE	STATE					
	DISABLE	STANDBY	ACQUIRE	COMM	CHANNEL RECOVERY	INACTIVE(1) (MASTER ONLY)
Initialize	**Valid**(2)	N/A - Slave(3) **Valid** - master	N/A	N/A	N/A	N/A
Discover (Master only)	N/A	**Valid**	N/A	N/A	N/A	N/A
Polling (Slave only)	N/A	**Valid**	**Valid** (4)	N/A	N/A	N/A
Open (Master only)	N/A	**Valid**	N/A	N/A	N/A	N/A
Data	N/A	N/A	N/A	**Valid**	N/A - master **Valid**-slave(5)	N/A
Emergency (Master only)	N/A	N/A	N/A	**Valid**	N/A	N/A
Diagnostic (Local) (Master only)	N/A	**Valid**	N/A	**Valid**	**Valid**	**Valid**
Diagnostic (Remote) (Master only)	N/A	N/A	N/A	**Valid**	N/A	N/A
Diagnostic (Mapping) (Master only)	N/A	**Valid**	N/A	N/A	N/A	N/A
Medical Event	N/A	**Valid - Master** N/A - Slave(6)	N/A	N/A	N/A	N/A
Raw Mode (master packet attached) (Master only)	N/A	N/A	N/A	**Valid**	N/A	N/A
Network (Sub-type - Configure)	**Valid**(7)	**Valid**	**Valid**	**Valid**	**Valid**	**Valid** - Master N/A - Slave

TABLE 7: EXTERNAL SERVICES REQUEST VERIFICATION

Network (Sub-type - Data)	N/A	**Valid** - Master N/A - slave	N/A - Master **Valid** - Slave	N/A	N/A	N/A
Security - assert EBAN key duration (Slave only)	**Valid**	**Valid**	**Valid**	**Valid**	**Valid**	N/
Security - Authorize Key Delivery and Update Proximity Status (Slave only)	N/A	**Valid**	**Valid**(8)	N/A	N/A	N/A
Security - Request Key via Proximity Switch (Master only)	N/A	**Valid**	N/A	N/A	N/A	N/A
Security - Create Key	N/A	**Valid**	N/A	N/A	N/A	N/A
Security - Assert Key Local and Request Key Local	N/A	**Valid**	N/A - Master **Valid** - Slave(9)	N/A	N/A	N/A
Power inhibit (Slave only)	**Valid**	**Valid**	**Valid**	**Valid**	**Valid**	N/A
Close (Master only)	N/A	**Valid**	N/A	**Valid**	**Valid**	**Valid**
Shutdown	**Valid**	**Valid**	**Valid**	**Valid**	**Valid**	**Valid**

1. Applies to master ONLY - unspecified Channel Recovery is triggered when data, emergency, open, or remote diagnostic master requests are received.
2. **Valid** means that the external service request will be processed at this state.
3. N/A - means that the external service request is not allowed at this state. (Both ‘Valid’ and ‘N/A’ may occur in a single cell, meaning the logic is conditional, dependant on whether the External Service request is coming from an slave or from a master station.)
4. Polling request (sub-type - 0x06 - Poll upon Tel M) will be valid only if slave is uplinking medical event during acquire state. For details, see **TLMM_S_440 - MEDICAL EVENT - POLL UPON TELM REQUEST**. Polling requests (sub-type -0x01 - Poll upon POR, 0x04 - Poll upon Medical event, 0x05 - poll upon patient name and 0x06 - Poll upon Tel M) are invalid if slave is already polling during acquire state.
5. Where in recovery state, if bit “Enable WF in recovery” is set to “1” (via the initialization request), the data request (the sub-type = 0x02 - waveform data) is valid. Slave will save the last two waveform requests (one from service request3 and one from service request4) received during Recovery state and uplink the waveform when the session is recovered. For details, see **TLMM_S_356 - ARQ - RETRANSMIT WAVEFORM PACKET**. Where in Recovery state, if bit “Enable WF in recovery” is cleared (via the initialization request), the data request (the sub-type = 0x02 - waveform data) is invalid. Other data requests (sub-type-0x00 - Memory data, 0x01-unrequested data) are invalid during recovery state.
6. Slave transmits medical event by using the service called Poll upon medical event service.
7. It’s valid only after initialization request has been processed once after Mozart FW is enabled.For example, after standby timer times out, FW goes to Disable state.
8. Valid on Slave in the acquire state when slave is polling. Invalid during Medical event and Network services.
9. Valid on Slave in the acquire state when slave is polling. Invalid during Medical event and Network services.

External service requests will be processed in a serialized manner. Once an External Service is issued to the Data Link Layer its processing will be completed by the Data Link Layer. (Exception: see **5.2.13 Emergency**.)

TLMC_MS_006 - EXTERNAL SERVICE REQUEST PROCESSING

External service requests will be accepted in a serialized manner, with one exception: Emergency can interrupt any service in progress.

TLMC_M_237 - SERVICE DURATION TOLERANCE

All service requests are issued with a service duration. The tolerance for all service durations (measured at the application) is [-1 second, +1 second].

The state machine for External Service requests and timeout events is shown in **Fig. 22 (Master Data Link Layer State Transition Diagram)**.

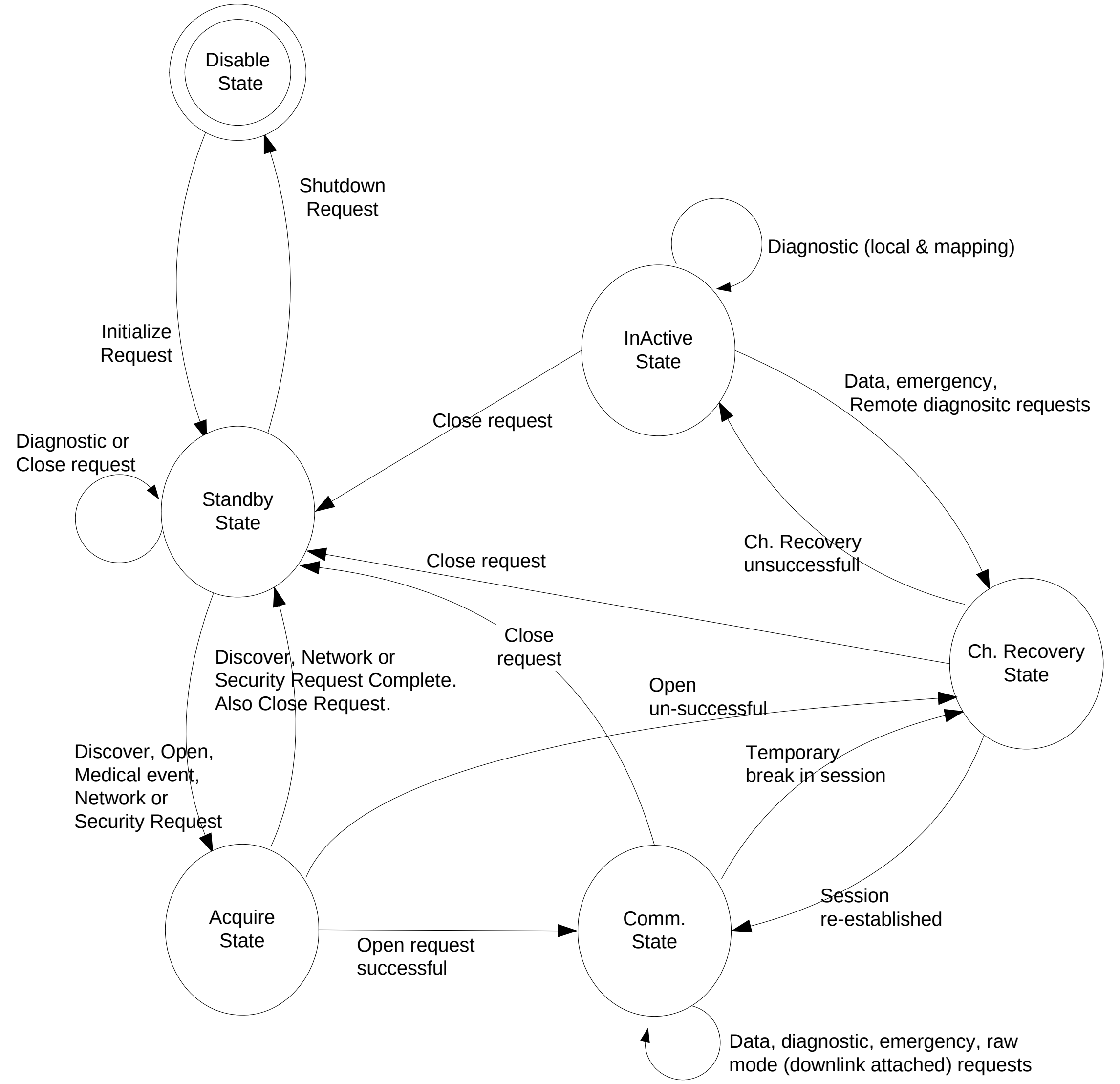


Figure 22: Master Data Link Layer State Transition Diagram

The Data Link Layer is in the Acquire State to perform the Discover or the Open services. For the most part, the Upper Layer has to wait for the completion of these services, and the Data Link Layer transition back to the Standby State before requesting another service, with the exception of the Diagnostic service request.

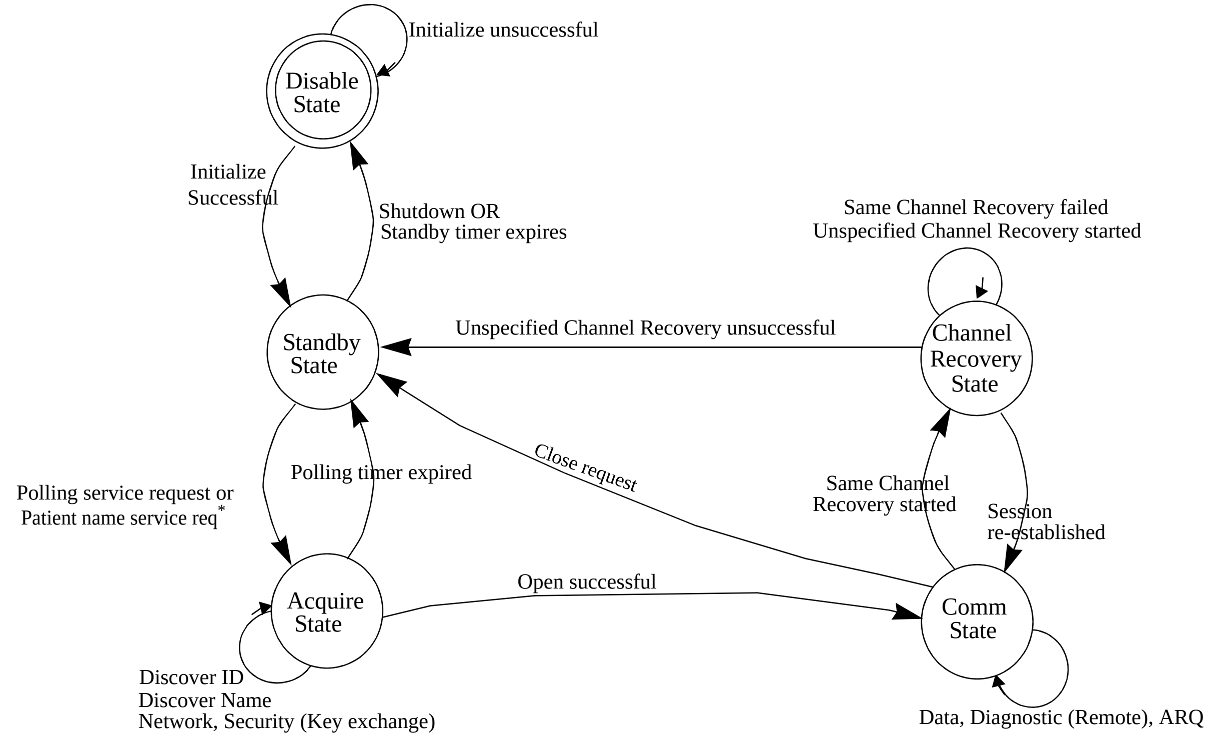
Table 8 defines the external service requests and the events which cause the state transition of the master.

TABLE 8: MASTER STATE TRANSITION TABLE

EXTERNAL SERVICE REQUEST(1) OR EVENT(2)	STATE					
	DISABLE	STANDBY	ACQUIRE	COMM	CHANNEL RECOVERY	INACTIVE
Initialize *Successful*	**STANDBY**	N/C	N/A	N/A	N/A	N/A
Discover Request	N/A	**ACQUIRE**	N/A	N/A	N/A	N/A
Discover - Complete	N/A	N/A	**STANDBY**	N/A	N/A	N/A
Open Request(3)	N/A	**ACQUIRE**	N/A	N/A	N/A	N/A
Open Request(3) - *Successful*	N/A	N/A	**COMM**	N/A	N/A	N/A
Open Request(3) - *Unsuccessful*	N/A	N/A	**RECOVERY(4) OR STANDBY**	N/A	N/A	N/A
Close Request - *Successful*	N/A	N/C(5)	**STANDBY**	**STANDBY**	**STANDBY**	**STANDBY**
Channel Recovery - *Successful*	N/A	N/A	N/A	N/A	**COMM**	N/A
Channel Recovery - *Unsuccessful*	N/A	N/A	N/A	N/A	**INACTIVE**	N/A
Communication failure	N/A	N/A	N/A	**RECOVERY**	N/A	N/A
Data Request	N/A	N/A	N/A	N/C	N/A	**RECOVERY**
Emergency Request	N/A	N/A	N/A	N/C	N/A	**RECOVERY**
Remote Diagnostic Request	N/A	N/A	N/A	N/C	N/A	**RECOVERY**
Medical Event Request	N/A	**ACQUIRE**	N/A	N/A	N/A	N/A
Network request	N/A	**ACQUIRE**	N/A	N/A	N/A	N/A
Network service complete	N/A	N/A	**STANDBY**	N/A	N/A	N/A
Security request (Remote request key)	N/A	**ACQUIRE**	**N/A**	N/A	N/A	N/A
Security - complete (remote key request)	N/A	N/A	**STANDBY**	N/A	N/A	N/A
Shutdown request	**DISABLE**	**DISABLE**	**DISABLE**	**DISABLE**	**DISABLE**	**DISABLE**

1. The external service requests are in **Bold** face. The external service requests are from the Upper Layer of the local station.
2. The events are not in bold face. These events may have resulted from processing of packets received from remote station or from internal Data Link events.
3. Open request includes the open (unsecure) and open (secure). Open (secure) is described in the security section of this document.
4. The master will go to standby if master lacks the appropriate key, or if the open response packet indicates the slave does not have the appropriate key, otherwise master will go to recovery. In the first case user needs to exchange the appropriate key before open can be successful.
5. N/C - no change.

Fig. 23 (Slave Data Link Layer State Transition Diagram) illustrates the state transitions as the result of the External Service requests and timeout events..



* A patient name service request is an implied polling service request. It is sent (by the host) after recovery fails or a close request. This allows another session to be established with minimal latency.

Figure 23: Slave Data Link Layer State Transition Diagram

The slave does not sniff nor listen in native mode during standby state.

Table 9 defines the external service requests and the events which cause the state transition of the slave.

TABLE 9: SLAVE STATE TRANSITION TABLE

EXTERNAL SERVICE REQUEST[(1)] OR EVENT	STATE				
	DISABLE	STANDBY	ACQUIRE	COMM	CHANNEL RECOVERY
Initialize Request Successful	**=> STANDBY**	N/A	N/A	N/A	N/A
Polling Request(2)	N/A	**=> ACQUIRE**	**=> ACQUIRE**(3)	N/A	N/A
Patient Name Request(4)	N/A	**=> ACQUIRE**	N/A	N/A	N/A
Polling timer expired	N/A	N/A	**=> STANDBY**	N/A	N/A
Medical Service Duration timer expired	N/A	N/A	**=> STANDBY**	N/A	N/A
Open packet(5) - successful	N/A	N/A	**=> COMM**	N/A	N/A
Channel Recovery - successful	N/A	N/A	N/A	N/A	**=> COMM**
Channel Recovery - unsuccessful	N/A	N/A	N/A	N/A	**=> STANDBY**
Close packet	N/A	N/A	N/A	**=> STANDBY**	N/A
Standby timer expired	N/A	**=> DISABLE**	N/A	N/A	N/A
Same Channel Recovery started	N/A	N/A	N/A	**=> CHANNEL RECOVERY**	N/A
Shutdown request	**DISABLE**	**DISABLE**	**DISABLE**	**DISABLE**	**DISABLE**

1. The external service requests are in **Bold** face. The external service requests are from the Upper Layer of the local station.
2. This includes the poll upon medical event, poll upon Tel M and Poll upon POR subtypes.
3. See footnote (4) in Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)
4. It's a subtype of polling request.
5. Open request includes open (unsecure) and open (secure).

Note the network request, security request and data request will not cause the slave to change state.

Most of the Telemetry M packets contain an master ID and/or slave ID. A Telemetry M packet received by the master is valid if the packet has a correct message authentication code (MAC) and is targeted to this station. A Telemetry M packet received by the slave is valid if the packet has a correct MAC and is targeted to this station.

The slave shall accept the master packets as defined by **Table 10** to be the valid master packets.

TABLE 10: MASTER PACKET VALIDATION

STATE	CONTROL BYTE FIELDS							
	PKT TYPE	Xfer MODE	SEQ#(1)	FRAGMENT NUMBER	LAST FRAGMENT	M/S bit	SIZE	PACKET (CONTENT) SPECIFIC VALIDATIONS
Acquire	Discover ID	NA	NA	NA	NA	m	NA	Master ID(2)
Acquire	Discover Name	NA	NA	NA	NA	m	NA	Master ID(2), Slave ID(3), Discover ID collision avoidance duration
Acquire/ Channel Recovery(4)	Open(5)	NA	NA	NA	NA	m	NA	Master ID(2), Slave ID(3)
Acquire	Network	NA	0	NA	NA	m	1-21	Network ID
Acquire	Security	NA	NA	NA	NA	m	NA	Master ID(2), Slave ID(3)
Comm	Close	NA	NA	NA	NA	m	NA	Master ID(6), Slave ID(3)
Comm	Diagnostic	0-2	NA	NA	NA	m	NA	Master ID(7)
Comm	ARQ	0-2	NA	NA	NA	m	NA	Master ID(7)
Comm	Mem Data(8)	0-2	0 - 255	0	True if last, False otherwise	m	1-24	Master ID(7)
		0-2		1-15		m	1-30	None

1. Sequence Number: This field only applies to the Memory Data packets.
2. The master ID must be a valid ID.
3. The slave ID has to match the ID of this (slave) station.
4. Open packets may occur in Comm State during Channel Recovery.
5. Open packet includes open (unsecure) and open (secure).
6. The master ID has to match that of the master with which the slave station has a session open.
7. The master ID has to match that of the master with which the slave station has a session open. If not, Channel Recovery is initiated.
8. The sequence number has to fall within the window allowed by ARQ. (For details, see **5.3.1 Packet Retransmission (ARQ)**.)

The master shall accept slave packets as defined by **Table 11** to be the valid slave packets.

TABLE 11: SLAVE PACKET VALIDATION

STATE	CONTROL BYTE FIELDS[1]					Content
	Pkt Type	Fragment #	Seq [2]	M/S bit	Size	Packet (Content) Specific Validations
Acquire	Discover ID	NA	NA	s	NA	Master ID[3], Master ID[4]
Acquire	Discover Name	0	NA	s	1-18[5]	Master ID, Slave ID, Station Name String
		1-4	NA	s	1-30	Station Name String
Acquire/ Channel Recovery[6]	Open[7]	NA	NA	s	NA	Master ID[3], Slave ID[8]
Acquire	Network	NA	0	s	0-21	Network ID
Acquire	Security	N/A	N/A	s	N/A	Master ID[3], Slave ID[8]
Acquire	Medical Event	N/A	N/A	s	N/A	Slave ID[8]
Comm	Close	NA	NA	s	NA	Master ID[3], Slave ID[8]
Comm	Diagnostic	0	0-255[9]	s	NA	Slave ID[8]
		1-15		s	NA	None
Comm	Mem Data	0	0-255[10]	s	1-24	Slave ID[8]
		1-15		s	1-30	None
Comm	WF Data	0	0-255	s	1-24	Slave ID[8]
		1-15		s	1-30	None
Comm	UR[11] Data	0	0-255	s	1-24	Slave ID[8]
		1-15		s	1-30	None

1. NA means not applicable. A field with NA is not used to decide the validity of the packet.
2. Sequence Number. This field applies only to the various slave Data Packets.
3. The master ID must match the ID of this (master) station.
4. The slave ID must be a valid slave ID; see **TLMC_M_011 - VALID SLAVE ID**.
5. The discover name size control byte field should include the proximity status byte. See **Fig. 34 (Discover Name Response Packet)**.
6. Open packets may occur in Comm State during Channel Recovery.
7. Open packet include open (unsecure) and open (secure).
8. The slave ID must be the ID of the (slave) station with whom the master is communicating.
9. The master ignores the diagnostic sequence number.
10. The sequence number has to fall within the window allowed by the ARQ. (For details, see **5.3.1 Packet Retransmission (ARQ)**.)
11. UR = Unrequested.

5.2.6.1 FAULT INDICATION

TLMM_MS_404 - FAULT INDICATION - INDICATION TO UPPER LAYER

The Fault indication will be generated to the Upper Layer when a fault reset occurs. See **Table 12 (FAULT INDICATION (MASTER OR SLAVE))**

TABLE 12: FAULT INDICATION (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x33 - Fault indication

TABLE 12: FAULT INDICATION (MASTER OR SLAVE)

Sub-type	N/A	1	The indication sub-type	0xF0 - Fault 0 (HW register) 0xF1 - Fault 1 (HW register) 0xFF - Firmware faults (see firmware error codes). See **Table 130 (SUMMARY OF FIRMWARE FAULTS)**
Size	N/A	2	The size of the info field	4
Info	Fault Reason	1	Status of the indication	One of the following: • The content of IRQC fault0 register at reset (if sub-type is 0xf0), or • The content of IRQC fault1 register at reset (if sub-type is 0xf1), or • The firmware reset code at reset (if sub-type is 0xff)
	Firmware State	1	State of firmware at time of fault	0x00 - 0xFF
	Firmware Event	1	Event generated at time of fault	0x00 - 0xFF
	IRQC bus master	1	Bus master state at time of fault	0x00 - 0x0F
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

5.2.6.2 SHUTDOWN INDICATION

TLMM_S_263 - SHUTDOWN INDICATION - INDICATION TO UPPER LAYER

When the standby timer, ephemeral BAN key timer, and security (key exchange authorization) timer have all expired, a shutdown indication will be generated to the upper layer, a transition will be made to the disable state. See **Table 13 (SHUTDOWN INDICATION (SLAVE, MASTER))**

[Then the slave will return to synchronous receptor wakeup operation. If the wakeup disable bit is set to 1 in the initialization request firmware does not set the wakeup enable bit bp_wakeup_en bit in REG_BP_WU_CTRL0 register. See requirement **TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP**.

Finally, the slave will wait for “Timer between Shutdown Indication and Turn off 2V” duration and then turn off the 2 volt supply. See **Table 129 (PARAMETERS FROM EEPROM)** for “Timer between Shutdown Indication and Turn off 2V”.]

TLMM_MS_413 - SHUTDOWN INDICATION - DISABLE TIMER EXPIRES

Upon the expiration of the disable ON timer shutdown indication will be generated to the upper layer and 2 volt supply will be turned off.

[A shutdown indication can occur when Mozart is enabled via reset, host dummy write or wakeup and the host does not send a subsequent initialize request.

Note that the 2 volt supply is not turned off until the “Timer between Shutdown Indication and Turn off 2V” (see **Table 129 (PARAMETERS FROM EEPROM)** expires after the expiration of the disable ON timer. In addition, during the “ Timer between Shutdown Indication and Turn off 2V” that follows the disable ON timer expiration, the mozart will not respond to wakeups.]

TABLE 13: SHUTDOWN INDICATION (SLAVE, MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x26 - Shutdown indication
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	The size of the info field	1
Info	Shutdown indication status	1	Status of the indication	0x01 - successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

5.2.7 INITIALIZE

This service is used by the Upper Layer to activate the telemetry system on a local station. Upper layer shall pass configuration data. When this service is invoked various physical and Data Link Layer parameters are initialized to a known state.

Fig. 24 (Initialize Sequence Diagram) illustrates the Initialize service operations.

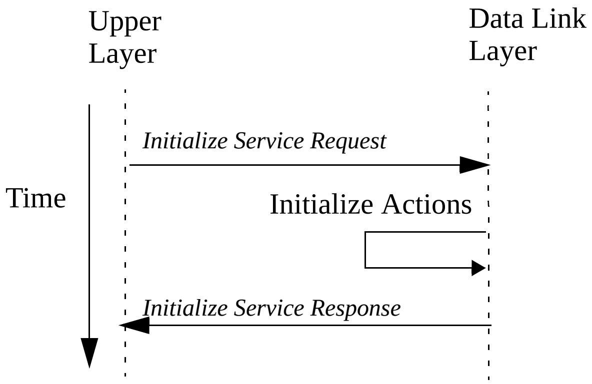


Figure 24: Initialize Sequence Diagram

TLMM_MS_428 - INITIALIZE - UPPER LAYER TO DATA LINK LAYER PARAMETERS

When invoking the Initialize service request, the Upper Layer will provide information as specified in **Table 14 (INITIALIZE SERVICE REQUEST (MASTER OR SLAVE))**.

TABLE 14: INITIALIZE SERVICE REQUEST (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x81 - Master Initialize service request 0x01 - Slave Initialize service request
Sub-type	N/A	1	The service request sub-type	N/A
Size	N/A	2	The size of the info field	Sizeof(master config file) or sizeof(slave config file)
Info	Configuration file	N	See **Table 123 (Master parameters from host)** or **Table 124 (SLAVE PARAMETERS FROM HOST)**	See **Table 123 (Master parameters from host)** or **Table 124 (SLAVE PARAMETERS FROM HOST)**
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

TLMM_MS_265 - INITIALIZE ADDRESS WITH VALUES

Upon receiving the initialization data, firmware will copy each value to the specified address up to, but not including, an address value of 0. Format for master see **Table 123 (Master parameters from host)** and for slave see **Table 124 (SLAVE PARAMETERS FROM HOST)**.

[Initialization data are address-data pairs contained in the parameters received from the host. This could be used to write any value to the specified address.]

5.2.7.1 MASTER

TLMM_M_266 - MASTER INITIALIZE SERVICE REQUEST FROM UPPER LAYER

When invoking an Initialize service request, the Upper Layer-in the master will provide information contained in the configuration file (see **Table 123 (Master parameters from host)**).

TLMM_M_267 - MASTER INITIALIZE SERVICE RESPONSE TO UPPER LAYER

An Initialize service response will be generated to the Upper Layer after the initialize service request is processed.

[The content of the response to the Upper Layer is specified in **Table 15 (INITIALIZE SERVICE RESPONSE (MASTER))**]

.

TABLE 15: INITIALIZE SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service response	0x81 - Master Initialize service response
Sub-type	N/A	1	The service response sub-type	N/A
Size	N/A	2	The size of the info field	1

TABLE 15: INITIALIZE SERVICE RESPONSE (MASTER)

Info	Initialize service status	1	Status of the service request	0x01 - Request successful 0x02 - Request invalid in current state (state is Comm) 0x03 - Request invalid in current state (state is Recovery; slave scanning)(1) 0x04 - Request invalid in current state (state is Recovery; slave not scanning)(1) 0x05 - Request invalid in current state (state is Inactive 0x07 - Request invalid in current state (listening for Medical Event) 0x0A - Parameter out of range(2) 0x12 - Invalid size (info size >506)(2) 0x0F - Request not serial (another service is in progress) 0x10 - Invalid CRC 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(3)
CRC	N/A	2	CRC of all above fields.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration). Note that a service duration timeout will result in a Data Link response status of 3 or 4 (invalid state, state is Recovery). A Data Link status of 9 (service duration timeout) would not provide any benefit.
2. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
3. This status can occur if the master host attempts to send a initialize service request using service request buffer 1 (instead of buffer 0).

5.2.7.2 SLAVE

TLMC_S_008 - SLAVE INITIALIZE SERVICE REQUEST FROM UPPER LAYER

When invoking an initialize service request, the Upper Layer in the slave will provide information as specified in **Table 124 (SLAVE PARAMETERS FROM HOST)**

TLMC_S_009 - INITIALIZE SERVICE RESPONSE TO UPPER LAYER.

An initialize service response will be generated to the Upper layer after the initialize service request is processed, and a standby timer is started.

[The content of the response to the Upper Layer is specified in **Table 16 (INITIALIZE SERVICE RESPONSE (SLAVE))**.]

TABLE 16: INITIALIZE SERVICE RESPONSE (SLAVE)

FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	1	The type of service response	0x01 - Slave Initialize service response
Sub-type	1	The service response sub-type	N/A
Size	2	The size of the info field	1
Info	1	Status of the service request	0x01 - Initialization Successful 0x12 - Invalid size 0x13 - Invalid request type (This code can occur if the service request is sent using an interrupt other than service request0 (there is a buffer/ service request mismatch). 0x15 - Request in invalid state(1) 0x24 - Request invalid during EEPROM read 0x10 - Invalid CRC
CRC	2	CRC of all above fields.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. See **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)** for the invalid states.

5.2.8 POLLING

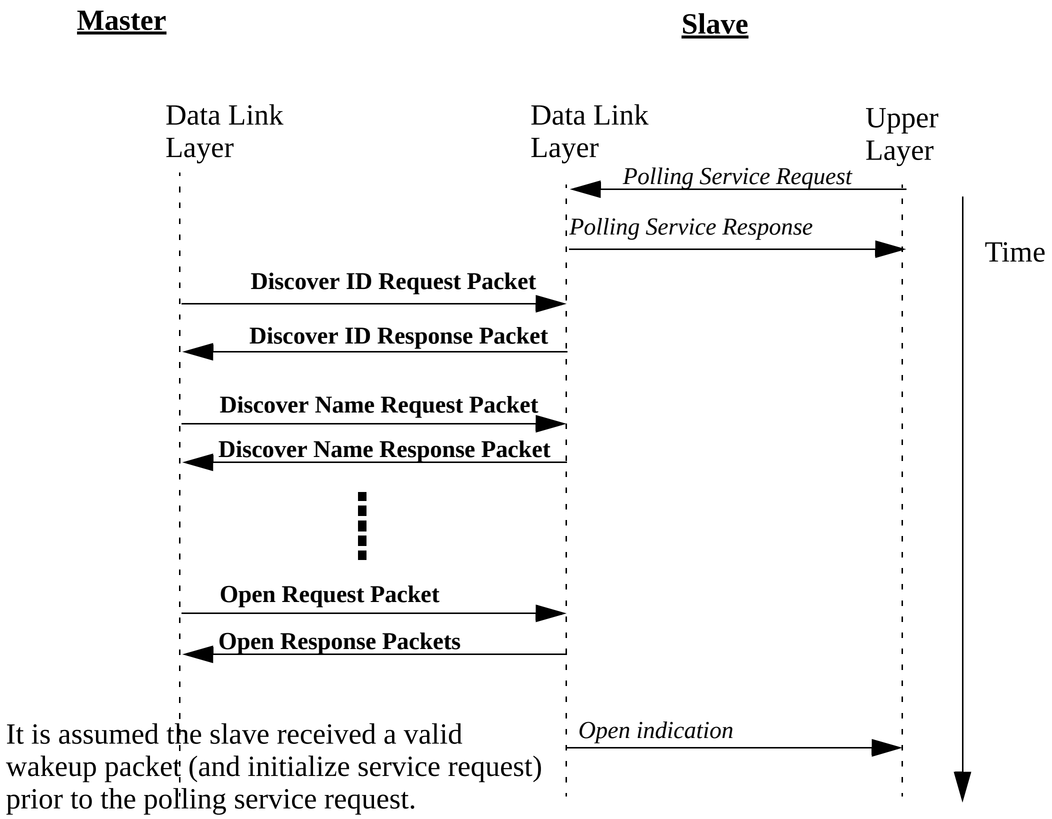
The purpose of the polling service is to allow a slave to “listen/respond” to a master station for a total polling time duration. The Total Polling Duration Time can be specified two ways:

1. When Telemetry M is enabled as part of initialize service Total Polling Duration Time is provided to Telemetry M.
2. A master provides the Total Polling Duration Time as part of open master packet.

A station uses this functionality to listen for other stations using Discover, Open, Network or Security services. The following bullets briefly summarize this functionality:

- The slave receives a valid wakeup packet (during the disable state). It sends a wakeup indication to the upper layer. The upper layer sends an initialize (slave) service request.
- The Upper Layer then sends a Polling request to the Data Link Layer.
- The Data Link Layer listens for Discover, Open, Network or Security packet.
- If a valid Discover ID or a Discover Name request is received, the Data Link Layer records the associated information and transmits the appropriate Discover ID or Discover Name response.
- If a valid Open, Network or Security request is received, the Data Link Layer transmits the appropriate response.

Fig. 25 (Polling Sequence Diagram) illustrates the operations.



It is assumed the slave received a valid wakeup packet (and initialize service request) prior to the polling service request.

Figure 25: Polling Sequence Diagram

The polling service can be configured in one of two ways:

1. scanning on
2. scanning off (alternating native/ async receptor wakeup)

In the scanning on case, the data link layer listens for a discover, open, network or security packet. If these valid packets are not received (channel listen timeout), a new channel is selected for (native mode) listening. This process continues until the polling duration timer expires or a valid open packet is received. Fig. 26 (Polling (scanning on) example) illustrates the Discover and Open services on a master and polling (scanning on) on a slave. The stations successfully exchange Discover ID/Discover Name request/response packets. At some later time the master switches from Discover requests to Open requests. Eventually, the slave lands on the channel selected by the master. The slave receives an Open request from this master. The slave verifies the Open request, and transmits an Open response to establish communication with the other station.

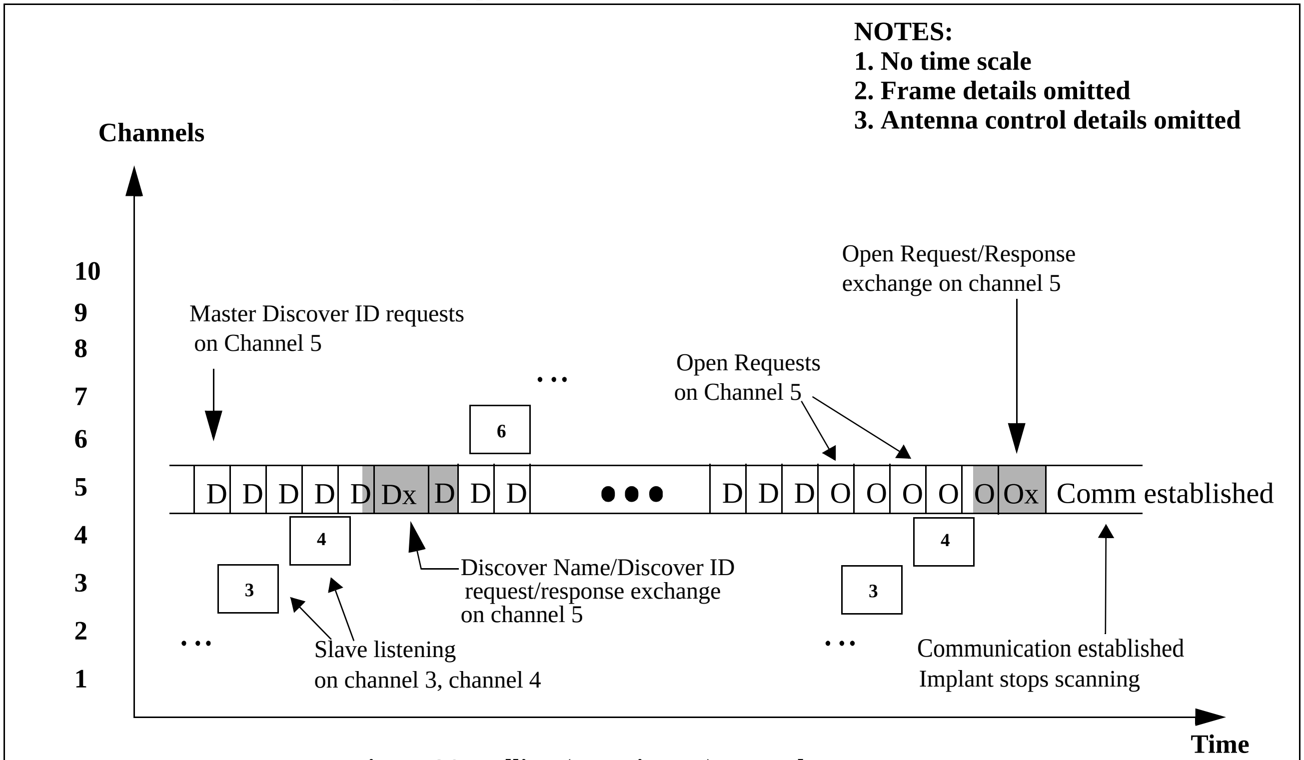


Figure 26: Polling (scanning on) example

When polling is invoked (scanning off case), the Data Link Layer dwells on the selected channel for a period of time, listening for a Discover, Open, Network or Security packet. If these valid packet types are not received (native mode timeout), the Data Link Layer begins listening for a wakeup packet. If a valid wakeup packet is received, the slave begins listening for a valid native mode packet again. This process continues until the polling duration expires, or a valid open packet is received.

A polling station only responds to Discover, Open, Network or Security packet. All other packet types are ignored during polling. When the polling station detects a valid Open request, it stops polling its channels and remains in native mode while two way communication continues.

Total duration of the polling mode is configurable.

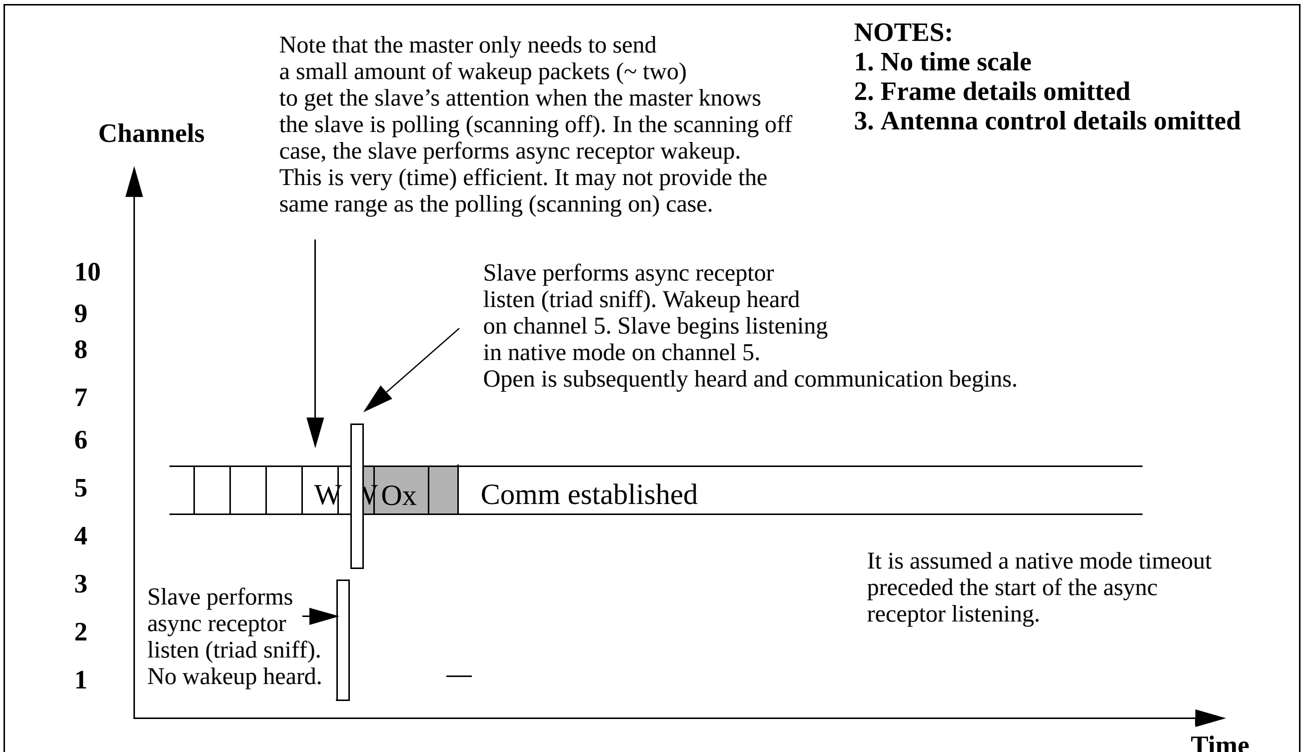


Figure 27: Polling (scanning off) example

5.2.8.1 REQUIREMENTS

TLMM_S_241 - POLLING ENABLING CRITERIA

If the polling feature is enabled then it will be activated:

1. Pre-session – when a polling service request is received from the upper layer (i.e., valid wakeup packet received),
2. Post-session – when channel recovery is terminated unsuccessfully and the patient name service request is received from the upper layer.
3. Post-session – when a close request packet is received and the patient name service request is received from the upper layer.

The slave will use the polling duration in the open packet

[If session is not opened then default polling time value from the initialization data provided by the host will be used.]

TLMM_S_035 - POLLING - SERVICE REQUEST FROM UPPER LAYER

When invoking a Polling service request before a session has been established, use the Total Initial Polling Duration time provided by the host as part of initialization data when initialization service request is invoked.

[The polling timer can also be programmed as via Open master packet (see **5.2.10 OPEN SESSION**), this timer value specifies the duration of the polling period.]

TABLE 17: POLLING SERVICE REQUEST (SLAVE)

FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	1	The type of service request	0x0A -Polling service
Sub-type	1	The service request sub-type	0x01 - Poll upon POR 0x02 - Poll upon Tel B ID (C only) 0x03 - Poll upon schedule time (C only)(1) 0x06 - Poll upon TelM
Size	2	The size of the info field	Size of **Table 126 (SLAVE POLLING REQUEST)**
Info	Size of **Table 126 (SLAVE POLLING REQUEST)**	See **Table 126 (SLAVE POLLING REQUEST)**	See **Table 126 (SLAVE POLLING REQUEST)**
CRC	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

TLMM_S_256 - POLLING - SERVICE RESPONSE TO UPPER LAYER

When poll request is received, the Data Link Layer will validate the Poll service request.

[The content of the response to the Upper Layer is specified in **Table 18 (POLLING SERVICE RESPONSE (SLAVE))**]

TABLE 18: POLLING SERVICE RESPONSE (SLAVE)

FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	1	The type of service request	0x0A -Polling service

TABLE 18: POLLING SERVICE RESPONSE (SLAVE)

Sub-type	1	The service response sub-type	0x01 - Poll upon POR 0x02 - Poll upon Tel B ID (C only) 0x03 - Poll upon schedule time (C only) 0x06 - Poll upon Tel M
Size	2	The size of the info field	1
Info	1	Status of the service request	0x01 - Successful 0x10 - Invalid CRC 0x12 - Invalid size(1) 0x13 - Invalid request type (This code can occur if the service request is sent using an interrupt other than service request0 (there is a buffer/ service request mismatch). 0x14 - Invalid Poll subtype 0x15 - Request in invalid state(2) 0x24 - Request invalid during EEPROM read 0x2A - Request invalid during EEPROM write 0x2B - Request invalid as host requires secure but local wake up does not include security
CRC	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. It’s invalid size if the size is zero or greater than what’s indicated in **Table 126 (SLAVE POLLING REQUEST)**.
2. See **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)** for the invalid states.

The polling mode (scanning on or off) is selected using the parameter “slave scans during polling” in **TABLE 123: Master parameters from host**. This is communicated to the slave via a wakeup packet.

If slave is authorized to open an “unsecure” by the host, and a valid local wakeup “secure or unsecure user data/network packet” request is received, when polling request (upon Tel M) is received, then slave will listen in native mode for the network data request packet. See **TLMM_S_434 - NETWORK DATA - PROCESSING POLLING REQUEST** for details.

TLMM_S_432 - POLLING - HOST REQUIRES SECURE BUT LOCAL WAKEUP PACKET DOES NOT INCLUDE SECURITY

If slave’s “authorized to talk unsecurely” parameter is false as provided by the host, and a valid local wakeup packet with “security user data/network packet” set false is received, when polling request (upon Tel M) is received, slave will send an invalid polling response to upper layer.

[Host authorizes Tel M to talk secure/unsecure as part of initialization.]

TLMM_S_240 - POLLING INITIATION

Once the polling feature (scanning off) is activated Pre-session the slave will start the polling timer, start the native mode timer and start listening for a native mode packet. The native mode listen channel will come from the recently received wakeup packet.

Once the polling feature (scanning off) is activated Post-session the slave will start the polling timer and start listening for wakeup packets.

When the polling feature (scanning on) is activated (pre or post session) the slave will start the polling timer, start the channel listening timer and start listening for a native mode packet on a channel.

[For the polling (scanning off) case, if no wakeup packet has been received (i.e. Poll upon POR), the listen channel will be chosen randomly.

For the polling (scanning on, pre session) case, if Unicast wakeup packet has been received, the first listen channel will be the channel number indicated in the Unicast wakeup packet; if Global wakeup packet has been received, the first listen channel will be a randomly selected channel number.

For the polling (scanning on, post session) case, the first listen channel will be the channel number used in the last session.

Note that the slave’s polling mode (scanning on or scanning off) comes from EEPROM until a wakeup packet is received.]

While polling, the slave will process valid Discover ID, Discover Name, Network, Security and Open master (request) packets. Other master packets will be ignored.

TLMC_S_252 - POLLING - RESTART NATIVE MODE TIMER OR CHANNEL LISTEN TIMER

The slave will do the following upon the receipt of a valid discover ID, or discover name packet:

1. Restart the native mode timer for the polling (scanning off) case. Restart the channel listen timer for the polling (scanning on) case.
2. Restart the polling timer using the “last polling” value for the polling. “Last polling” is the latest polling duration via the initialize service or polling duration via downlink open packet.

TLMC_S_037 - POLLING - NATIVE MODE TIMEOUT/ CHANNEL LISTEN TIMEOUT

The slave will do the following upon a native mode timeout/ channel listening timeout for the scanning off/ scanning on cases respectively:

1. Begin listening for wakeup for the scanning off case. Utilize continual asynchronous receptor wakeup listens to do this. It will continue to listen for wakeup until a valid wakeup packet is received, or the polling timer expires.
2. Select a new channel for (native mode) listening for the scanning on case. Note that this is done using two phases:
   a. Phase 1 - random channel listening - the slave will randomly select a channel and will restart the channel listening time and listen for a valid packet from the master. If no valid open packet has been received and the service duration timer is still active, the slave will switch to the phase 2 approach.
   b. Phase 2 - sequential channel listening - the slave will sequentially select the next channel and will restart the channel listening time and listen for a valid packet from the master. When every channel has been selected once the slave will switch to the Phase 1 approach.

[Note that the slave will scan on the bands specified in **Table 124 (SLAVE PARAMETERS FROM HOST)**. See the parameter called “telemetry scheme and waveform setup”.]

TLMM_S_268 - POLLING - VALID WAKEUP PACKET

The slave will start the native mode timer and begin listening for a valid native mode packet if a valid wakeup packet is received during polling (scanning off).

The Discover ID and Discover Name response is described in the Discover section. The Data Link Layer maintains a list of Station IDs received from Discover Name request packets. It is used to decide the appropriate response of the Discover ID request. This list shall be discarded upon completing the polling duration.

TLMC_S_038 - POLLING - TERMINATION

The polling will terminate when Polling duration timer expires, or when a valid Open request is received. The slave shall discard all the master IDs stored during the polling operation. The slave will maintain the ID of the master with which communication is established.

TLMC_S_039 - POLLING INDICATION TO UPPER LAYER

A polling timeout event indication to the Upper Layer will be generated when the polling timer expires. See **TABLE 19: POLLING INDICATION (SLAVE)**.

TABLE 19: POLLING INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x21 - Polling indication
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	The size of the info field	1
Info	Polling indication status	1	Status of the indication	0x03 - timeout

TABLE 19: POLLING INDICATION (SLAVE)

CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

5.2.9 DISCOVER

This service is invoked by the Upper Layer in the master to identify multiple slave stations within range. Slave stations identified during the Discover Service Duration Time will be passed on to the Upper Layer.

The Discover service is divided into two types:

- **Discover Full -** is used to identify ALL the slave stations whose ID matches the slave ID specified in the service request. The Discover Full service obtains the device ID and the name for each device identified. The six bytes of slave ID may contain wildcard.
- **Discover Stop -** is used to stop previously invoked discover service.

The slave must be listening in order to receive the Discover request packet(s). To facilitate the Open service request, the master may request the slave to extend its scanning duration.

The following bullets summarize the operations of this service.

- Upper Layer generates the Discover request to the Data Link Layer.
- The Data Link Layer uses the best available channel for communication.
- The Data Link Layer transmits wakeup packet(s) to wakeup the slave(s).
- The Data Link Layer periodically transmits Discover ID request packets alternating between the two antennas, and listens for Discover ID responses from one or more slaves.
- The Data Link Layer requests the Station Name String from the slaves responding to the Discover ID request.
- The Data Link Layer records information received from the slave(s) and presents the information to the Upper Layer.

Fig. 28 (Discover (Full) Sequence Diagram) illustrates the Discover service operations.

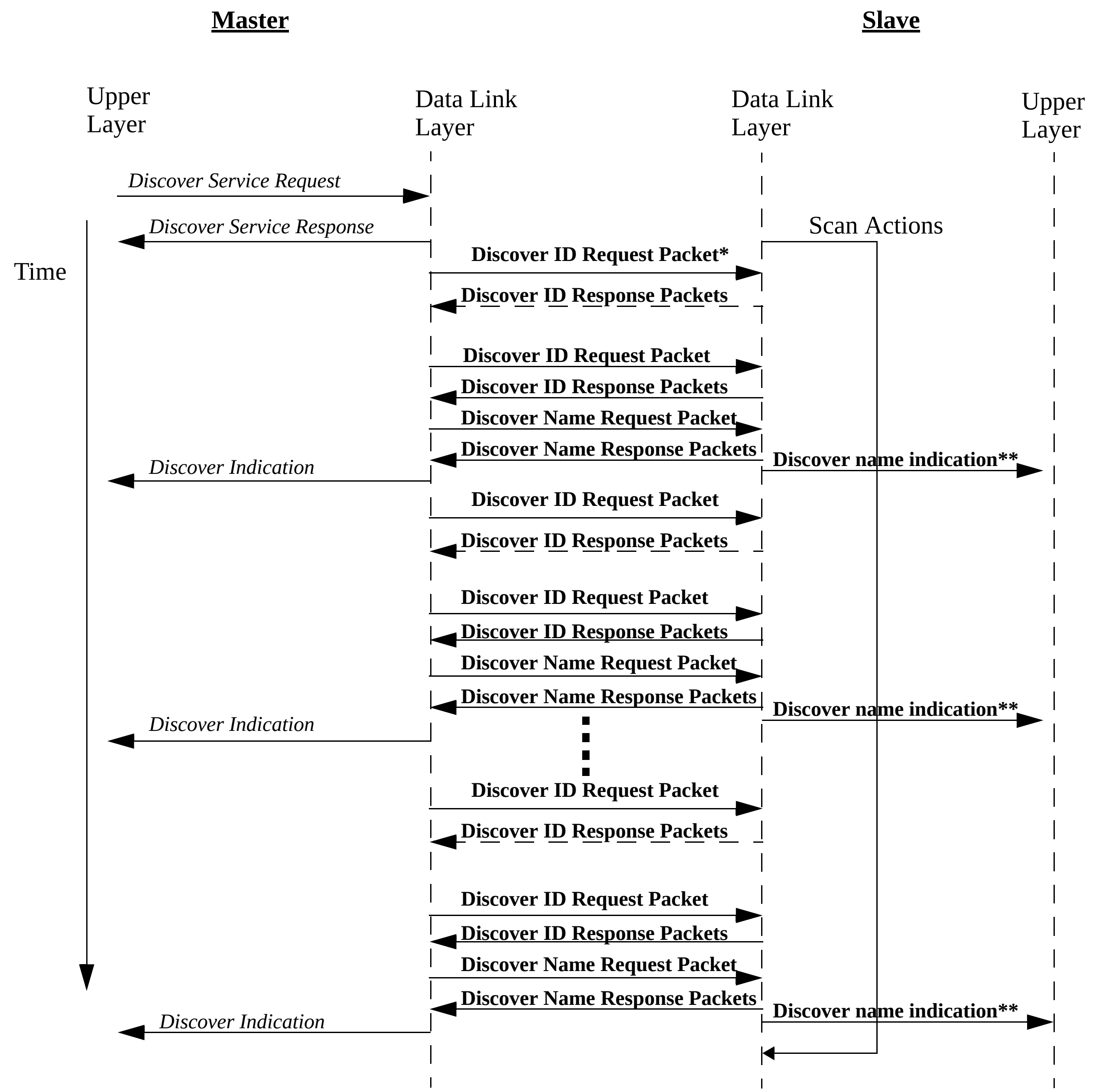


Figure 28: Discover (Full) Sequence Diagram

*Channel mapping and wakeup are not shown.

** Implicit security indication

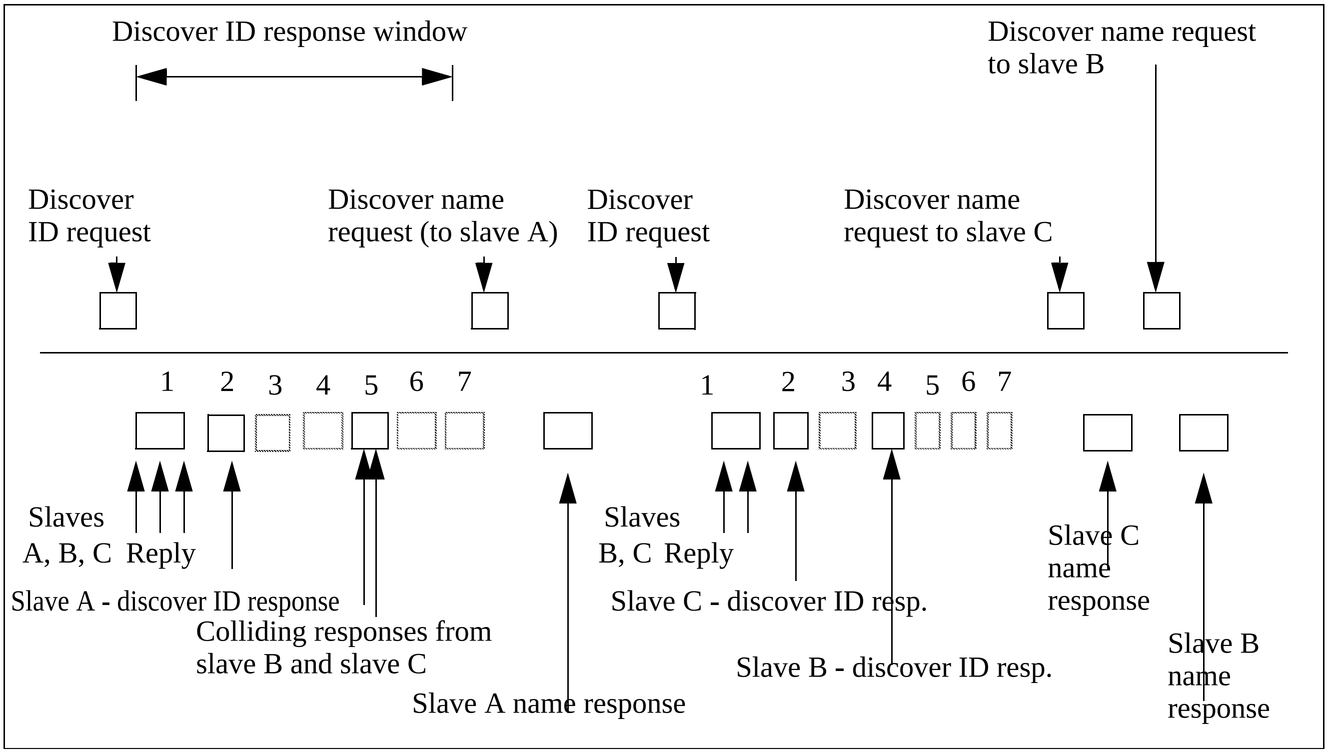


Figure 29: Discover Full example (slave doesn't scan during polling)

In telemetry M, the wakeup packet instructs the slave to use a particular channel for native mode when the slave does not scan during polling. If multiple slaves wake up, they may all hear and respond to the first discover ID request. This may cause collisions. In telemetry M, the discover full response can be configured to occur with a random delay (to reduce collisions). The slave responds randomly in "slot" 1, 2, 3, 4, or 5 (see **Fig. 29 (Discover Full example (slave doesn't scan during polling))**). The master listens for the discover ID response window after each discover ID request.

When the slave scans during polling, the discover ID response window is one uplink frame. Note that the slave begins scanning on a random channel. This is the collision avoidance mechanism for that configuration.

5.2.9.1 REQUIREMENTS

TLMC_M_010 - DISCOVER SERVICE REQUEST FROM UPPER LAYER

When invoking a Discover service request, the Upper Layer will provide information as specified in **Table 20 (DISCOVER SERVICE REQUEST (MASTER))**.

TABLE 20: DISCOVER SERVICE REQUEST (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x82 - Discover service request
Sub-type	N/A	1	The service request sub-type	0 - Discover Full, 1- Discover Stop
Size	N/A	2	The size of the info field	6
Info[1]	Discover Slave ID	6	Slave ID consists of six bytes: Model ID - two bytes, SubModel ID - one byte, Serial # - three bytes.	Six bytes of slave ID: For Discover Full. The Discover slave id may contain wild cards.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

[1] The discover service has two additional parameters which control its behavior: discover service duration and discover ID collision avoidance duration. See **Table 123 (Master parameters from host)**.

The Discover Slave ID may contain a wild card specification in its Model, Sub-Model and/or Serial Number. However, the Slave ID with wild card specification must follow the hierarchy indicated in **Table 21**

TLMC_M_011 - VALID SLAVE ID

The valid slave IDs with wild card specification are defined in **Table 21 (VALID SLAVE ID WITH WILD CARD)**.

TABLE 21: VALID SLAVE ID WITH WILD CARD

MODEL NUMBER	SUB-MODEL NUMBER	SERIAL NUMBER
Specified	**Wild Card (0x00)**	**Wild Card (0x00)**
Specified	Specified	**Wild Card (0x00)**
Wild Card (0x00)	**Wild Card (0x00)**	**Wild Card (0x00)**

Any Slave ID with wild card specification not in **Table 21** is considered an invalid target slave ID.

5.2.9.2 DISCOVER FULL

This section describes the requirements for the Discover service type Full. The Discover Full service is used to identify all the devices within telemetry range. It will also try to obtain the Device Name for each device identified.

TLMC_M_024 - DISCOVER FULL - RESPONSE TO UPPER LAYER

A Discover service response to the Upper Layer will be generated when a Discover Full service request is received.

[The content of the response to the Upper Layer is specified in **Table 22 (DISCOVER SERVICE RESPONSE (MASTER))**.]

TABLE 22: DISCOVER SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x82 - Discover service response
Sub-type	N/A	1	The service response sub-type	0 - Discover Full 1 - Discover Stop
Size	N/A	2	The size of the info field	1 byte
Info	Discover service status	1	Status of the service request	0x01 - Request initiated 0x02 - Request invalid in current state (state is Comm) 0x03 - Request invalid in current state (state is Recovery; slave scanning)[1] 0x04 - Request invalid in current state (state is Recovery; slave not scanning)[1] 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x0A - Parameter value error[2] 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)[2] 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch[3]
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration).
2. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
3. This status can occur if the master host attempts to send a discover service request using service request buffer 1 (instead of buffer 0).

TLMC_MS_012 - DISCOVER FULL - DISCOVER ID PACKET FORMAT

The Discover ID request packet has the layout indicated in **Fig. 30 (Discover ID Request Packet)**.

Discover ID Control (5.25 bytes)	**Source ID (master ID) (6 bytes)**	**(zero fill) (24 bytes)**	**MAC[a] (3 bytes)**	**Reed Solomon (9 bytes)**

a. Message authentication code

Figure 30: Discover ID Request Packet

TLMC_M_013 - DISCOVER FULL - DISCOVER ID REQUEST PACKET

When Discover service request is invoked by the Upper Layer, start a service duration timer and transmit a Discover ID request packet.

[A service duration timer will be started; select the least interfered channel (takes approximately 275 +/ 50 msec(1)); start the Listen Before Talk (LBT) timer; perform instigator wireless wakeup, send a “first request packet sent” indication to the Upper Layer (see **Table 23**); start the native mode timer (if the slave does not scan during polling), transmit a discover ID request packet.

Note the 'first request packet sent' indication is configurable. See the parameter called 'transmit notification enabled' in **Table 123 (Master parameters from host)**

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TABLE 23: FIRST REQUEST PACKET SENT INDICATION (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xAA -First request packet sent
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	The size of the info field	1
Info	indication status	1	N/A	0x01 - successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

---

(1) CCA of the MICS band takes approximately 275 msec. CCA of the MEDS band takes approximately 550 msec. CCA of both bands takes approximately 825 msec. Note that if HDR mode is enabled, assessment of each channel, antenna pair is lengthened by 5 msec (to allow the HDR synthesizer to stabilize). This means the CCA of the MICS band is lengthened from 275 msec to 375 msec.

TLMM_M_269 - DISCOVER FULL - DISCOVER ID RESPONSE WINDOW

The master shall continue listening for more slave responses upon hearing any response (valid or not) provided the device ID response window is not complete.

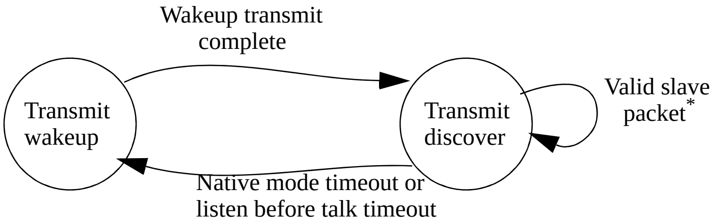
TLMM_M_270 - DISCOVER FULL - RESTART NATIVE MODE TIMER

The master will restart the native mode timer whenever it receives a valid discover ID (or discover name) packet from the slave (if the slave does not scan during polling).

[Note that a channel is selected and instigator wakeup is performed when the native mode timer expires. Once instigator wakeup is complete, discover ID is transmitted again. See

Fig. 31 (Simplified master discover full state machine)

Note that the native mode timer is not used when the slave scans during polling.]



The master transitions between native mode discover and wakeup using the native mode timer if the slave does not scan during polling. The transition is done using the LBT timer otherwise.

Note that discover full will alternate between instigator wakeup and discover transmissions until the service duration timer expires (or stop discover is received).

Figure 31: Simplified master discover full state machine

TLMC_M_014 - DISCOVER FULL - RETRY DISCOVER ID REQUEST

If no valid Discover ID slave response packet is received within the discover ID response window, the master shall transmit another Discover ID request packet. The Discover ID retry will continue until:

1. A valid Discover ID response packet is received, or
2. The service duration timer expires, or
3. The native mode timer expires, or
4. The listen before talk timer expires.

[In order to retransmit the Discover ID request due to no slave response from a previous discover ID request data link layer will switch to a different antenna and will retransmit the discover ID request.]

Discover ID slave response packets with mismatched master ID will be ignored.

TLMC_S_015 - DISCOVER FULL - DISCOVER ID RESPONSE

Upon receiving a valid Discover ID request packet the slave will perform as follows:

1. If the master packet contains an master ID which the slave received previously from a valid Discover Name request packet, then slave response will not be generated.
2. If the master packet contains a master ID which the slave has not received previously from a valid Discover Name request packet, the slave will transmit a Discover ID response packet in the first slave timeslot. It will transmit a second Discover ID response packet in slave timeslot 2, 3,..., or 7. This timeslot will be randomly chosen (to reduce collisions). Note that the second response is not used if the slave scans during polling.

The slave will restart the native mode timer to allow the receipt of another master packet if the slave does not scan during polling.

TLMC_M_016 - DISCOVER FULL - LBT OR NATIVE MODE TIMEOUT

While the service duration timer is active and LBT timer expires or the native mode timer expires, the master will reselect the least interfered channel, toggle the antenna, perform instigator wireless wakeup, and transmit a discover ID packet.

[Note that the master will perform clear channel assessment before and after transmitting wakeup (if the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMC_MS_017 - DISCOVER FULL - DISCOVER ID RESPONSE PACKET FORMAT

The Discover ID response packet has the layout indicated in **Fig. 32 (Discover ID Response Packet)**.

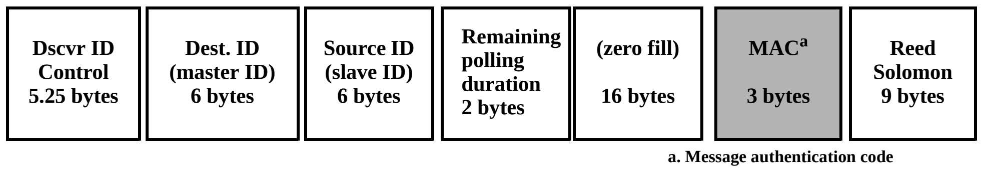


Figure 32: Discover ID Response Packet

The Discover Name request is to request the Station Name String from the slave. The Discover Name request is also used to acknowledge the reception of the Discover ID response from the slave.

TLMC_MS_018 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET FORMAT

The Discover Name request packet has the following layout indicated in **Fig. 33 (Discover Name Request Packet)**.

Discv Name Control 5.25 bytes	**Source ID (master ID) 6 bytes**	**Destination ID (slave ID) 6 bytes**	**Collision Avoidance Duration 1 byte**	**(zero fill) 17 bytes**	**MAC[a] 3 bytes**	**Reed Solomon 9 bytes**

[a] Message authentication code

Figure 33: Discover Name Request Packet

TLMC_M_019 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET

After the receipt of a valid Discover ID response, the master will transmit a Discover Name request packet.

[The master will keep track of all the unique slave IDs, the discover name request will be transmitted on the same antenna as was used for Discover ID.

If multiple slave respond during the discover ID response window, the master will sequentially transmit a name request (and wait for a name response) to each slave.]

The Discover Name request is only issued by the master during the Discover service processing. If a valid Discover Name response is not received by the master, the master shall request the Station Name String again by issuing another Discover Name packet.

TLMC_M_020 - DISCOVER FULL - RETRY DISCOVER NAME REQUEST

If no valid slave Discover Name response is received, the master will retry the Discover Name request packet until the discover name retry timer expires. Note that this timer was started when the initial discover name downlink was sent. Once the timer expires, the master will continue the Discover service by transmitting a new Discover ID request packet.

[The retry attempt of the discover name request will use the same antenna as was used for the first discover name request. If no response is received from the retry attempts then discover ID will be transmitted using a different antenna.]

The new Discover ID request packet will be transmitted via the Asynchronous Mode operation, but from a different antenna.

The Discover Name request acknowledges the Discover ID response sent by the slave earlier. The slave shall remember this master such that it will not respond to any subsequent Discover ID request from this master.

TLMC_S_021 - DISCOVER FULL - DISCOVER NAME RESPONSE

After the receipt of a valid master Discover Name request packet, the slave shall do the following:

1. Transmit the Discover Name response packet(s). see **Table 112 (DISCOVER NAME SLAVE RESPONSE FORMAT)** for details. The proximity status shall be the byte immediately following the slave ID see **Fig. 35 (Proximity status type byte)**.
2. The slave shall store the master station ID from the Discover Name packet for the collision avoidance duration indicated in the Discover Name request packet. The slave shall not transmit any Discover ID response for the subsequent Discover ID request from this master for the duration of the collision avoidance timer.
3. Send a discover name indication (sub-type of 0x01) to the host (see **Table 24 (DISCOVER INDICATION (SLAVE))**). This is an implicit security indication. The indication to the host will not be sent more than once a second.

[The slave shall update its current remaining Scan duration with that from the Discover Name request packet (discover ID collision avoidance duration) if the current remaining Scan duration is shorter. Also extend the listen time by 50ms if the remaining listen time is less than 50ms. The size of the collision avoidance buffer is ten, a circular buffer is used to keep track of number of masters issuing Discover full]

TABLE 24: DISCOVER INDICATION (SLAVE)(1)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x22 - Discover indication
Sub-type	N/A	1	The indication sub-type	0x01 = Discover Name received 0x02 = Discover ID received while uplinking medical event
Size	N/A	2	The size of the info field	1
Info	Discover name indication status	2	Status of the indication	1st byte: 0x01 - successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

1. The same Discover Indication will not be sent to the host more than once per second. For example, while uplinking medical event, if discover ID is received over and over within 1 second after sending the previous Discover ID received indication, slave shall not send the Discover ID received indication.

TLMM_S_271 - DISCOVER FULL - CANCEL COLLISION AVOIDANCE TIMER

The collision avoidance timer will be canceled upon receiving of an “authorize BAN key exchange service request or update proximity status service request”.

TLMC_MS_022 - DISCOVER FULL - DISCOVER NAME RESPONSE PACKET FORMATS

During the slave response of Discover Name the slave will use one or more of the packet formats as shown in **Fig. 34 (Discover Name Response Packet)**.

TLMM_M_272 - DISCOVER FULL - SUCCESSFUL NAME

The master will send a discover indication to the upper layer (with a status of success) if it successfully obtains the slave’s name. The format of the indication is shown below:

TABLE 25: DISCOVER INDICATION (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xA2 - Discover indication
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	Number of data bytes	1 or N+8[1]
Info	Status	1	The indication status	0x01 - success 0x02 - cancelled[2] 0x03 - timeout
	Slave ID	6	The slave ID (6 bytes)	The ID of the slave
	Proximity status	1	The security proximity status	**Fig. 35 (Proximity status type byte)**
	Slave name	N	Station name string for the slave ID	The slave name
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

1. The size of 1 if the status is not successful. The size of N+8 is otherwise.
2. This status can occur if the master receives a discover stop service request (while the discover service duration is active). It can additionally occur if emergency interrupts discover.

First Packet

Discvr Name Control (5.25 bytes)	**Destination ID (master ID) (6 bytes)**	**Source ID (slave ID) (6 bytes)**	**Proximity Status (1 bytes)**	**Device Name n bytes**	**Zero fill (17 -n bytes)**	**MAC[a] (3 bytes)**	**Reed Solomon (9 bytes)**

Discover Name (and packet is not full)

Discvr Name Control (5.25 bytes)	**Destination ID (master ID) (6 bytes)**	**Source ID (slave ID) (6 bytes)**	**Proximity Status (1 byte)**	**Device Name (17 bytes)**	**MAC[a] (3 bytes)**	**Reed Solomon (9 bytes)**

Discover Name (and packet is full)

[a] Message authentication code

Ensuing Packet(s)

Discvr Name Control (5.25 bytes)	**Device Name (n bytes)**	**Zero fill (30 - n bytes)**	**MAC[a] (3 bytes)**	**Reed Solomon (9 bytes)**

Discover Name (and packet is not full)

Discvr Name Control (5.25 bytes)	**Device Name (30 bytes)**	**MAC[a] (3 bytes)**	**Reed Solomon (9 bytes)**

Discover Name (and packet is full)

[a] Message authentication code

Figure 34: Discover Name Response Packet

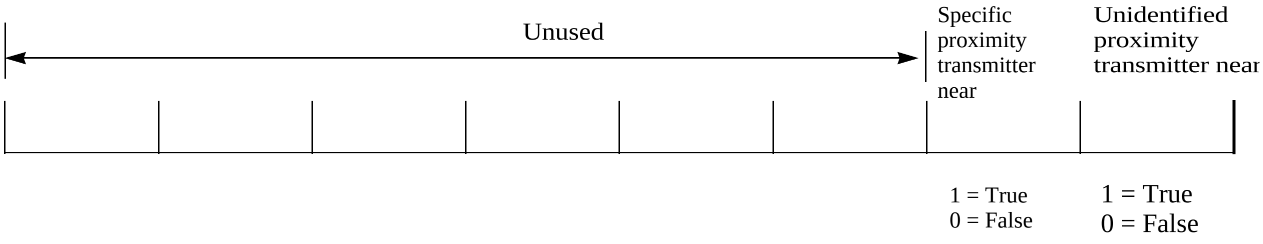


Figure 35: Proximity status type byte

TLMC_S_023 - DISCOVER FULL - RESPONSE PARTITIONING

The slave will partition the Discover Name response packet(s) using **Table 26 (DISCOVER NAME RESPONSE FRAGMENTATION)**.

[Note that if a Discover Name (ensuing, not full) packet is needed, it must be the last packet of the transaction.]

TABLE 26: DISCOVER NAME RESPONSE FRAGMENTATION

INPUTS		RESPONSE PACKETS			
NAME SIZE (N)	(N-17) MOD 30 EQUAL 0	NAME (FIRST, FULL)	NAME (ENSUING, FULL)	NAME (FIRST, NOT FULL)	NAME (ENSUING, NOT FULL)
N < 17 BYTES	-	0	0	1	0
N >= 17 BYTES	TRUE	1	INT[(D-17)/30]	0	0
N>= 17 BYTES	FALSE	1	INT[(D-17)/30]	0	1

If a valid Discover Name response is not received by the master, the master shall request the Station Name String again by issuing another Discover Name packet.

TLMM_M_273 - DISCOVER FULL - TIMEOUT INDICATION

The master will send a discover indication (with a status of timeout) when the discover full service duration expires. See **Table 25 (DISCOVER INDICATION (MASTER))**.

TLMM_M_274 - DISCOVER FULL - STOP

The master will stop the discover full service and send a service response if it receives a stop discover full service request. Additionally it will send a discover indication (status of cancelled). See **Table 25 (DISCOVER INDICATION (MASTER))**.

5.2.9.3 DISCOVER EXAMPLE

Fig. 36 (Discover Service - Packets Exchange in Async Mode) illustrates the exchange of request packets by the Data Link Layer for the Discover services. The response packets may or may not be received properly. At ‘Setup’, the Data Link Layer invokes the physical layer to transmit the request packet. The physical layer provides an event when the packet is transmitted and another event if the packet is correctly received. The time between the slave packet and the next master packet is used by the master Data Link Layer to process the slave packet. (In **Fig. 36 (Discover Service - Packets Exchange in Async Mode)**, Disc ID Req = Discover ID request, Disc ID Resp = Discover ID response, and so forth.)

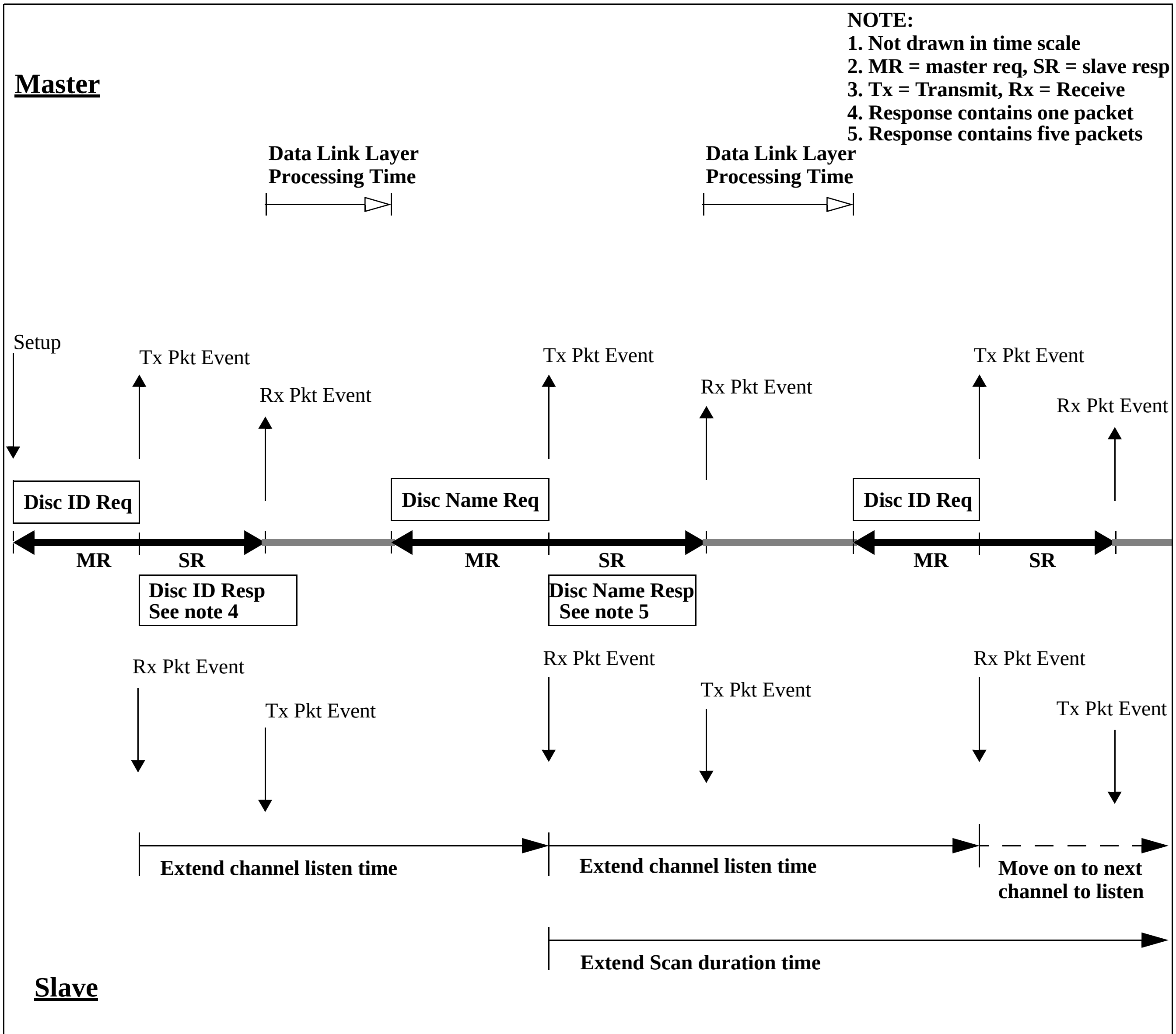


Figure 36: Discover Service - Packets Exchange in Async Mode

Fig. 37 (Discover Service - Discover ID request antenna switching) shows antenna switching for Discover ID request packets in Async mode.

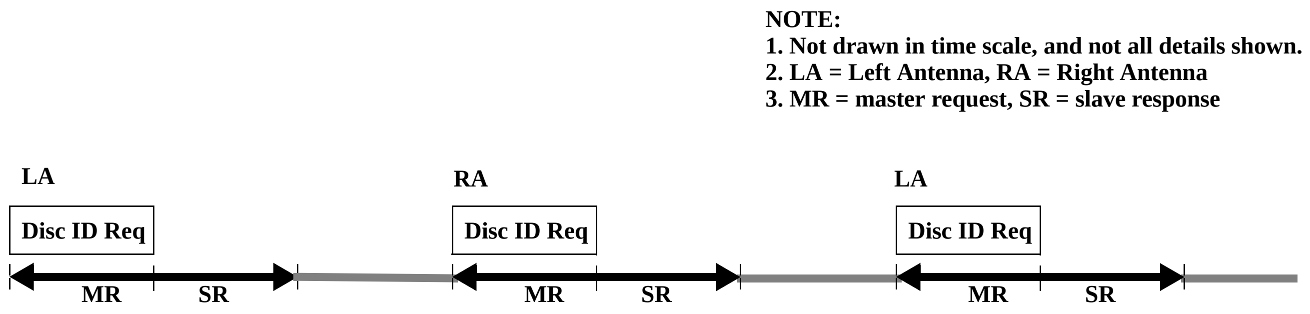

NOTE:
1. Not drawn in time scale, and not all details shown.
2. LA = Left Antenna, RA = Right Antenna
3. MR = master request, SR = slave response

Figure 37: Discover Service - Discover ID request antenna switching

Fig. 38 (Discover Service - Discover ID/Discover Name request/response) shows antenna switching for Discover ID/Discover Name request/response master packets.

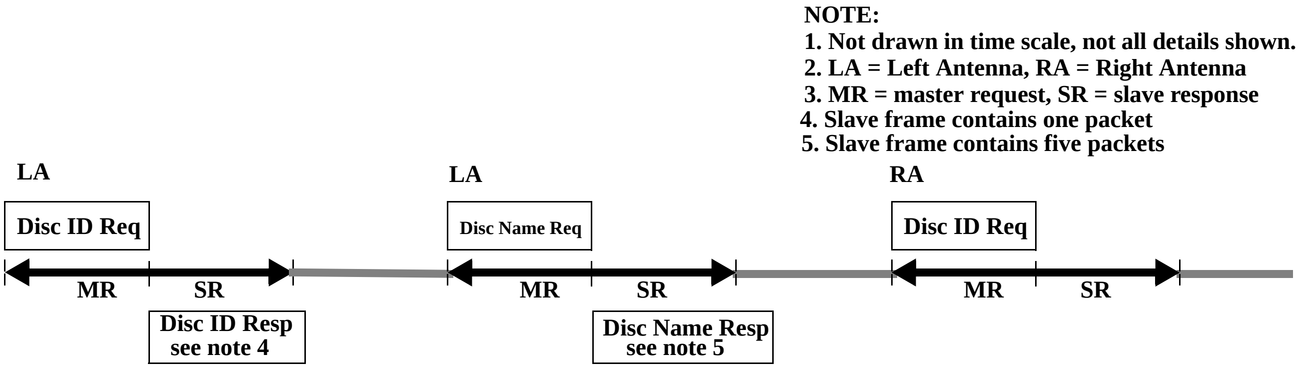

NOTE:
1. Not drawn in time scale, not all details shown.
2. LA = Left Antenna, RA = Right Antenna
3. MR = master request, SR = slave response
4. Slave frame contains one packet
5. Slave frame contains five packets

Figure 38: Discover Service - Discover ID/Discover Name request/response

The data collected by the master during the Discover services for each slave include the slave ID, the Station Name String, the slave Telemetry M Protocol version, and the antenna on which the responses were received for each slave.

5.2.10 OPEN SESSION

A station uses this service to establish communications with another station. Normally a master uses the open service to establish communications with a slave. The following bullets briefly summarize this service:

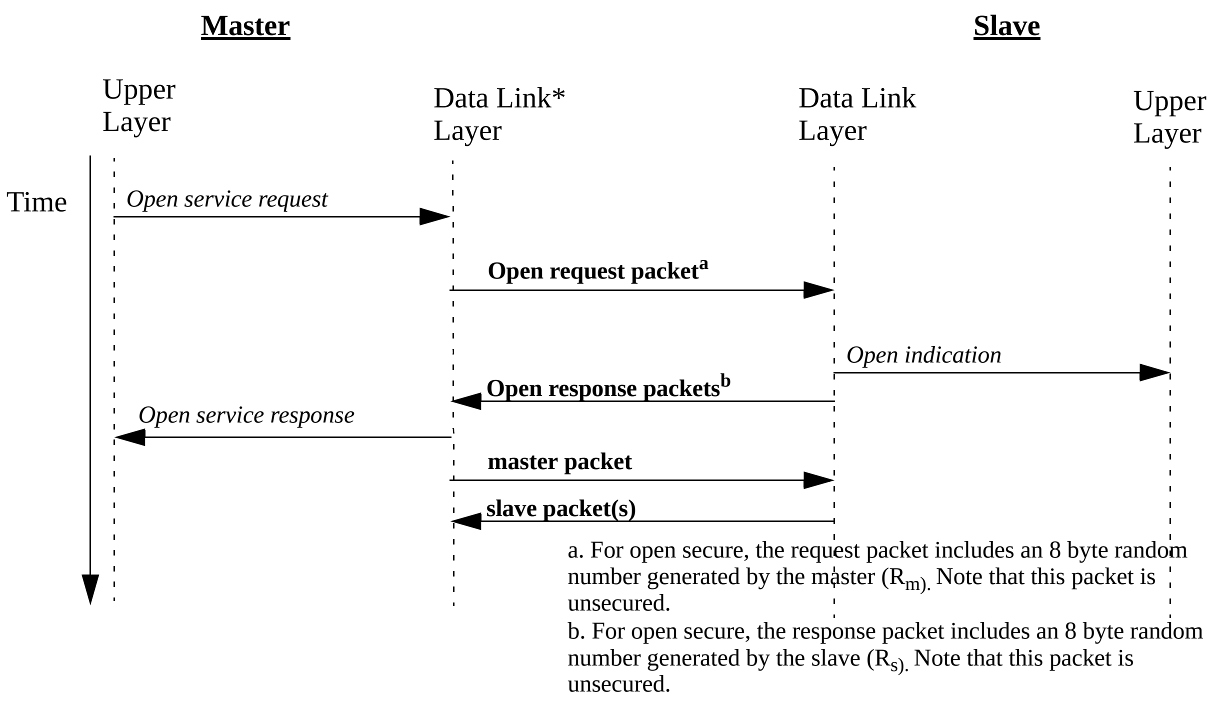
- Upper Layer generates the Open request to the Data Link Layer.
- The Data Link Layer selects a channel and transmits Open requests. The station remains on the selected channel until communication is established or a Service Duration Timeout occurs.
- If an Open response is received, the Data Link Layer verifies the response contents. If the contents are correct, then communication is established between the two stations.
- The Data Link Layer notifies the Upper Layer with the service results.
- To open a secured communication session requires the use of one the two keys:

  K_{BAN} - Key used during a secure communication session, or

  K_{EBAN} - Temporary key is used for a duration of a session. This key expires based on a timer or when session is closed.
- A fixed known token value is used to open an unsecured communication session

  K_{Token}- A fixed known token value is used among the intended stations.

When invoking this service the Upper Layer provides the station ID of the slave station to establish communication. An operational example of the Open service is shown in **Fig. 39 (Open Sequence Diagram)**.



a. For open secure, the request packet includes an 8 byte random number generated by the master (R_m). Note that this packet is unsecured.

b. For open secure, the response packet includes an 8 byte random number generated by the slave (R_s). Note that this packet is unsecured.

* Channel mapping and wakeup are not shown.

Figure 39: Open Sequence Diagram

5.2.10.1 OPEN — GENERAL REQUIREMENTS

TLMC_M_040 - OPEN - UPPER LAYER PARAMETERS

When invoking an Open service request, the Upper Layer will provide information as specified in **Table 27 (OPEN SERVICE REQUEST (MASTER))**.

TABLE 27: OPEN SERVICE REQUEST (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x83 =Open service request
Sub-type	N/A	1	The service request sub-type	0x00 = Open unsecure 0x02 = Open secure using K_{BAN} 0x04 = Open unsecure (for Tel. M only) 0x06 = Open secure using K_{EBAN} Note: there are no illegal sub-types. The host supplied sub-type will be “anded” with 0x06. Stated differently bits 1 and 2 are pertinent; bits 0 and 3-7 are impertinent.
Size	N/A	2	The size of the info field	6
Info(1)	Open Slave ID	6	Slave ID consists of six bytes: Model ID - two bytes, SubModel ID - one byte, Serial # - three bytes.	Slave ID (Model ID, SubModel ID, and Serial# must all be specified.). Model ID must match the model ID in the config file. Slave ID may not change from one open request to another (without an intervening close service request).
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

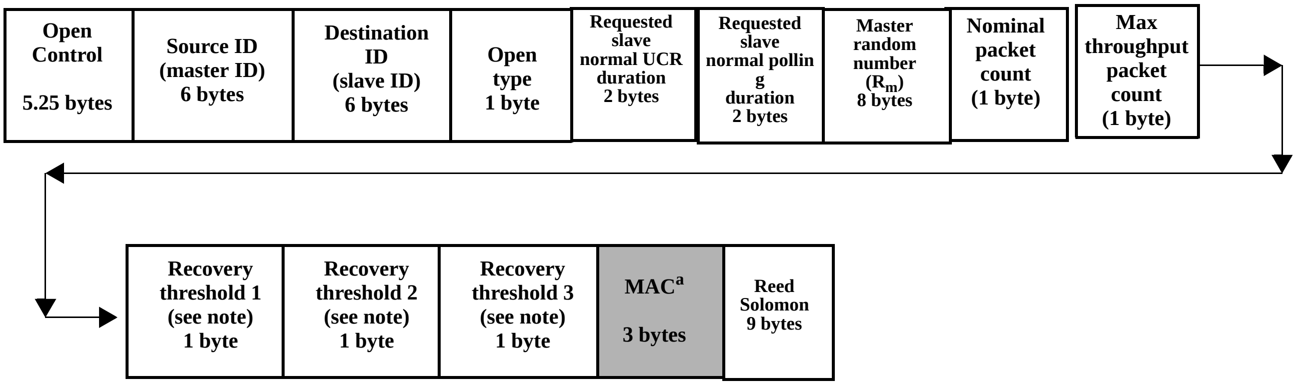
1. The open service has one additional parameters which control its behavior: open service duration. See **Table 123 (Master parameters from host)**.

The open master packet is used to establish a bidirectional communication link between a master and a slave where the open request type field is initialized to zero.

TLMM_MS_281 - OPEN - REQUEST PACKET FORMAT

The open request packet has the format indicated in **Fig. 40 (Open Secure/Unsecure Request Packet)**

The “normal values” for the slave’s unspecified channel recovery (UCR) duration and slave’s polling duration will come from the initialize service request from the master host (**Table 123 (Master parameters from host)**).



a. Message authentication code

Note: The effective channel recovery threshold is controlled by the power inhibit mode. Recovery threshold 1 is used when the power inhibit mode is idle immediate or roll through mode 1. Recovery threshold 2 is used when the power inhibit mode is roll through mode 2. Recovery threshold 3 is used when the power inhibit mode is roll through mode 3.

Figure 40: Open Secure/Unsecure Request Packet

The description of open type byte is as follows:

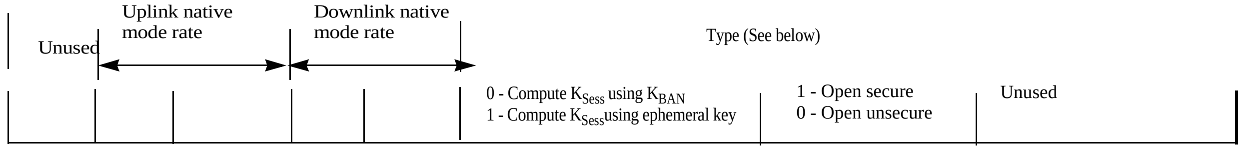
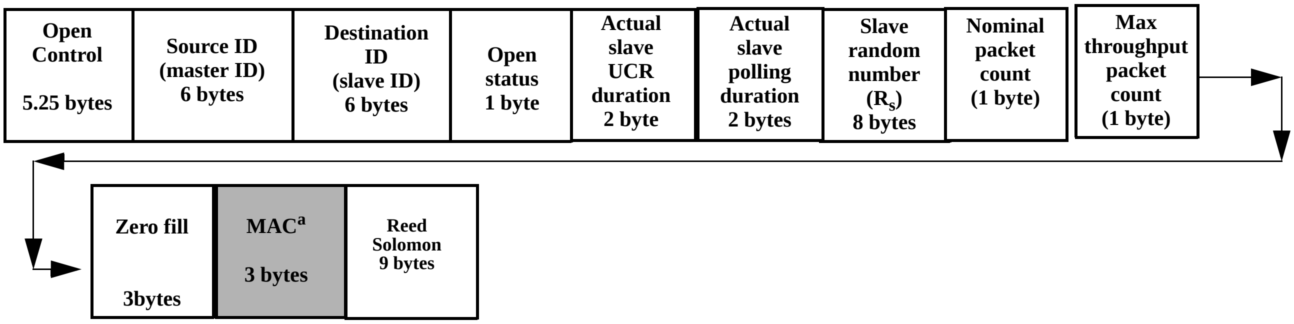


Figure 41: Open type byte (Request)

TLMM_MS_282 - OPEN - RESPONSE PACKET FORMAT

The open response packet has the format indicated in **Fig. 42 (Open Secure/Unsecure Response Packet)**

[Nominal packet count and Max. throughput packet count in the open response packet is to support verification testing when Protocol Analyzer is used.]



a. Message authentication code

Figure 42: Open Secure/Unsecure Response Packet

The description of open status byte is as follows

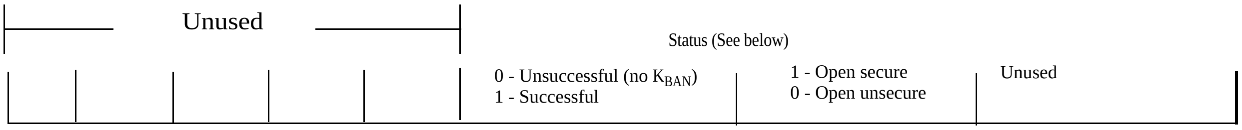


Figure 43: Open status byte (Response)

TLMM_M_283 - OPEN - TRANSMIT MASTER PACKET

Upon receiving an open service request, the master will start a service duration timer and transmit the open request packet.

[Start the service duration timer; select the least interfered channel (takes approximately 275 +/- 50 msec(2)); start the Listen Before Talk (LBT) timer; perform instigator wireless wakeup (including unsecured or secured user data if any exists); transmit the open request packet (including the 8 byte random number used to compute a session key); send a “first request packet sent” event indication to the Upper Layer (see **Table 23 (FIRST REQUEST PACKET SENT INDICATION (MASTER))**).

Note that instigator wireless wakeup is suppressed if the master knows the slave is scanning. Time now, time of last uplink, and remaining polling duration at the time of the last uplink are used to determine this.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup is not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

[The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**). Note that the master random number (R_m) can be re-used for the entire open secure service. When communication is established and subsequently lost, a new random number (R_m) must be generated during open and session recovery.]

TLMM_M_284 - OPEN - NO VALID SLAVE RESPONSE RETRY REQUEST

If no response is received, the master will resend the open request till the Service Duration timer is expired.

[If no valid open response packet is received, the master will perform the following:

1. Transmit an open request packet on the alternate antenna.

This process shall be continued until a valid open response packet is received, a Service Duration Timeout occurs, or a Listen Before Talk timer expires.]

TLMM_M_285 - OPEN - LISTEN BEFORE TALK TIMER OPERATIONS

If the listen before talk timer expires during an open service, the master will transition to unspecified channel recovery.

[As part of unspecified channel recovery following actions are performed: Re-select the least interfered channel for communication, Start the Listen Before Talk timer, toggle the antenna, perform instigator wireless wakeup, and transmit an open request.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

2. CCA of the MICS band takes approximately 275 msec. CCA of the MEDS band takes approximately 550 msec. CCA of both bands takes approximately 825 msec. Note that if HDR mode is enabled, assessment of each channel, antenna pair is lengthened by 5 msec (to allow the HDR synthesizer to stabilize). This means the CCA of the MICS band is lengthened from 275 msec to 375 msec

TLMC_S_286 - OPEN - RESPONSE TO A VALID OPEN REQUEST

Upon receiving a valid open request packet, during polling operation, the slave will fill up the transmit frame with N open response packets ("N" = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**. Slave will update the unspecified channel recovery duration and polling duration using the values in the open request packet.

Upon completion of uplinking all the open response packets, the slave will terminate the polling operation; It will then transition to the Comm State. An open indication will be sent to the upper layer (see indication **Table 28 (OPEN INDICATION (SLAVE))**). Additionally the slave will transition to the desired downlink rate and uplink rate specified in the open downlink packet. Finally the slave will reset (to zero) the waveform sequence number.

[Note that the open response packets will include the actual UCR duration and actual polling duration and an 8 byte slave random number (R_S).

Slave will restart the native mode time, channel listen time or the medical event transmit time if the timer (native mode time, channel listen time or the medical event transmit time) times out during the uplinking of the open response packets.]

TLMM_S_287 - OPEN - NO RESPONSE TO A VALID OPEN REQUEST

When slave's "authorized to talk unsecurely" parameter is false as provided by the host, and a valid open "unsecure" request is received then slave will not transmit any response packets.

[When this happens, slave will continue polling.]

[Host authorizes Tel M to open and/or unsecure session as part of initialization]

TLMM_S_288 - OPEN - RESPONSE TO A VALID OPEN UNSECURE REQUEST

When slave is authorized to open an "unsecure" session by the host, and a valid open "secure or unsecure" request is received then slave will accordingly transmit the response packets.

[Host authorizes Tel M to open secure and/or unsecure session as part of initialization]

The security block of the hardware will be enabled during the communication session.

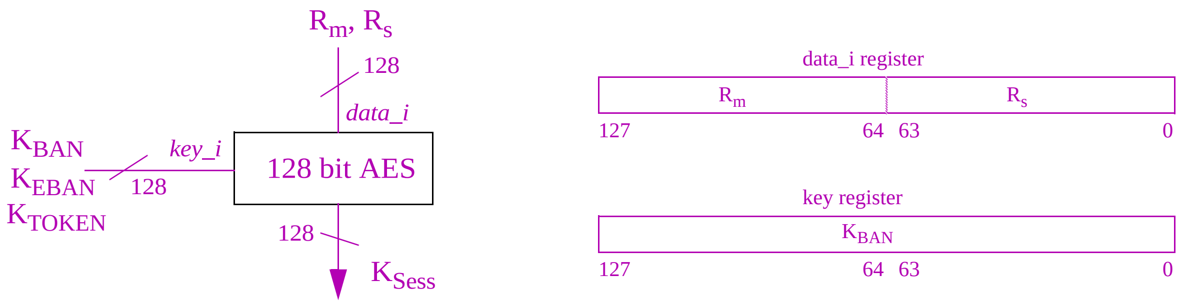


Figure 44: Computing a session key

TLMM_S_289 - OPEN - SESSION KEY COMPUTATION UPON VALID OPEN REQUEST

Upon the receipt of a valid open request the slave will compute a session key using the eight byte master random number (R_m), the eight byte slave random number (R_s) and one of the K values using the block cipher: the BAN key (K_{BAN}), ephemeral BAN key (K_{EBAN}), a fixed known token value (K_{token}). See **Fig. 44 (Computing a session key)**.

[The eight byte master random number (R_m), slave random number (R_s) and BAN Key (K_{BAN}) or Ephemeral K_{EBAN} will be loaded into the appropriate registers of the AES hardware block to generate the Session Key $K_{Sess.}$ For an unsecure session - A fixed known token value (K_{Token}) is loaded to the AES hardware block instead of the K_{BAN} or K_{EBAN} key(s)]

TLMM_S_290 - OPEN - RESPONSE SUCCESSFUL

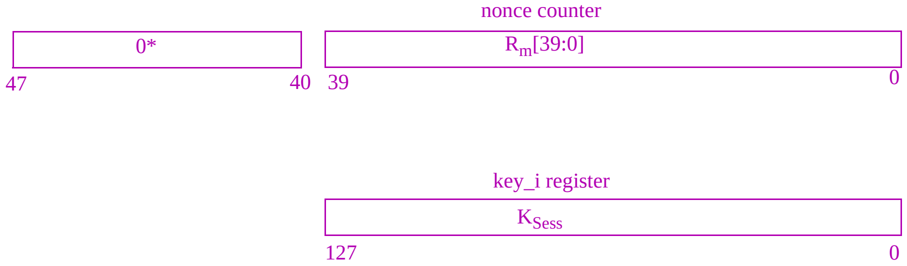
Open successful status will be transmitted in the open response packet if:

1. Slave has the appropriate Ban Key as request by the master, or
2. Slave is authorized to communication unsecurely and open request is for unsecure open.

TLMM_S_292 -OPEN - ENABLE SECURITY BLOCK

Upon transmitting the open response the slave will enable the hardware's security block (security register to on). The slave will start the block cipher for message integrity, privacy, and freshness checking. This will be done by writing the session key to the block cipher key register. Additionally, the slave will initialize the block cipher nonce counter using the lower 5 bytes of the master random number (R_m). The most significant byte of the nonce counter is set to zero to allow a very long session because once the nonce reaches its max value it stops.

[Slave enables the hardware security block even if an unsecured session is opened - See **Fig. 45 (Open - block cipher initialization)**.]



* Note: 2^{48} - 2^{40} packets to be exchanged during a session

Figure 45: Open - block cipher initialization

TABLE 28: OPEN INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x23 - Open indication
Sub-type	N/A	1	The indication sub-type	0x01 - Open secured 0x02 - Open unsecure
Size	N/A	2	The size of the info field	1
Info	open indication status	1	status of the indication	0x01 - successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

[Note: An open indication is sent to the upper layer indicating that a session is opened; continue the Synchronous Mode operation and wait for a valid master packet. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).]

TLMM_S_293 - OPEN - UNSUCCESSFUL - MISSING KEY

The slave will send an unsuccessful open response if the following occurs:

1. The slave receives a valid open request and the open type is secured, and
2. The slave does not have the K_{BAN} or K_{EBAN} filled that was specified in the open type byte (See **Fig. 41 (Open type byte (Request))**

The slave will set the open response type byte to unsuccessful. See **Fig. 42 (Open Secure/Unsecure Response Packet)**.

TLMM_M_294 - OPEN - A VALID SLAVE RESPONSE

If a valid open response packet is received, the master will perform the following:

1. Return a successful open service response to the Upper Layer.
2. Transition to the Comm State.
3. Compute a session key using the eight byte master random number (R_m), the eight byte slave random number (R_s) and the BAN key (K_{BAN}) using the block cipher. See **Fig. 44 (Computing a session key)**.
4. Transition to the desired downlink rate and uplink rate. Note that these originally came from **Table 123 (Master parameters from host)**.
5. Enable the hardware’s security block (security register to on).
6. Start the block cipher for message integrity, privacy and freshness checking. This is done by writing the block cipher key register with the session key. Additionally, the nonce counter is initialized using the lower 5 bytes of the master random number (R_m). The most significant byte of the nonce counter is set to zero to allow a very long session because once the nonce reaches its max value it stops. See **Fig. 45 (Open - block cipher initialization)**.
7. Clear the wakeup internal service user data buffer to prevent retransmission of stale user data.
8. Reset (to zero) the expected waveform sequence number.

[Note that the recently exchanged ephemeral BAN key will be used (to calculate a session key) if the ephemeral key is still valid. The ephemeral BAN key is valid until the ephemeral BAN key timer expires. Note: if open type is unsecure, K_{token} is used.]

TLMM_M_295 - OPEN - RESPONSE WITHOUT K_{BAN} OR K_{EBAN}

If a valid open secure response packets is received with unsuccessful status, then master will stop transmitting open requests.

[The master will transition to the standby state if the open uplink indicates the slave lacks the appropriate key.]

The open secure service status will be returned to the Upper Layer.

TLMM_M_296 - OPEN - RESPONSE TO UPPER LAYER

An open service response will be generated to the Upper Layer when the open service is completed.

[The content of the response to the Upper Layer is specified in **Table 29 (OPEN SERVICE RESPONSE (MASTER))**.]

TABLE 29: OPEN SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x83 - Open service response
Sub-type	N/A	1	The service response sub-type	0x00 = open unsecure 0x02 = open secure (with Kban) 0x04 = open unsecure 0x06 = open secure (with Keban)
Size	N/A	2	The size of the info field	1
Info	Open secure service status	1	Status of the service request	0x01 - Request successful 0x02 - Request invalid in current state (state is Comm) 0x03 - Request invalid in current state (state is Recovery; slave scanning)(1),(2) 0x04 - Request invalid in current state (state is Recovery; slave not scanning)(1),(2) 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x0A - Parameter value error(3) 0x0C - Service cancelled (interrupted by emergency) 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)(3) 0x18- Slave has no ephemeral BAN key 0x1E - slave has no permanent key 0x1F - master has no EBAN key 0x20 - master has no permanent BAN key 0x25 - Buffer/ service request mismatch(4)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration). Note that a service duration timeout will result in a Data Link response status of 3 or 4 (invalid state, state is Recovery). This is done to aid the upper layer (application) in case the open immediate request fails. A Data Link status of 9 (service duration timeout) would not provide any benefit.
2. Note that this status can also occur when open unsecure is attempted and the slave is not authorized to talk unsecurely.

3. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table or the slave ID is incorrect, 0x0A - Parameter value error will be used.
4. This status can occur if the master host attempts to send a open service request using service request buffer 1 (instead of buffer 0).

5.2.11 DATA

The data service request is used for four purposes in Mozart:

1. Transmit a transaction from the master host to the slave host. Upon receipt, the slave host answers with a data service request of its own. The original (master) data service request is typically a read or write request (read slave host memory or write slave host memory). This is the most common use of the data service request.
2. Transmit a RAMware patch from the master host directly to the slave Mozart module. Note the the slave host may never know of this service request.
3. Transmit a RAMware patch from the master host to the master Mozart module. Note that no telemetry packets are needed in this instance.
4. Transmit a RAMware patch from the slave host to the slave Mozart module. Note that no telemetry packets are needed in this instance.

Refer to **7.6 Ramware** for reading or writing to the telemetry module.

This service is used for both master and slave to transfer user data. The following bullets briefly summarize this service operation.

- The user on the source station requests the Data Link Layer to transmit data to the destination station.
- The Data Link Layer on both stations ensures the packets are transmitted and received without error.
- In the destination station, the Data Link Layer notifies the Upper Layer that data was received.
- In the source station, the Data Link Layer notifies the Upper Layer of the data service result.

The Data Link Layer does not generate or interpret the contents of the user data. The Data Link Layer merely transfers the user data between the two stations. The Data Link Layer is responsible for error free delivery of the user data between Data Link Layers of the stations.

The Data service indication is used to inform the Upper Layer that data are received and available. This data includes master and slave Memory Data; slave Supplemental Markers; and Waveform data.

The timing of user data transfers initiated by the master is handled by the master. User data transfers initiated by the slave, such as supplemental markers, are constrained by the parameters of that data transfer.

User data transfers are prioritized in the following manner:

1. Master data transfers have priority over slave data transfers.
2. Slave real-time data transfers have priority over slave Memory Data transfers.

Slave real-time data is interleaved with master communications or within slave communications.

[Note that **5.2.13 Emergency** is implemented by borrowing significantly from components of the Data service.]

Fig. 46 (Memory Data Master/ Slave Sequence Diagram) is a sequence diagram; it depicts an master transmitting a Memory Data request and the slave responding with Memory Data. In this case, the ARQ algorithm is used to acknowledge the slave memory data.

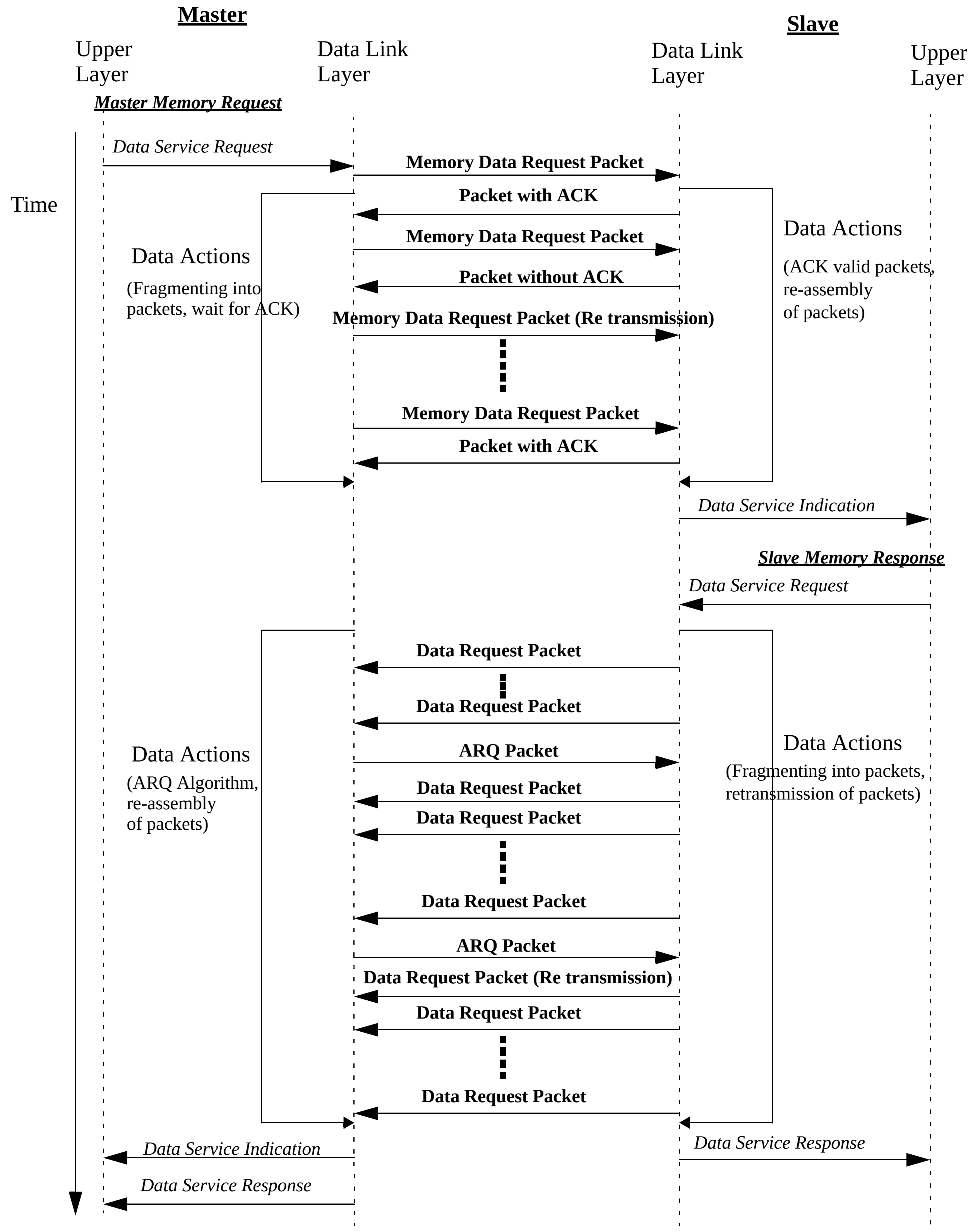
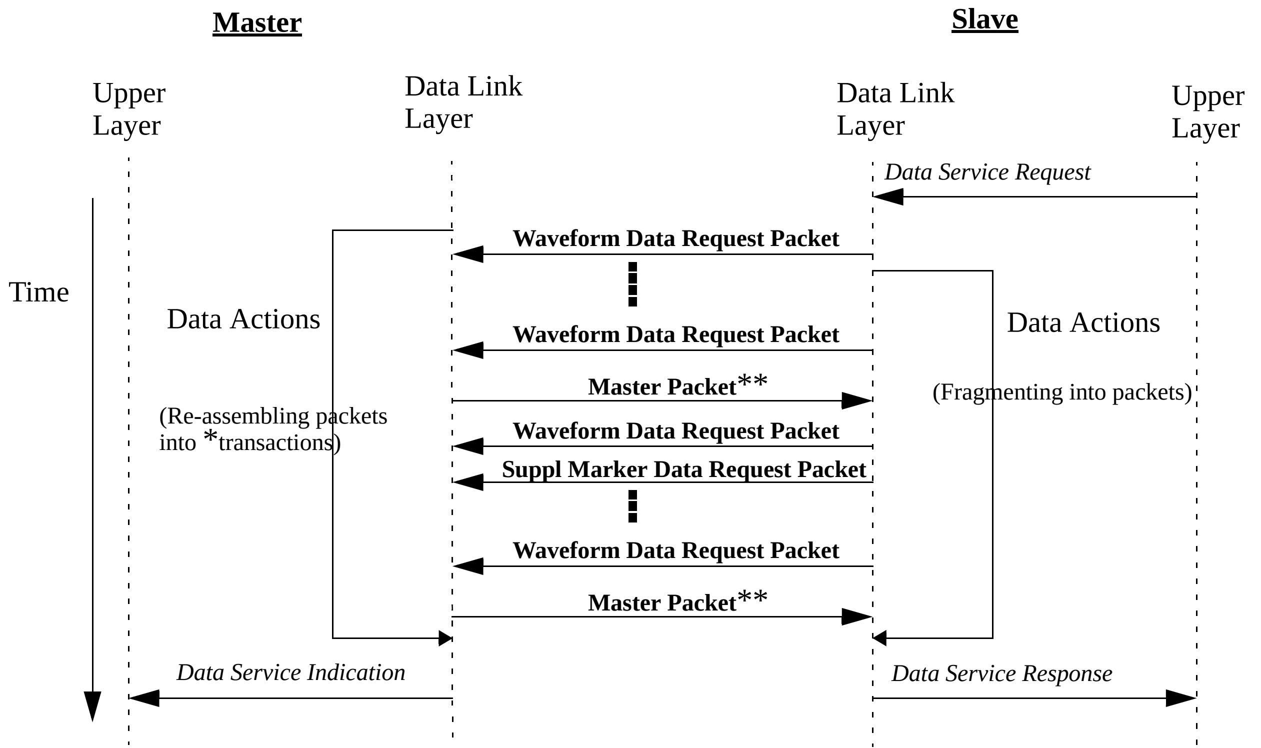


Figure 46: Memory Data Master/ Slave Sequence Diagram

Fig. 47 (Slave Unrequested Data Sequence Diagram) illustrates the slave transmitting unrequested data packets which do not use the ARQ algorithm.



* ‘transaction’ is defined in section 5.2.11.1

** In this illustration, the slave data packets are interleaved with two master packets of unspecified type and content.

Figure 47: Slave Unrequested Data Sequence Diagram

5.2.11.1 REQUIREMENTS

Each Data service request is considered a ***transaction***. The maximum number of data bytes that can be transmitted by the master via one Data service request is the MAX_DATA_TRANS_SIZE. The maximum number of data bytes for a slave is dependent on the maximum transaction buffer size (MAX_DATA_TRANS_SIZE).

User data to be transmitted via a Data request shall have the Upper Layer Header (ULH).

5.2.11.2 MASTER’S UPPER LAYER HEADER (ULH) BYTE FORMAT

The data service, emergency service, and network data service request is used to transmit a transaction from the master to the slave. An upper layer message from the master consists of one or more transactions. This message is used to read or write a slave node’s memory.

The master’s upper layer will add an upper layer header (ULH) byte to the start of each transaction. The master’s upper layer header byte contains two active bits:

- **Transaction Type -** This allows the slave to distinguish between a Data and Emergency request.
- **Max throughput data transfer mode enabled -** allows the master to utilize max throughput mode when receiving memory data from the slave. This mode is typically disabled when the slave is performing periodic power inhibit.
- **Reset -** Cancel an (in-progress) master or slave message. A cancelled master message occurs when a transaction (with reset) is received before the final transaction of a multi-transaction master message.
- **Final transaction -** This allows the slave to assemble a multi-transaction request message. Note that this bit must be set for emergency (multi-transaction emergency requests are not allowed).
- **Destination -** These bits are used to indicate the intended destination of the data being sent. The intended receipt could be slave host. Data intended for telemetry module slave, or telemetry module master see section **7.6 Ramware** in **7.0 Appendices** for details.
- **Application reply requested -** If true, the master is requesting an application response from the slave. This is typically true. It can be false in certain cases (to save energy). This field is pertinent to the network service only.

The format of the master’s upper layer header byte is given in **Fig. 48 (Master’s ULH byte)**.

bit 7	**bit 6**	**bit 5**	**bit 4**	**bit 3**	**bit 2**	**bit 1**	**bit 0**

bit 7, bit 6 - Transaction Type (01 = Emergency, 00 = Data)
bit 5 - max throughput data transfer mode enabled
bit 4 - Reset
bit 3 - Final Transaction (1 = Final, 0 = More)
bit 1,2 = destination
    00 - Slave host
    01 - Tel_module slave data
    10 - Tel_module master data
    11 - Slave host
bit 0 - Application reply requested (pertinent to the network service only)

Figure 48: Master’s ULH byte

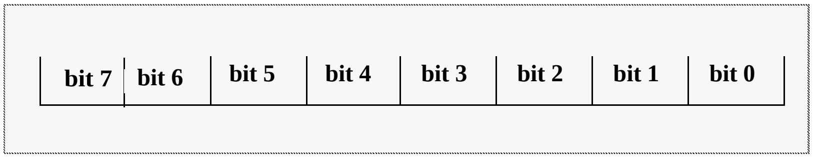
5.2.11.3 SLAVE’S UPPER LAYER HEADER BYTE FORMAT

The data service and network data service request is also used to transmit a transaction from the slave to the master. A network data message from the slave consists of one transaction. This message is used to respond to a read or write request.

The slave’s upper layer will add an upper layer header (ULH) byte to the start of each transaction. The slave’s upper layer header byte contains two active bits:

- **Final transaction -** This allows the master to assemble a multi-transaction response message.
- **Transaction type -** This allows the master to distinguish between a Data and Emergency response.
- **Destination -** These bits are used to indicate the intended destination of the data being sent. The intended receipt could be master host. Data intended for telemetry module slave, see section **7.6 Ramware** in **7.0 Appendices** for details.
- **First transaction -** This is set and used by the master, it is not used by the slave.
- **Buffer transaction number from host-** These are set by the slave host. This helps the telemetry module to determine which transaction should be transmitted first[3] The slave host should start with a transaction number of zero and increment thereafter.

The format of the slave’s upper layer header byte is given in **Fig. 49 (Slave’s ULH byte)**.



bit 7 - Transaction Type (1 = Emergency, 0 = Data)
bit 4 - Final Transaction (1 = Final, 0 = More)
bit 5, 6 - destination:

00 - Master host
01 - Tel_module slave data
10 - Master host
11 - Master host

bit 3 - First transaction (1 = first, 0 = not first)
bits 0, 1, 2 - Buffer transaction numbers from host

Figure 49: Slave’s ULH byte

---

[3] If the slave host does not utilize this field, the intended response may be received out of order at the master host.

5.2.11.4 MASTER DATA

Memory Data requests from the Upper Layer are serialized. One Data request from the Upper Layer has to be completed before the next Data request can be processed.

TLMC_MS_049 - DATA - UPPER LAYER TO DATA LINK LAYER PARAMETERS

When invoking the Data service request, the master Upper Layer will provide information as specified in **Table 30 (DATA SERVICE REQUEST (MASTER OR SLAVE))**.

TABLE 30: DATA SERVICE REQUEST (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x84 - Master Data service request(1) 0x04 - Slave Data service request
Sub-type	N/A	1	The service request sub-type	0 - Memory data(2), 1 - Unrequested data(3), 2 - Waveform data(4) Note, this field applies to Slave only
Size	N/A	2	The size of the info field	1 to MAX_DATA_TRANS_SIZE (480 - size of STATION_ID) 480 - sizeof (ID) = memory and unrequested data 240 - sizeof (ID) = waveform data
Info(5)	Data	N	Transaction to be transmitted	The data to be transmitted. Memory data includes the Upper Layer Header (ULH) as the first byte. See **5.2.11.2** and **5.2.11.3** for the master's ULH byte and the slave's ULH byte respectively. Supplemental marker and waveform data do not have a ULH byte.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. A master data service request is used to request the slave to write memory (or to read the slave's memory).
2. A (slave) memory data service request is used to respond to the memory read or write request.
3. A (slave unrequested) supplemental marker data request is used to send the slave's therapy status.
4. A (slave) waveform data service request is used to send the slave's waveform data.The table content will not be checked when data link layer receives the waveform service request interrupt. Data link layer will move the data into the info field and clear the corresponding service request interrupt.
5. The data service has one additional parameters which control its behavior: reply duration. See **Table 123 (Master parameters from host)**.

TLMC_MS_050 - DATA - MASTER REQUEST PACKET FORMAT(S)

The master data request has the format shown in **Fig. 50 (Master data packet formats)**

Figure 50: Master data packet formats

TLMC_M_051 - DATA - TRANSMISSION OF MASTER DATA PACKET

When the Data service request is invoked, the master will start the reply timer, format the user data into data fragment(s)/packet(s) and transmit the data packet(s).

[Initiate the Internal Service Reset if the reset parameter is True, Add the master ID in the first fragment, transmit the packet(s), send a first request packet sent event indication to the Upper Layer after sending the first data packet (see **Table 23**)]

TLMC_S_052 - DATA - ACKNOWLEDGEMENT OF MASTER DATA PACKET

The slave shall set the ACK bit in the slave packet(s) when it receives a master packet with good message authentication code (MAC). The ACK bit in the slave packet(s) will not be set if the master packet was received with a bad MAC.

[Note: A packet with a bad sequence number will also be ACKed and the data will be ignored.]

[See requirement **TLMC_S_055 - DATA - MASTER PACKETS WITH DUPLICATE SEQUENCE NUMBERS**.]

Each master data packet has a unique sequence number. The Last Fragment bit in the control byte indicates the end of a master transaction.

The master shall perform all the Internal Services necessary for the successful transmission of the packets to the slave station. The packet will be retransmitted until positively acknowledged or a Service Duration Timeout occurs.

TLMC_M_053 - DATA - PACKETIZATION OF THE MASTER DATA

The master will fragment the user data into packets per **Table 31 (PACKETIZATION OF MASTER DATA)**.

[Note that if a data (ensuing, not full) packet is needed, it must be transmitted as the last packet of the transaction.]

TABLE 31: PACKETIZATION OF MASTER DATA

INPUTS		MASTER PACKETS			
DATA SIZE[1] (D)	(D-24) MOD 30 EQUAL 0	DATA (FIRST, FULL)	DATA (ENSUING, FULL)	DATA (FIRST, NOT FULL)	DATA (ENSUING, NOT FULL)
D < 24 bytes	-	0	0	1	0
D >= 24 bytes	TRUE	1	int[(D-24)/30]	0	0
D >= 24 bytes	FALSE	1	int[(D-24)/30]	0	1

1. Excludes master ID.

[For example, if the size of the user data is 53 bytes, the transaction is fragmented as follows:

a. One data (first, full) packet - 30 bytes (6 bytes master ID, and 24 bytes of user data)

c. One data (ensuing, not full) packet - 29 bytes of user data]

TLMC_M_054 - DATA - RETRANSMISSION OF MASTER PACKET - NO ACK RECEIVED

After transmitting a data packet, if no slave packet with the ACK bit set is received, that master data packet shall be retransmitted. Only after an slave packet with the ACK bit set is received, the next data packet shall be transmitted.

[The master data service uses the Stop and Wait algorithm.]

TLMC_S_055 - DATA - MASTER PACKETS WITH DUPLICATE SEQUENCE NUMBERS

If the master data request packet has the same sequence number as the previous master data request packet, the slave will toss the data packet and retransmit unacknowledged Memory Data packets, if any exist.

TLMC_S_056 - DATA - VALIDATION OF MASTER DATA REQUEST

The slave shall process the master data packet if it is valid per **Table 10 (MASTER PACKET VALIDATION).**

TLMC_S_057 - DATA - MASTER PACKETS INTO TRANSACTION

The master data packets are assembled into a transaction and are passed on to the slave's upper layer (application).

[The data indication to the slave Upper Layer will be generated by the slave data link layer, after it successfully receives and assembles all the data packets of a given transaction. The content of the Indication to the Upper layer is defined in **Table 32 (DATA INDICATION (MASTER/SLAVE))**]

TABLE 32: DATA INDICATION (MASTER/SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xA4 - Master Data indication 0x24 - Slave Data indication
Sub-type	N/A	1	The indication sub-type	0 - Memory data (Master/Slave) 1 - Unrequested data (Master) 2 - Waveform data (Master). See **Fig. 51 (Waveform indication sub-type byte)**(1),(2)
Size	N/A	2	Number of data bytes	1 to K1(3) for data and emergency.
Info	Data	N	Data received	The received transaction. Memory data includes the Upper Layer Header (ULH) as the first byte. See **5.2.11.3** for the slave's ULH byte and See **5.2.11.2** for the Master's ULH byte. Supplemental marker data and waveform data do not have a ULH byte.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

1. The master will copy a decimated version of the general sequence number into the most significant bits of the waveform sub-type. The master host can then detect any missing waveform indications. If the master host knows there is one packet/ indication, the sequence numbers should increment by "1". Similarly if the master host knows there are two packets/ indication, the sequence numbers should increment by "2" (0, 2, 4,...).
2. If there are 8 packets/transaction, a "false rest" can occur in the missed waveform indication logic as follows: master host receives waveform indication with sequence number 0, it subsequently misses sequence number 8, 16, 24, 32, 40, 48, 56, and 0. If it finally gets 8, it may not realize it missed any waveform indications. To mitigate this issue the master host could use a timer as well. If "too much" time goes by, a waveform has been missed (even if the sequence number advanced in an expected manner).
3. K1 is equal to 480 - sizeof(STATION-ID). (480 = 30 bytes/packet * 16 packets/transaction)

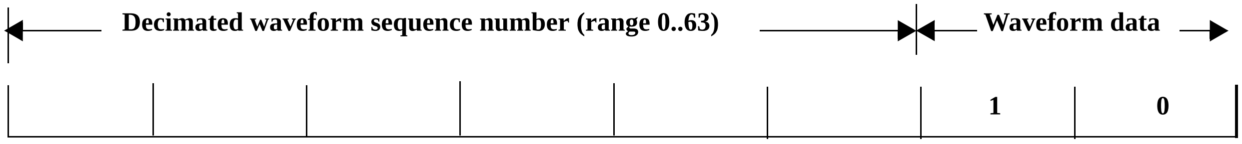


Figure 51: Waveform indication sub-type byte

The information in the control bytes of the data packets are used to reassemble the packets into the user data, and sequence numbers control the relative order of the data packets. The Last Fragment bit in the control bytes indicates the end of each transaction.

TLMC_M_058 - DATA - RESPONSE TO UPPER LAYER

When slave's upper layer response is completely received or a reply timeout occurs then a data service response to the master upper layer will be generated. No slave upper layer response is expected when the master transaction is not final. In this case the data service response is generated when all the master data packets have been acknowledged by the slave.

[The content of the response to the Upper Layer is defined in **Table 33 (MASTER DATA SERVICE RESPONSE)**.]

TABLE 33: MASTER DATA SERVICE RESPONSE

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x84 - Data service response

TABLE 33: MASTER DATA SERVICE RESPONSE

Sub-type	N/A	1	The service response sub-type	0 - Memory data
Size	N/A	2	The size of the info field	1
Info	Data service status	1	Status of the service request	0x01 - Request successful 0x02 - Request invalid in current state (state is communication)(1) 0x03 - Request invalid in current state (state is Recovery; device scanning)(2) 0x04 - Request invalid in current state (state is Recovery; device not scanning)(2) 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x09 - Request invalid in current state (state is standby) 0x0A - Parameter value error(3) 0x0C - Service cancelled(4) (interrupted by emergency) 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x11 - Reply timeout 0x12 - Invalid size (info size >506)(3) 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(5) 0x29 - Request invalid when max throughput transfer mode is enabled and power inhibit roll through is enabled (Telemetry M only)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. This status occurs when a local data service request (destination is master mozart). A local data service request is only valid in the standby state.
2. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration). Note that a reply timeout will result in a Data Link response status of 3 or 4 (invalid state, state is Recovery). This is done to aid the upper layer (application). A Data Link status of 9 (service duration timeout) would not provide any benefit.
3. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
4. This can occur if an emergency service interrupts “this” data service.
5. This status can occur if the master host attempts to send a data service request using service request buffer 1 (instead of buffer 0).

TLMM_M_297 - DATA - WAIT FOR INDICATION

The master will send a data indication to the Upper Layer when the data transaction completes.

- For the (master) Memory Data service, when all the data packets are acknowledge by the slave (and the master transaction was the message's final transaction) the master will wait for slave data packet(s). It will re-assemble these packets into complete transactions (indications). It will forward these indications (in-order) to the Upper Layer. It will cancel the reply timer if it forwards the final slave indication to the Upper Layer;
- For the (slave) Unrequested Data (or Waveform) service, the master will reassemble all the slave data packets into a complete indication and forward it to the Upper Layer.

[The content of the Indication to the Upper layer is defined in **Table 32 (DATA INDICATION (MASTER/SLAVE))**.

If the reply timer expires while waiting for the final indication an (unsuccessful) data response will be sent to the Upper Layer.]

TLMM_M_298 - DATA - RECOVERY

The master will "suspend" the data service request if channel recovery is triggered. If channel recovery is successful (and the reply timer is still active), the master will retry the data service request.

TLMM_M_299 - DATA - REPLY TIMEOUT

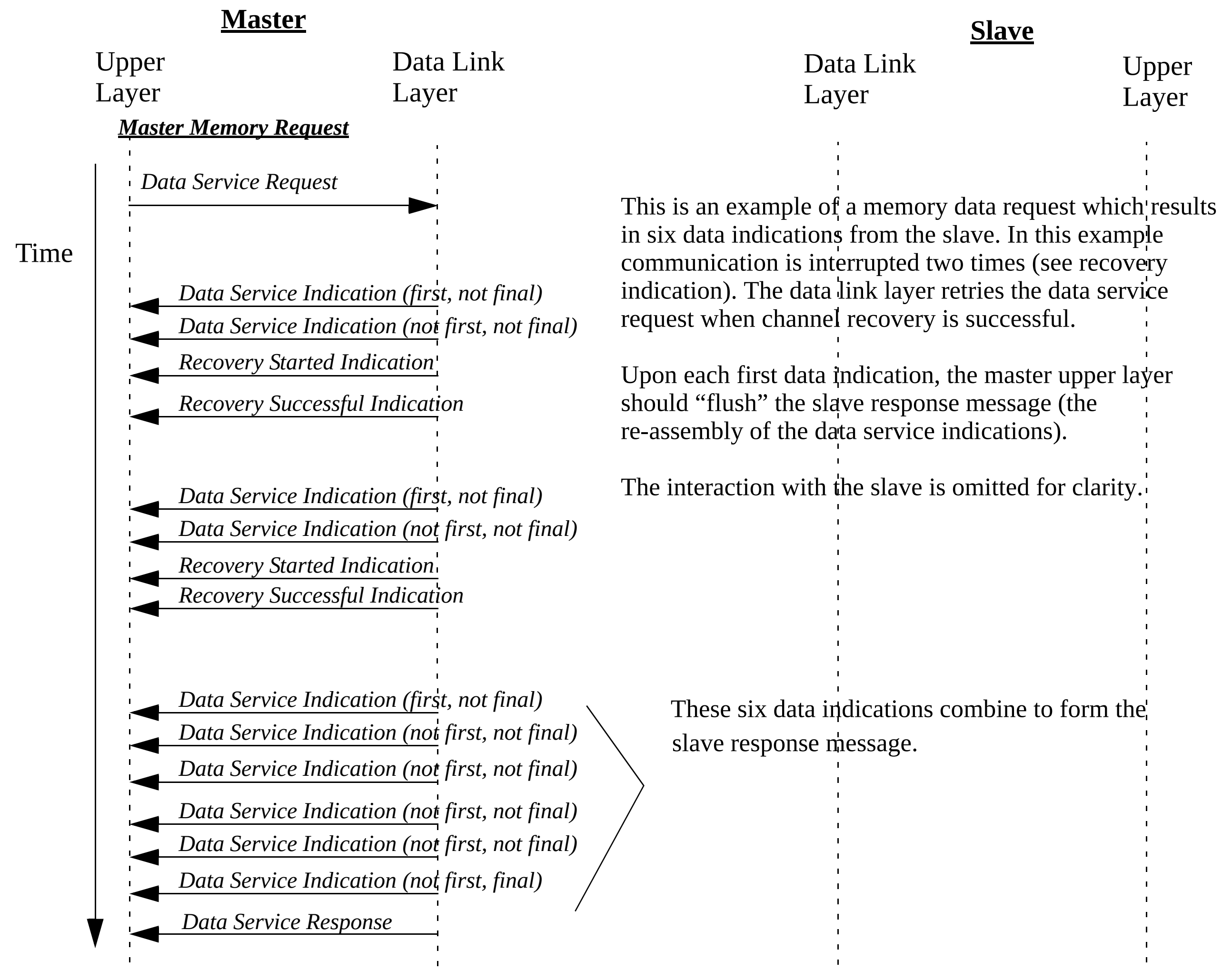
The master will send an (unsuccessful) data service response to the upper layer if the reply timer expires during the data service request.

[The Upper Layer should consider the data "message" service unsuccessful if it receives a failed data service response.]

TLMM_M_300 - DATA - FIRST INDICATION

The master will set the first indication bit in the slave upper layer header byte before it forwards the first indication to the upper layer.

[Note: upper layer may receive multiple “first” data indications. This can occur during retry. The upper layer should restart the re-assembly of the response message each time it receives a “first” data indication.]



This is an example of a memory data request which results in six data indications from the slave. In this example communication is interrupted two times (see recovery indication). The data link layer retries the data service request when channel recovery is successful.

Upon each first data indication, the master upper layer should “flush” the slave response message (the re-assembly of the data service indications).

The interaction with the slave is omitted for clarity.

These six data indications combine to form the slave response message.

Figure 52: Memory Data - master data link layer and upper layer interaction

5.2.11.5 SLAVE DATA

The slave uses the Data service (each time) to transmit Memory Data, Supplemental Marker Data, and Waveform Data. The ARQ algorithm is used only for Memory Data retransmission. The slave data is prioritized as follows: Waveform Data, Supplemental Marker Data, Memory Data. For more details see section **7.5 Firmware Interaction between host and Telemetry**.

TLMC_S_059 - DATA - SLAVE UPPER LAYER PARAMETERS

When invoking the Data service request, the slave Upper Layer will provide information as specified in **Table 30 (DATA SERVICE REQUEST (MASTER OR SLAVE))**.

TLMC_S_058 - DATA - RESPONSE TO UPPER LAYER

When Data request is received, the Data Link Layer will validate the Data service request and will send the response to the upper layer.

[The content of the response to the Upper Layer is defined in **Table 34 (SLAVE DATA SERVICE RESPONSE).**]

TABLE 34: SLAVE DATA SERVICE RESPONSE

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x04 - Data service response
Sub-type	N/A	1	The service response sub-type	0 - Memory data 1 - Unrequested data 2 - Waveform data(1)
Size	N/A	2	The size of the info field	1
Info	Data service status	1	Status of the service request	0X01 - Successful 0X15 - Request in invalid state(2) 0x0C - Cancelled due to new data request downlink recvd(3) 0x0D - Cancelled due to channel recovery started, close packet received, or initialization request is received(4) 0X10 - Invalid CRC
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. No response will be generated - only the corresponding service request interrupt will be cleared.
2. See **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)** for the invalid states.
3. Only applicable to Sub-type = Memory data.
4. While in Disable state, if Data request interrupt was pending (a valid data service request had been previously generated by host), then 0x0D will occur if initialization request was received.

TLMC_S_060 - DATA - SLAVE TRANSMISSION OF DATA PACKETS

When the Data service request is invoked the data link layer will fragment the user data into slave packet(s) which then will be transmitted. The first fragment of the user data will contain the slave ID.

Note that if bit “Enable WF in Recovery” (see **Table 124 (SLAVE PARAMETERS FROM HOST)**) is set to “1”, Data service requests (sub-type Waveform data) received during recovery state will be transmitted when the channel recovery is successful.

[The data link layer will fragment the user data and format them into slave packets, add slave ID in the first fragment, format the packet control byte according the Data type specified, transmit the packets via Synchronous Mode operation.]

Slave Mozart will always save the last two Waveform data requests (one from service data request 3 and one from service data request 4) as they are received during Communication state, and recovery state if bit “Enable WF in Recovery” (see **Table 124 (SLAVE PARAMETERS FROM HOST)**) is set to “1”. Slave will start to transmit the waveform data during communication state. Slave host should select the slave transmit and receive data rate properly based on the amount of the real time data they have. If more waveform data requests are received before the saved Waveform data requests have been transmitted, the saved Waveform data requests will be lost.

All the slave data packets of the same external data service request will have the same transaction number, but each packet will have its own fragment number.

TLMC_S_061 - DATA - SLAVE DATA PACKET FORMAT(S)

The slave will use one or more of the slave packet(s) formats shown in **Fig. 53 (Slave Data Packet(s) Format)**.

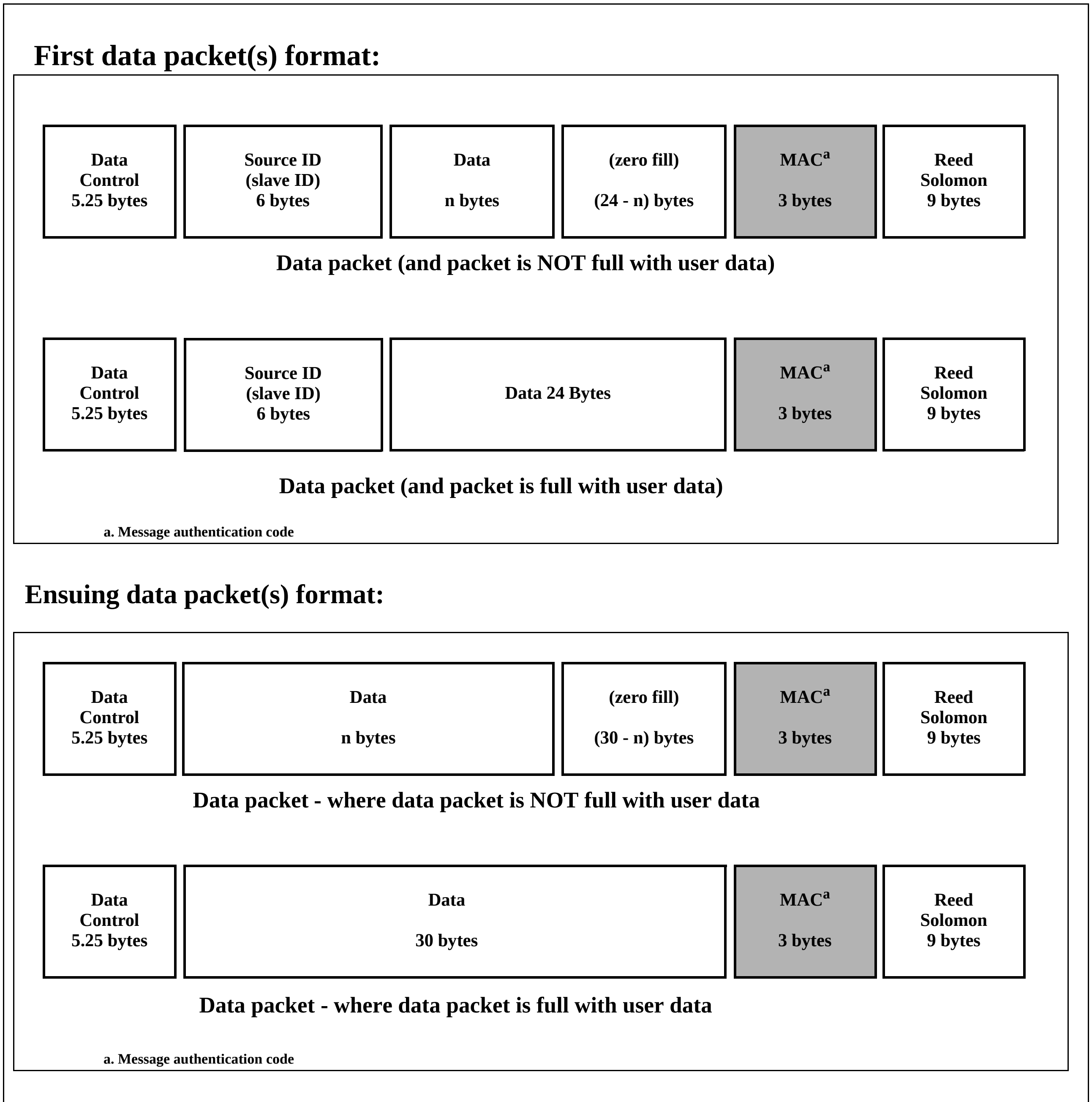


Figure 53: Slave Data Packet(s) Format

TLMC_S_062 - DATA - PACKETIZATION OF SLAVE DATA

The slave will packetize the user data into slave packets per **Table 35 (DATA PACKET RECEPTION VALIDATION).**

[Note: If a data (ensuing, not full) packet is needed, it must be the last packet of the transaction.]

TABLE 35: DATA PACKET RECEPTION VALIDATION

INPUTS		SLAVE PACKETS			
DATA SIZE (D)	(D-24) MOD 30 EQUAL 0	DATA (FIRST, FULL)	DATA (ENSUING, FULL)	DATA (FIRST, NOT FULL)	DATA (ENSUING, NOT FULL)
D<24 bytes	-	0	0	1	0
D>=24 bytes	True	1	int[(D-24)/30]	0	0
D>=24 bytes	False	1	int[(D-24)/30]	0	1

TLMC_M_063 - DATA - RECEIPT OF AN INVALID SLAVE PACKET

The master shall use the ARQ algorithm to acknowledge or request the retransmission of invalid or not received Memory Data packets See **Table 11 (SLAVE PACKET VALIDATION)**.

[A packet with an invalid sequence number is ignored.]

This slave response received may be an unrequested packet or the result of a previous master request. After all the packets of one transaction are received correctly, the Data Link Layer shall assemble all the packets in order and return the data via the Data service indication to the Upper Layer.

5.2.12 DIAGNOSTIC

The Diagnostic service is intended to be used during development to analyze the performance of the system. It is not intended for Medtronic's customers.

There are three types of diagnostic services:

- Remote Diagnostic - this service returns the most recent mapping information, the current slave information, and the current master Data Link information.
- Local Diagnostic - this service returns the most recent mapping information, the most recent slave information, and the current master Data Link information.
- Mapping Diagnostic - this service returns the current mapping information.

Fig. 54 (Remote Diagnostic Request) illustrates the Remote Diagnostic request.

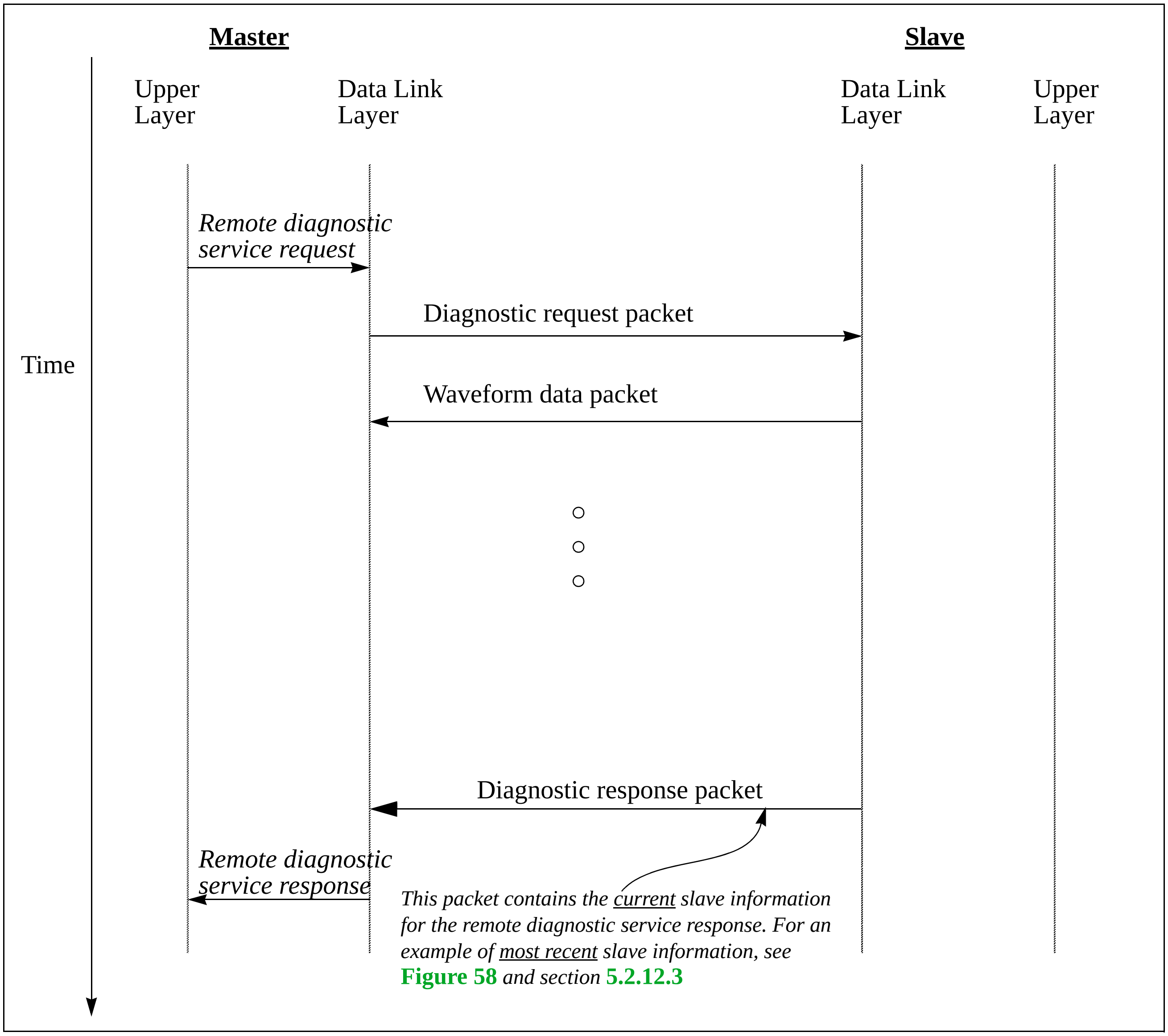


Figure 54: Remote Diagnostic Request

5.2.12.1 REQUIREMENTS

TLMC_M_066 - DIAGNOSTIC SERVICE REQUEST FROM UPPER LAYER

When the Upper Layer invokes the diagnostic service it will provide information as specified in **Table 36 (DIAGNOSTIC SERVICE REQUEST (MASTER))**.

TABLE 36: DIAGNOSTIC SERVICE REQUEST (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x89 - Diagnostic service request
Sub-type	N/A	1	The service request sub-type	0 - Remote, 1 - Local, 2 - Mapping
Size	N/A	2	The size of the info field	2
Info(1)	Address(2)	2	Remote address for diagnostic data	2 byte address Address value: 0x0F00(3)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. The diagnostic service has one additional parameters which control its behavior: diagnostic service duration. See **Table 123 (Master parameters from host)**.
2. This applies to Remote only.
3. The address should be 0x0f00. The slave portion of the diagnostic response is then defined by **Table 106 (DIAGNOSTIC SERVICE RESPONSE — IMPLANT DATA)**. If the address is not 0x0F00, the slave will set the invalid address bit in the Slave diagnostic status byte, see TLMC_S_069 - REMOTE DIAGNOSTIC - STATUS BYTE FORMAT for detail.

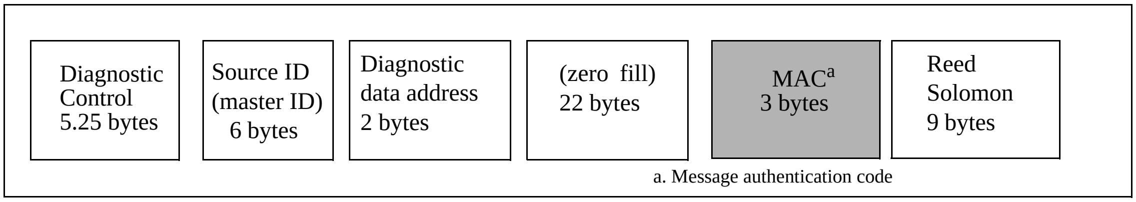
5.2.12.2 REMOTE DIAGNOSTIC

TLMC_M_067 - REMOTE DIAGNOSTIC - MASTER PACKET FORMAT

When a remote diagnostic service request is invoked by the Upper Layer, the master will:

1. Start the service duration timer.
2. Transmit a diagnostic packet as defined by **Fig. 55 (Master diagnostic (request) packet)**. The diagnostic packet sub-type bit (in master control byte 2) should be set to one (request packet, not filler packet).

[Send a “first request packet sent” event indication to the Upper Layer (see **Table 23**).]

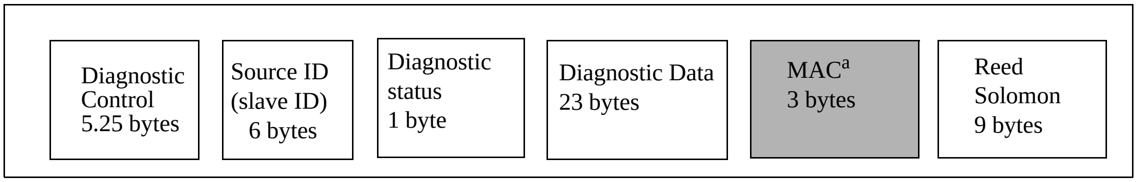


a. Message authentication code

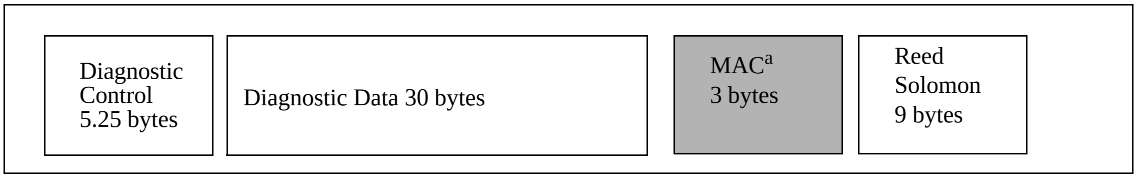
Figure 55: Master diagnostic (request) packet

TLMC_S_068 - REMOTE DIAGNOSTIC - SLAVE PACKET FORMAT

The slave will respond to a valid Diagnostic request packet with one or more diagnostic response packet. The diagnostic packet sub-type bit (in the diagnostic status byte) should be set to one (response packet, not filler packet). The format of the slave Diagnostic packet is indicated in **Fig. 56 (Slave diagnostic packet)**.



First slave diagnostic packet



Ensuing slave diagnostic packet

a. Message authentication code

Figure 56: Slave diagnostic packet

TLMC_S_069 - REMOTE DIAGNOSTIC - STATUS BYTE FORMAT

Upon receipt of a diagnostic request packet, the slave will update the diagnostic status byte. If the diagnostic data address (in the request packet) is valid, the slave will set the request address bit to valid, and the diagnostic sub-type bit to response. If the diagnostic data address (in the request packet) is invalid, the slave will set the request address bit to invalid, and the diagnostic sub-type bit to filler. The format of the diagnostic status byte is shown in **Fig. 57 (Slave diagnostic status byte)**

.

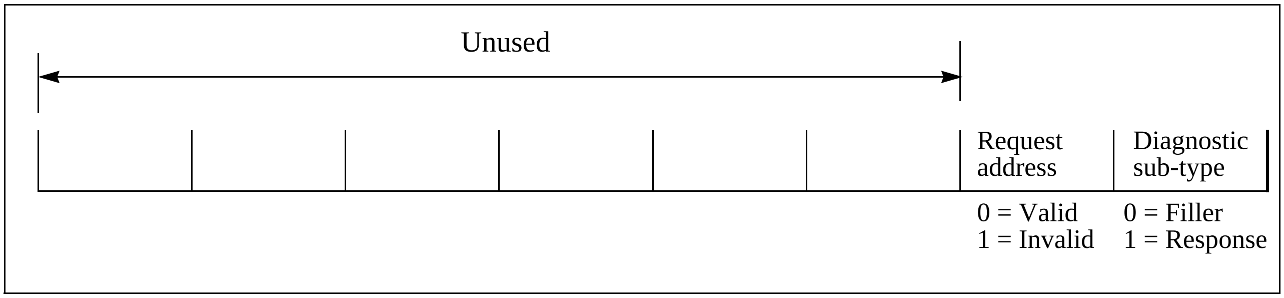


Figure 57: Slave diagnostic status byte

TLMC_S_070 - REMOTE DIAGNOSTIC - SLAVE RESPONSE PACKET

The diagnostic response will be transmitted when there is available room in the frame, based on the slave packet priorities. The diagnostic response is cancelled if Channel Recovery is triggered. The contents of this transaction are shown in **Table 106 (DIAGNOSTIC SERVICE RESPONSE — IMPLANT DATA)**.

A diagnostic response transaction will utilize these bytes as follows:

1. Diagnostic data byte 1 = content of diagnostic (request) address
2. Diagnostic data byte 2 = content of diagnostic (request) address+1
3. .
4. .
5. Diagnostic data byte n = content of diagnostic (request) address + (n-1)

[A diagnostic transaction shall be limited in size to (240 bytes - size of (slave station ID))]

TLMC_M_071 - REMOTE DIAGNOSTIC - MASTER REQUEST RETRANSMISSION

The master will continue to transmit the diagnostic request packet until it is ACKed, service duration timeout, channel recovery, or interrupted by emergency. Once ACKed, the master will wait for the slave diagnostic (response) transaction. The master will transmit filler packets while waiting for the slave transaction. Transmission of diagnostic filler packets continues until successful, service duration timeout, channel recovery, or interrupted by emergency.

TLMC_M_072 - REMOTE DIAGNOSTIC - RESPONSE TO UPPER LAYER

A Remote Diagnostic service response to the Upper Layer will be generated at the completion of the Remote Diagnostic service.

[The content of the response is shown in **Table 37 (REMOTE DIAGNOSTIC SERVICE RESPONSE (MASTER))**. Note that a service response status of 0x26 occurs if the service is interrupted by a PLL lock error (and a subsequent successful trim).]

TABLE 37: REMOTE DIAGNOSTIC SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x89 - Diagnostic service response
Sub-type	N/A	1	The service response sub-type	0 - Remote diagnostic
Size	N/A	2	The size of the info field	1 to 480
Info	Diagnostic service status	1	Status of the service request	0x01 - Request successful 0x03 - Request invalid in current state (state is Recovery; slave scanning)(1) 0x04 - Request invalid in current state (state is Recovery; slave not scanning)(1) 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x09 - Request invalid in current state (state is Standby) 0x0A - Parameter value error(2) 0x0B - Service duration timeout 0x0C - Service cancelled (interrupted by emergency) 0x0D - Service cancelled (interrupted by recovery) 0x0E - Invalid diagnostic address(3) 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)(2) 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(4)
	Mapping info	20	Most recent mapping information	For further details, see **7.0 Appendices**.
	Master info	N2	Current master Data Link information	For further details, see **7.0 Appendices**.
	Slave info	N1	Current slave information, (5)	For further details, see **7.0 Appendices**.

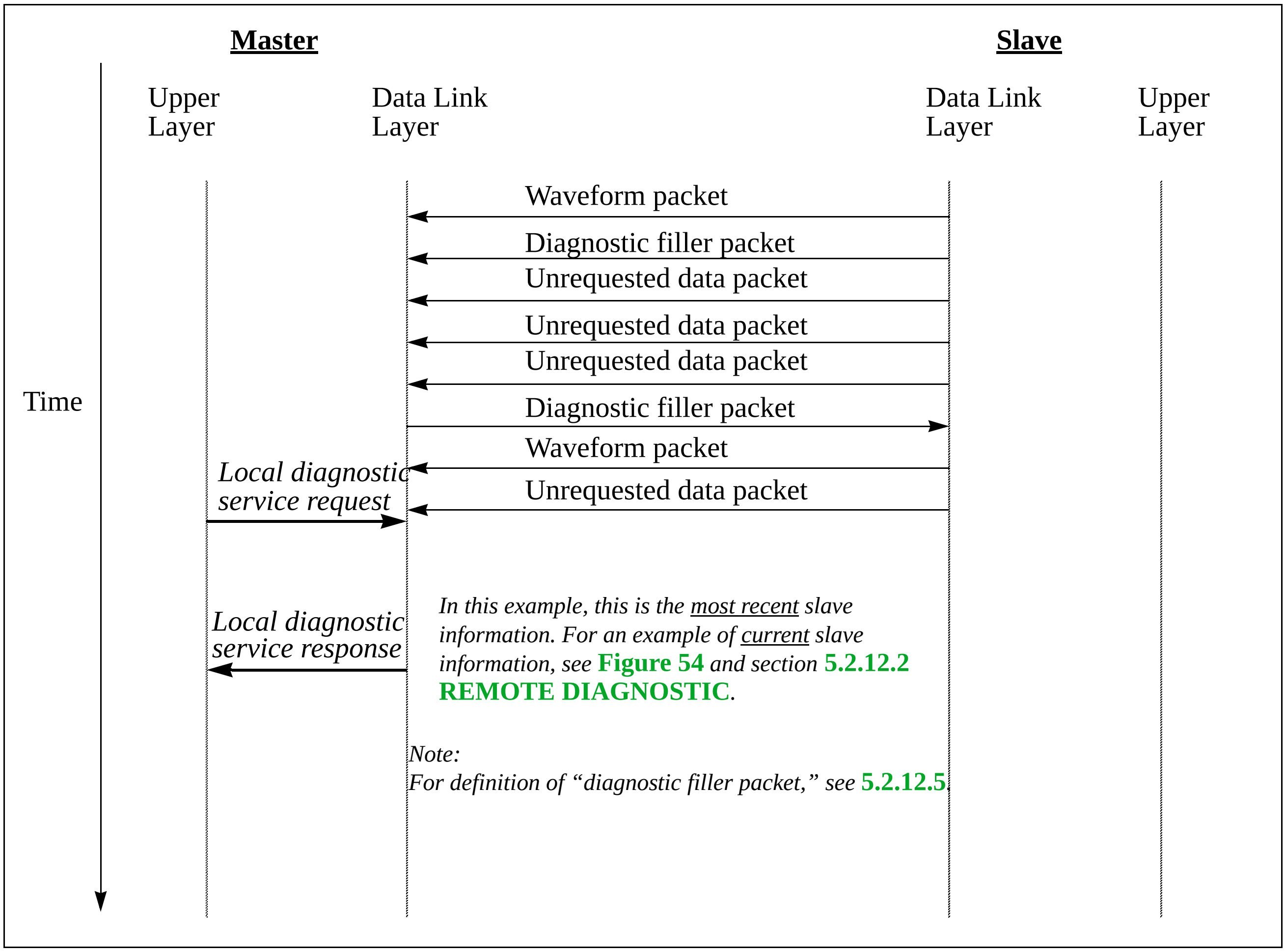
TABLE 37: REMOTE DIAGNOSTIC SERVICE RESPONSE (MASTER)

CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration).
2. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
3. This occurs when the slave diagnostic response packet indicates an invalid address in the status byte.
4. This status can occur if the master host attempts to send a diagnostic service request using service request buffer 1 (instead of buffer 0).
5. The current slave information is a copy of the slave diagnostic response packet (content field). Note that this packet has just been received from the slave.

5.2.12.3 LOCAL DIAGNOSTIC

When the Upper Layer invokes the Local Diagnostic service, the Data Link Layer will respond immediately. See **Fig. 58 (Local diagnostic sequence)**.



In this example, this is the most recent slave information. For an example of current slave information, see **Figure 54** *and section* **5.2.12.2 REMOTE DIAGNOSTIC**.

Note:
For definition of "diagnostic filler packet," see **5.2.12.5**.

Figure 58: Local diagnostic sequence

TLMC_M_073 - LOCAL DIAGNOSTIC - RESPONSE TO UPPER LAYER

The local diagnostic service response to the Upper Layer will be generated immediately.

[The content of the response is shown in **Table 38 (LOCAL DIAGNOSTIC SERVICE RESPONSE (MASTER))**.]

TABLE 38: LOCAL DIAGNOSTIC SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x89 - Diagnostic service response
Sub-type	N/A	1	The service response sub-type	1 - Local diagnostic
Size	N/A	2	The size of the info field	1 to 480 bytes
Info	Diagnostic service status	1	Status of the service request	0x01 - Request successful 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x0A - Parameter value error(1) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)(1) 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(2)
	Mapping info	20	Most recent mapping information(3)	For further details, see **7.0 Appendices**.
	Master info	N2	Current master Data Link information	For further details, see **7.0 Appendices**.
	Slave info	N1	Current slave information,(4)	For further details, see **7.0 Appendices**.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
2. This status can occur if the master host attempts to send a diagnostic service request using service request buffer 1 (instead of buffer 0).
3. This is part of the “attached data”. It is included only if the service is successful.
4. The current slave information is a copy of the slave diagnostic response packet (content field). Note that this packet has just been received from the slave.

When a local diagnostic service is invoked by the slave’s Upper Layer, the Data Link Layer will provide the stored diagnostic data.

5.2.12.4 MAPPING DIAGNOSTIC

When the master’s Upper Layer invokes the Mapping Diagnostic service, the Data Link Layer will initiate diagnostic mapping. When complete, the Data Link Layer responds back to the Upper Layer.

Diagnostic mapping consists of a “long” screen (but no evaluate phase). A “short” screen and an evaluate phase is used in channel (selection) mapping.

Diagnostic mapping can be used to characterize the noise in a room. See **Fig. 59 (Mapping diagnostic sequence).**

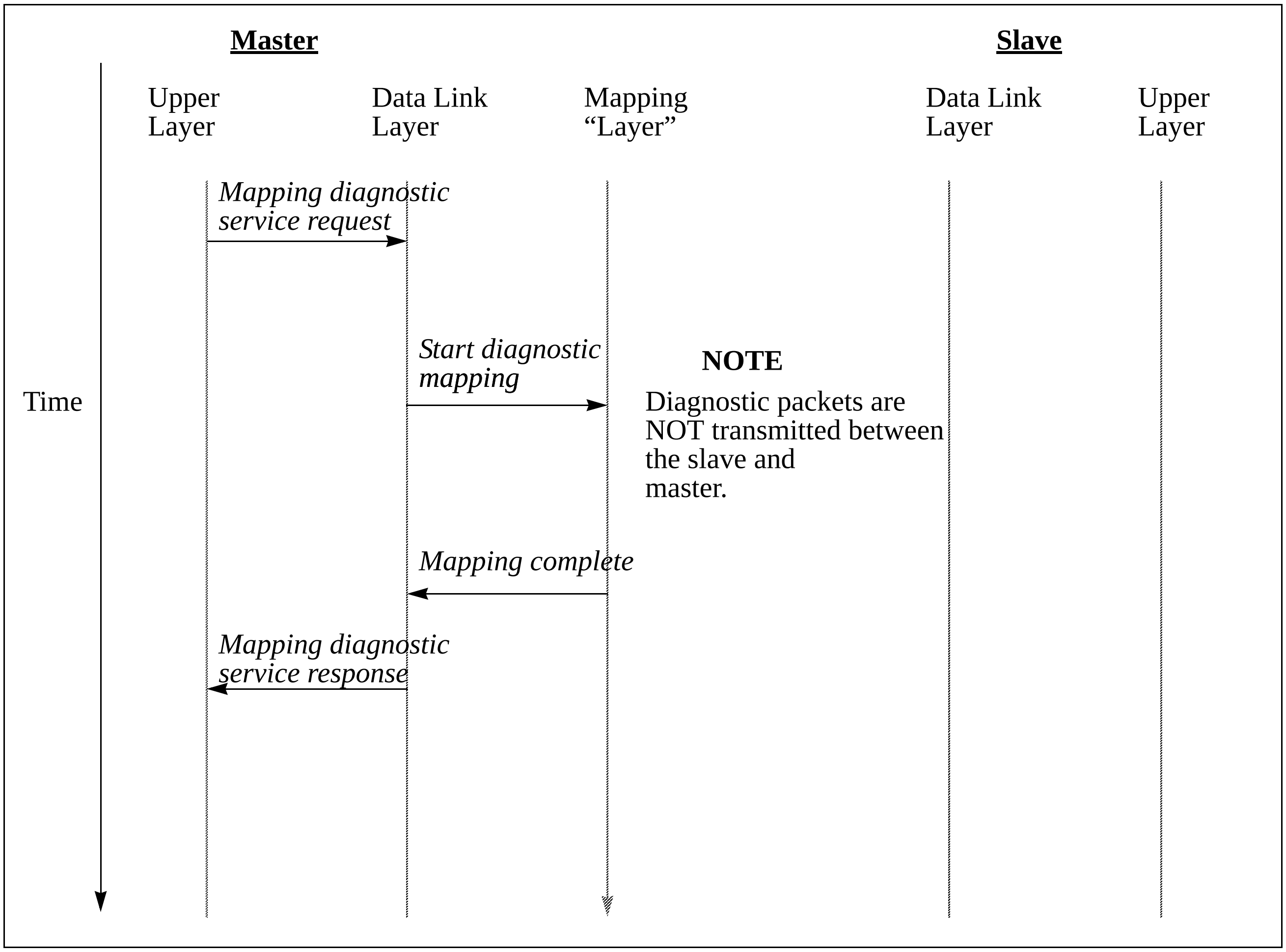


Figure 59: Mapping diagnostic sequence

TLMC_M_074 - MAPPING DIAGNOSTIC - RESPONSE TO UPPER LAYER

When the master’s Upper Layer invokes the Mapping Diagnostic service, the Data Link Layer will initiate diagnostic mapping. When complete, the Data Link Layer responds back to the Upper Layer.

[The content of the response is shown in **Table 39 (MAPPING DIAGNOSTIC SERVICE RESPONSE (MASTER))**. Note that a service response status of 0x26 occurs if the service is interrupted by a PLL lock error (and a subsequent successful trim).]

TABLE 39: MAPPING DIAGNOSTIC SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x89 - Diagnostic service response
Sub-type	N/A	1	The service response sub-type	2 - Mapping diagnostic
Size	N/A	2	The size of the info field	21 bytes if successful, 1 byte if unsuccessful
Info	Diagnostic service status	1	Status of the service request	0x01 - Request successful 0x02 - Request invalid in current state (state is Comm) 0x03 - Request invalid in current state (state is Recovery; slave scanning)(1) 0x04 - Request invalid in current state (state is Recovery; slave not scanning)(1) 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x0A - Parameter value error(2) 0x0B - Service duration timeout 0x0C - Service cancelled (interrupted by emergency) 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)(2) 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(3)
	Mapping info	20	Most recent mapping information	For further details, see **7.0 Appendices**.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration).
2. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.

3. This status can occur if the master host attempts to send a diagnostic service request using service request buffer 1 (instead of buffer 0).

5.2.12.5 DIAGNOSTIC FILLER PACKET

A diagnostic (filler) packet is used as a “space filler” when there is no active External Service in progress.

A diagnostic request packet and a diagnostic filler packet are differentiated by a single sub-type bit in the master control byte header. The diagnostic sub-type bit can be set or cleared by the master. The setting of the diagnostic sub-type bit to a one is an indication of a diagnostic request to the slave. The clearing of the bit to a zero is an indication to the slave that it is a filler packet.

TLMC_M_075 - DIAGNOSTIC FILLER - MASTER FILLER PACKETS

When the master does not have any other packets to transmit it will transmit diagnostic filler packet(s) as follows:

1. The diagnostic sub-type bit (in master control byte 2) will be cleared to a zero.
2. The filler packet will be transmitted to the slave during the absence of user data (and an ARQ packet).
3. The filler packet format will be as indicated in **Fig. 60 (Master diagnostic (filler) packet)**.

Diagnostic Control 5.25 bytes	Source ID (master ID) 6 bytes	(zero fill) 24 bytes	MAC[a] 3 bytes	Reed Solomon 9 bytes

a. Message authentication code

Figure 60: Master diagnostic (filler) packet

A filler packet will be transmitted by the slave when user Memory Data is not being transmitted. A filler packet will not be retransmitted as part of the ARQ algorithm. Each filler packet will contain data from a default (pre-defined) address. The size of the filler packets is limited by the size of a slave packet.

TLMC_S_076 - DIAGNOSTIC FILLER - SLAVE FILLER PACKETS

The slave will transmit filler transactions when there is no Data Packet to transmit.
See **Fig. 56 (Slave diagnostic packet)** for the packet format. The diagnostic sub-type bit (in the status byte) will be set to a zero (filler packet, not response packet). The contents of this transaction are shown in **Table 106 (DIAGNOSTIC SERVICE RESPONSE — IMPLANT DATA)**.

A diagnostic response transaction will utilize these bytes as follows:

1. Diagnostic data byte 1 = content of diagnostic (default) address
2. Diagnostic data byte 2 = content of diagnostic (default) address+1
.
.
n. Diagnostic data byte n = content of diagnostic (default) address + (n-1)

[The slave does not respond with a filler packet each time it receives a filler packet from the master. The diagnostic filler packets are not transmitted during the low energy transfer mode.]

5.2.13 EMERGENCY

Emergency can only be transmitted or received during a bidirectional communication session. Telemetry M uses a data packet (with the Reset bit set) to transmit the emergency request. The master will send Emergency only if it is in the Comm State. The slave device will process Emergency only if it is in the Comm State (already).

A separate Emergency service allows the Data Link Layer to enforce serialization of service requests. Without an Emergency service, Data service #2 could interrupt data service #1 (provided data service #2 was for Emergency). The Data Link Layer would have to “look at the content of the Data service” (to see if it is OK to interrupt data service #1).
[Another benefit is that it makes the upper layer design simpler.]

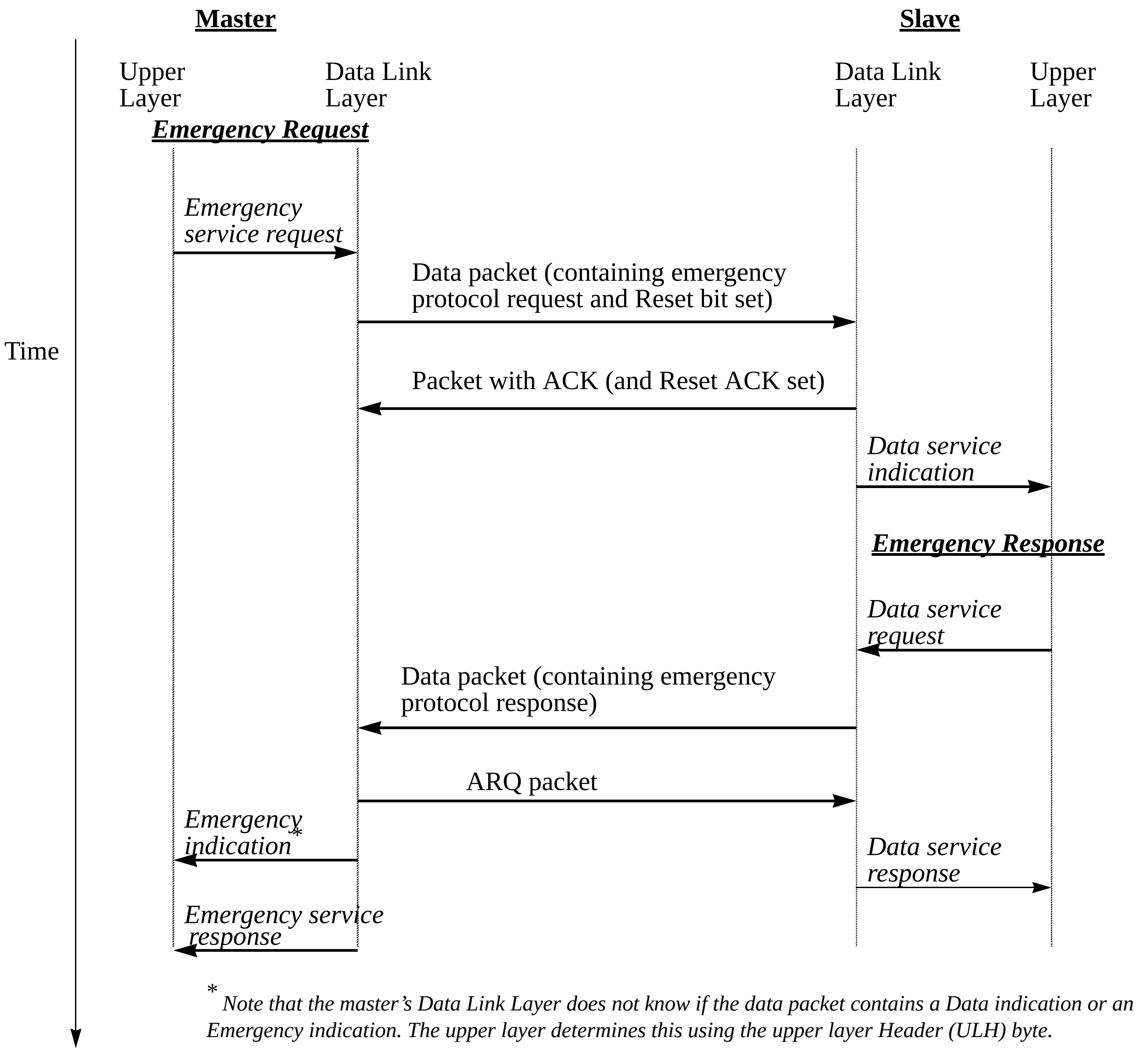


Figure 61: Emergency master/slave sequence diagram

5.2.13.1 REQUIREMENTS

Each Emergency service request is considered a *transaction* (defined in **5.2.11.1**).

5.2.13.2 MASTER

Service requests from the Upper Layer must be serialized. Emergency is an exception. It can be received while processing another request (most likely a data service request). If so, it will stop the service request in progress (so it can do “this” emergency request). Note that Emergency will never interrupt Emergency (consecutive Emergency button “presses” are separated by at least two seconds; an Emergency response must be sent within two seconds).

TLMC_M_077 - EMERGENCY - UPPER LAYER PARAMETERS

When invoking the Emergency service request, the master Upper Layer will provide information as specified in **Table 40 (EMERGENCY SERVICE REQUEST (MASTER))**.

TABLE 40: EMERGENCY SERVICE REQUEST (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x86 - Emergency service request
Sub-type	N/A	1	The service request sub-type	N/A
Size	N/A	2	The size of the info field	1 to MAX_DATA_TRANS_SIZE (480 - size of STATION_ID) 480 - 30 bytes/packet * 16 packets/transaction
Info(1)	Data	N	Message to be transmitted(2)	The data to be transmitted. This includes the Upper layer header as the first byte. See **5.2.11.2 MASTER’S UPPER LAYER HEADER (ULH) Byte Format**.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. The emergency service has one additional parameters which control its behavior: reply duration. See **Table 123 (Master parameters from host)**.
2. For further details, see requirement DEVICE PROG_INST EMERGENCY REQUEST DEFINITION in Medtronic’s Document 217420 (System B Telemetry Communication Protocol Specification)

TLMC_M_078 - EMERGENCY - TRANSMISSION OF EMERGENCY REQUEST

When the Emergency service request is invoked, the master shall:

1. Stop the service request in progress (if there is one in progress)
2. Generate a “cancelled” response to the Upper Layer for the cancelled service (if any)
3. Start the reply duration timer
4. Format the user data into a data packet
5. Set the Reset bit to 1 in control byte1 of the data packet
6. Set the sequence number to 0 in control byte2 of the data packet
7. Transmit the packet to the slave

The master retransmits the emergency protocol request until successful (or until timeout occurs); see data requirement(s) in **5.2.11.4**.

The slave sets the ACK bit in the next slave packet(s) upon receiving the emergency protocol request; see **5.2.11.4**.

The slave sets the Reset ACK bit in the next slave packet(s) upon receiving the data packet (with the Reset bit set); see reset requirement(s) in **5.3.5**.

The master resets several Data Link parameters when it receives the Reset ACK bit set; see **5.3.5**.

The slave sends a data indication to the Upper Layer after it receives the Emergency protocol request. See **5.2.11.4**.

TLMC_M_079 - EMERGENCY - MASTER REQUEST NOT ACKNOWLEDGED

The master Emergency service request is considered not successful if the Emergency request is not acknowledged by the slave within the reply duration.

TLMC_M_081 - EMERGENCY - STATUS RESPONSE TO UPPER LAYER

An Emergency service response to the master’s Upper Layer will be generated. When the slave’s upper layer response is completely received or a reply timeout occurs.

[The content of the response to the Upper Layer is defined in **Table 41 (EMERGENCY SERVICE RESPONSE (MASTER))**.]

TABLE 41: EMERGENCY SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x86 - Emergency service response
Sub-type	N/A	1	The service response sub-type	N/A
Size	N/A	2	The size of the info field	1 byte
Info	Emergency service status	1	Status of the service request	0x01 - Request successful 0x03 - Request invalid in current state (state is Recovery; slave scanning)(1) 0x04 - Request invalid in current state (state is Recovery; slave not scanning)**(1)** 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x09 - Request invalid in current state (state is standby) 0x0A - Parameter out of range(2) 0x0F - Request non-serial (emergency already in progress) 0x10 - Invalid CRC 0x11 - Reply timeout 0x12 - Invalid size (info size >506)**(2)** 0x29 - Request invalid when max throughput transfer mode is enabled and power inhibit roll through is enabled (Telemetry M Only)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The device is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration). Note that a reply timeout will result in a Data Link response status of 3 or 4 (invalid state, state is Recovery). This is done to aid the upper layer (application). A Data Link status of 9 (service duration timeout) would not provide any benefit.
2. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used. This status will also be used if the final bit of the upper layer header byte is not set (multi-transaction emergency requests are not allowed).

The slave will invoke the Data service request to transmit the Emergency protocol response (receive ready or reject); see slave data requirement(s) in **5.2.11.5**.

The slave will use a data packet to execute the Data service; see **5.2.11.5**.

TLMM_M_301 - EMERGENCY - WAIT FOR INDICATION

The master will wait for slave packets when all emergency request packets are acknowledged by the slave (and the master transaction was the message's final transaction). It will re-assemble these packets into complete transactions (indications). It will forward these indications (in-order) to the Upper Layer. It will cancel the reply timer if it forwards the final slave indication to the Upper Layer.

[The content of the Indication to the Upper layer is defined in **Table 42 (EMERGENCY INDICATION (MASTER))**.]

TLMM_M_302 - EMERGENCY - RECOVERY

The master will "suspend" the emergency service request - if channel recovery is triggered. If channel recovery is successful (and the reply timer is still active), the master will retry the emergency service request.

TLMM_M_303 - EMERGENCY - REPLY TIMEOUT

The master will send an (unsuccessful) emergency service response to the upper layer if the reply timer expires during the emergency service request.

[The Upper Layer should consider the data "message" service unsuccessful if it receives a failed emergency service response.]

The master will send an Data indication to the Upper Layer when it receives the Emergency protocol response. Note that both Memory Data and Emergency are sent (to the Upper Layer) in a Data indication. The Upper Layer uses the Upper Layer header (ULH) byte to differentiate between Memory Data and Emergency; see 5.2.11.5.]

TABLE 42: EMERGENCY INDICATION (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xAC - Master emergency indication
Sub-type	N/A	1	The indication sub-type	n/a
Size	N/A	2	Number of data bytes	1 to K1(1)
Info	Data	N		The received transaction. Memory data includes the Upper Layer Header (ULH) as the first byte. See 5.2.11.3 for the slave's ULH byte. Supplemental marker data and waveform data do not have a ULH byte.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

1. K1 is equal to 480 - sizeof(STATION-ID). (480 = 30 bytes/packet * 16 packets/transaction)

TLMC_M_304 - EMERGENCY DURING CLOSE

The data link layer will send an unsuccessful emergency service response if the emergency service request is received while processing a close service request. The emergency service response status is Request invalid in current state (state is recovery; slave scanning).

TLMC_M_305 - EMERGENCY - SECOND EMERGENCY REQUEST WHILE PROCESSING FIRST

The master will do the following if it receives an Emergency service request (if an emergency service request is already in progress):

1. Send the (unsuccessful) Emergency service response to the upper layer.

[The response status will be "Emergency already in progress".]

5.2.14 RAW MODE (*DEVELOPMENT USE ONLY*)

This service provides a mechanism for sending and receiving data directly, without mediation by the lower layers. It is intended to be used only in the development and testing environments. This service is divided into a single type:

- **Master Packet Attached** — This allows the user to transmit a raw packet to the slave.

  Note: the user can set the control bytes as well as the information field. For example, the user can transmit a diagnostic filler packet with a bad master ID.

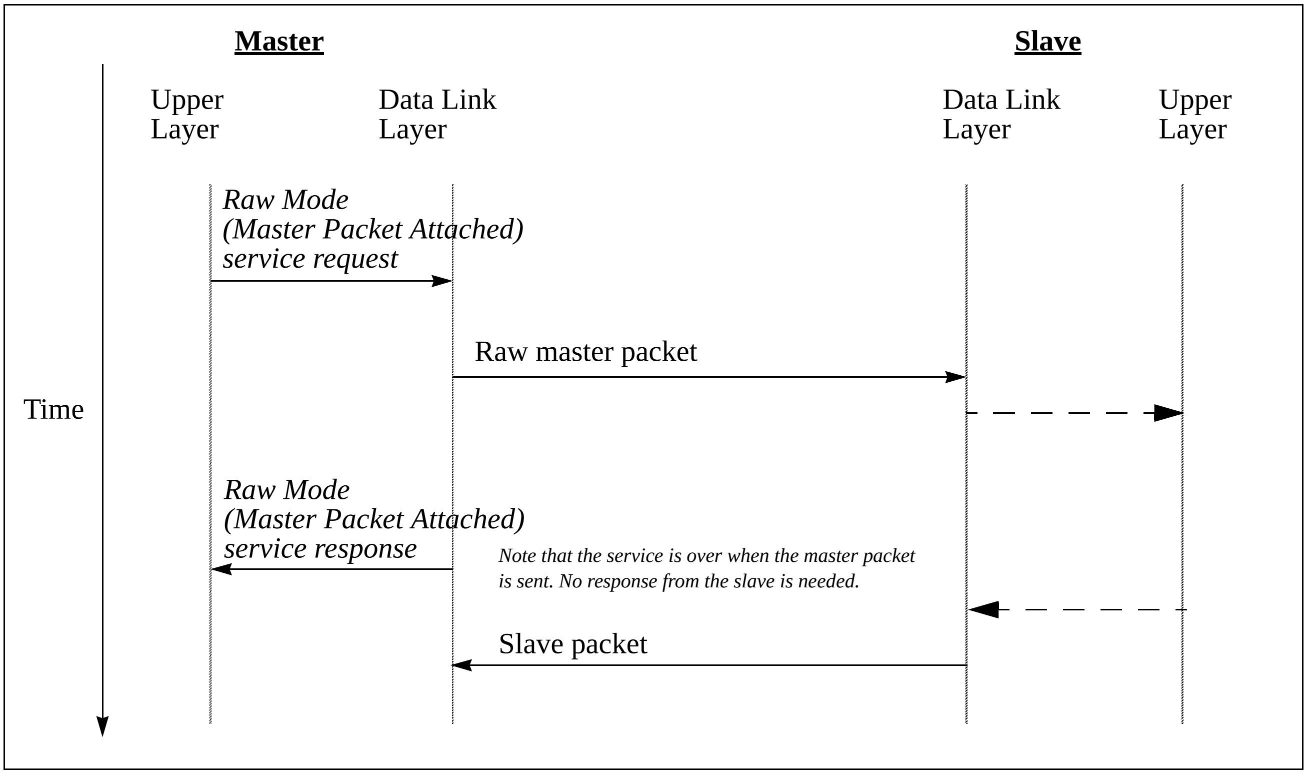


Figure 62: Raw Mode (Master Packet Attached) Sequence Diagram

5.2.14.1 REQUIREMENTS

TLMC_M_082 - RAW MODE - UPPER LAYER RAW MODE PARAMETERS

When the Upper Layer invokes the Raw Mode service it will provide information as specified in **Table 43 (RAW MODE SERVICE REQUEST (MASTER))**.

TABLE 43: RAW MODE SERVICE REQUEST (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x8B - Raw mode service request
Sub-type	N/A	1	The service request sub-type	N/A
Size	N/A	2	The size of the info field	36 bytes
Info(1)	Data	36	Packet to be transmitted	Raw master packet to be transmitted including the control bytes.
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. The emergency service has one additional parameters which control its behavior: reply duration. See **Table 129 (PARAMETERS FROM EEPROM)**.

5.2.14.2 RAW MODE (MASTER PACKET ATTACHED)

TLMC_M_083 - RAW MODE - MASTER PACKET ATTACHED OPERATION

When a valid Raw Mode (Master Packet Attached) service is invoked, the master shall:

1. Start the service duration timer (comes from **Table 129 (PARAMETERS FROM EEPROM)**)
2. Transmit the raw packet
3. Send the Raw Mode service response to the Upper Layer (status of successful)

[Note that the service is complete after the master packet is sent. This service does not wait for positive acknowledgement from the slave.]

TLMC_M_084 - RAW MODE - MASTER PACKET ATTACHED RESPONSE

The Raw Mode (Master Packet Attached) service response to the Upper Layer will be generated at the completion of the service.

[The content of the response is shown in **Table 44 (RAW MODE SERVICE RESPONSE (MASTER))**.]

TABLE 44: RAW MODE SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x8B - Raw mode service response
Sub-type	N/A	1	The service response sub-type	N/A
Size	N/A	2	The size of the info field	1 byte
Info	Raw mode service status	1	Status of the service request	0x01 - Request successful(1) 0x03 - Request invalid in current state (state is Recovery; slave scanning)(2) 0x04 - Request invalid in current state (state is Recovery; slave not scanning)(2) 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x09 - Request invalid in current state (state is Standby) 0x0A - Parameter value error(3) 0x0B - Service duration timeout 0x0C - Service cancelled(4) (interrupted by emergency) 0x0D - Service cancelled(5) (interrupted by recovery) 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)(3) 0x25 - Buffer/ service request mismatch(6) 0x27 - Raw mode not enabled via initialize request
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. Master packet transmitted and acknowledged successfully.
2. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration).
3. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
4. This can occur if an emergency service interrupts “this” raw mode service.
5. This can occur if Channel Recovery interrupts “this” raw mode service.
6. This status can occur if the master host attempts to send a raw mode service request using service request buffer 1 (instead of buffer 0).

5.2.15 CLOSE

When this service is invoked, both Telemetry M stations can quickly terminate Telemetry M operations, saving energy and preventing unnecessary RF transmissions see **Fig. 63 (Close Sequence Diagram)**.

The following bullets briefly summarize this service:

- The upper layer on the master generates the Close request to the Data Link Layer.
- The Data Link Layer on the slave informs the Upper Layer after acknowledging the Close request.
- The upper layer on the slave terminates the communication.

No Channel Recovery shall be performed after a close request has been transmitted.

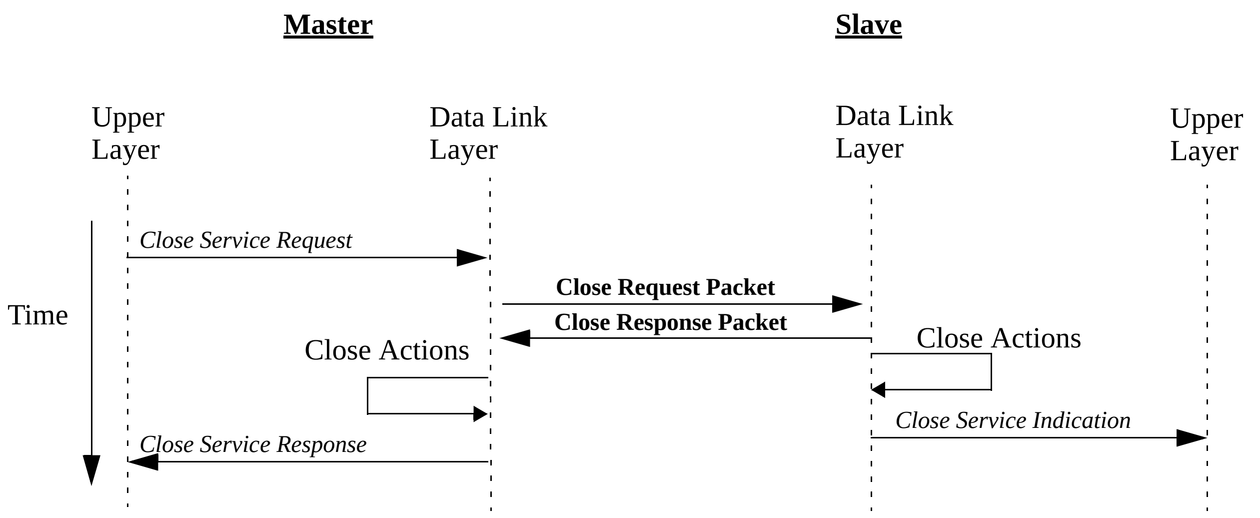


Figure 63: Close Sequence Diagram

5.2.15.1 REQUIREMENTS

TLMC_M_089 - CLOSE - UPPER LAYER CLOSE PARAMETERS

When invoking a Close service request, the Upper Layer will provide information as specified in **Table 45 (CLOSE SERVICE REQUEST (MASTER))**.

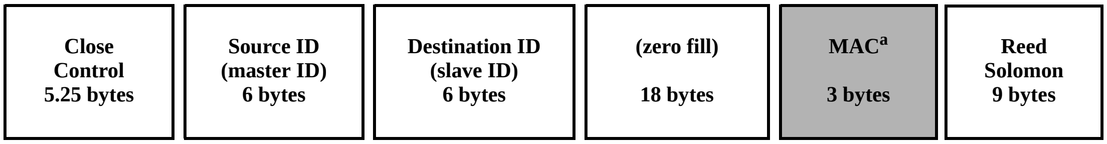
TABLE 45: CLOSE SERVICE REQUEST (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x87 - Close service request
Sub-type	N/A	1	The service request sub-type	N/A
Size	N/A	2	The size of the info field	0 bytes
Info(1)	N/A	0	N/A	N/A
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. The close service has one additional parameters which control its behavior: close service duration. See **Table 123 (Master parameters from host)**.

TLMC_MS_090 - CLOSE - MASTER PACKET FORMAT

The Close Request packets have the layout indicated in **Fig. 64 (Master Close Request Packet)**.

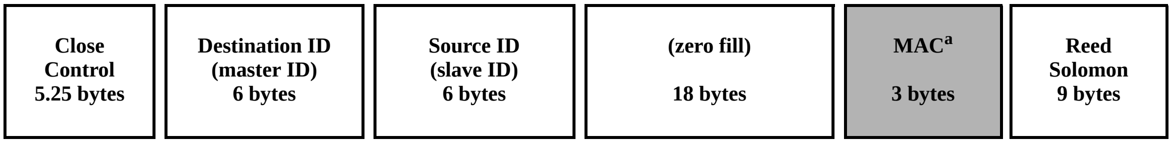


a. Message authentication code

Figure 64: Master Close Request Packet

TLMC_MS_091 - CLOSE - SLAVE PACKET FORMAT

The Close Response packets have the layout in **Fig. 65 (Slave Close Response Packet)**.



a. Message authentication code

Figure 65: Slave Close Response Packet

TLMC_M_092 - CLOSE - MASTER REQUEST

The actions performed upon receiving the Close service request when the state is Comm:

1. Start the service duration timer.
2. Transmit Close Request Packet and wait for the Close Response Packet.

[Send a “first request packet sent” event indication to the Upper Layer (see **Table 23**).]

TLMC_M_093 - CLOSE - RETRANSMISSION OF MASTER REQUEST

If no valid Close Response Packet is received from the slave, the Data Link Layer shall retransmit the Close Request packet. The master shall continue this until a valid Close Response packet is received or a Service Duration Timeout occurs or until recovery is triggered. Note that recovery is not initiated (if triggered).

TLMC_S_094 - CLOSE - RECEIPT OF A VALID CLOSE REQUEST

Upon receiving a valid master Close Request packet, the slave will transmit a close response packet and will stop the transmission of data packets.

[The security hardware will be disabled if slave starts to poll again upon polling request or sniff when standby timer times out.]

TLMC_S_095 - CLOSE - INDICATION TO SLAVE UPPER LAYER

The Close service indication will be generated to the Upper Layer when a successful Close response packet is transmitted.

[The indication to the Upper Layer may consist of Polling ON or Polling OFF based on the value of the Total Polling Duration Time in the open request (i.e., Zero meaning Polling OFF, otherwise Polling ON.]

TABLE 46: CLOSE INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x25 - Close indication
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	The size of the info field	2
Info	Close indication status	2	Status of the indication	1st byte: 0x01 - successful 2nd byte: 0x01 - without poll, or 0x02 - with poll
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

TLMC_M_096 - CLOSE - RECEIPT OF A VALID RESPONSE

Upon receiving a valid Close Response Packet from the slave, or if a Close Service Duration Timeout occurs, the master will return a successful close service response to the Upper layer.

TLMC_M_097 - CLOSE - IMPLICITLY SUCCESSFUL

Upon receiving a valid close service request from the Upper Layer when the state is Channel Recovery or Standby or Inactive or Medical Event, the master will disable the security hardware and return a successful close service response to the Upper layer.

[Terminate physical layer communication, return a successful Close service response to the Upper Layer, transition to Standby State - Note: Close packet is not transmitted. The Close service is implicitly successful.]

TLMC_M_098 - CLOSE - RESPONSE TO UPPER LAYER

The security hardware will be disabled and a close service response will be generated after the close service is complete. The content of the response to the Upper Layer is specified in **Table 47 (CLOSE SERVICE RESPONSE (MASTER))**.

TLMC_M_099 - CLOSE - RECOVERY DURING CLOSE

If channel recovery is triggered during the close service, the service is implicitly successful. The master will disable the security hardware and return successful to the Upper layer.

[Terminate physical layer communication, return success to upper layer, transition to Standby.]

TABLE 47: CLOSE SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x87 - Close service response
Sub-type	N/A	1	The service response sub-type	N/A
Size	N/A	2	The size of the info field	1 byte
Info	Close service status	1	Status of the service request	0x01 - request successful(1) 0x08 - Request in invalid state (state is disable, not reading EEPROM) 0x0A - Parameter value error(2) 0x0F - Request not serial (another service is in progress) 0x10 - Invalid CRC 0x12 - Invalid size (info size >506)(2) 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(3)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. Note that request successful is returned:
   if a valid Close packet is received from the slave, or
   if a service duration timeout occurs, or
   if Channel Recovery is triggered, or
   if the Close service request occurs when the telemetry state is Standby or Channel Recovery or Inactive or Medical Event.
2. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.

3. This status can occur if the master host attempts to send a close service request using service request buffer 1 (instead of buffer 0).

5.2.16 MEDICAL EVENT

The medical event service is somewhat analogous to an Open service except the roles of the master and slave are reversed. During a medical event the slave will duty cycle between transmitting and listening for the medical event service duration. First slave will send medical event packet as a “beacon” signal and then will listen for an Open request. The master will listen indefinitely for medical event packets. Upon hearing a medical event packet it will inform the host via indication, upon which the application could then issue an open service request.

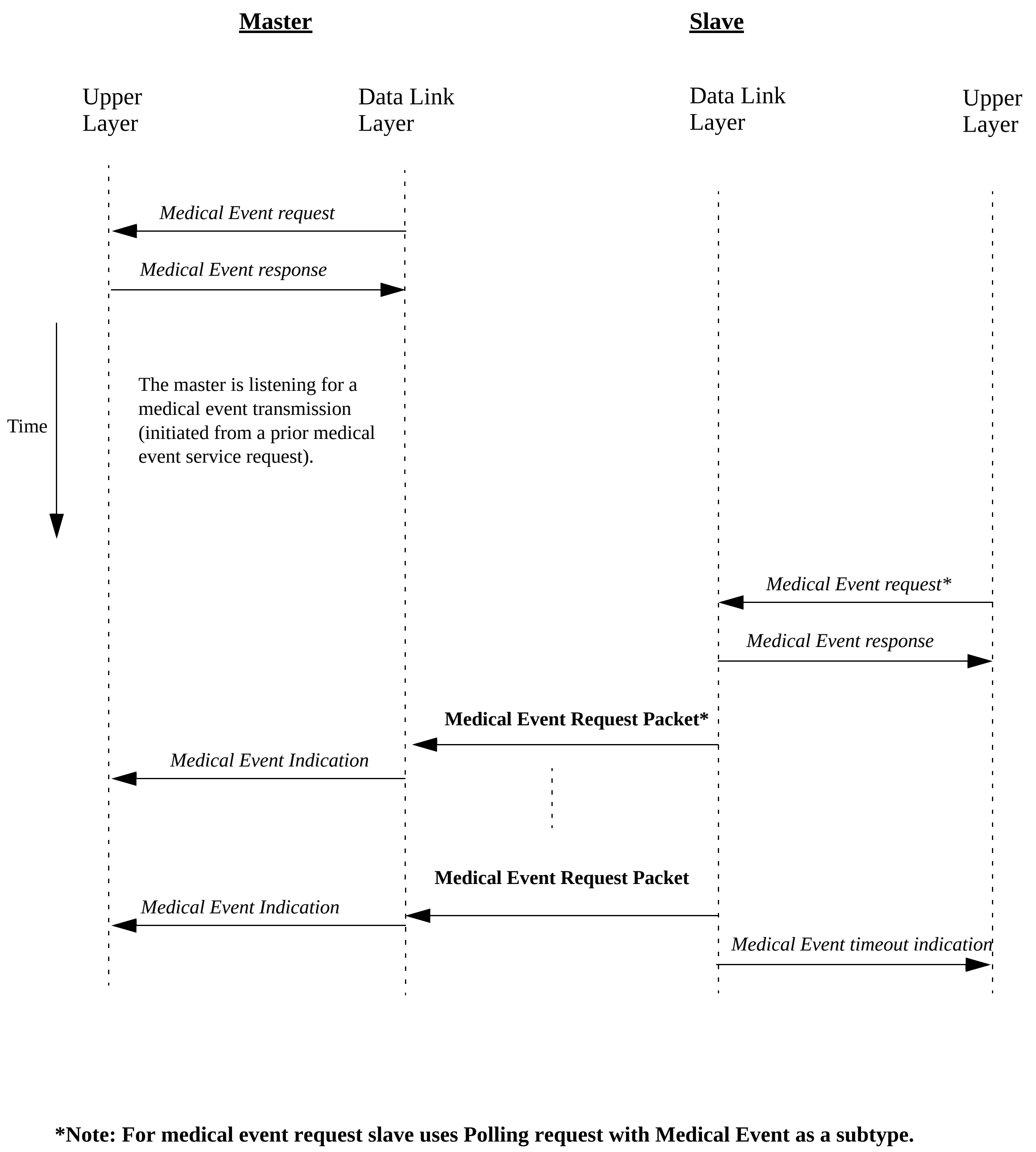
Master’s role:

- The data link layer send a successful service response to the host upon receipt of a medical event service request, selects a channel and listens for a Medical Event transmission. Every five seconds it will re-select the best channel and listen for a medical event.
- If a medical event packet is received, the data link layer verifies the content. If the contents are correct, the data link layer will send a medical event indication to the host.
- The master host should send an open service request.

Slave’s role:

- When a medical event is triggered the host will enable Mozart and send a medical event request (polling request with medical event as a sub-type)
- The data link layer will transmit one medical event packet (as specified by medical event channel transmit time **Table 124 (SLAVE PARAMETERS FROM HOST)** on each channel for the service duration.
- Upon receipt of an open request an open successful indication will be send to the host and the data link layer will transition to communication state.

When invoking this service the Upper Layer provides the station ID of the sequentially transmitting station to establish communication. An operational example of the Medical Event service is shown in **Figure 66**.



***Note: For medical event request slave uses Polling request with Medical Event as a subtype.**

Figure 66: Medical Event Sequence Diagram

5.2.16.1 REQUIREMENTS

TLMC_S_414 - MEDICAL EVENT - SLAVE PACKET FORMAT

The slave Medical Event response packet has the format is indicated in **Fig. 67 (Medical Event Packet)**

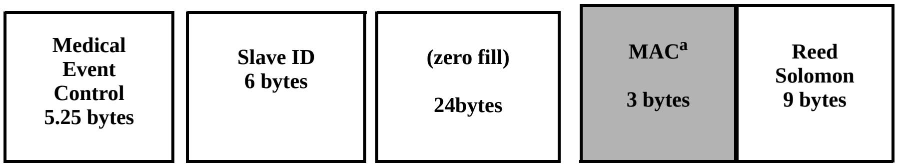


Figure 67: Medical Event Packet

TLMC_S_229 - MEDICAL EVENT - REQUEST

When medical event request is received, the Upper Layer will provide information as specified in **Table 48 (MEDICAL EVENT SERVICE REQUEST (SLAVE))**.

TABLE 48: MEDICAL EVENT SERVICE REQUEST (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x0A - Polling Service
Sub-type	N/A	1	The service request sub-type	0x04 - Poll upon Medical Event
Size	N/A	2	The size of the info field	Size of **Table 126 (SLAVE POLLING REQUEST)**
Info(1)	N/A	N	Polling info.	See **Table 126 (SLAVE POLLING REQUEST)**
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. The medical event service has one additional parameters which control its behavior: medical event service duration. See **Table 124 (SLAVE PARAMETERS FROM HOST)**

TLMC_S_307 - MEDICAL EVENT - CHANNEL TRANSMIT TIMOUT

The slave will restart native mode (transmit first) operation and re-send the Medical Event packet upon medical event channel transmit timeout. It will re-send this packet on a new channel.

[Slave will send Medical Event packet at a regular intervals as defined in **Table 124 (SLAVE PARAMETERS FROM HOST)**]

TLMC_S_416 - MEDICAL EVENT - INVALID OPEN REQUEST

The slave will restart native mode (transmit first) operation and re-send the Medical Event packet if it receives an Invalid Open packet.

If a valid Discover ID packet is received, Slave shall send a Discover Indication (i.e., Discover ID received while uplink medical events) to the host. See **Table 24 (DISCOVER INDICATION (SLAVE))**.

[An Invalid Open packet includes Open with mismatched slave ID, a packet other than Open, and packet with MAC error. Receipt of an invalid open packet causes the slave to transmit another medical event packet (and listen for an open packet).It will resend this packet on a new channel.]

TLMM_S_440 - MEDICAL EVENT - POLL UPON TELM REQUEST

If a poll upon Tel M request is received while uplink medical events, slave will stop uplink medical event, send polling response to host, and then slave shall start polling.

[See **TLMM_S_035 - POLLING - SERVICE REQUEST FROM UPPER LAYER**,
TLMM_S_256 - POLLING - SERVICE RESPONSE TO UPPER LAYER and
TLMM_S_241 - POLLING ENABLING CRITERIA for details]

TLMC_S_417 - MEDICAL EVENT - SERVICE DURATION

The slave will send Medical Event packets (and listen for open packets) until the Medical Event service duration timer expires.

TLMC_S_418 - MEDICAL EVENT - RESPONSE

When medical event request is received, the Data Link Layer will validate the medical event service request. A medical event service response will be sent to the host (see **TABLE 49: MEDICAL EVENT SERVICE RESPONSE (SLAVE)**). If the service status is successful, the slave will start the Medical Event Service Duration timer, and send a medical event packet.

TABLE 49: MEDICAL EVENT SERVICE RESPONSE (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x0A - Polling Service
Sub-type	N/A	1	The service response sub-type	0x04 - Poll upon Medical Event
Size	N/A	2	The size of the info field	1 byte
Info	Medical event service status	1	Status of the service request	0x01 - Successful 0x12 - Invalid size 0x14 - Invalid Poll subtype 0x15 - Request in invalid state(1) 0x10 - Invalid CRC 0x2A - Request invalid during EEPROM write
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. See **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)** for the invalid states.

TLMC_S_257 - MEDICAL EVENT - INDICATION TO UPPER LAYER

A medical event service will be stopped under the following conditions:

1. Medical event service duration timer expires medical event timeout indication will be generated to the upper layer **Table 50 (MEDICAL EVENT INDICATION (SLAVE))**.
2. A valid open packet is received an open successful indication will be generated to the upper layer.**Table 28 (OPEN INDICATION (SLAVE))**

TABLE 50: MEDICAL EVENT INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x28 - Medical event indication
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	The size of the info field	1
Info	Medical event indication status	1	Status of the indication	• 0x03 - timeout
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

TLMC_M_230 - MEDICAL EVENT - HOST PARAMETERS

When invoking a Medical Event service request, host will provide information as specified in **Table 51 (MEDICAL EVENT SERVICE REQUEST (Master))**

TABLE 51: MEDICAL EVENT SERVICE REQUEST (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x85 - Medical Event Service
Sub-type	N/A	1	The service request sub-type	n/a
Size	N/A	2	The size of the info field	6 bytes
Info	N/A	6	Slave ID	6 bytes of slave ID
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

TLMC_M_231 - MEDICAL EVENT - RESPONSE TO HOST

A Medical Event response will be generated when the Medical Event service is initiated.

[The content of the response is shown in **Table 52 (MEDICAL EVENT SERVICE RESPONSE (Master))**.

TABLE 52: MEDICAL EVENT SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x85 - Medical Event Service Response
Sub-type	N/A	1	The service response sub-type	n/a
Size	N/A	2	The size of the info field	1 byte
Info	Medical event service status	1	Status of the service request	1 = request successful 2 = request invalid in current state (state is Comm) 3 = request invalid in current state (state is Recovery, device scanning) 4 = request invalid in current state (state is Recovery, device not scanning)(2) 5 = request invalid in current state (state is Inactive) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x0A - Parameter value error(1) 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)**(1)** 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(2)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
2. This status can occur if the master host attempts to send a medical event service request using service request buffer 1 (instead of buffer 0).

TLMC_M_232 - MEDICAL EVENT - MEDICAL EVENT PACKET

If the master receives a valid medical event packet it will send a medical event indication to the host as specified in **Table 53 (MEDICAL EVENT INDICATION (Master))**

[The indication will be generated to the host no more frequently than one/second]

TABLE 53: MEDICAL EVENT INDICATION (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xA8 - Medical event indication
Sub-type	N/A	1	The indication sub-type	N/A
Size	N/A	2	The size of the info field	1
Info	Medical event indication status	1	Status of the indication	• 0x01 - Successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

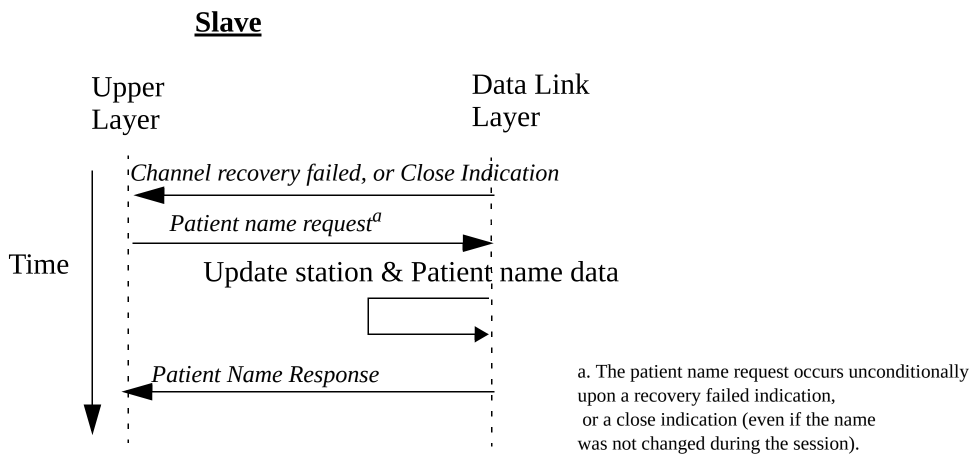
5.2.17 PATIENT NAME REQUEST

The device and patient name data is provided by the host to Telemetry M when:

- Telemetry M issues following indications:

  Channel recovery failed with polling ON and Close with polling ON.

Fig. 68 (Patient Name Sequence Diagram) illustrates the Patient Name request operations.



a. The patient name request occurs unconditionally upon a recovery failed indication, or a close indication (even if the name was not changed during the session).

Figure 68: Patient Name Sequence Diagram

The patient name request is an “implied” polling request. The patient name and polling functionality are “tied” so that a stale name is not shown to the user after recovery fails or after close. The slave state machine returns to standby after recovery fail or close. It transitions to acquire upon a patient name (implied polling). Slave will start to poll once poll upon patient name request is received. See **TLMM_S_240 - POLLING INITIATION** for details.

5.2.17.1 REQUIREMENTS

TABLE 54: PATIENT NAME SERVICE REQUEST (SLAVE)

FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	1	The type of service request	0x0A -Polling service
Sub-type	1	The service request sub-type	0x05 -Poll upon Patient Name
Size	2	The size of the info field	
Info		See **Table 126 (SLAVE POLLING REQUEST)**	See **Table 126 (SLAVE POLLING REQUEST)**
CRC	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

TLMC_S_258 - PATIENT NAME REQUEST

When Patient name request is invoked host will provide the patient name information.

[In the case of CRM this information resides in the device-specific system parameters database.]

TLMC_S_259 - PATIENT NAME SERVICE RESPONSE TO UPPER LAYER

When patient name request is received the Data Link Layer will validate the Patient Name service request (see **Table 54 (PATIENT NAME SERVICE REQUEST (SLAVE))**) and will generate an appropriate response (see **Table 55 (PATIENT NAME SERVICE RESPONSE (SLAVE))**).

TABLE 55: PATIENT NAME SERVICE RESPONSE (SLAVE)

FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	1	The type of service request	0x0A -Polling service
Sub-type	1	The service response sub-type	0x05 -Poll upon Patient Name
Size	2	The size of the info field	1 byte
Info	1	Status of the service request	0x01 - Successful 0x12 - Invalid size 0x15 - Request in invalid state(1) 0x10 - Invalid CRC 0x14 - Invalid subtype 0x2A Request invalid during EEPROM write
CRC	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. See **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)** for the invalid states.

5.2.18 NETWORK

The term “network” used in this section is not an industry standard as used in the telecommunication industry. The network in this document refers to an external service which could be invoked by the user of this communication protocol specification. This network service allows the communication between two nodes (known as one to one communication). The network service also allows communication between 1 to N nodes. Time slots are used to avoid collisions between multiple nodes. See **Fig. 78 (Network Multiple Nodes Response)**.

The network service does not require a communication session to be opened via open service.There are two sub-services which are associated with the network service.

- Configure Network: is used to configure the network.
- Network Data: is used to exchange data between nodes. It uses wireless local wakeup.

In order to define and setup a network, the node(s) wakeup internal service (Section 5.3.6) along with a Discover external service (Section 5.2.7) are used to identify the nodes within the vicinity.

Facilitator’s (network node or third party) upper layer is responsible for building the network table and insuring that table is securely transmitted to all the network nodes as identified in the configuration table. The host for each node will use the configure network service to provide the table to the data link layer each time the content of the configuration table is changed.

5.2.18.1 BUILDING LOCAL NETWORK

The network is initially built by first performing the Discover Full Service request on a facilitator, to find nodes who do not belong to a network. A facilitator can be a network node or a third party. An example of the third party is a clinician’s programmer. The user (clinician) selects the nodes for the local network and identifies one node as the network coordinator. The network table is sequentially transmitted to each network node using the data service. When all network nodes have received the network table, a local network is considered formed. Once the local network is formed host on each node uses the configure network request to pass the network table to its data link layer.

5.2.18.2 NETWORK SERVICES

The network service consists of two sub-services:

1. Configure network - service is used by the host to provide the network configuration table to the data link layer. Upon a successful completion of the network service the configuration table is stored in the data link layer, it may also be stored by the upper layer (host). When network config values change or a reset occurs host must provide the network configuration table. A typical example of network table is shown in **Table 56 (NETWORK TABLE)**.

TABLE 56: NETWORK TABLE

TABLE INDEX (SHORT ID)	ID
0	Station ID of coordinator(1)
1	Station ID of child #1
2	Station ID of child #2(2)
3	Station ID of child #3(2)

(1) The coordinator's ID is the network ID. The coordinator must always be the first ID in the network table.

(2) Unused nodes should be set to a station ID of all zeroes.

2. Network data - a service used by a network node to exchange a small amount of data. This exchange may be a one-to-one exchange. It may also be a one-to-N. An example of a one-to-one exchange follows: an external sensor transmits its sensor data to a device. The device transmits a network data indication to its (the device's) upper layer when it receives the network data packet from the sensor. The device's upper layer interprets the data (for therapy decisions, etc.). The service is complete when the device transmits a response message and it is acknowledged by the sensor.

5.2.18.3 CONFIGURE NETWORK

The configure network service is used by this node's host to provide the network table to its data link layer. An example is shown in **Table 56 (NETWORK TABLE)**.

5.2.18.3.1 CONFIGURE NETWORK SERVICE REQUIREMENTS

TABLE 57: CONFIGURE NETWORK SERVICE REQUEST

FIELD	INFO. FIELD	BYTES	DESCRIPTION	VALUES
Type	n/a	1	Type of service.	Network 0x0C - Slave 0x8C - Master
Sub-type	n/a	1	Sub-type of the service.	0x03 - Configure network service request
Size	n/a	2	Size of Info	26
Info	Network table	24	The IDs of all the network members.	See **Table 56 (NETWORK TABLE)**. 24 - 4 nodes
	Network table CRC	2	Network table CRC	
CRC	n/a	2	CRC of all fields above.	This is given by the upper layer.

Note the Network table is fixed at 24 bytes, if there are less than four nodes in the network, zero "00" should be used to fill the unused bytes. For example, if there are only two nodes in the network table, last 12 bytes should be "00" and the Network table CRC is the CRC of the 24bytes.

TLMM_MS_309 - CONFIGURE NETWORK - UPPER LAYER PARAMETERS

When invoking a configure network service request, the Upper Layer shall provide information as specified in **Table 57 (CONFIGURE NETWORK SERVICE REQUEST)**.

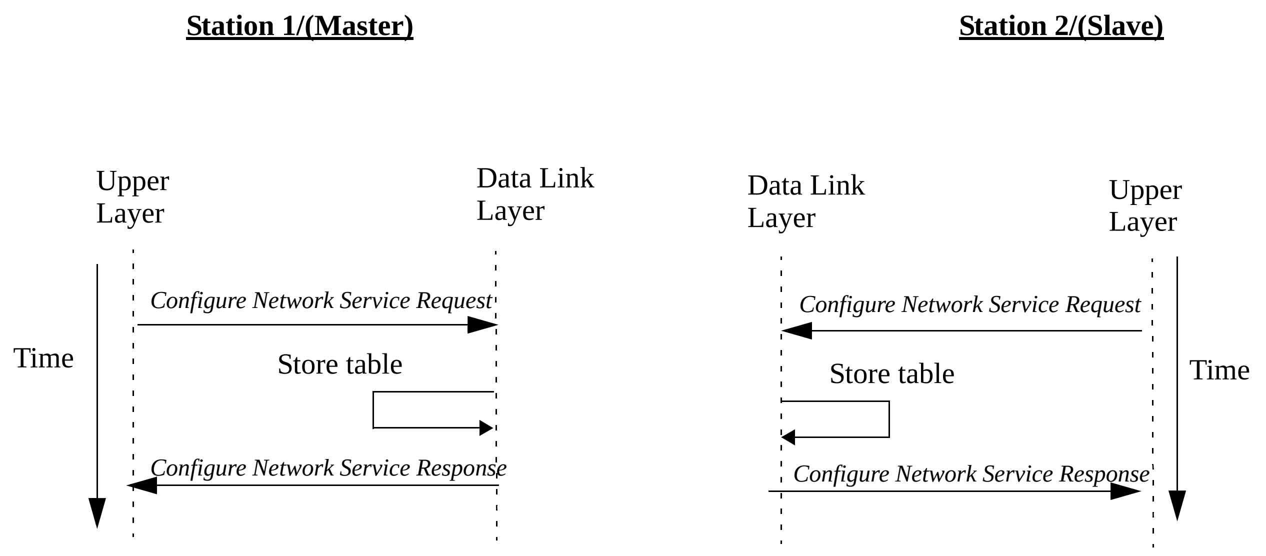


Figure 69: Configure Network Sequence Diagram

TLMM_MS_310 - CONFIGURE NETWORK - RESPONSE TO UPPER LAYER

Upon the receipt of the network configuration table data link layer will generate a response to upper layer as specified in **Table 58 (CONFIGURE NETWORK SERVICE RESPONSE)**.

TABLE 58: CONFIGURE NETWORK SERVICE RESPONSE

FIELD	INFO FIELD	BYTES	DESCRIPTION	VALUES
Type	n/a	1	Service type.	Network 0x0C - Slave 0x8C - Master
Sub-type	n/a	1	Sub-type of the service.	0x03 - Configure network service response.

TABLE 58: CONFIGURE NETWORK SERVICE RESPONSE

Size	n/a	2	Size of Info	1
Info	Configure Network Service Result	1	Status of the service request	0x01 - request successful 0x0A - parameter value error[1] 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)[1] 0x13 - Invalid Request (This code can occur if the service request is sent using an interrupt other than service request0 (there is a buffer/ service request mismatch). (Slave Only) 0x19 - Invalid configuration (Tel M only) 0x24 - request invalid during EEPROM read (slave only) 0x25 - Buffer/ service request mismatch (master only)[2]
CRC	n/a	2	CRC of all fields above.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
2. This status can occur if the master host attempts to send a network service request using service request buffer 1 (instead of buffer 0).

5.2.18.4 NETWORK DATA

A network node uses the network data service (secured) to communicate with another local network node (one-to-one exchange). A network node may also use this service for a one-to-N exchange. This service differs from the data service (during communication) in that a small amount of data is transferred (typically less than 20 bytes and fit into one network data request packet) and it does not require open communication session.

- Upper layer generates the request to the data link layer.
- A channel is selected, wakeup packet(s) are transmitted and the data link layer transmits the request. The station remains on the selected channel until the response is received, or a service duration occurs, or the LBT timeout occurs.
- If the response is received, the service is declared a success.
- The data link layer notifies the upper layer with the service results.

The network data service can be any of the following:

- Network data service with message level (upper layer) response.**Fig. 70 (Network data (with message level acknowledgement) diagram)**.
- Network data service with data link layer response. **Fig. 71 (Network data (with data link acknowledgement) diagram)**.
- Network data service with no response. A typical example of this service is a transmission-only device. See **Fig. 72 (Network data (no acknowledgement) diagram)**.

TLMM_M_311 - NETWORK DATA - UPPER LAYER PARAMETERS

When invoking a network data service request, the upper layer shall provide information as specified in **Table 59 (NETWORK DATA SERVICE REQUEST(MASTER))**.

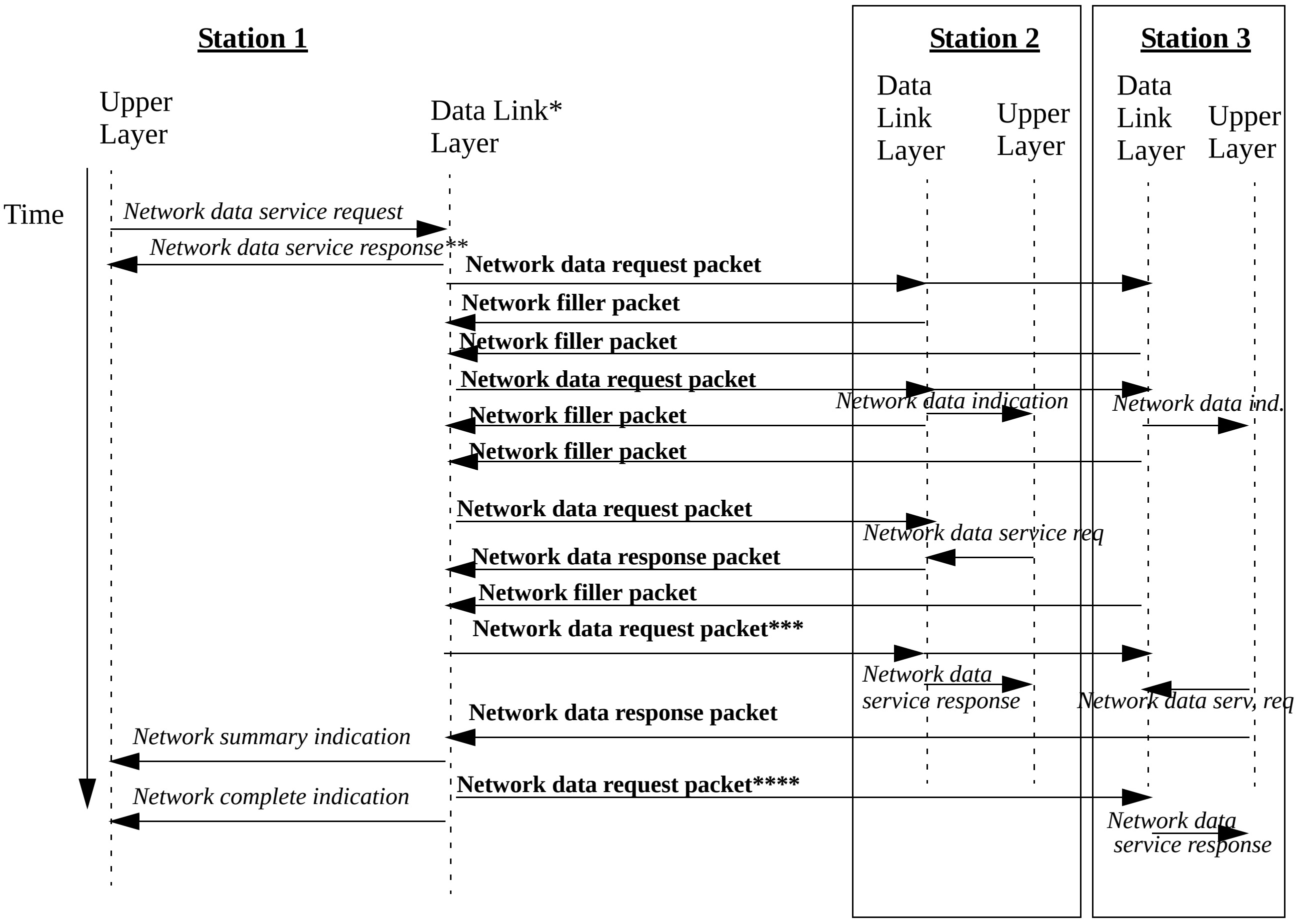
TABLE 59: NETWORK DATA SERVICE REQUEST(MASTER)

FIELD	INFO FIELD	BYTES	DESCRIPTION	VALUES
Type	n/a	1	The type of service network	0x8C - Master Network Request
Sub-type	n/a	1	The sub-type of service.	0x04 - Network data request.
Size	n/a	2	Size of Info.	4 - 13
Info.(1) .	Deliver to coordinator(2)	1 bit	A flag telling if the network request should be delivered to the coordinator.	True = this network request should be delivered to the network coordinator False = otherwise
	Deliver to child #1(2)	1 bit	A flag telling if the network request should be delivered to child #1.	True = this network request should be delivered to child #1 False = otherwise
	Deliver to child #2(2)	1 bit	A flag telling if the network request should be delivered to child #2.	True = this network request should be delivered to child #2 False = otherwise

	Deliver to child #3(2)	1 bit	A flag telling if the network request should be delivered to child #3.	True = this network request should be delivered to child #N
				False = otherwise

	Expect response	1	A flag indicating if a data link response is desired. This is communicated to the slaves via a bit in the control bytes.	True = the master expects an data link layer response from each slave. The master will re-transmit the request packet until all slaves have responded it or a service duration occurs, or the LBT timeout occurs. False = The master will transmit the network data request packets for a period of time, no response is required.
	User Data size	1	Number of user data bytes	1 to 10
	User Data	1 - 10	User data bytes to transmit	The data to be transmitted. This includes the upper layer header (ULH) as the first byte. See 5.2.11.2 for master ULH byte.
CRC	n/a	2	CRC for all fields above.	This is given by the upper layer.

1. The is one more parameter which controls the operation of the network data service: service duration. See **Table 123 (Master parameters from host)**. This value should be approximately 2 seconds longer than the total wakeup burst duration. Twu + Tcca <= Service Duration <= Twu + 2. Twu is the wakeup burst duration, Tcca is the Clear-Channel-Assessment time. 2 seconds is coming from LBT 5 seconds minus wakeup last block time 3 seconds.
2. The combination of deliver to coordinator, deliver to child #1, deliver to child #2,deliver to child #3 yields the receptor bit map used by the wireless wakeup internal service (a field of the wakeup packet). The upper 4bits are unused.



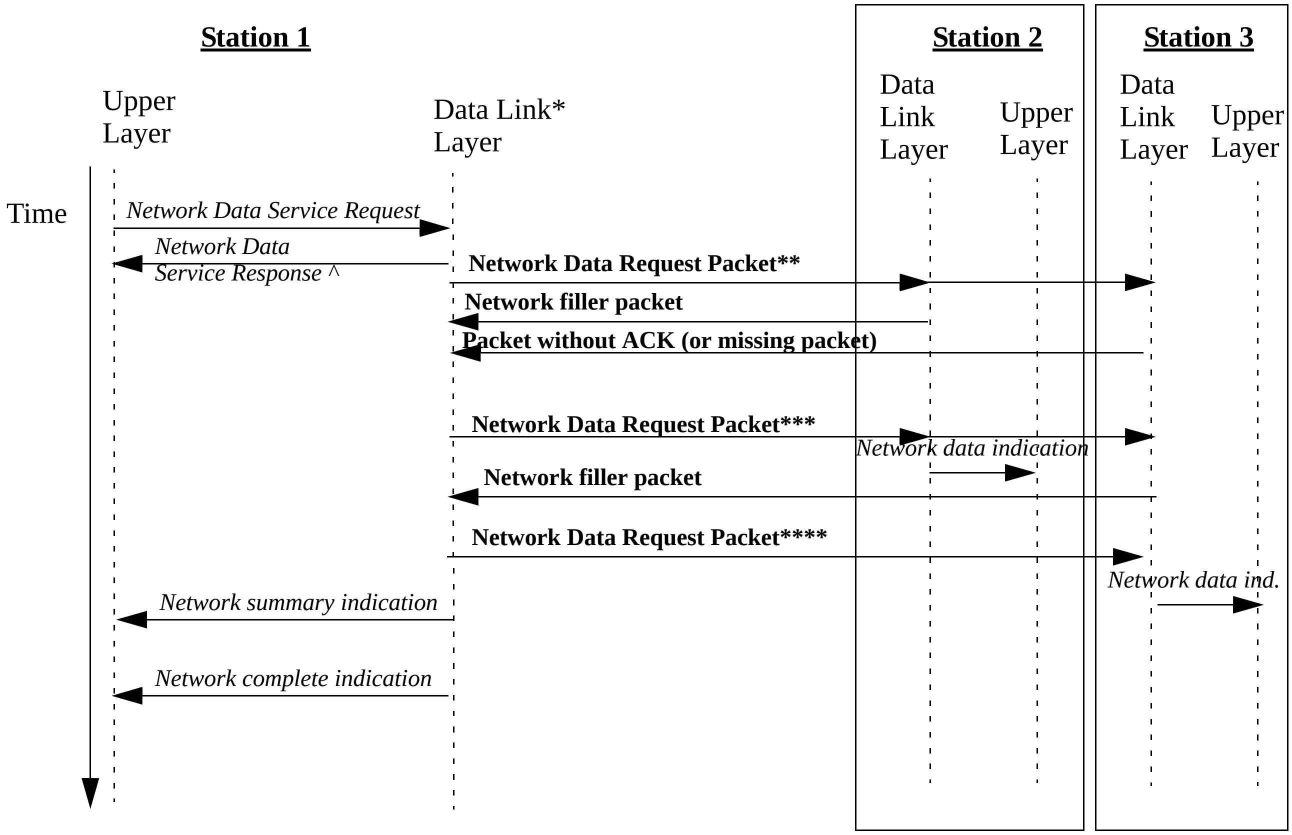
* Channel mapping and wakeup are not shown.

** Response to upper layer for received service request.

*** This packet acknowledges station 2’s message level response. Station 2 stops listening (it is done).

**** This packet acknowledges station 2 and 3’s message level response. Station 3 stops listening (it is done).

Figure 70: Network data (with message level acknowledgement) diagram



* Channel mapping and wakeup are not shown.

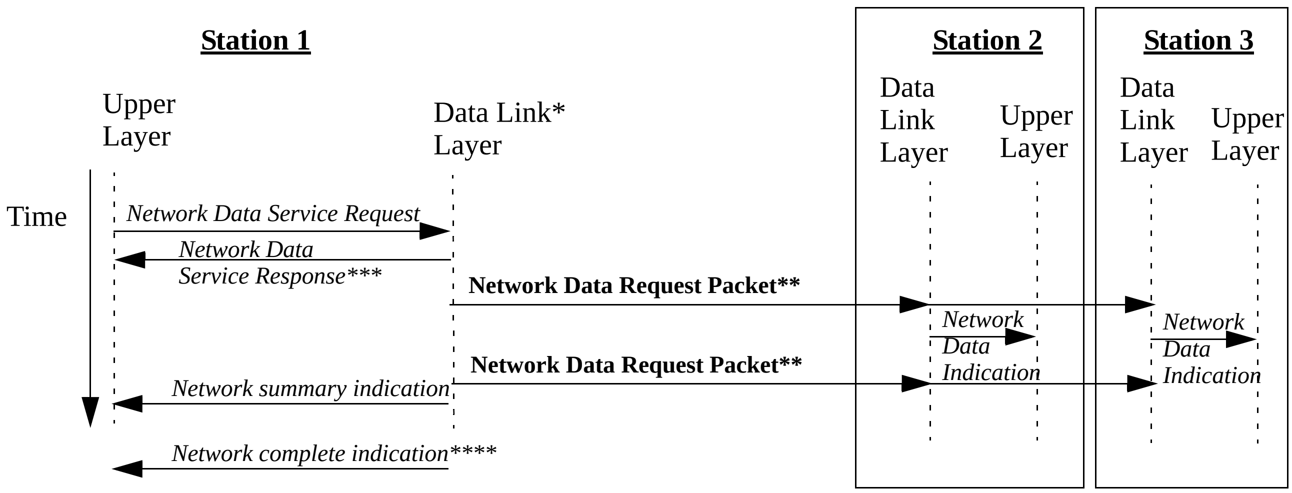
** The upper layer header (ULH) byte indicates no message level response is needed.

*** This packet acknowledges station 2’s data link acknowledgement (ack-ack). Station 2 stops listening.

**** This packet acknowledges station 2 and 3’s data link acknowledgement (ack-ack). Station 3 stops listening.

^ Response to upper layer for received service request.

Figure 71: Network data (with data link acknowledgement) diagram



* Channel mapping and wakeup are not shown.

** Packet control byte has “expect response” equal false. Station 2 and station 3 stop listening.

Note that network data packets continue until any one of the timers expires (service duration timer and LBT timer).

*** Response to the upper layer for received service request.

**** Indication to the upper layer that service request has been transmitted for a while and a timer expired (the service duration timer, or the LBT timer).

Figure 72: Network data (no acknowledgement) diagram

TLMM_M_312 - NETWORK DATA - REQUEST AND RESPONSE PACKET FORMAT

The network data request and response packet shall have the format shown in **Fig. 73 (Network data Request packet)** and **Fig. 74 (Network data Response packet)**. In the request packet the source network table index shall be the index of the master. Master shall echo the slave's random numbers received in the network filler packets. In the response packet the source network table index shall be the index of a slave. The network ID shall be the station ID of the coordinator.

Network data control. 5.25 bytes	**Network ID 6 bytes**	**Source network table index[b]**	**Destination bitmap[c] 1 byte**	**Ack bitmap[d] 1 byte**	**Network Table CRC (2 byte)**	**Slave's Random Number[e] (9bytes)**	**Data N bytes**	**Zero fill 10-N bytes**	**MAC[a] 3 bytes**	**Reed Solomon 9 bytes**

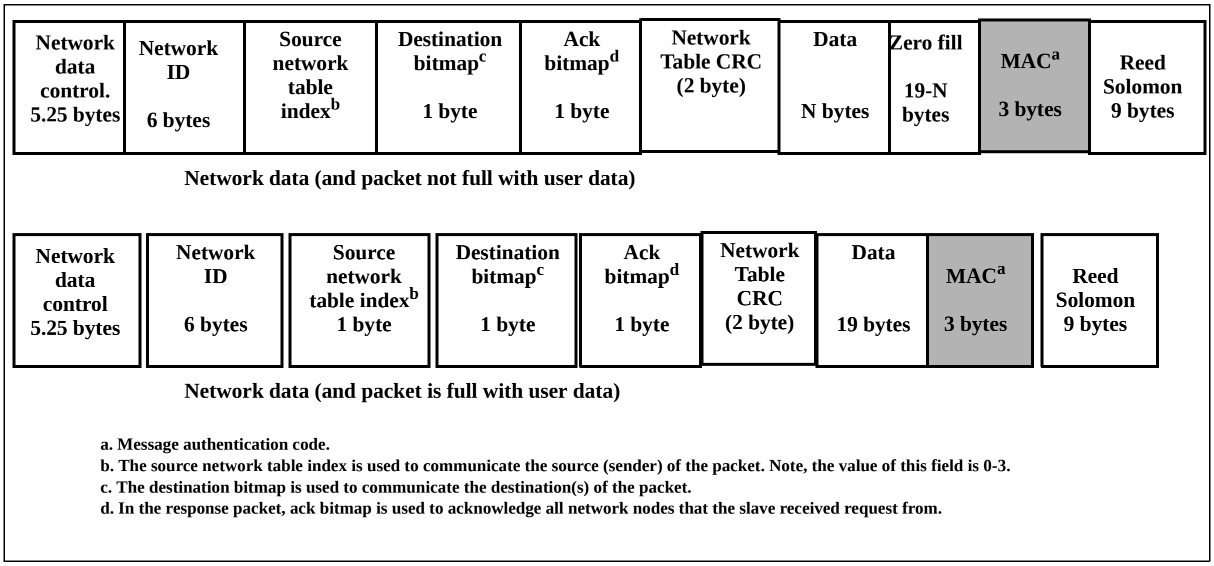
Network data (and packet not full with user data)

Network data control 5.25 bytes	**Network ID 6 bytes**	**Source network table index[b] 1 byte**	**Destination bitmap[c] 1 byte**	**Ack bitmap[d] 1 byte**	**Network Table CRC (2 byte)**	**Slave's Random Number (9bytes)**	**Data 10 bytes**	**MAC[a] 3 bytes**	**Reed Solomon 9 bytes**

Network data (and packet is full with user data)

a. Message authentication code.
b. The source network table index is used to communicate the source (sender) of the packet. Note, the value of this field is 0-3.
c. The destination bitmap is used to communicate the destination(s) of the packet.
d. The ack bitmap in the Network Data request is to provide the "highest needed" acknowledgement.
e. Slave's random numbers are in the order : Random number received in slot #1 (3bytes), slot #2 (3bytes) and slot #3 (3bytes). They are zero unless the random number has been received from the slot.

Figure 73: Network data Request packet



Network data control. 5.25 bytes	Network ID 6 bytes	Source network table index[b]	Destination bitmap[c] 1 byte	Ack bitmap[d] 1 byte	Network Table CRC (2 byte)	Data N bytes	Zero fill 19-N bytes	MAC[a] 3 bytes	Reed Solomon 9 bytes

Network data (and packet not full with user data)

Network data control 5.25 bytes	Network ID 6 bytes	Source network table index[b] 1 byte	Destination bitmap[c] 1 byte	Ack bitmap[d] 1 byte	Network Table CRC (2 byte)	Data 19 bytes	MAC[a] 3 bytes	Reed Solomon 9 bytes

Network data (and packet is full with user data)

a. Message authentication code.
b. The source network table index is used to communicate the source (sender) of the packet. Note, the value of this field is 0-3.
c. The destination bitmap is used to communicate the destination(s) of the packet.
d. In the response packet, ack bitmap is used to acknowledge all network nodes that the slave received request from.

Figure 74: Network data Response packet

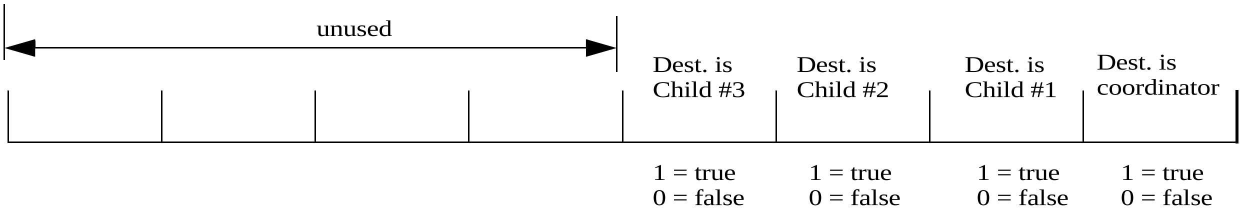


Figure 75: Destination bitmap (Network Data Request and Response Packet)

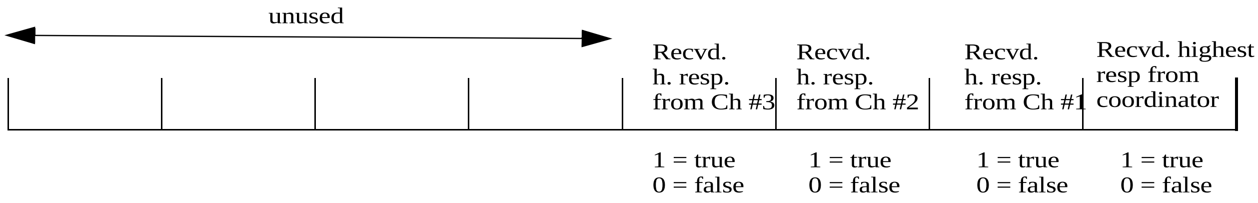
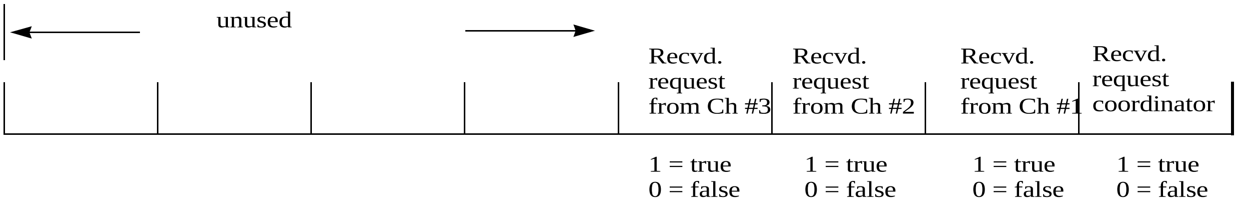


Figure 76: Network Data Request Packet Ack bitmap



* Note this is used in simultaneous multiple communications among network nodes.

Figure 77: Network Data Response Packet Ack bitmap

An example of one to N communication:

A local network contains 4 device nodes, Child #1 as master, Child #2, Child #3 are slave nodes. This is a network data service with upper layer (message level) response required. See **Fig. 78 (Network Multiple Nodes Response)**. The example below shows eight nodes:

1) First Network Data request packet:

Network ID = 0x112233445566 (local network coordinator’s ID)

Source network table index = 0x01 (Child #1)

Destination bitmap = 0x0C (Child #2, #3)

Ack bitmap = 0x00

Data 1st byte ULH = 0x09 (Transaction Type(bit6,7)= 0 | Final Transaction(bit3)= 1 |

Application response requested(bit0) = 1)

Random Numbers = 9bytes of 00.

Network Filler packet from Child #2:

Network ID = 0x112233445566 (local network coordinator’s ID)

Source network table index = 0x02 (Child #2)

Destination bitmap = 0x02 (Child #1)

Ack bitmap = 0x00 (Child #1)

Control Byte 1 bit7(ACK) = 1, this is the acknowledgement to the master to indicate Good MAC packet has been received.

Response time slot = 1

Random Number of slot 1 = Random number of child 2

Network Filler packet from Child #3:

Network ID = 0x112233445566 (local network coordinator's ID)

Source network table index = 0x03 (Child #3)

Destination bitmap = 0x02 (Child #1)

Ack bitmap = 0x00 (Child #1)

Control Byte 1 bit7(ACK) = 1, this is the acknowledgement to the master to indicate Good MAC packet has been received.

Response time slot = 2

Random Number of slot 2= Random number of child 3

2)Second Network Data request packet (Child #2 and Child #3 not respond yet):

Network ID = 0x112233445566 (local network coordinator's ID)

Source network table index = 0x01 (Child #1)

Destination bitmap = 0x0C (Child #2, #3)

Ack bitmap = 0x00

Data 1st byte ULH = 0x09 (Transaction Type(bit6,7)= 0 | Final Transaction(bit3)= 1 |

Application response requested(bit0) = 1)

Random Numbers = 3 bytes received from of slot1, 3 bytes received from of slot2, 3bytes of 00.

Network Data response packet from Child #2:

Network ID = 0x112233445566 (local network coordinator's ID)

Source network table index = 0x02 (Child #2)

Destination bitmap = 0x02 (Child #1)

Ack bitmap = 0x02 (Child #1)

Control Byte 1 bit7(ACK) = 1, this is the acknowledgement to the master to indicate Good MAC packet has been received.

Data 1st byte ULH = 0x10 (Transaction Type(bit6,7)= 0| Final Transaction(bit4)= 1)

Response time slot = 1

3) Third Network Data request packet (Child #3 not respond yet):

Network ID = 0x112233445566 (local network coordinator's ID)

Source network table index = 0x01 (Child #1)

Destination bitmap = 0x0C (Child #2,#3)

Ack bitmap = 0x04 (Child #2 responded)

Data 1st byte ULH = 0x09 (Transaction Type(bit6,7)= 0 | Final Transaction(bit3)= 1 |

Application response requested(bit0) = 1)

Network Data response packet from Child #3:

Network ID = 0x112233445566 (local network coordinator’s ID)

Source network table index = 0x03 (Child #3)

Destination bitmap = 0x02 (Child #1)

Ack bitmap = 0x02 (Child #1)

Control Byte 1 bit7(ACK) = 1, this is the acknowledgement to the master to indicate Good MAC packet has been received.

Data 1st byte ULH = 0x10 (Transaction Type(bit6,7)= 0| Final Transaction(bit4)= 1)

Response time slot = 2

4)Last Network Data request packet (all Child #2,#3 responded):

Network ID = 0x112233445566 (local network coordinator’s ID)

Source network table index = 0x01 (Child #1)

Destination bitmap = 0x0C (Child #2,#3)

Ack bitmap = 0x0C (Child #2,#3 responded)

Data 1st byte ULH = 0x09 (Transaction Type(bit6,7)= 0 | Final Transaction(bit3)= 1 |

Application response requested(bit0) = 1))

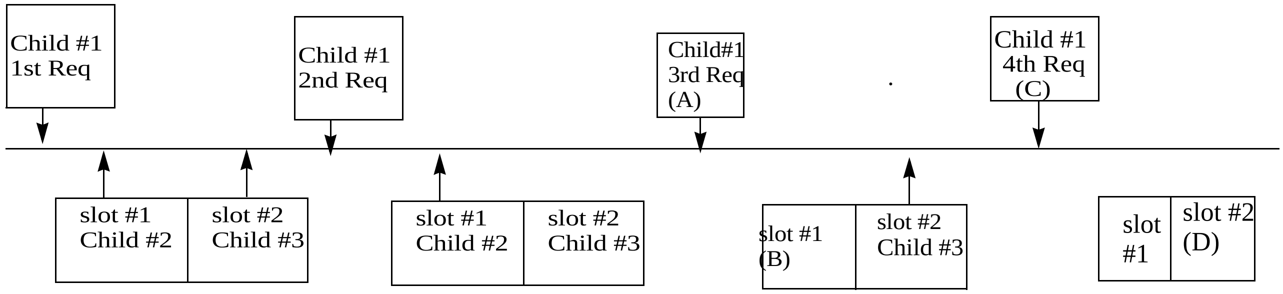


Figure 78: Network Multiple Nodes Response

Note:
(A) Child #1 3rd Req. acked Child #2. (B) Child#2 receives the ack from Child #1 and goto standby.
(C) Child #1 4th Req. acked Child #2 and Child #3.
(D) Child#3 receives the ack from Child #1 and goto standby.

TLMM_S_313 - NETWORK DATA - FILLER PACKET FORMAT

The network filler packet shall be used by the slave to transmit its own random number to the master as well as respond to the network data request before the slave host responds to the network data request.

[The filler packet is to occupy the channel format of this packet is shown in **Fig. 79 (Network Filler Packet)**].

Network filler packet format:

Network filler control. 5.25 bytes	**Network ID 6 bytes**	**Source network table index[b] 1 byte**	**Destination bitmap[c]**	**Ack bitmap[d] 1 byte**	**Network Table CRC (2 byte)**	**Random Number (3bytes)**	**Zero fill 16 bytes**	**MAC[a] 3 bytes**	**Reed Solomon 9 bytes**

[a] Message authentication code.
[b] The source network table index is used to communicate the source (sender) of the packet.
[c] A destination bitmap is used to communicate the destination(s) of the packet.
[d] Ack bitmap is used to acknowledge other network nodes in simultaneous multiple communications among network nodes.

Figure 79: Network Filler Packet

5.2.18.4.1 NETWORK DATA SERVICE REQUIREMENTS

5.2.18.4.2 MASTER

TLMM_M_314 - NETWORK DATA - TRANSMITTING REQUEST PACKETS

Upon receiving a network data service request from the upper layer master will:

1. Start the service duration timer.
2. Select the least interfered channel (takes 275 +/- 50 msec(4))
3. Start LBT timer.
4. Perform instigator wireless wakeup. See parameters in **Table 123 (Master parameters from host)** for Wakeup user data present, secure wakeup and user data.
5. Transmit the network data request packet.

---

(4) CCA of the MICS band takes approximately 275 msec. CCA of the MEDS band takes approximately 550 msec. CCA of both bands takes approximately 825 msec. Note that if HDR mode is enabled, assessment of each channel, antenna pair is lengthened by 5 msec (to allow the HDR synthesizer to stabilize). This means the CCA of the MICS band is lengthened from 275 msec to 375 msec

TLMM_M_315 - NETWORK DATA - UPPER LAYER RESPONSE (MESSAGE LEVEL RESPONSE)

The network data request packet shall be transmitted until any one of the timers expires (the service duration timer or the LBT timer). When any of the timers expire a network complete indication will be sent to the host. See **Table 60 (NETWORK COMPLETE INDICATION (MASTER))**

TABLE 60: NETWORK COMPLETE INDICATION (MASTER)

FIELD	INFO FIELD	BYTES	DESCRIPTION	VALUES
Type	n/a	1	Type of indication	0xBA - Network complete indication
Sub-type	n/a	1	The indication sub-type	n/a
Size	n/a	2	The size of the info field	1
Info	status of the indication	1	The status of the indication	0x03 - timeout
CRC	CRC of all the above fields. Two bytes.	2	A two byte CRC of all above fields	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

[See **Fig. 70 (Network data (with message level acknowledgement) diagram)** for an example.

The upper layer sets “application reply requested” bit in the ULH byte to request upper layer (message level) response from the slave.

The antenna will be toggled each time a network packet is transmitted.]

TLMM_M_316 - NETWORK DATA - DATA LINK LAYER RESPONSE (PACKET LEVEL RESPONSE)

The network data request packet shall be transmitted until any one of the timers expires (the service duration or the LBT timer). When any of the timers expire a network complete indication will be sent to the host. See **Table 60 (NETWORK COMPLETE INDICATION (MASTER))**.

[See **Fig. 71 (Network data (with data link acknowledgement) diagram)** for an example.

The upper layer sets the “Expect response” field in the network data service request and clears the “application reply requested” bit in the ULH byte to request data link layer (packet level) response from the slave.

The data link layer will set the “Expect response” bit in control bytes of the network data request packet accordingly.

The antenna will be toggled each time a network packet is transmitted.]

TLMM_M_317 - NETWORK DATA - NO ACK

The network data request packet shall be transmitted until any one of the timer expires (the service duration or the LBT timer), if no response is required from the slave(s). When any of the timers expire a network complete indication will be sent to the host. See **Table 60 (NETWORK COMPLETE INDICATION (MASTER))**.

[The upper layer clears “Expect response” field in the network data service request and clears the “application reply requested” bit in the ULH byte to suppress any response from the slave(s).

The data link layer will clear the “Expect response” bit in the control bytes of the network data request packet accordingly.

The antenna will be toggled each time a network packet is transmitted.]

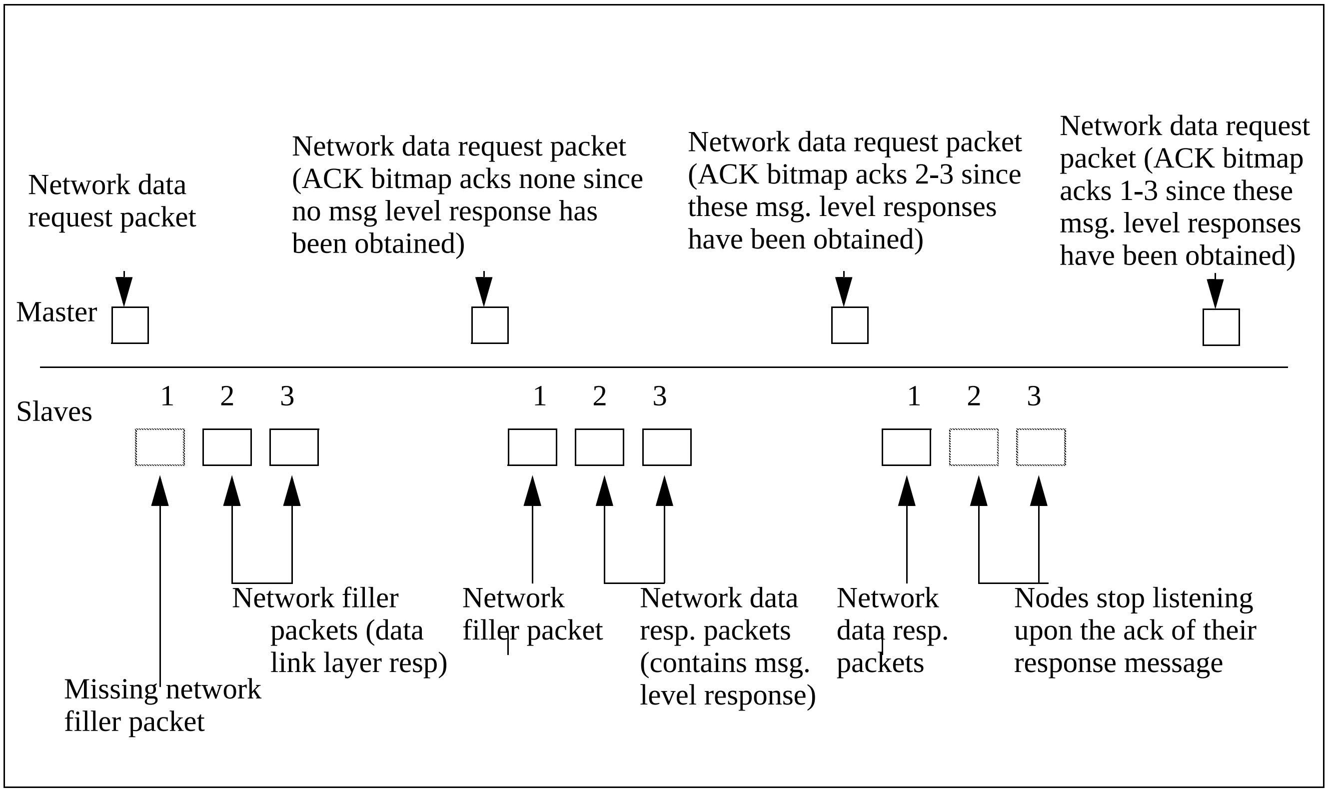


Figure 80: Network data example (with data link & upper layer response)

TLMM_M_318 - NETWORK DATA - SUCCESSFUL RESPONSE TO UPPER LAYER

Upon receiving a valid network data service request, the data link layer shall send a successful network data service response to the upper layer. See **Table 61 (NETWORK DATA SERVICE RESPONSE (MaSter))**

[The data link layer checks CRC and the parameters of the service request.]

TLMM_M_319 - NETWORK DATA - UNSUCCESSFUL RESPONSE TO UPPER LAYER

Upon receiving an invalid network data service request, the data link layer shall send an unsuccessful network data service response to the upper layer. **Table 61 (NETWORK DATA SERVICE RESPONSE (MaSter))**

[The data link layer checks CRC and the parameters of the service request.]

TABLE 61: NETWORK DATA SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	BYTES	DESCRIPTION	VALUES
Type	n/a	1	Service type Network	0x8C - Master
Sub-type	n/a	1	Sub-type of the service.	0x04 - Network data response.
Size	n/a	2	Size of Info.	1
Info	Network Data Service Result	1	Status of the service request	0x01 - Request successful(1) 0x02 - Request invalid in current state (state is Comm) 0x03 - Request invalid in current state (state is Recovery; device scanning)(2) 0x04 - Request invalid in current state (state is Recovery; device not scanning) 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM) 0x0A - Parameter value error(3) 0x0F - Request not serial (another service is in progress) 0x10 - Bad CRC on service request 0x12 - Invalid size (info size >506)(3) 0x19 - Invalid configuration (Tel M only) 0x20 - Master has not BAN key(4) 0x24 - Request invalid in current state (state is disable, reading EEPROM) 0x25 - Buffer/ service request mismatch(5)
CRC	n/a	2	CRC of all fields above.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. All master data packets transmitted and acknowledged successfully.
2. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration).
3. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used.
4. This status only occurs if the master has no BAN key and the “security for wakeup user data/ network packet” parameter is true (see the master parameters to host in the appendix).
5. This status can occur if the master host attempts to send a network service request using service request buffer 1 (instead of buffer 0).

TLMM_M_320 - NETWORK DATA - INDICATION TO UPPER LAYER

A network summary indication shall be sent to the upper layer if any one of the timers expires (the service duration and the LBT timer). See **Table 62 (NETWORK SUMMARY INDICATION (MASTER))**

TABLE 62: NETWORK SUMMARY INDICATION (MASTER)

FIELD	INFO FIELD	BYTES	DESCRIPTION	VALUES
Type	n/a	1	Type of indication	0xAD - Network summary indication
Sub-type	n/a	1	The indication sub-type	n/a
Size	n/a	2	The size of the info field	92 bytes
Info	Coordinator status	1	The indication status of the coordinate	0 - unused 1 - successful, 2 - cancelled, 3 - timeout
	Coordinator data size	1	Coordinator’s date size	0 - 19
	Coordinator network table CRC(1)	2	Coordinator’s network table CRC	
	Coordinator data	19	The indication response message from the coordinator	Network response from coordinator. (i.e. if child #3 is the instigator then coordinator is the destination these bytes are the coordinator’s response) otherwise all bytes are zero.
	Child #1 status	1	The indication status of child #1	0 - unused 1 - successful, 2 - cancelled, 3 - timeout
	Child #1 data size	1	Child #1 data size	0 - 19
	Child #1 network table CRC(1)	2	Child #1’s network table CRC	
	Child #1 data	19	The indication response message from child #1	Network response from child #1. (i.e. if child #3 is the instigator then child #1 is the destination. These bytes are the response of child #1, otherwise all bytes are zero.
	...	...	...	...
	Child #3 status	1	The indication status of Child #3	0 - unused 1 - successful, 2 - cancelled, 3 - timeout
	Child #3 data size	1	Child #3’s data size	0 - 19
	Child #3 network table CRC(1)	2	Child #3’s network table CRC	
	Child #3 data	19	The indication response message from child #3	Network response from child#3 (i.e. if child #1 is the instigator then child #3 is the destination. These bytes are the response of child #3, otherwise all bytes are zero.
CRC	CRC of all the above fields. Two bytes.	2	A two byte CRC of all above fields	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. It is the host's responsibility to check the network table CRC on each responding node. A mismatched CRC indicates the node does not share the same network table as the master. A new network table should be sent to that node.

TLMM_M_321 - NETWORK DATA - RESPONSE COMPLETE AT MASTER

Upon receiving the response from all slaves a network summary indication will be sent to the host (see **Table 62 (NETWORK SUMMARY INDICATION (MASTER))**). The final acknowledgement shall be re-sent until any one of the timers expires (the service duration timer or the LBT timer).

[This is done for the case that a slave does not receive the final acknowledgement and it is still listening in the native mode. Note, the slaves will not respond to the master's final acknowledgement either it receives it or it does not receive it.]

TLMM_M_431 - NETWORK DATA - TRANSMIT SECURED NETWORK DATA PACKET

When the "security user data/network packet" bit is set in the local wake up packet, master should do the following before the transmission of every network data packet:

1. Start the block cipher for message integrity, privacy and freshness checking. The block cipher's key input is set to AES_out[127:64] concatenated with K_{BAN}[63:0]. The block cipher's nonce counter is initialized using master random number (CTR$_{i}$) See **Fig. 81 (Wakeup security - block cipher initialization (to secure native mode packets))**..

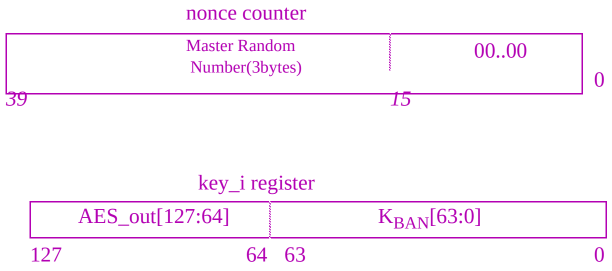


Figure 81: Wakeup security - block cipher initialization (to secure native mode packets)

2. Enable the hardware's security block (security register to on).

[For AES_out[127:64] and master Random number (CTRi), see **Fig. 116 (Calculate AES_out for wakeup security)**. In case there is no user data, a known pattern (0x00,0x00,0x00,0x00) will be used.]

[An idle immediate is performed before 1) and 2)]

[When the "security user data/network packet" bit is cleared, only idle immediate is performed]

5.2.18.4.3 SLAVE

The slave shall go to the standby state after being woken up and completing the initialization service. It shall transfer to the acquire state after completing the polling service.

TLMM_S_434 - NETWORK DATA - PROCESSING POLLING REQUEST

After enabled by a valid local wakeup, if a polling request (upon TEl M) is processed, slave shall start to poll, i.e., listen on the channel with the data rate provided by the local wakeup packet.

[Both the channel and transmit/receive data rate are from local wakeup packet.

“SlaveScansDuringPolling” bit does not apply in this case. There are no channel change in native mode and no async sniffing.

Slave will generate and save 3 bytes of random number. The 3bytes random number will be transmitted in the following Network Filler packets and also used to verify the Network Data Request packets]

TLMM_S_322 - NETWORK DATA - NO ACK OF INVALID REQUEST PACKET

No response packet shall be transmitted to the master if the invalid network data request packet is received.

[Then idle immediate is performed and the slave starts to listen in native mode again.]

TLMM_S_323 - NETWORK DATA - NO ACK OF REQUEST PACKET

No response packet shall be transmitted to the master if the received network data request packet has the “Expect response” bit set to zero in the control bytes.

[The master shall be the transmission-only device. The slave shall go to the disable state after processing the network data request packet.]

After slave starts to listen for network data packet, slave will do the following if the received network data request packet requires a data link response or message level response:

- If the received random numbers in the network data packet does not match to its own, transmit network filler packet (with “ACK bitmap” not set), else send Network data indication to slave host.
- if slave does not receive the network data service request from host, transmit network filler packet (with “ACK bitmap” set), else transmit network data packet to master.

TLMM_S_324 - NETWORK DATA - ACK OF REQUEST PACKET

Upon receiving a valid network data request packet with the “expect response” bit set in control bytes or with the “application response requested” bit set in ULH, a network filler packet shall be transmitted with its own random number. If the random number in the received network data request packet matches to its own random number, the master’s bit in the ack bit map of the network filler response packet shall be set accordingly.

[Note: A network data response packet will be sent in lieu of a network filler packet (if a network service request occurs before the slave has had a chance to transmit a network filler packet).]

TLMM_S_325 - NETWORK DATA - SERVICE INDICATION TO THE UPPER LAYER

For Network Data Request packet “with no acknowledgement”, Network Data indication shall be sent to the upper layer upon receiving a valid network data request packet.

For Network Data Request packet “with message level acknowledgement” or “with data link acknowledgement”, upon receiving a valid network data request packet, slave shall also compare its own random number with the random number in the received Network Data Request packet. If they match, Network Data indication shall be sent to the upper layer.

[For Network Data Request packet “with no acknowledgement”, there is no random number comparison as slave does not transmit at all.

The content of the indication is defined **Table 63 (NETWORK Data INDICATION (SLAVE))**.

It is the host’s responsibility to check the network table CRC of the instigator with the network table CRC of its own. It is also host’s decision on what to do if the two network tables’ CRCs do not match. From a system's point of view it is probably best if the slave host ignores the network data from a therapy point of view. Additionally it is probably best if the slave attempts to respond to the instigator with some application failed status. The master can recognize the network table CRCs are mismatched and correct the slave table. If the slave host did not respond, the instigator wouldn't know if the tables were mismatched or if the instigator and slave were out of telemetry range.]

TABLE 63: NETWORK DATA INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x2D - Network Data Indication
Sub-type	N/A	1	The service request sub-type	0x01 = Network Data Received with message level acknowledgement or with data link acknowledgement 0x02 = Network Data Received with no acknowledgement
Size	N/A	2	The size of the info field	9 to 18 (2bytes CRC+ 1 to 10bytes User Data+6bytes ID)
Info	Network Table CRC (received in the Network data packet)	2	the network table CRC of the instigator	two byte crc
	Data	N	Data received	The received transaction. Network data includes the Upper Layer Header (ULH) as the first byte.
	Source ID	6	The source node’s ID given at manufacturing	source ID (model, sub-model, serial number)

TABLE 63: NETWORK DATA INDICATION (SLAVE)

CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

TLMM_S_326 -NETWORK DATA- ENABLE SECURITY BLOCK

1. Upon receipt of a poll service request or whenever network data response or network filler packet is transmitted, if the "security user data/network packet" bit is set in the local wakeup packet, the hardware's security block will be enabled (security register to on).
2. At that time, the block cipher will be started for message integrity, privacy and freshness checking. The block cipher's key input is set to AES_out[127:64] concatenated with K_{BAN}[63:0]. The block cipher's nonce counter is initialized using CTR_i (see **Fig. 81 (Wakeup security - block cipher initialization (to secure native mode packets))**

[When the "security user data/network packet" bit is cleared, only idle immediate is performed]

TLMM_S_327 - NETWORK DATA - NETWORK DATA RESPONSE

Upon receiving a valid network data request from the upper layer, the data link layer shall transmit a network data response packet to the master. See **Table 64 (NETWORK DATA SERVICE REQUEST (SLAVE))**

The slave will retransmit the network data packet if it receives a network data packet with "this slave's" ack bit not set.

[Note that a network data response packet will be sent in lieu of a network filler packet (if a network service request occurs before the slave has had a chance to transmit a network filler packet).]

TABLE 64: NETWORK DATA SERVICE REQUEST (SLAVE)

FIELD	INFO FIELD	BYTES	DESCRIPTION	VALUES
Type	n/a	1	The type of service network	0x0C - Slave Network Request
Sub-type	n/a	1	The sub-type of service.	0x04 - Network data request.
Size	n/a	2	Size of Info.	1 - 19
Info. .	User Data	1 - 19	User data bytes to transmit	The data to be transmitted. This includes the upper layer header (ULH) as the first byte. See **5.2.11.3** for slave ULH byte.
CRC	n/a	2	CRC for all fields above.	This is given by the upper layer.

TLMM_S_328 - NETWORK DATA - RESPONSE TIME SLOT

The network data filler packets or the network data response packets shall be transmitted to the sender in the designated time slot. The slave's designated time slot shall be obtained using the equation below:

$$\text{Slave Designated Time Slot \#} = \sum_{i=0}^{\substack{\text{Slave's network} \\ \text{table index (this value ranges from 0 to 3)}}} \text{bit value[i] in the destination bitmap}$$

$$\text{Total Response Time Slots} = \sum_{i=0}^{3} \text{bit value[i] in the destination bitmap}$$

Note: This number is used to determine the physical layer timing.

TLMM_S_329 - NETWORK DATA - SEQUENCE NUMBER

The slave shall ignore a duplicate network data request packet.

[A master shall transmit the same network data request packet multiple times to all designated slaves if it does not receive expected the response from any one of the slaves. A master will always transmit the network data request packet with a sequence number of zero. The slave will ignore duplicate network data request. The slave will only process one network data packet and send one Network data indication to host after it receives a valid wakeup packet.]

TLMM_S_330 - NETWORK DATA - INVALID RESPONSE TO INDICATION

The slave data link layer will send an immediate unsuccessful service response to the upper layer if the slave data link layer receives a network data service request when the master does not expect an application response (see **Fig. 48 (Master's ULH byte)** - application reply requested bit). See **Table 65 (NETWORK DATA SERVICE RESPONSE (SLAVE))**.

[Note that the slave host uses a network data service request to send an application response to the master host. This requirement describes the case where the master host does not want an application response, yet the slave host is trying to provide one.]

TLMM_S_331 - NETWORK DATA - SERVICE SUCCESSFUL AT SLAVE

Once acknowledge is received from the master:

For packet level response slave will terminate polling,

For message lever response a successful service response will be sent to the upper layer and polling will be terminated. See **Table 65 (NETWORK DATA SERVICE RESPONSE (SLAVE))**.

[The slave shall stop native mode listening, cancel all timers and return to the standby state.]

TLMM_S_332 - NETWORK DATA - SERVICE UNSUCCESSFUL AT SLAVE

When the polling timer expires while a network data service request has been received but the service response has not been generated, an unsuccessful network service response will be generated to the upper layer. See **Table 65 (NETWORK DATA SERVICE RESPONSE (SLAVE))**.

Slave will also generate polling timeout indication to the upper layer.

Slave will generate polling timeout indication to the upper layer when the polling duration timer expires. see **TLMC_S_039 - POLLING INDICATION TO UPPER LAYER**

[The slave shall stop native mode listening, cancel all timers and return to the standby state.]

TABLE 65: NETWORK DATA SERVICE RESPONSE (SLAVE)

FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	1	The type of service request	0x0C - Network Request

TABLE 65: NETWORK DATA SERVICE RESPONSE (SLAVE)

Sub-type	1	The service request sub-type	0x04 - Network Data response
Size	2	The size of the info field	1
Info	1	Status of the service	0x01 - Request successful 0x10 - Bad CRC on service request; 0x12 - invalid (info) size; 0x13 - invalid request type (This code can occur if the service request is sent using an interrupt other than service request0 (there is a buffer/ service request mismatch). 0x14 - Invalid Request subtype 0x15 - invalid state(1); 0x19 - invalid configuration; 0x1A- polling timeout 0x24 - request invalid during EEPROM read
CRC	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. See **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)** for the invalid states.

5.2.19 SECURITY

The security feature provides for privacy, integrity, freshness, and authenticity.

1. Privacy - communicated messages are to be kept private (encrypted) such that an attacker cannot read the message. For example, an attacker cannot acquire information on the cardiac health of an ICD/IPG patient.
2. Integrity - the integrity of received messages is assured such that a legitimate recipient of a message is able to detect if a message has been manipulated. For example, an attacker cannot change a diabetes message from “deliver 1 unit bolus” to “deliver 100 units bolus”.
3. Freshness - the freshness of received messages is to be assured, such that the legitimate recipient of a message is able to detect if a message is a copy of the old message. For example, an attacker cannot replay a recorded message instructing the delivery of a neurological stim.
4. Authenticity - the authenticity of the message author is to be assured such that the legitimate users will not accept messages from illegitimate sources.

The security service provides a method to deliver (or accept) a body area network (BAN) key. Once the BAN key has been delivered, the security service provides a method to open a secure session (using a normal session key). The security service allows for a proximity exchange of the BAN key or ephemeral BAN key. In addition to a normal session, the BAN key can also be used to secure user data contained in a wakeup packet.

5.2.19.1 SECURITY SERVICE — GENERAL REQUIREMENTS

The security service consists of several sub-services:

1. Create BAN key/ephemeral BAN key local
2. Assert BAN key/ephemeral BAN key local
3. Request BAN key/ephemeral BAN key local
4. Request BAN key/ephemeral BAN key via proximity switch
5. Authorize delivery of BAN key/ephemeral BAN key
6. Assert EBAN key duration
7. Update proximity status

TLMM_MS_403 - SECURITY - UPPER LAYER PARAMETERS

When invoking a Security service request, the Upper Layer will provide information as specified in **Table 66 (SECURITY SERVICE REQUEST (MASTER OR SLAVE))**. If the service type is correct but the sub-type is not equal to any of the values specifed for the master and slave respectively, the status of “invalid security sub-type” will be returned in the service response.

[The service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))** and **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.]

TABLE 66: SECURITY SERVICE REQUEST (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	Security service request 0x0D - Slave 0x8D - Master
Sub-type	N/A	1	The service request sub-type	0x00 - Create BAN key/ephemeral BAN key local (Master/Slave) 0x01 - Assert BAN key/ephemeral BAN key local (Master/Slave) 0x02 - Request BAN key/ephemeral BAN key local (Master/Slave) 0x03 - Request BAN key/ephemeral BAN key via proximity switch (Master) 0x04 - Authorize delivery of BAN key/ephemeral BAN key (Slave) 0x05 - Assert EBAN duration (Slave) (Note master does not start an EBAN timer) 0x06 - Update proximity status (slave)
Size	N/A	2	The size of the info field	25
Info[1]	Slave ID	6	Station ID which is the destination of the request	Slave ID (Model ID, SubModel ID, and Serial# must all be specified.). Model ID must match the model ID in the config file. Slave ID may not change from one open request to another (without an intervening close service request). Note: this field is only pertinent to subtype 0x03.
	BAN key type	1	The type of the BAN key: this service request or the data link layer's current BAN key	0 - BAN key 1 - ephemeral BAN key Note: Only the least significant bit of this byte is pertinent. All other bits are ignored. This parameter is NOT pertinent to Assert EBAN duration and Update proximity status service requests. Note: this field is pertinent to all subtypes except subtype 0x05 and 0x06.
	BAN key / ephemeral BAN key	16	The user supplied BAN key	This parameter is only pertinent to the assert BAN key/ephemeral BAN key local service (subtype 0x01). The BAN key type field determines whether this value represents the BAN key or ephemeral BAN key.
	EBAN key Duration	2	EBAN key duration	1 second resolution (valid for Assert EBAN key duration)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

1. The security service has two additional parameter which control its behavior: security service & ephemeral K_{BAN} timer duration. See **Table 123 (Master parameters from host)**.

5.2.19.2 CREATE BAN KEY / EPHEMERAL BAN KEY LOCAL

A station’s host can ask its data link layer to create a BAN or EBAN key. The following briefly summarize this service:

- A station’s upper layer generates the create BAN key or ephemeral BAN key service request to its data link layer.
- The station’s data link layer uses the AES block cipher to create a pseudo random number. This pseudo random number is the new BAN key or ephemeral BAN key.
- The station’s data link layer returns a successful create BAN key or ephemeral BAN key service response to its upper layer. Note that the BAN key or ephemeral BAN key is not returned to the upper layer; the upper layer is just told that a new key has been successfully created.

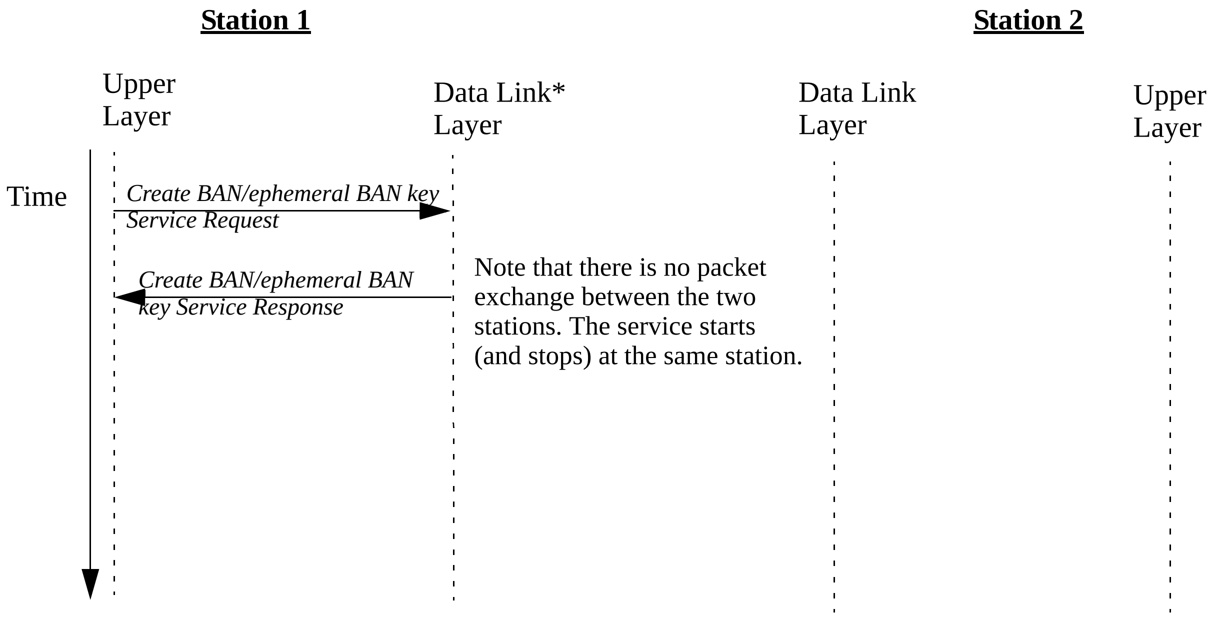


Figure 82: Create BAN/EBAN Key Sequence

TLMM_MS_334 - SECURITY - CREATE BAN KEY/EPHEMERAL BAN KEY LOCAL

The master or slave will do the following upon the receipt of a create BAN key or ephemeral BAN key service request:

1. Create pseudo random number. See the appendix, section **7.3 Pseudo Random Number Generation.**
2. Save the pseudo random number as the new BAN key or ephemeral BAN key.
3. Send a successful create BAN key or ephemeral BAN key service response to the upper layer.
4. The EBAN key duration timer will also be started on the slave when an EBAN key is created on the slave due to this service request.

[The slave service response is generated per **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))** and the master service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))**]

5.2.19.3 ASSERT BAN KEY/EPHEMERAL BAN KEY LOCAL

A station’s host can assert a BAN key or ephemeral BAN key to its data link layer. Upon receiving the service request, the data link layer will save the BAN key or ephemeral BAN key and returns a successful service response to its upper layer.

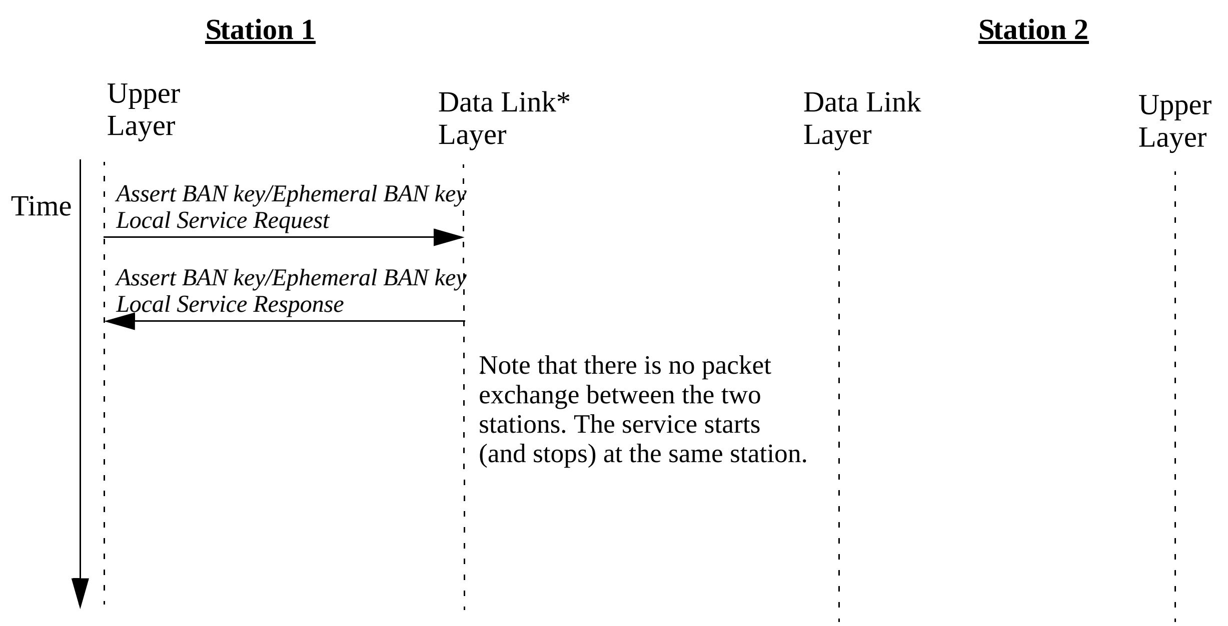


Figure 83: Assert BAN Key/Ephemeral BAN Key Sequence Diagram

TLMM_MS_335 - SECURITY - ASSERT BAN KEY/EPHEMERAL BAN KEY LOCAL

The master or slave will do the following upon the receipt of an Assert BAN key or ephemeral BAN key service request:

1. Save the BAN key or ephemeral BAN key contained in the service request as the new BAN key or ephemeral BAN key.
2. Send a successful assert BAN key or ephemeral BAN key service response to the upper layer.
3. The EBAN key duration timer will also be started on the slave when an EBAN key is asserted to the slave.

[The slave service response is generated per **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))** and the master service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))**]

5.2.19.4 REQUEST BAN KEY/EPHEMERAL BAN KEY LOCAL

A station's host can request a BAN key /ephemeral BAN key be provided from its data link layer. Upon receiving the service request, the station's data link layer will read the BAN key/ephemeral BAN key from the memory and returns it to its upper layer via a service response.

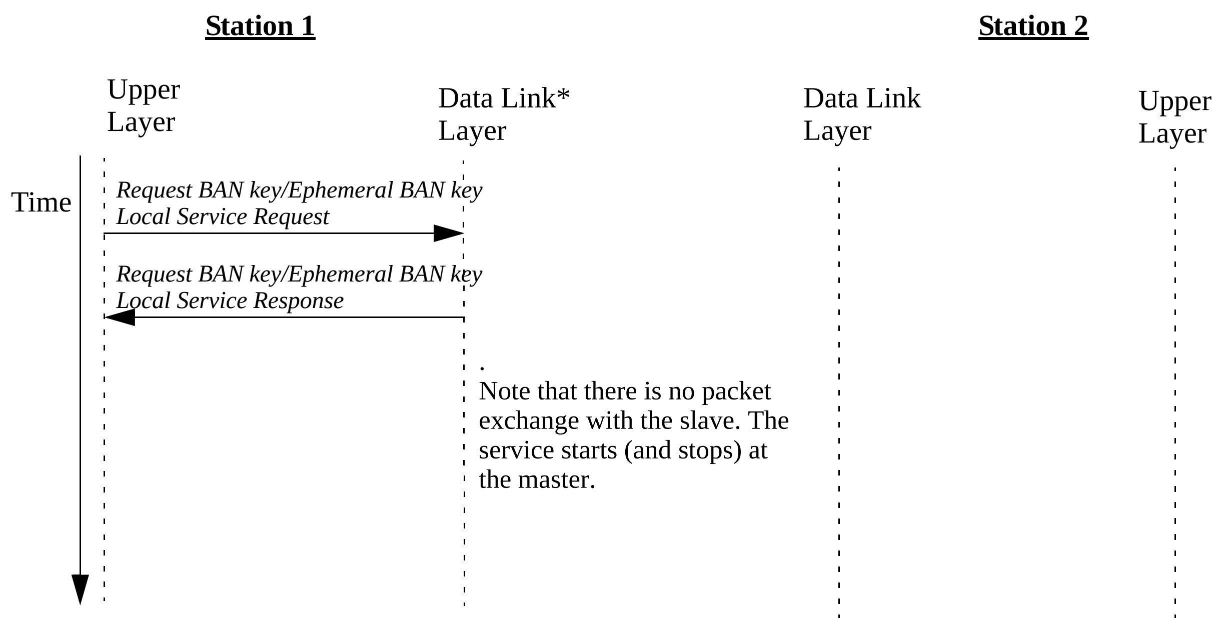


Figure 84: Request BAN Key/Ephemeral BAN Key Sequence Diagram

TLMM_MS_336 - SECURITY - REQUEST BAN KEY OR EPHEMERAL BAN KEY LOCAL

The master or slave will do the following upon the receipt of a Request BAN key service request:

1. Read the BAN key or ephemeral BAN key from the memory;
2. Return the BAN key or ephemeral BAN key to its upper layer via a service response.

[The service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))** and **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**]

TLMM_MS_401 - SECURITY - MASTER/SLAVE REJECTS KEY REQUEST IF IT HAS NO BAN/EBAN KEY

The slave/master will send an unsuccessful request BAN/EBAN key local response if the slave/master has no corresponding permanent BAN key or EBAN key as requested.

[This will occur if the host has never created or asserted a permanent BAN key or EBAN key since the last reset]

TABLE 67: SECURITY SERVICE RESPONSE (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	Security service response 0x8D - Master
Sub-type	N/A	1	The service response sub-type	0x00 - Create BAN key/ephemeral BAN key local 0x01 - Assert BAN key/ephemeral BAN key local 0x02 - Request BAN key/ephemeral BAN key local 0x03 - Request BAN key/ephemeral BAN key via proximity switch
Size	N/A	2	The size of the info field	17(1)
Info	Security service status	1	Status of the service request	0x01 - Request successful 0x02 - Request invalid in current state (state is Comm) 0x03 - Request invalid in current state (state is Recovery; slave scanning)(2) 0x04 - Request invalid in current state (state is Recovery; slave not scanning) 0x05 - Request invalid in current state (state is Inactive) 0x07 - Request invalid in current state (listening for Medical Event) 0x08 - Request invalid in current state (state is disable, not reading EEPROM); 0x0A - Parameter value error(3); 0x0B - Service duration timeout(4)(5); 0x0C - Service cancelled (interrupted by emergency)(6); 0x0F - Request not serial (another service is in progress); 0x10 - Bad CRC on service request; 0x12 - invalid (info) size(7) 0x14 - invalid security sub-type; 0x19 - invalid configuration; 0x1D - Cancelled due to PLL lock error with trim fail(8) 0x1E - Slave has no permanent BAN key(9); 0x1F - Master has no EBAN key (10); 0x20 - Master has no permanent BAN key(11); 0x22 - Slave host never authorized the key delivery(12) 0x23 - Slave delivered key type mismatch(13); 0x24 - Request invalid in current state (state is disable, reading EEPROM) (not applicable to sub-type = 0x03 - Request BAN key/ephemeral BAN key via proximity switch) 0x25 - Buffer/ service request mismatch(14)
	BAN key / ephemeral BAN key	16	the value of the BAN key/ephemeral BAN key	This is only pertinent to the successful response (status = 0x01) to the request BAN key/ephemeral BAN key local service request;

CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. This value only applies when the correct service request buffer is used.
2. The slave is considered to be still scanning if (time now - time of last valid slave packet) is less than (slave recovery duration + slave polling duration). Note that a service duration timeout will result in a Data Link response status of 3 or 4 (invalid state, state is Recovery).
3. This is only pertinent to remote key request service. This status will occur if the slave ID in the service request is incorrect.
4. The status service duration timeout occurs if the master received no telemetry packets from the slave during the remote key delivery service. Note that if telemetry packets were received and service duration expired the master host will receive a service status (0x22) slave host never authorized key delivery.
5. Note that these status values are pertinent only to the remote key request service. They are not pertinent to local create, local request, or local assert.
6. Note that these status values are pertinent only to the remote key request service. They are not pertinent to local create, local request, or local assert.
7. If the request info size is greater than 506 or does not match to the request info size, 0x12 will be used.
8. Note that these status values are pertinent only to the remote key request service. They are not pertinent to local create, local request, or local assert.
9. Note that these status values are pertinent only to the remote key request service. They are not pertinent to local create, local request, or local assert.
10. Note that this status value is only pertinent to the local request EBAN key service.
11. Note that this status value is only pertinent to the local request BAN key service
12. Note that these status values are pertinent only to the remote key request service. They are not pertinent to local create, local request, or local assert.
13. Note that these status values are pertinent only to the remote key request service. They are not pertinent to local create, local request, or local assert.
14. This status can occur if the master host attempts to send a security service request using service request buffer 1 (instead of buffer 0).

TABLE 68: SECURITY SERVICE RESPONSE (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	Security service response 0x0D - Slave
Sub-type	N/A	1	The service response sub-type	0x00 - Create BAN key/ephemeral BAN key local 0x01 - Assert BAN key/ephemeral BAN key local 0x02 - Request BAN key/ephemeral BAN key local 0x04 - Authorize delivery of BAN key via proximity switch 0x05 - Assert EBAN key duration 0x06 - Update proximity status
Size	N/A	2	The size of the info field	17 (For all cases except Security service status =0x13) or 1 (when Security service status =0x13)
Info	Security service status	1	Status of the service request	0x01 - Request successful 0x10 - Bad CRC on service request; 0x12 - invalid (info) size(1); 0x13 - invalid request type (This code can occur if the service request is sent using an interrupt other than service request0 (there is a buffer/ service request mismatch). 0x14 - invalid security sub-type; 0x15 - invalid state(2); 0x18 - Slave has no EBAN key(3); 0x19 - invalid configuration; 0x1E - Slave has no permanent BAN key(4); 0x24 - request invalid during EEPROM read
	BAN key / ephemeral BAN key	16	the value of the BAN key/ephemeral BAN key	This is only pertinent to the successful response (status = 0x01) to the request BAN key/ephemeral BAN key local service request;
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. If the request info size is greater than 506 or does not match to the request info size, 0x12 will be used.
2. See **Table 7 (EXTERNAL SERVICES REQUEST VERIFICATION)** for the invalid states.
3. Note that this status value is only pertinent to the request EBAN key local service request;
4. Note that this status value is only pertinent to the request BAN key local service request;

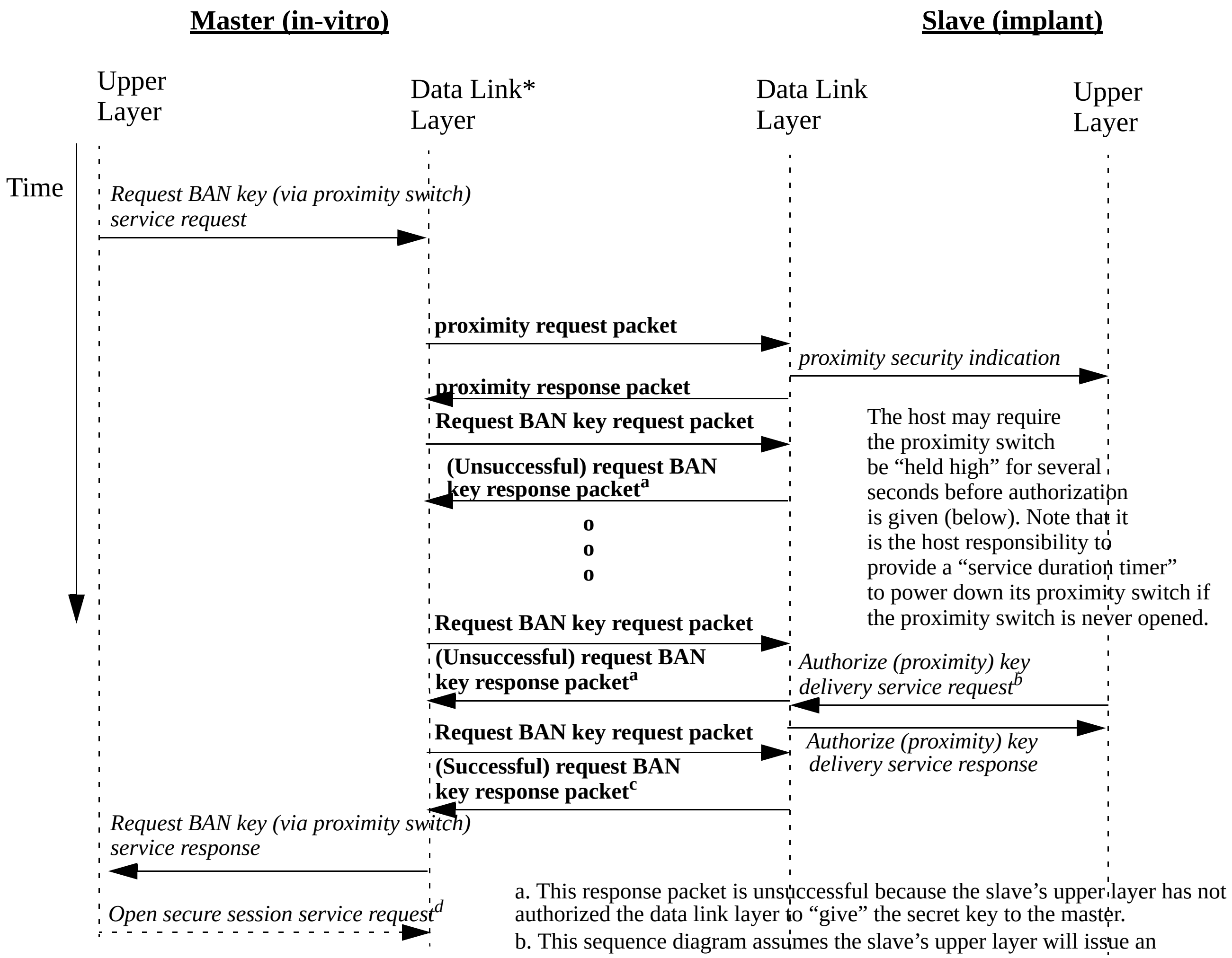
5.2.19.5 REQUEST BAN KEY/EBAN KEY VIA PROXIMITY SWITCH

Clinicians who lack appropriate authentication material may need to communication with an implanted device. A short-range "proximity switch" service is provided.

The following bullets briefly summarize this service:

- The master's upper layer generates the request BAN/EBAN key (via proximity) service request to the data link layer.
- The master's data link layer selects a channel and transmits proximity switch security requests. The station remains on the selected channel until a response is heard or a service duration timeout occurs.
- The slave's data link layer sends a proximity switch security event indication to the slave's upper layer (host).
- The master's data link layer transmits the request BAN/EBAN key packet if the proximity switch security packet is acknowledged.
- The slave's data link layer rejects the request BAN/EBAN key requests until the slave's upper layer sends an authorize (proximity switch) key delivery service request. The slave's upper layer will do this when the user has physically opened the proximity switch. The user may open the proximity switch by holding a magnet by the patient's device for a finite amount of time. This could be detected with a reed switch. Note that the proximity switch is short range. The clinician needs to be within the patient's personal space to open the proximity switch. This means the proximity switch cannot be opened anonymously.
- The master's data link layer sends a (successful) request BAN/EBAN key service response to the master's upper layer if it obtains the BAN/EBAN key from the slave.

An operational example of the request BAN/EBAN key (via proximity switch) service is shown in **Fig. 85 (Request BAN/EBAN key (via proximity switch) service request)**.



a. This response packet is unsuccessful because the slave’s upper layer has not authorized the data link layer to “give” the secret key to the master.
b. This sequence diagram assumes the slave’s upper layer will issue an “authorize (proximity) BAN key delivery” service request (upon detecting the user has opened the proximity). The authorization will end when the security service duration times out.
c. This response packet is now successful because the slave’s upper layer has authorized the data link layer to “give” the secret key to the master.
d. This sequence diagram assumes the master’s upper layer will issue an open secure session service request after obtaining the BAN key.

* Channel mapping and wakeup are not shown.

Figure 85: Request BAN/EBAN key (via proximity switch)

TLMM_M_337 - SECURITY - PROXIMITY SECURITY REQUEST PACKET FORMAT

The proximity security request packet has the format indicated in **Fig. 86 (Proximity Security Request Packet)**

.

proximity Security Control 5.25 bytes	Source ID (master ID) 6 bytes	Destination ID (slave ID) 6 bytes	BAN/EBAN Key type 1 byte	Zero fill (17 bytes)	MAC[a] 3 bytes	Reed Solomon 9 bytes

a. Message authentication code

Figure 86: Proximity Security Request Packet

TLMM_S_338 - SECURITY - PROXIMITY RESPONSE PACKET FORMAT

The proximity security response packet has the format indicated in **Fig. 87 (Proximity Security Response Packet)**.

proximity Security Control 5.25 bytes	Destination ID (master ID) 6 bytes	Source ID (slave ID) 6 bytes	BAN/EBAN Key type 1 byte	Remaining polling duration 2 bytes	Zero fill (15 bytes)	MAC[a] 3 bytes	Reed Solomon 9 bytes

a. Message authentication code

Figure 87: Proximity Security Response Packet

TLMM_M_339 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH REQUEST PACKET FORMAT

The request BAN/EBAN key via proximity switch request packet has the format indicated in **Fig. 88 (Request BAN/EBAN key via proximity switch request packet)**.

.

request key Control 5.25 bytes	Source ID (master ID) 6 bytes	Destination ID (slave ID) 6 bytes	BAN/EBAN Key type 1 bytes	EBAN Key duration (2 bytes)	Zero fill 15 bytes	MAC[a] 3 bytes	Reed Solomon 9 bytes

a. Message authentication code

Figure 88: Request BAN/EBAN key via proximity switch request packet

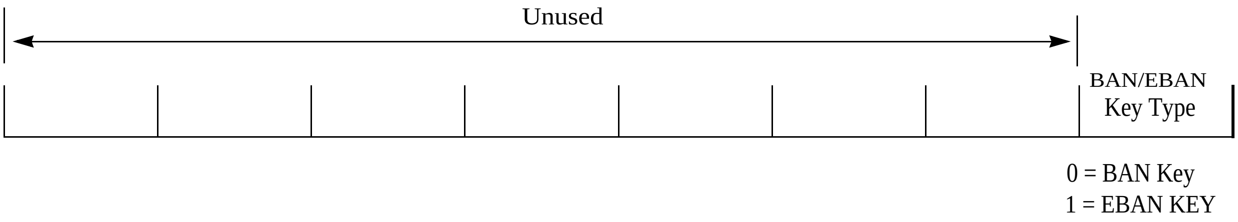


Figure 89: BAN/EBAN key type byte

TLMM_S_340 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH RESPONSE PACKET FORMAT

The request BAN/EBAN key via proximity switch response packet has the format indicated in **Fig. 90 (Request BAN/EBAN key via proximity switch response packet)**.

and **Fig. 91 (BAN/EBAN key status byte)**.

request key Control 5.25 bytes	**Destination ID (master ID) 6 bytes**	**Source ID (slave ID) 6 bytes**	**BAN key status[a] 1 bytes**	**BAN key[b] 16 bytes**	**Remaining polling duration 1 byte**	**MAC[c] 3 bytes**	**Reed Solomon 9 bytes**

[a] The BAN key status can indicate the request is rejected if the proximity is not open.
[b] The BAN key will be all zero if the request is rejected.
[c] Message authentication code.

Figure 90: Request BAN/EBAN key via proximity switch response packet

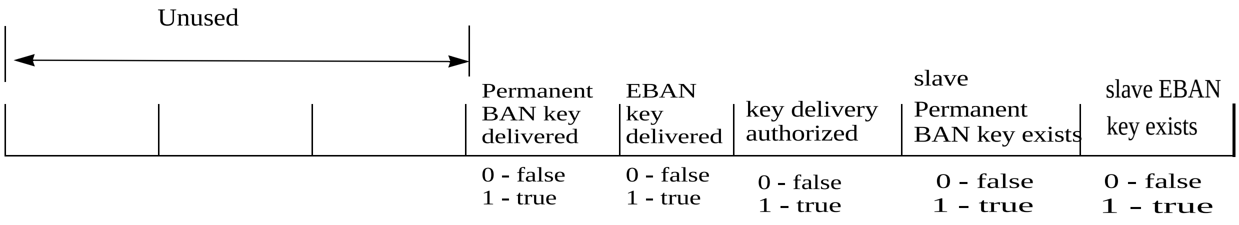


Figure 91: BAN/EBAN key status byte

[Note that if the remaining polling duration is greater than 255 seconds, this field will be set to 255 seconds. Also note that the “slave EBAN key exists” bit is only set when the EBAN key duration timer is running.]

5.2.19.5.1 MASTER

TLMM_M_341 - SECURITY - TRANSMIT PROXIMITY SECURITY REQUEST PACKET

Upon receiving a request BAN key via proximity switch service request, the master will start a service duration timer and transmit the proximity switch security request packet.

[Start the service duration timer; select the least interfered channel (takes approximately 275 +/- 50 msec(5)); start the Listen Before Talk (LBT) timer; perform instigator wireless wakeup, transmit the proximity switch security request packet, send a “first request packet sent” event indication to the Upper Layer (see **Table 23 (FIRST REQUEST PACKET SENT INDICATION (MASTER))**).

Note that instigator wireless wakeup is suppressed if the master knows the slave is scanning. Time now, time of last uplink, and remaining polling duration at the time of last uplink are used to determine this.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup if not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMM_M_342 - SECURITY - NO RESPONSE TO PROXIMITY SECURITY REQUEST

If no response is received, the master will resend the proximity switch security request until the Service Duration timer is expired.

[If no valid proximity switch security response packet is received, the master will re-transmit the proximity switch security request packet on the alternate antenna.

This process shall be continued until a valid proximity switch security response packet is received, a Service Duration Timeout occurs, or a Listen Before Talk timer expires.]

TLMM_M_343 - SECURITY - VALID RESPONSE TO PROXIMITY SECURITY REQUEST

If a valid proximity security response packet is received, the master will transmit a request BAN/EBAN key packet to the slave.

TLMM_M_344 - SECURITY - NO RESPONSE OR REJECTION TO REQUEST BAN/EBAN KEY VIA PROXIMITY SWITCH

If no response is received or if the BAN/EBAN key request is rejected (because the proximity switch is not yet open), the master will resend the request BAN/EBAN key packet until successful or until the Service Duration timer is expired.

[The master will re-transmit the request BAN/EBAN key packet on the alternate antenna if no valid BAN/EBAN key response packet is received. The master will re-transmit the request BAN/EBAN key packet on the same antenna if the BAN/EBAN key request is rejected (since the proximity switch is not yet open).

This process shall be continued until a valid BAN/EBAN key response packet is received, a Service Duration Timeout occurs, or a Listen Before Talk timer expires.]

5. CCA of the MICS band takes approximately 275 msec. CCA of the MEDS band takes approximately 550 msec. CCA of both bands takes approximately 825 msec. Note that if HDR mode is enabled, assessment of each channel, antenna pair is lengthened by 5 msec (to allow the HDR synthesizer to stabilize). This means the CCA of the MICS band is lengthened from 275 msec to 375 msec

TLMM_M_346 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH - RESPONSE TO UPPER LAYER

A request BAN/EBAN key via proximity switch service response will be generated to the Upper Layer when the service is completed.

[The content of the response to the Upper Layer is specified in **Table 67 (SECURITY SERVICE RESPONSE (MASTER))**.]

TLMM_M_347 - SECURITY - LBT TIMEOUT DURING REQUEST BAN/EBAN KEY EXCHANGE VIA PROXIMITY SWITCH SERVICE

If the listen before talk (LBT) timer expires during a request BAN/EBAN key via proximity switch service, the master will do the following:

1. Re-select the least interfered channel for communication,
2. Start the Listen Before Talk timer when a good channel is found,
3. Toggle the antenna,
4. Transmit a wakeup packet,
5. Transmit a proximity security request.

[Note that “transmit wakeup packet” is suppressed if the master knows the slave is scanning. Time now, time of last uplink, and remaining polling duration at the time of last uplink are used to determine this.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup is not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

5.2.19.5.2 SLAVE

TLMM_S_348 - SECURITY - SLAVE RESPONSE TO A VALID PROXIMITY SWITCH SECURITY REQUEST PACKET

The slave will send a proximity switch security response packet upon receiving a valid proximity switch security request packet. Additionally the slave will send a proximity security indication to the slave’s upper layer (host). The indication to the upper layer (host) will not be generated more than once a second. See **Table 69 (PROXIMITY SECURITY INDICATION (SLAVE))**

[Upon receipt of the proximity security indication, the slave’s upper layer should do the following:

1. Power up the proximity switch circuitry

2. Determine if the user has “opened the proximity switch”. This may be polling a reed switch, etc. The proximity switch is short range. This ensures that the proximity switch is not opened anonymously.

3. Send an authorize (proximity switch) BAN/EBAN key delivery to the data link layer

4. Power down the proximity switch circuitry]

TABLE 69: PROXIMITY SECURITY INDICATION (SLAVE)

FIELD	FIELD INFO	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x36 - proximity security
Sub-type	N/A	1	The indication sub-type	n/a
Size	N/A	2	The size of the info field	1
Info	Proximity Security indication status	1	Status of indication	0x00 - BAN key requested 0x01 - EBAN key requested
CRC	N/A	2	CRC of all the above field	This value is computed by data link layer and is used by upper layer to verify the integrity of the indication

TLMM_S_349 - SECURITY - SLAVE REJECTS A VALID REQUEST BAN/EBAN KEY VIA PROXIMITY SWITCH REQUEST

The slave will send an (unsuccessful) request BAN/EBAN key (via proximity switch) response packet under the following conditions:

1. The slave receives a request BAN/EBAN key (via proximity switch) request packet, and
2. The slave is not authorized to deliver the BAN/EBAN key to the master. Specifically, the security (authorize key delivery) service duration timer is not active.

TLMM_S_350 - SECURITY - SLAVE ACCEPTS A VALID REQUEST BAN/EBAN key VIA PROXIMITY SWITCH REQUEST

The slave will generate a EBAN key, for EBAN key requested, and will send a (successful) request BAN/EBAN key (via proximity switch) response packet under the following conditions:

1. Attenuate the transmit power according to the settings in slave configuration table. Once the transmitting process of response is finished original transmission power should be restored.
2. The slave receives a request BAN/EBAN key (via proximity switch) request packet, and
3. The slave is authorized to deliver the BAN/EBAN key to the master (it has “recently” received an authorize BAN/EBAN key delivery service request from the upper layer (host)).

When delivering the BAN/EBAN key the transmit power should be attenuated according to the settings in slave configuration **Table 124 (SLAVE PARAMETERS FROM HOST)**, once the transmitting process of response is finished original transmission power should be restored. BAN/EBAN key delivery is authorized upon receipt of the authorize BAN/EBAN key delivery (via proximity switch) service request. A service request timer is started upon receipt of the service request. BAN/EBAN key delivery is no longer authorized when the service duration timer expires.

[The EBAN key is generated when slave is entering the Acquire state and delivered when sending the request (EBAN) key response packet.]

TLMM_S_351 - SECURITY - AUTHORIZATION TO GIVE BAN/EBAN KEY SERVICE RESPONSE

When authorize BAN/EBAN key delivery request is received then response will be generated See **Table 66 (SECURITY SERVICE REQUEST (MASTER OR SLAVE))**.

[The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE)).**The security service duration will be started upon the receipt of a authorize key delivery service request. The security service duration timer will be re-started upon the receipt of an authorize key delivery service request if the security service duration timer is already running (from a previous update proximity status service request.) When the key delivery is authorized, the slave will set the “unidentified proximity transmitter near” bit and the “specific proximity transmitter near” bit. See **Fig. 35 (Proximity status type byte)** for the bit definition.

Also note that the EBAN key duration timer will be restarted on the slave upon the delivery of the EBAN Key to the master.]

TLMM_S_352 - SECURITY - USAGE OF EPHEMERAL BAN KEY

The use of ephemeral BAN key is limited by the ephemeral BAN key duration parameter. Once this timer expires a secure session may not be opened (via the ephemeral BAN key). If a secure session is opened before the timer expires, the timer is cancelled. The ephemeral BAN timer is restarted upon a close session or channel recovery timeout. If the timer expires, the ephemeral BAN key is no longer valid. Note that the timer is cancelled if a new session is successfully started using the ephemeral BAN key. The ephemeral BAN key timer also expires when the slave receives a Shut Down request from its host.

[The ephemeral BAN key timer value is received from Master as part of proximity switch request packet. The EBAN key duration timer can also be updated via assert EBAN duration service request]

5.2.19.6 SECURITY - ASSERT EBAN KEY DURATION

This service provides an ability for slave host to change the EBAN key duration under various circumstances (i.e. longer implant times, etc.).

TLMM_S_405 - SECURITY - ASSERT EBAN KEY DURATION

The EBAN key duration will be updated upon the receipt of an assert EBAN key duration service request. If the EBAN key duration timer is running, the new value will take effect immediately. A service response will also be sent to the upper layer.

[Note: The EBAN key duration should be greater than the channel recovery duration by the host. When the existing EBAN key duration timer is running and new values is provided via assert EBAN key duration service request the new value will take effect immediately - the expiration of the running timer will be based on the new value. Also note that because the EBAN key duration timer has a resolution of one second, it may take up to one second for it to expire if the newly asserted EBAN key duration implies the timer should expire immediately.

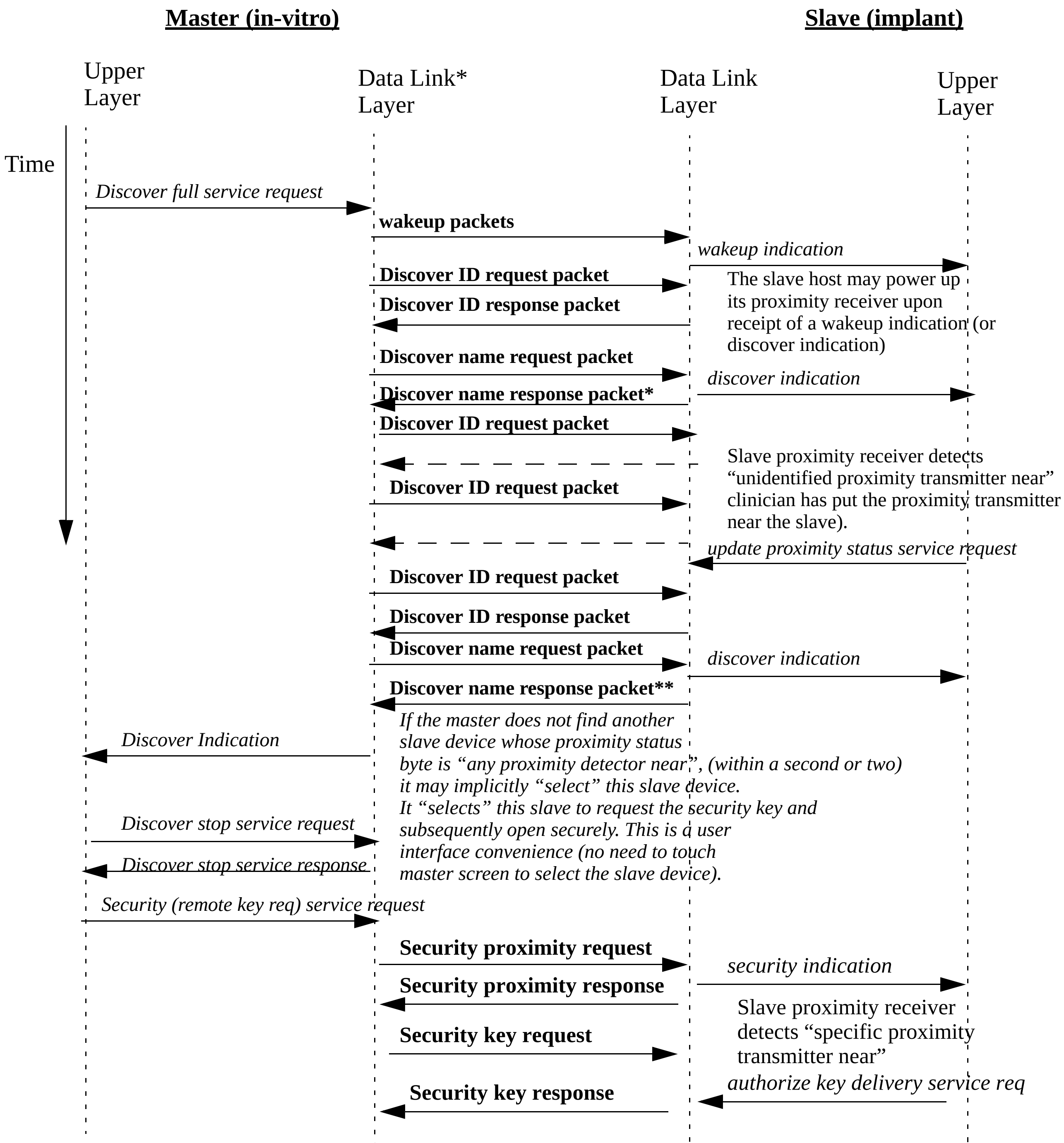
The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.]

5.2.19.7 SECURITY - UPDATE PROXIMITY STATUS

A short-range proximity receiver is provided on some slave devices. This proximity receiver may be used during the security key delivery process. The proximity receiver (when enabled) can detect one of the following:

1. No proximity transmitters near.
2. Unidentified proximity transmitter near. Detection occurs when a signal is received by the proximity receiver whose amplitude exceeds a pre-defined threshold. Note that the transmitter must be fairly close to the receiver (approximately 6 inches). Also note that “nearness” may be detected very shortly after a proximity transmitter comes within range of the slave’s proximity receiver (~100 msec).
3. Specific proximity transmitter near. Detection occurs when the slave’s proximity receiver detects a specific data pattern. This may require that the proximity transmitter is held close to the proximity receiver for a significant amount of time (possibly many seconds). The specific data pattern is recommended to be the master station ID. This makes the subsequent task of security key delivery easier

An operational example of the update proximity status service is shown in **Fig. 85 (Request BAN/EBAN key (via proximity switch) service request)**.



The slave host may power up its proximity receiver upon receipt of a wakeup indication (or discover indication)

Slave proximity receiver detects “unidentified proximity transmitter near” clinician has put the proximity transmitter near the slave).

If the master does not find another slave device whose proximity status byte is “any proximity detector near”, (within a second or two) it may implicitly “select” this slave device. It “selects” this slave to request the security key and subsequently open securely. This is a user interface convenience (no need to touch master screen to select the slave device).

Slave proximity receiver detects “specific proximity transmitter near”

* Proximity status byte says no proximity transmitter near.
** Proximity status byte says any proximity transmitter near. Note that the collision avoidance timer is cancelled upon the receipt of the update proximity status service request.

Figure 92: Update proximity status service request

TLMM_S_419 - SECURITY - UPDATE PROXIMITY STATUS

The slave will do the following upon receipt of an update proximity status service request:

1. Start the security service duration timer
2. Update the proximity status byte (set the “unidentified proximity transmitter near” bit). See **Fig. 35 (Proximity status type byte)**.
3. Send an update proximity service response to the upper layer

[The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.

Also note that the discover collision avoidance timer is cancelled upon the receipt of the update proximity status service request]

TLMM_S_420 - SECURITY - SERVICE DURATION TIMEOUT

The slave will do the following when the security service duration timer expires:

1. Clear the proximity status byte (clear the “unidentified proximity transmitter near” bit and the “specific proximity transmitter near” bit). See **Fig. 35 (Proximity status type byte)**.

[The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.

Also note that the security key delivery is no longer allowed once this timer expires.]

5.2.20 POWER INHIBIT

Power inhibit is a unique feature which is an interaction between the host and telemetry module to better manage peak current constraints. Presently this feature is intended for the slave (implant) based on Neuro applications. This feature could be used by other application as well. A host applications may not be able to support simultaneous (high current) data collection and telemetry (due to peak current constraints). A host can assert power inhibit while its (host's) high current data is collected. It can then deassert power inhibit to transmit the collected data.

The power inhibit feature allows a host to pause telemetry M during its increase power consumption periods. A power inhibit bit is used by the host to assert and deassert the power inhibit condition by generating an interrupt to the telemetry M slave. When host asserts a power inhibit interrupt it causes telemetry functionality to pause, thus allowing host to support its own high current features. The key goal is to:

- Minimize the impact on the power consumption as demand increases for both the host and telemetry module (i.e. during battery voltage measurements, sounding of an alarm, or the use of electronic port finding).

An example of telemetry M feature(s) during which power consumption may be high are: EEPROM access, Telemetry session (transmit and receive), sniff periods (cpu off), and housekeeping periods. The host features may include EEPROM/Flash access, and Neuro's electronic port finding, alarm, or pump stroke.

Power inhibit functionality is supported by two separate algorithms:

- **Idle Immediate**: is a default option when the host asserts power inhibit CPU speed is reduced to 1.9MHz and when power inhibit is deasserted then channel recovery will be attempted to recover the link. The channel recovery is unconditionally entered each time power inhibit is asserted thus impacting the data throughput. Note that if (idle immediate) power inhibit is asserted before a session is established, the radio will be disabled. When power inhibit is deasserted the radio will be re-enabled (sniff or native mode listening).
- **Roll through:** in this option the RF analog is turned off (the telem M digital state machine and firmware is allowed to run at its programmed CPU speed). This roll through mode uses slightly more current, but can provide increased slave throughput (since channel recovery can be avoided). Channel recovery is avoided by dynamically lengthening the recovery thresholds (see **Table 71 (POWER INHIBIT MODES)** for details). Note that the lengthened recovery thresholds are retained until power inhibit roll through is disabled. Frequency agility will be compromised if the power inhibit is enabled and left enabled indefinitely. This means the user may experience longer than normal interruptions in their telemetry (because it takes longer to switch to a new channel).

If power inhibit is asserted during polling operation (when the parameter slave scans during polling is true), the slave will not change the channel but will keep the polling timer active. When the host deasserts power inhibit, the channel is scanned again. Note that the power inhibit deassertation duration must be greater than the channel listen duration or the slave will not change channels during polling. If short, assertion/ deassertion intervals are needed, the parameter slave scans during polling should be set to false. The table below quantifies the definition of "short" deassertion periods.

TABLE 70: RECOMMENDED POWER INHIBIT DEASSERTION DURATION

NON-COMMUNICATION RATE	SUPER FRAME DURATION(1)	RECOMMENDED CHANNEL LISTEN TIME(2)	RECOMMENDED POWER INHIBIT DEASSERTION DURATION(3)
48 Kbps (1 DOWNLINK, 5 UPLINK)	57.7 msec	173.1 msec	173.1 msec or greater
97 Kbps (1 DOWNLINK, 5 UPLINK)	34.4 msec	103.2 msec	103.2 msec or greater
190 Kbps (1 DOWNLINK, 5 UPLINK)	16.2 msec	48.6 msec	48.6 msec or greater
380 Kbps (1 DOWNLINK, 5 UPLINK)	10.3 msec	30.9 msec	30.9 msec or greater
48 Kbps (1 DOWNLINK, 1 UPLINK)	26.7 msec	80.1 msec	80.1 msec or greater
97 Kbps (1 DOWNLINK, 1 UPLINK)	18.8 msec	56.4 msec	56.4 msec or greater
190 Kbps (1 DOWNLINK, 1 UPLINK)	8.2 msec	24.6 msec	24.6 msec or greater
380 Kbps (1 DOWNLINK, 1 UPLINK)	6.3 msec	18.9 msec	18.9 msec or greater

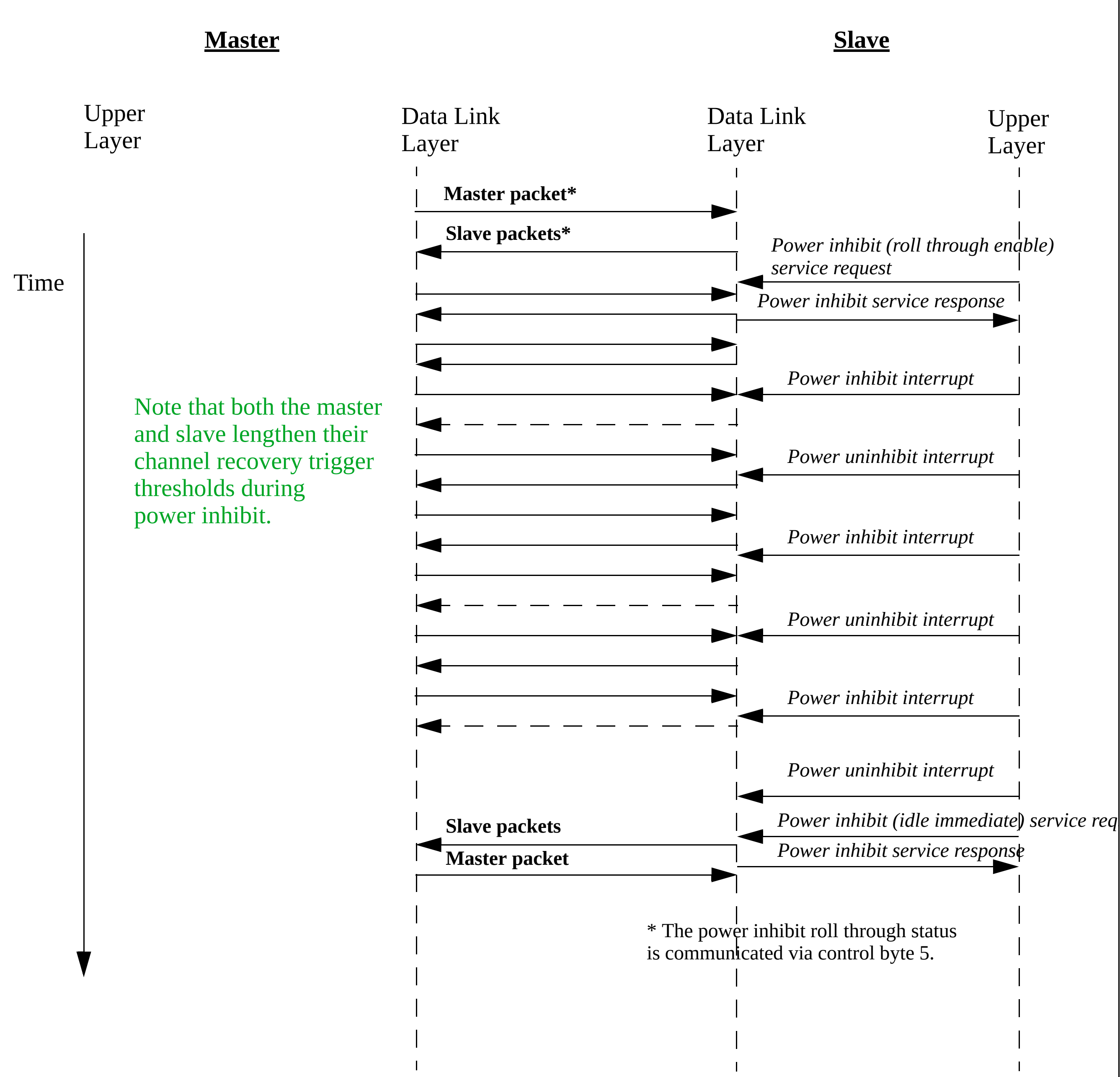
1. Open is sent using one downlink. The response uses a configurable number of uplink packets.
2. This allows two complete opportunities to hear an open downlink on each channel (one opportunity on each antenna). Note the first downlink may be missed if the slave changes channels in the middle of the downlink.
3. If the power inhibit deassertion duration is less than the channel listen time, the slave will never change channels (slave scans during polling). The same holds true for medical event. If the deassertion duration is too small, the slave will continue to uplink medical event on the same channel.

An analogous situation occurs for medical event. The power inhibit deassertion duration should be greater than the medical event channel transmit time.

The following describes the power inhibit roll through operation:

1. The master host sends one or more data service requests to the slave host to establish the power inhibit pattern.
2. The slave host sends a power inhibit roll through enable service request to the slave data link layer. This will be communicated to the master data link layer via telemetry.
3. Once enabled, the slave host may periodically assert a power inhibit interrupt to the slave data link layer (based on the pattern communication in step 1).
4. The slave host will send a power inhibit roll through disable service request to the slave data link layer when it wants roll through to be disabled (power inhibit idle immediate enabled). This will be communicated to the master data link layer via telemetry.

.



* The power inhibit roll through status is communicated via control byte 5.

Figure 93: Power Inhibit Sequence Diagram

*Data service request defining power inhibit pattern are not shown.

Key:

◄– – Packet sent w/ RF off

Figure 93 illustrates the Power Inhibit service operations.

TABLE 71: POWER INHIBIT MODES

POWER INHIBIT MODE	DESCRIPTION	RECOVERY THRESHOLD
0	Idle immediate/Original threshold	Recovery threshold 1(1)
1	Roll through	Recovery threshold 1**(1)**
2	Roll through	Recovery threshold 2**(1)**
3	Roll through	Recovery threshold 3**(1)**
4(2)	New threshold	Recovery threshold 4(3)

1. The recovery thresholds #1, #2, and #3 come originally from the master initialization request. They are communicated to the slave via an open packet.
2. Power inhibit mode 4 is pertinent only to the slave. This value cannot be communicated to the master since the power inhibit mode is only two bits (see **Fig. 17 (Packet Control Bytes)**).
3. The recovery threshold #4 come originally from the slave initialization request.

5.2.20.1 REQUIREMENTS

There are two notes which apply to master and slave requirements:

- A longer recovery threshold prevents channel recovery when the slave is inhibited (packet received at master as MAC error). For example if the slave host is allowing telemetry for 50 msec and inhibiting for 100 msec, the “number of frames without a valid packet” should be slightly longer than 100 msec.
- Also note that the recovery threshold 1, recovery threshold 2 and recovery threshold 3 are communicated to the slave via an open packet.

5.2.20.2 MASTER

TLMM_M_422 - POWER INHIBIT - CHANNEL RECOVERY THRESHOLD

The master data link layer will set its (the master’s) channel recovery threshold (number of consecutive frames without a valid packet) using the received power inhibit bits (control byte 5, bits 5 and 6 from the slave). See **Table 71 (POWER INHIBIT MODES)**

[Note that the original recovery thresholds come from the master initialization request. The power inhibit mode is just selecting threshold 1, threshold 2 or threshold 3.]

5.2.20.3 SLAVE

TLMM_S_264 - POWER INHIBIT - UPPER LAYER TO DATA LINK LAYER PARAMETERS

When invoking the power inhibit service request, the Upper Layer will provide information as specified in **TABLE 72: POWER INHIBIT SERVICE REQUEST (SLAVE).**

TABLE 72: POWER INHIBIT SERVICE REQUEST (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	0x0E - Slave power inhibit service request
Sub-type	N/A	1	The service request sub-type	See **Table 71 (POWER INHIBIT MODES)**
Size	N/A	2	The size of the info field	0
Info	N/A	0	The info field	N/A
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

TLMM_S_421 - POWER INHIBIT - TRANSMIT PACKET

The slave data link layer will do the following upon the receipt of a power inhibit service request:

1. Communicate the power inhibit sub-type to the master by setting the power inhibit bits on every transmit packet (see control byte 5)

[The slave transmit packet’s control byte 5, bits 5 and 6 shall mirror the power inhibit service request sub-type.]

TLMM_S_427 - POWER INHIBIT SERVICE RESPONSE TO UPPER LAYER

A power inhibit service response will be generated to the Upper Layer after the power inhibit service request is complete.

[The content of the response to the Upper Layer is specified in **Table 73 (POWER INHIBIT SERVICE RESPONSE (SLAVE))**].

TABLE 73: POWER INHIBIT SERVICE RESPONSE (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service response	0x0E - Slave power inhibit service response
Sub-type	N/A	1	The service response sub-type	See **Table 71 (POWER INHIBIT MODES)**
Size	N/A	2	The size of the info field	1

TABLE 73: POWER INHIBIT SERVICE RESPONSE (SLAVE)

Info	Power inhibit service status	1	Status of the service request	0x01 - Request successful 0x10 - Invalid CRC 0x13 - Invalid request type (This code can occur if the service request is sent using an interrupt other than service request0 (there is a buffer/ service request mismatch). 0x14 - Invalid subtype (subtype >= 5 for Tel M, subtype other than (0 and 4) for TelC) 0x24 - Request invalid during EEPROM read
CRC	N/A	2	CRC of all above fields.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

TLMM_S_423 - POWER INHIBIT IDLE IMMEDIATE - ASSERTED

If the power inhibit interrupt is asserted (while power inhibit mode is idle immediate),

1. While slave is in communication state, Slave will start same channel recovery immediately. (see **TLMC_S_130 - CH RECOVERY - START SAME CHANNEL RECOVERY**).
2. While slave is not in communication state. Slave will stop sniffing if sniffing is enabled or perform idle immediate if transmission or receiving is progress.

[For #1, note that the only difference from **TLMC_S_130 - CH RECOVERY - START SAME CHANNEL RECOVERY** is slave will not restart sync mode.]

TLMM_S_424 - POWER INHIBIT IDLE IMMEDIATE - DEASSERTED

If the power inhibit is deasserted (while power inhibit mode is Idle immediate),

1. While the same channel recovery time is active, slave will attempt to recover the communication session on the same channel.
2. While the unspecified channel recovery time is active, slave will attempt to recover the communication session following the unspecified channel recovery algorithm (see **5.3.3 Channel Recovery**).
3. While the polling duration is active, slave will restart to listen in native mode or sniffing. (see **5.2.8 Polling**).
4. While the disable ON timer is active, slave will restart the sync sniffing.
5. While the medical service duration is active, slave will restart to uplink medical event packet.(see **5.2.16 Medical Event**)

[For #1, the slave will restart sync mode when power inhibit is deasserted.]

TLMM_S_425 - POWER INHIBIT - RESPONSE

The slave data link layer will set its (the slave’s) channel recovery threshold (number of consecutive frames without a valid packet) using the last power inhibit service request from the slave host. Additionally it will mirror the power inhibit mode on all its subsequent transmit packets.

[The radio will be disabled when power inhibit is asserted. This will cause the slave to receive packets with MAC error. In order to avoid channel recovery, the slave host will use power inhibit roll through (mode 1, 2, or 3). The recovery threshold (paired with the roll through mode) should be chosen to be slightly longer than the power inhibit duration. Once power inhibit is deasserted, the slave should receive packets without MAC error.]

TLMM_S_442 - POWER INHIBIT - WAKEUP COMPLETE INDICATION DURING SYNC SNIFF

If power inhibit interrupt is asserted while slave is synchronously sniffing and waiting for the “last packet last block” wakeup packet (after the “first packet last block” wakeup packet has already been received), a wakeup indication “Wakeup Complete”, containing the wakeup packet contents, shall be sent to the upper layer. (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[After the “first packet last block” wakeup packet has already been received, Mozart hardware will turn the radio off and start a timer. When the timer expires, Mozart hardware will generate the “last packet last block” wakeup packet interrupt. If power inhibit is asserted, the timer will be turned off and no “last packet last block” wakeup packet interrupt will be generated by the hardware. Instead of waiting for the next synchronous sniffing window after power inhibit is deasserted, a wakeup indication “Wakeup Complete” is sent to the upper layer when the power inhibit is asserted.]

TLMM_S_447 - POWER INHIBIT - WAKEUP COMPLETE INDICATION DURING ASYNC SNIFF

If power inhibit interrupt is asserted while slave is asynchronously sniffing and waiting for the “last packet last block” wakeup packet (after the “first packet last block” wakeup packet has already been received), a wakeup indication “Wakeup Complete”, containing the wakeup packet contents, shall be sent to the upper layer. (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[Slave will perform asynchronously sniffing during polling and unspecified channel recovery if Slave scans during polling = 0x00 (see **Table 123 (Master parameters from host)**).]

TLMM_S_448 - POWER INHIBIT - CHANNEL LISTEN TIMER FOR SCAN DURING POLLING

While slave is configured to do scan during polling, if power inhibit interrupt is asserted while slave is listening on one channel during polling or unspecified channel recovery, slave will stop the channel listen timer. Then when power inhibit interrupt is deasserted, slave will restart the channel listen timer and listen on the same channel.

[Slave will change the scan channel during polling or unspecified channel recovery only if the “channel listen timer” times out while power inhibit status is low (i.e., power inhibit has been deasserted or power inhibit has never been asserted.)]

TLMM_S_449 - POWER INHIBIT - MEDICAL EVENT TRANSMIT TIME

If power inhibit interrupt is asserted before the “medical event transmit time” timesout (while slave is uplinking medical event on one channel), slave will stop the medical event transmit timer. Then when power inhibit interrupt is deasserted, slave will uplink medical event on the same channel and restart “the medical event transmit time” .

[Slave will change the channel during uplinking medical event only if the “medical event transmit time” times out while power inhibit status is low (i.e., power inhibit has been deasserted or power inhibit has never been asserted.)]

TLMM_S_450 - POWER INHIBIT - OPEN RESPONSE PACKETS

If power inhibit interrupt is asserted while slave is uplinking open response packets,

- when slave is configured to do scan during polling - slave will perform idle immediate and stop the channel listen time. Then when power inhibit interrupt is deasserted, slave will listen on the same channel and restart the channel listen time;
- when slave is configured to do asynchronously sniffing/scan during polling - slave will perform idle immediate. Then when power inhibit interrupt is deasserted, slave will listen on the same channel.

[Note the native mode timer is not stopped and this is because native mode timer is much longer than power inhibited duration.]

- when slave is running the medical event service duration timer,slave will perform idle immediate and stop the medical event transmit time. Then when power inhibit interrupt is deasserted, slave will uplink medical event on the same channel and restart “the medical event transmit time”

5.2.21 CHARGING NOISE

An implantable cardiovascular defibrillator (ICD) delivers high voltage shock pulses to the heart to defibrillate or cardiovert the heart. One or more capacitors in the ICD are charged to a desired voltage to generate the shock pulses. Capacitor charging can cause noise interference on the downlink that compromises telemetry communication between the ICD and another device, such as an external programmer or home monitor or another implanted device. Note that charging has no impact to the uplink packets. Telemetry communication is important during capacitor charging since a clinician may be trying to deliver a command to the ICD to abort the shock pulse delivery. Telemetry communication during capacitor charging also allows a clinician to observe real time information uplinked from the ICD relating to sensed and detected cardiac events.

To reduce the impact of the charging noise interference to telemetry, the following charging noise algorithm can be used so that even if received downlink packets are corrupted slave will stay in communication session and continue to transmit waveforms and supplemental markers, and slave can respond to any downlink packet within certain amount of time.

Note that this design has no impact to master. This algorithm can also be used when there is a need for slave to change channel recovery threshold during communication session without any impact or changes on the master.

Based on the transmit data rate, the transfer mode and system requirement about how quickly slave should respond to a downlink, the following four thresholds are needed:

- New Channel Recovery Trigger Threshold For Nominal mode - Channel Recovery Trigger Threshold while slave mozart is in nominal transfer mode. This number should be selected to be greater than the original recovery threshold (i.e., Recovery Threshold 1) so that slave won’t trigger channel recovery when the original recovery threshold is reached.

- New Channel Recovery Trigger Threshold For Max. Transmit Mode - Channel Recovery Trigger Threshold while slave mozart is in Max. Transmit transfer mode. As much more packets will be uplinked in the Max. Transmit transfer mode, New Channel Recovery Trigger Threshold For Max. Transmit Mode shall be set less than New Channel Recovery Trigger Threshold For Nominal mode.

- New Channel Recovery Imminent Threshold For Nominal mode - This will be used to indicate host that channel recovery is going to occur while in slave is in Nominal Mode. The value should be less than “New Channel Recovery Trigger Threshold For Nominal mode”.

- New Channel Recovery Imminent Threshold For Max. Transmit Mode - This will be used to indicate host that channel recovery is going to occur while in slave is in Max. Transmit Mode. The value should be less than “New Channel Recovery Trigger Threshold For Max. Transmit Mode”.

Fig. 94 (Charging Noise Sequence Diagram) explains the algorithm sequence using charging as an example.

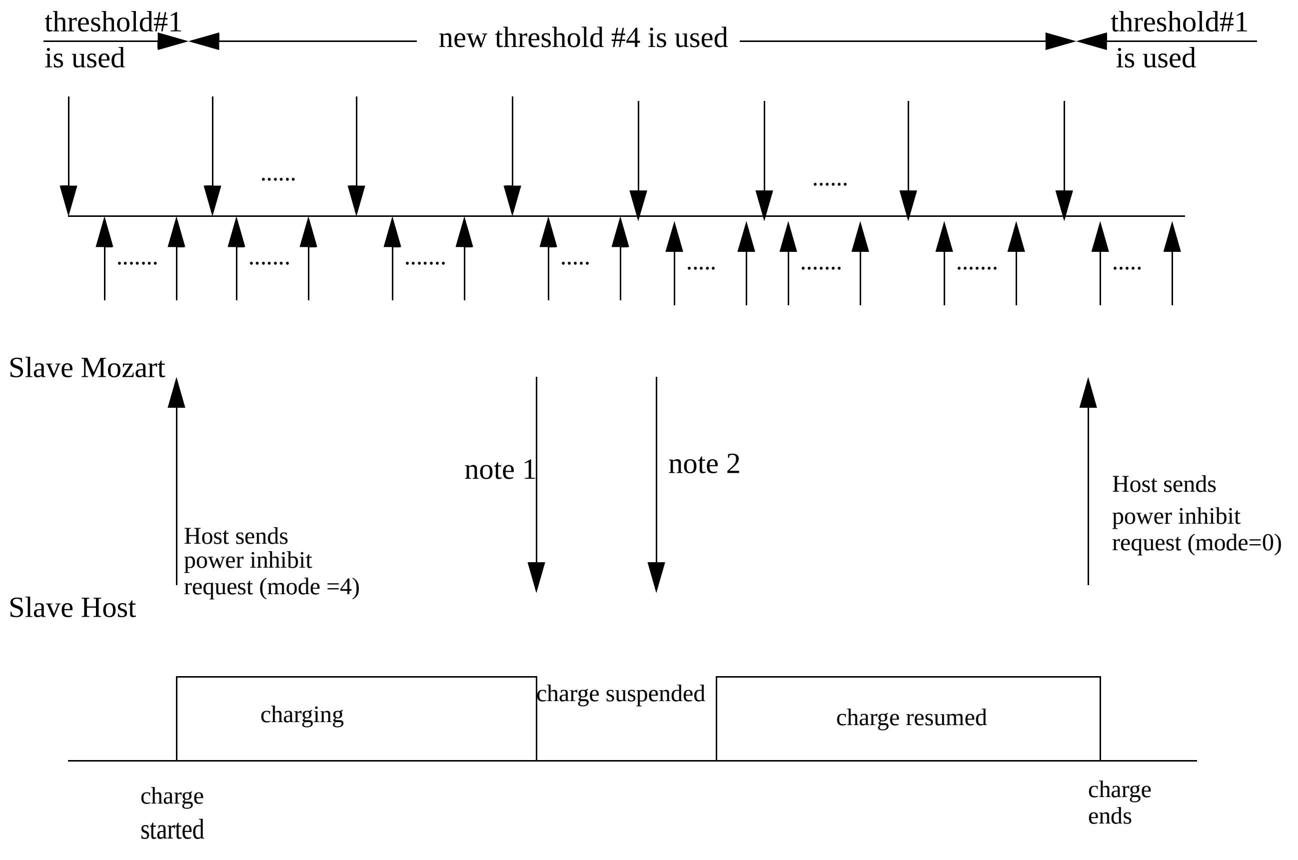


Figure 94: Charging Noise Sequence Diagram

Note:

1:Mozart sends channel recovery imminent indication to host when recovery imminent threshold is reached.

2. Mozart sends channel recovery imminent resumed indication to host when first downlink received with good MAC is received

When charge starts or when slave Host wants slave mozart to use the new channel recovery threshold (threshold #4), slave host sends the Power Inhibit Service Request with sub-type = 0x04 (see **Table 72 (POWER INHIBIT SERVICE REQUEST (SLAVE))**. Then slave mozart will use threshold #4 (“New Channel Recovery Trigger Threshold For Nominal mode” and “New Channel Recovery Trigger Threshold For Max. Transmit Mode” from **Table 124 (SLAVE PARAMETERS FROM HOST)**)to trigger the channel recovery.

When charge ends, aborts or when slave host wants slave mozart to use the original channel recovery threshold (threshold #1), slave host sends the Power Inhibit Service Request with sub-type = 0x00 (see **Table 72 (POWER INHIBIT SERVICE REQUEST (SLAVE))**. Then slave mozart will use the “Recovery Threshold 1” to trigger channel recovery.

In addition, slave host can configure slave mozart to indicate when channel recovery is going to occur. If configured (i.e., Send Channel Recovery Imminent Indication to host = 1, see Parameter “Power Inhibit” in **Table 124 (SLAVE PARAMETERS FROM HOST)**), then slave mozart sends Same Channel Recovery Imminent Indication to slave host when “New Channel Recovery Imminent Threshold For Nominal mode” or “New Channel Recovery Imminent Threshold For Max. Transmit Mode” is reached. Based on the indication, slave host can stop charging so that slave mozart will receive downlink packets with good MAC. When slave mozart receives the first downlink packet with good MAC, slave mozart will send Same Channel Recovery Imminent Resume Indication to slave host if configured by slave host (i.e., Send Channel Recovery Imminent Resume Indication to host = 1, see Parameter “Power Inhibit” in **Table 124 (SLAVE PARAMETERS FROM HOST)**).

TLMM_S_451- CHARGE NOISE - USE NEW CHANNEL RECOVERY THRESHOLD

When slave host sends the Power Inhibit Service Request with sub-type = 0x04 (see **Table 72 (POWER INHIBIT SERVICE REQUEST (SLAVE))**, then slave mozart will use the “New Channel Recovery Trigger Threshold For Nominal mode” and “New Channel Recovery Trigger Threshold For Max. Transmit Mode” from **Table 124 (SLAVE PARAMETERS FROM HOST)** to trigger the channel recovery.

[The slave transmit packet’s control byte 5, bits 5 and 6 shall NOT mirror the power inhibit service request sub-type when the sub-type = 0x04. Bits 5 and 6 will contain the last power inhibit mode uplinked prior to mode 4.]

When slave host sends the Power Inhibit Service Request with sub-type = 0x00 (see **Table 72 (POWER INHIBIT SERVICE REQUEST (SLAVE))**, then slave mozart will use the original channel recovery threshold, i.e., Recovery Threshold #1 in **Table 72 (POWER INHIBIT SERVICE REQUEST (SLAVE))** to trigger the channel recovery. This is the recommended usage of the charging noise algorithm. However if slave sends the Power Inhibit Service Request with sub-type = 1,2,or 3 (see **Table 72 (POWER INHIBIT SERVICE REQUEST (SLAVE))**, then slave mozart will change to the corresponding channel recovery thresholds.

TLMM_S_453- CHARGE NOISE - SAME CHANNEL RECOVERY IMMINENT

If the power inhibit mode is 4 and configured (i.e., Send Channel Recovery Imminent Indication to host = 1, see Parameter “Power Inhibit” in **Table 124 (SLAVE PARAMETERS FROM HOST)**), then slave mozart shall send Same Channel Recovery Imminent Indication (**Table 74 (SAME CHANNEL RECOVERY IMMINENT INDICATION (SLAVE))**)to slave host when “New Channel Recovery Imminent Threshold For Nominal mode” or “New Channel Recovery Imminent Threshold For Max. Transmit Mode” in **Table 124 (SLAVE PARAMETERS FROM HOST)** is reached.

TABLE 74: SAME CHANNEL RECOVERY IMMINENT INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x27 = Slave Channel recovery indication
Sub-type	N/A	1	The indication sub-type	0x01 = same
Size	N/A	2	The size of the info field	1
Info	Channel recovery indication status	1	status of the indication	0x13 = Imminent
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

TLMM_S_454- CHARGE NOISE - SAME CHANNEL RECOVERY IMMINENT RESUMED

If the power inhibit mode is 4 and configured (i.e., Send Channel Recovery Imminent Resumed Indication to host = 1, see Parameter “Power Inhibit” in **Table 124 (SLAVE PARAMETERS FROM HOST)**), and when the first packet is received with good MAC after Same Channel Recovery Imminent Indication has been sent, then slave mozart shall send Same Channel Recovery Imminent Resumed Indication (**Table 75 (SAME CHANNEL RECOVERY IMMINENT RESUMED INDICATION (SLAVE))**) to slave host.

TABLE 75: SAME CHANNEL RECOVERY IMMINENT RESUMED INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x27 = Slave Channel recovery indication
Sub-type	N/A	1	The indication sub-type	0x01 = same
Size	N/A	2	The size of the info field	1
Info	Channel recovery indication status	1	status of the indication	0x14 = Imminent resumed

TABLE 75: SAME CHANNEL RECOVERY IMMINENT RESUMED INDICATION (SLAVE)

CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

Slave Mozart will use the corresponding threshold based on the power inhibit mode in the last received power inhibit request.

If power inhibit is asserted when the power inhibit mode is 4, the slave will treat this like power inhibit roll through using recovery threshold 4. However it is not recommended to mix the usage of the charging noise algorithm with usage of the power inhibit. This is because the nature and impact of power inhibit and charging noise are different.

Power inhibit is used to stop slave mozart from using the power supply so that host can maximize its usage of the power supply. Power inhibit blocks both downlink and uplink on slave. Both master and slave should use the same channel recovery thresholds so that they will trigger recovery at about the same time. Slave mozart has a way to communicate the power inhibit mode (0 to 3) to master mozart.

Charging noise algorithm is to extend slave mozart to stay in communication session even if charging noise blocks the downlinks from received by slave mozart. Slave mozart does not communicate the power inhibit mode 4 to master mozart.

Host shall measure the battery before charge starts. If the battery is too low and host still want to charge, instead of extending the recovery threshold (using the charging noise algorithm), host may want to send power inhibit request with mode 0 and then assert power inhibit to totally stop Mozart from using the battery and therefore prevent host from a reset.

5.2.22 SHUTDOWN

This is a local service which is used between the host and the telemetry module. This service provides a mechanism to the host to gracefully power down the telemetry module. When this service is invoked all data link layer operation will cease, thus allowing to save energy and prevent unnecessary RF transmissions see **Fig. 95 (Shutdown Sequence Diagram)**

Note: If host fails to invoke this service for powering down the module - module could be left in a non-receptor mode, it is host’s responsibility to insure that module is in the intended state.

When shutdown service is invoked summary of data link layer will be as follows:

- Terminate any transmit/receive operations
- Cancel all the timers
- Transition to a disable state and configure the module as a receptor

- Start a disable timer and send a shutdown response to the host.

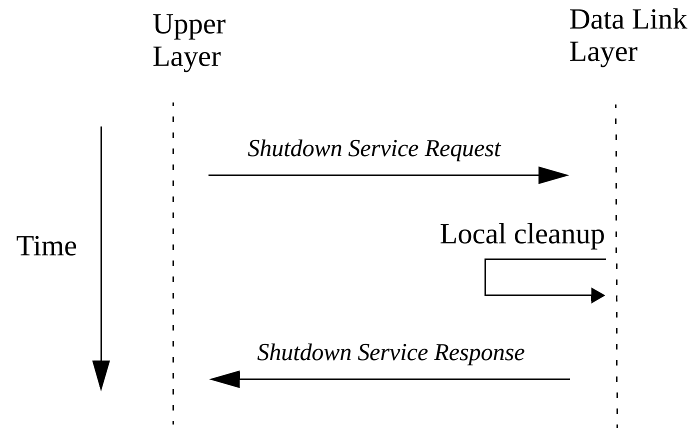


Figure 95: Shutdown Sequence Diagram

5.2.22.1 REQUIREMENTS

TLMM_MS_353 - SHUTDOWN - SERVICE REQUEST PARAMETERS

When invoking a shutdown service request, upper layer will information as specified in **Table 76 (SHUTDOWN SERVICE REQUEST)**.

TABLE 76: SHUTDOWN SERVICE REQUEST

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	Shutdown service request 0x0F - Slave 0x8F - Master
Sub-type	N/A	1	The service request sub-type	N/A
Size	N/A	2	The size of the info field	0
Info	N/A	0	N/A	N/A
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the upper layer. It is used by the data link layer verify the integrity of the service request.

TLMM_MS_354 - SHUTDOWN - RESPONSE TO UPPER LAYER

Upon the receipt of the shutdown service request, data link will perform the following actions:

1. Terminate any data transmission and/or receive operations,
2. Start a disable timer and send a shutdown response to host

[Cancel all the timers, Transition to a disable state, configure the module to be in a receptor mode (per initialization parameters). Terminate physical layer communication, transition to disable state, send a successful response. Note that before turning off the 2 volt supply, the slave will return to synchronous receptor wakeup operation. If the wakeup disable bit is set to 1 in the initialization request firmware does not set the wakeup enable bit bp_wakeup_en bit in REG_BP_WU_CTRL0 register. See requirement **TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP**]

TABLE 77: SHUTDOWN SERVICE RESPONSE

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of service request	Shutdown service response 0x0F - Slave 0x8F - Master

TABLE 77: SHUTDOWN SERVICE RESPONSE

Sub-type	N/A	1	The service response sub-type	N/A
Size	N/A	2	The size of the info field	1
Info	Close service status	1	Status of the service request	0x01 - request successful(1) 0x0A - Parameter value error(2) 0x10 - Invalid CRC 0x12 - Invalid size (info size >506)(2) 0x13 - Invalid request type(3) 0x24 - Request invalid during EEPROM read 0x25 - Buffer/ service request mismatch (master only)(4)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. Note that request successful is returned regardless of data link layer state.
2. If the request info size is greater than 506, 0x12 - Invalid size (info size >506) will be used; if the info size or subtype (if applicable) does not match the value in the request table, "Parameter out of range" will be used. Note “0x0A - Parameter value error” is only applicable to the master.
3. This value applies to the slave only.This code can occur if the service request is sent using an interrupt other than service request0 (there is a buffer/ service request mismatch)
4. This status can occur if the master host attempts to send a shutdown service request using service request buffer 1 (instead of buffer 0).

5.3 INTERNAL SERVICES

5.3.1 PACKET RETRANSMISSION (ARQ)

The packet retransmission algorithms allow the master and slave to retransmit individual Memory Data packets during data transmission. Telemetry M uses two types of ARQ algorithms for the Data packet retransmission:

- Stop and Wait - ARQ algorithm is used for master data packet retransmission.
- Selective Repeat - ARQ algorithm is used for slave Memory Data retransmission (acronym is SR-ARQ).

These ARQ algorithms are used only during the Nominal Mode and Max Slave Throughput Mode.

The retransmission of waveform packets are based on the Go-Back-N ARQ algorithm.

Supplemental marker packets are not retransmitted.

5.3.1.1 REQUIREMENTS

The following is an overview of the Go-Back-N, Stop and Wait and SR-ARQ and algorithms:

- The ***Go-Back-N*** - This ARQ algorithm is used to support the retransmission of the waveform packet(s).The slave builds and transmits the waveform packets. The master validates and processes each waveform packet received in the slave frame and transmits a master packet with the expected waveform sequence number. Then slave shall retransmit the waveform packets starting from the expected waveform sequence number based on whether there is any new waveform request or not.
- The ***Stop and Wait*** - This ARQ algorithm is used for master data packets. In this algorithm the slave acknowledges each data packet received with a valid message authentication code (MAC) by setting an acknowledge bit in the response packet(s). If the master does not receive an acknowledgment for the transmitted packet, it retransmits the packet in the next frame.
- The ***Selective Repeat*** - This ARQ algorithm (SR-ARQ) is used only for slave Memory Data packet(s). The slave builds and transmits the slave packets. The master validates and processes each slave packet received in the slave frame and transmits a master packet with acknowledgment information back to the slave. This acknowledgment information indicates the message authentication code (MAC) state of the applicable packet(s) received by the master. The slave uses the acknowledgment information to determine which packets to retransmit. The SR-ARQ algorithm uniquely identifies any slave packets requiring retransmission via ARQ information (general sequence number and bitmap). A sliding window mechanism is used to determine the range of the packet's (or packets') general sequence numbers that can be transmitted or retransmitted. The size of the sliding window is determined by the difference between the lower and upper limits. The computation formula for the sliding window size is shown in the formulas below:

$$\text{Windowsize} = (\text{MaxSeqNumber} + 1) / 4$$
$$\text{TelemetryCWindowsize} = (255 + 1) / 4 = 64$$

The lower and upper limits of the sliding window are advanced when the general sequence number of the received Memory Data packet is equal to the lower limit. Memory Data packets outside the sliding window are considered invalid and are ignored. The use of the sliding window mechanism is illustrated via two examples in **Fig. 96 (Sliding Window example)**. The first example illustrates the normal usage. The second example illustrates boundary conditions where the general sequence numbers are allowed to wrap.

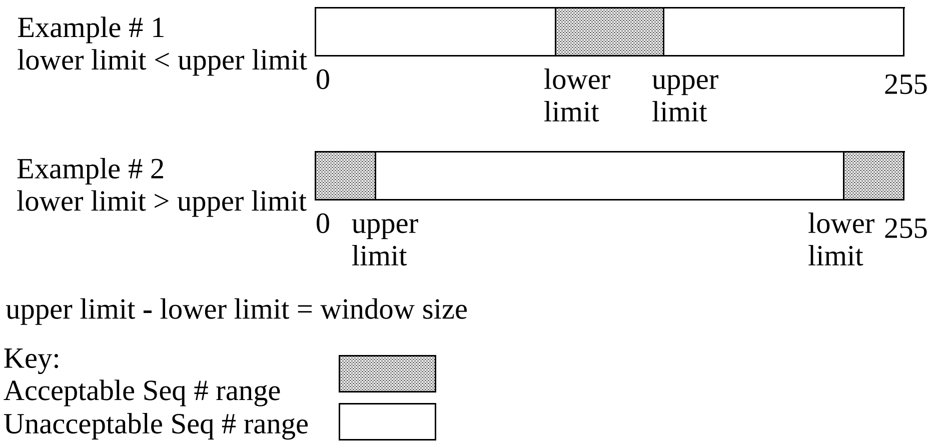


Figure 96: Sliding Window example

The ARQ packet is assembled by the master and is transmitted to the slave. The content of the ARQ packet information consists of two elements:

- First Outstanding Packet (FOP) sequence number.
- Bitmap indication of the received and outstanding packets.

5.3.1.2 MASTER

The Stop and Wait algorithm is used for master packet transmission. After transmitting a packet, the master checks the ACK bit in the slave packet(s). A set ACK bit in the slave packet(s) is an indication of a master packet received by the slave with a valid message authentication code (MAC). If the ACK bit is not set in the slave packet(s) or no valid slave packet(s) are received by the master, then the previous master packet is retransmitted.

TLMC_M_100 - ARQ - STOP AND WAIT MASTER ACKNOWLEDGMENT

After transmitting the Memory Data master packet, check for the ACK bit in the slave packet(s) and perform as follows:

1. If the ACK bit is set in at least one valid slave packet, then transmit the next Memory Data master packet (if the previous packet was not the final packet).
2. If the ACK bit is not set in any of the received slave packet(s) or no valid slave packets are received, then retransmit the previous master packet in the next frame.

The SR-ARQ algorithm is used for slave Memory Data retransmission. The master is responsible for:

- Managing its sequence number window
- Tracking the ARQ information for each packet, received or not received
- Building and sending ACK/ARQ information back to the slave

The FOP sequence number and the ARQ bitmap provides a received or not received indication of the Memory Data slave sequence numbers. The width of the bitmap is proportional to the sequence number window size for slave Memory Data packets.

The bitmap entries are initialized as not received until the corresponding packet is correctly received. When the FOP is correctly received, then the bitmap content is accordingly updated and the FOP sequence number is adjusted to reflect the next outstanding packet. The first example (**Fig. 97 (Ex. 1 - FOP and Bitmap Initialization)**) illustrates the state of the FOP and bitmap contents when this information is initialized or when there are no outstanding packets requiring retransmission. The FOP value represents the next packet expected by the master. The bitmap contents represent the default state of future packets, i.e. packets are assumed not received until correctly received.

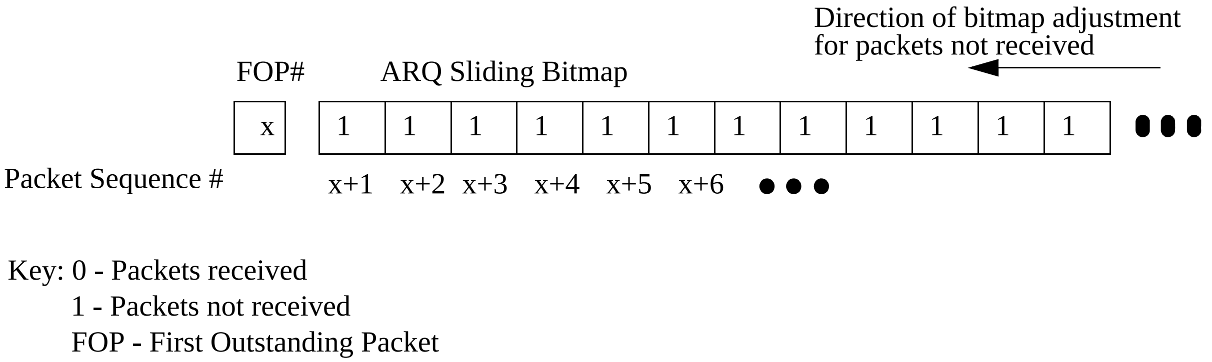


Figure 97: Ex. 1 - FOP and Bitmap Initialization

The second example (**Fig. 98 (Ex. 2 - Snapshot of ARQ Information)**) illustrates a specific snapshot of ARQ information on the master after the slave has transmitted packets 1 through 9.

- Successfully received packet#: 2, 3, 5, 8
- Not received packet#: 1 (FOP), 4, 6, 7, 9....

The master transmits the ARQ FOP and bitmap information back to the slave. The slave uses the ARQ information to determine which Memory Data packets require retransmission. In this example the slave retransmits Memory Data packets number 1, 4, 6, 7, and 9. Additional new packets may also be transmitted, resources permitting.

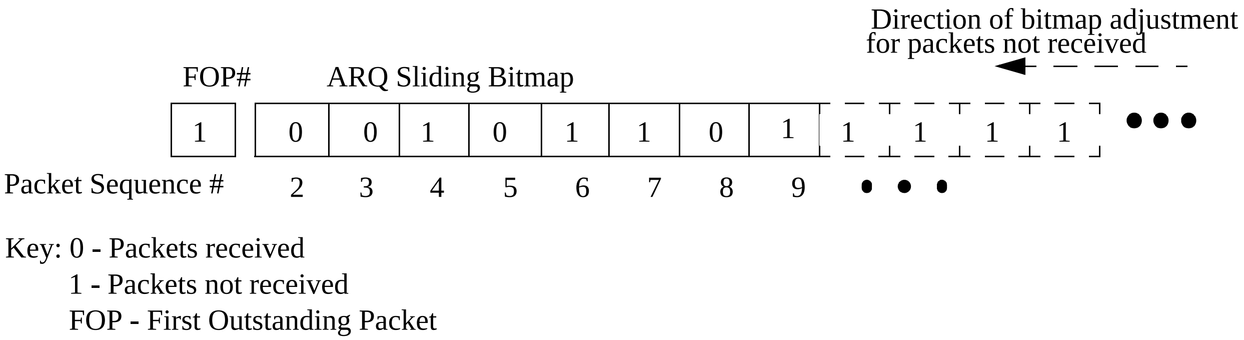


Figure 98: Ex. 2 - Snapshot of ARQ Information

The third example (**Fig. 99 (Ex. 3 - Next ARQ Snapshot)**) illustrates the next snapshot of the ARQ information after the slave has transmitted (or retransmitted) packet numbers 1, 4, 6, 7, 9, 10, 11:

- Sequence numbers of the packets successfully received: 1, 4, 9, 10, 11.
- Sequence numbers of the packets not received: 6 (FOP), and 7.

Several of the retransmitted packets were correctly received and several new packets were also correctly received. However, packets #6 and #7 were not correctly received. The updated ARQ information is transmitted to the slave for another retransmission attempt. The exchange of packets and ARQ information continues until all outstanding packets are correctly received or the link is reset.

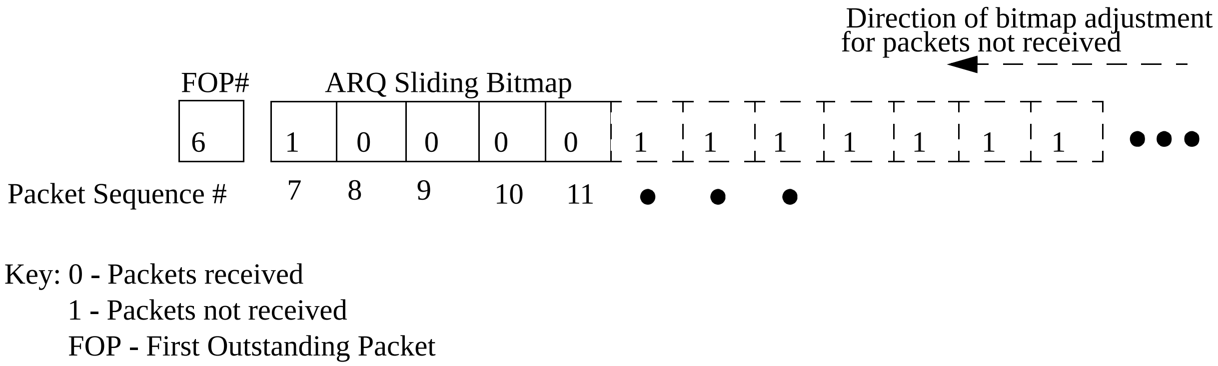


Figure 99: Ex. 3 - Next ARQ Snapshot

The example in **Fig. 100 (SR-ARQ Example During Nominal Transfer Mode)**illustrates the SR-ARQ mechanism for slave Memory Data packets in a nominal transfer mode. The solid up arrow lines represents the slave direction, and the solid down arrow represents the master direction of the data. In this example an slave message consists of nine Memory Data packets as represented by their sequence numbers (01, 02, 03...). In the slave frames, Memory Data packets and waveform packets are interleaved. The ARQ packet information is updated and transmitted back to the slave in the master frames.

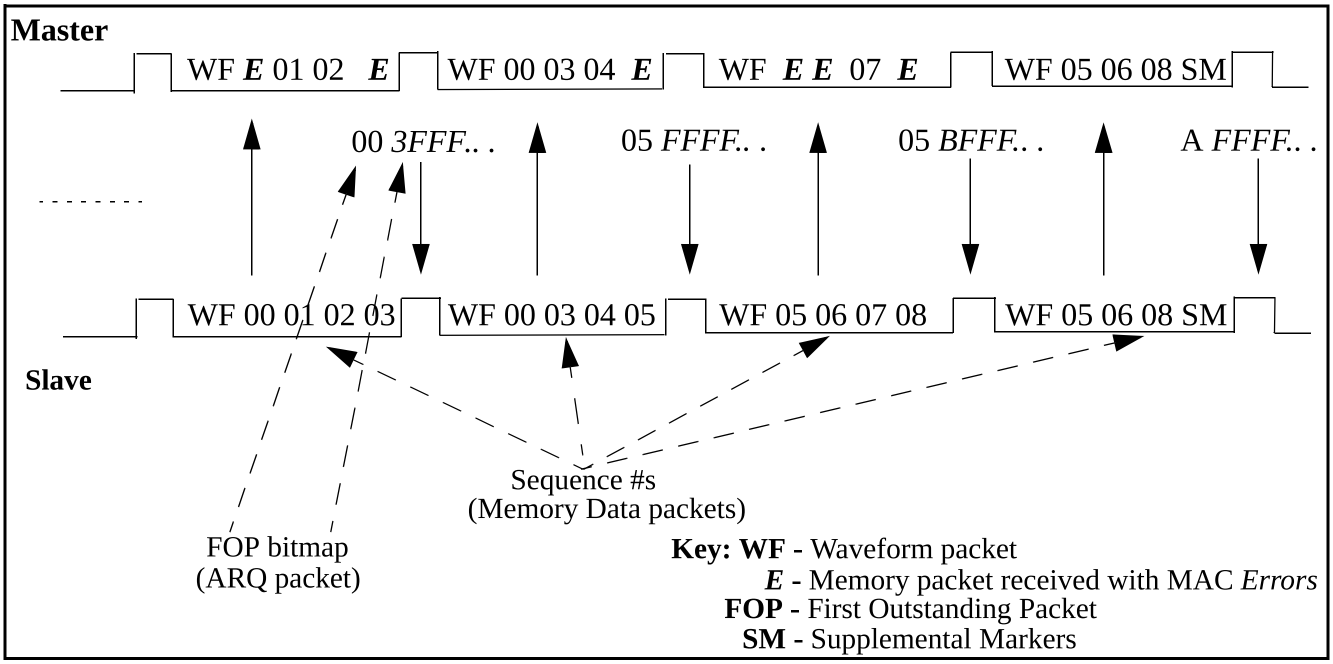
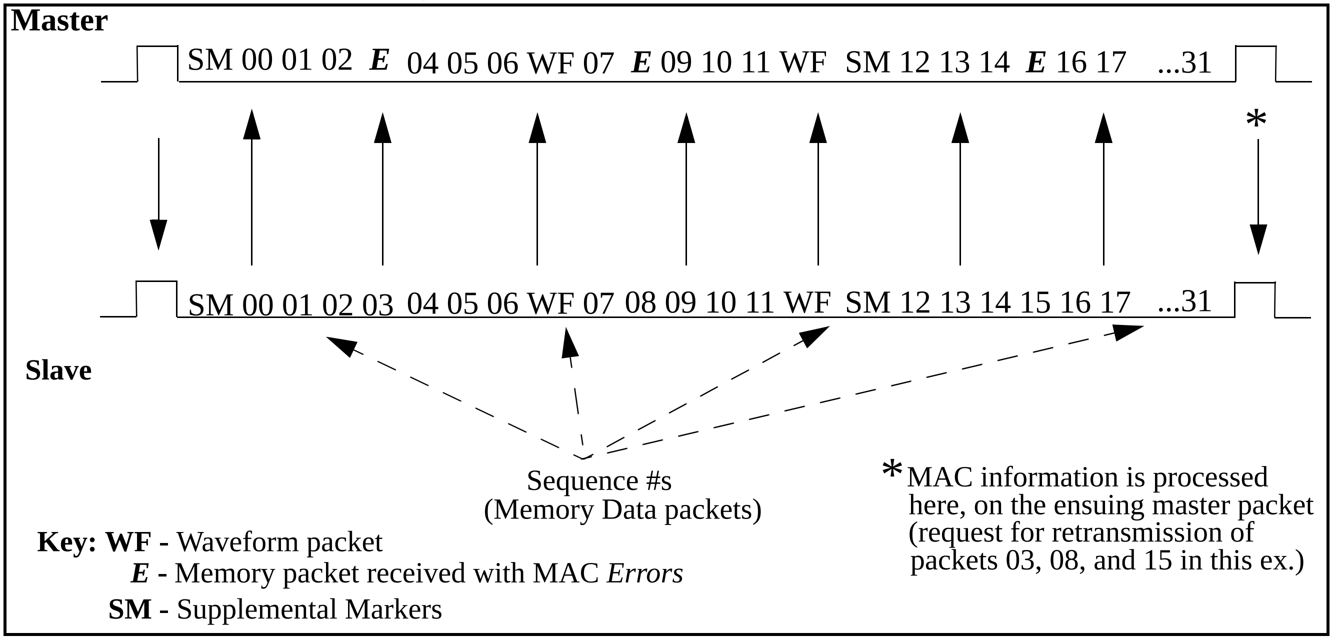


Figure 100: SR-ARQ Example During Nominal Transfer Mode

The example in **Fig. 101 (SR-ARQ Example During Maximum Slave Throughput Mode)** illustrates the max slave throughput mode.



*MAC information is processed here, on the ensuing master packet (request for retransmission of packets 03, 08, and 15 in this ex.)

Key: WF - Waveform packet
***E* -** Memory packet received with MAC *Errors*
SM - Supplemental Markers

Figure 101: SR-ARQ Example During Maximum Slave Throughput Mode

TLMC_MS_101 - ARQ - MASTER ARQ PACKET FORMAT

A master ARQ Request packet has the format indicated in **Fig. 102 (Master ARQ Request Packet)**.

ARQ Control 5.25 bytes	**Source ID (master ID) 6 bytes**	**FOP 1 byte**	**Bitmap 8 bytes**	**(zero fill) 15 bytes**	**MAC[a] 3 bytes**	**Reed Solomon n bytes**

a. Message authentication code

Figure 102: Master ARQ Request Packet

TLMC_M_102 - ARQ - UPDATING OF ARQ INFORMATION

The master will update the ARQ information if it receives a memory packet (if the master packet was not Discover, Open, Network, or Security).

[The state is Comm when the master packet is not Discover, Open, Network nor Security.]

TLMC_M_103 - ARQ - SLAVE MEMORY DATA PACKET ACKNOWLEDGMENT

For each slave Memory Data packet that is received:

1. Verify the sequence number of the received packet to ensure it is within the sliding window limits. A received packet with a sequence number outside the sliding window limits will be ignored.
2. Adjust the SR-ARQ information based on the received packet. Update the FOP field with the sequence number of the First Outstanding Packet required to be retransmitted. In the bitmap, clear the bit for the packet received.
3. Transmit the ARQ packet to the slave in the next master frame.

[Note that the last bit of the bitmap is never used (always 1).]

TLMC_M_104 - ARQ - SLAVE ACK NOT RECEIVED

When no Memory Data packet is received in the slave frame then perform the following:

1. If the previous master packet was an ARQ packet and it is not acknowledged (ACK bit not set) in the slave frame, then re-transmit the ARQ packet in the next master frame.
2. The ARQ packet will stop when ARQ packet is acknowledged via ACK bit set in at least one slave packet.

In general, the waveform packets are retransmitted using Go-Back-N ARQ algorithm. That is, all the transmitted packets starting with the expected waveform sequence number will be retransmitted. Below is the detailed waveform retransmission algorithm description.

Once a waveform request is received, the slave shall transmit the waveform packets and shall save all the transmitted waveform packets in the transmit frame.

The master provides the expected waveform sequence number in the master transmit packet. The expected waveform sequence number is initialized as zero whenever a session is opened or recovered.

The master can be configured to request retries (go back N) for the most recently received waveform transaction or the second most recently received waveform transaction. In the first case an incomplete waveform transaction is discarded when a fragment (packet) of any new waveform transaction is received (new transaction number not equal the old transaction number). In the second case an incomplete waveform transaction is discarded when a fragment (packet) is received with a transaction number of any value other than (N+1)%8 (where N is the transaction number of the older waveform transaction). For example, the master will continue to request retries from waveform transaction number 1 until it receives a packet from waveform transaction 3. The following rules apply:

- If the master has received all the fragments (packets) from the transaction, the expected waveform sequence number will be the transaction’s last waveform sequence number+1.
- If the master has not received all the fragments (packets) from the transaction, the expected waveform sequence number will be the waveform sequence number of the first missing fragment (packet).

The slave shall retransmit the saved waveform packets of the Nth waveform request based on the following rules:

- If there is no (N+2)th waveform request, then slave shall retransmit the saved waveform packets starting with the expected waveform sequence number.
- If there is (N+2)th waveform request, and if the “delay sending new waveform” is true (See “**Table 124 (SLAVE PARAMETERS FROM HOST)**”) then slave shall retransmit the saved waveform packets starting with the expected waveform sequence number.
- Else if the “delay sending new waveform” is false, then slave stops all the retransmission and transmission of the Nth waveform request and (N+1)th waveform request packets, slave shall transmit the (N+2)th waveform request packets.

TLMM_M_355 - ARQ - UPDATING THE EXPECTED WAVEFORM SEQUENCE NUMBER 1

When “delay new waveform detection” is false, the transaction number of the most recently received waveform packet will be used to set the expected waveform sequence number:

- If all the fragments (packets) from the transaction has been received, the expected waveform sequence number will be the received transaction’s (last waveform sequence number+1) modulus 256.
- If any of the fragments (packets) from the transaction have not been received, the expected waveform sequence number will be the waveform sequence number of the first missing fragment (packet).

[The “delay new waveform detection” parameter is contained in the master initialization table (see **Table 123 (Master parameters from host)**).

The master communicates the “expected waveform sequence number” using control byte 4. See **Fig. 17 (Packet Control Bytes)**.]

TLMM_M_429 - ARQ - UPDATING THE EXPECTED WAVEFORM SEQUENCE NUMBER 2

When "delay new waveform detection" is true, the expected waveform sequence number will set according to the table below:

TABLE 78: EXPECTED WAVEFORM SEQUENCE NUMBER

MISSING FRAGMENT IN 2ND MOST RECENTLY RECEIVED WAVEFORM TRANSACTION(1)	MISSING FRAGMENT IN MOST RECENTLY RECEIVED WAVEFORM TRANSACTION	EXPECTED WAVEFORM SEQUENCE NUMBER
True	-	The waveform sequence number of the first missing fragment of the second most recently received waveform transaction
False	True	The waveform sequence number of the first missing fragment of the most recently received waveform transaction
False	False	(last received waveform sequence number+1) modulus 256

1. By definition there is no 2nd most recently received waveform if there are two or more intervening incomplete waveforms between the most recent waveform and the last complete waveform transaction. For example if the last complete waveform transaction number is 4 and there are missing fragment(s) in both transaction 5 and 6 and a fragment is received with transaction number 7, the most recent waveform transaction becomes 7 and the 2nd most recent transaction becomes none.

Note that the second most recently received waveform transaction is discarded when the new waveform transaction number is any value other than (N+1) % 8, where N is the transaction number of the second most recently received waveform transaction.

[For example, the master will continue to request retries from waveform transaction 1 until it receives a packet from waveform transaction 3 (actually any transaction other than transaction 2).

The "delay new waveform detection" parameter is contained in the master initialization table (see **Table 123 (Master parameters from host)**).

The master communicates the "expected waveform sequence number" using control byte 4. See **Fig. 17 (Packet Control Bytes)**.]

In order to support different business needs for waveform transmission and to save power during low energy mode, bit "frequent waveform expected" (see **Table 123 (Master parameters from host)**) is used.

Note that if configured to run telemetry C, bit "frequent waveform expected" should be set if waveform packet(s) is expected in every uplink frame which will cause slave to retransmit the waveform packet(s) sent in the last uplink frame. (this is also backward compatible to Concerto).

5.3.1.3 SLAVE

The Stop and Wait algorithm is used for master Memory Data packets. Each master packet received with a valid message authentication code (MAC) is acknowledged via the ACK bit in the slave packet(s) of the next slave frame. An ACK bit will not be set in the slave packet(s) if the received master packet contains an invalid MAC.

TLMC_S_106 - ARQ - STOP AND WAIT MASTER PACKETS

A master packet received with a valid message authentication code (MAC) will be acknowledged by setting an ACK bit in all the slave packets of the next slave frame; otherwise, the ACK bit will not be set in any slave packet(s) of the slave frame.

[The slave frame may include the retransmitted packets.]

TLMC_S_107 - ARQ - RECEIPT OF AN ARQ PACKET

Upon the receipt of an ARQ packet the slave will perform as follows:

1. Retransmit the unacknowledged Memory Data packets using the FOP and bitmap as follows:
    - FOP indicates the sequence number of the packet not received or may need to be retransmitted, followed by a bitmap status of the remaining packets.
    - A set bit “1” in the bitmap indicates the packet(s) is not received or may need to be retransmitted.
    - A cleared bit “0” in the bitmap indicates a packet was successfully received, therefore should not be retransmitted.
2. Adjust the lower and upper limits of the sliding windows based on the ARQ information.

TLMC_S_108 - ARQ - MASTER ARQ PACKET NOT RECEIVED

If the ARQ packet is not received then re-transmit the unacknowledged Memory Data packet(s) from the previous frame(s).

[If packet space is available in the slave frame.]

When slave Memory Data transactions are complete, in the next master frame if a data service packet is received instead of an ARQ, this implies that all the previously transmitted packets are acknowledged thus allowing the sliding window to advance.

TLMC_S_109 - ARQ - IMPLICIT ACK OF VALID SLAVE MEMORY DATA

Once a slave packet is transmitted with a last fragment bit set, and the Memory Data request is received in the following master frame then the slave will perform as follows:

1. Memory data packets already transferred will be considered acknowledged.
2. The sliding window limits will be advanced accordingly. (New sequence #)

TLMC_S_110 - ARQ - MEMORY DATA ACKNOWLEDGEMENT NOT RECEIVED

If the slave does not receive acknowledgment for the transmitted Memory Data packets from the previous frame then the slave will retransmit the Memory Data packets from the previous frame if packet space is available and if the following apply:

1. No waveform packets to transmit.
2. No supplemental marker packets to transmit.

[For precedence rules, see **5.2.11** and **5.2.11.5 SLAVE DATA**.]

TLMM_S_356 - ARQ - RETRANSMIT WAVEFORM PACKET

The Nth transmitted waveform request packets shall be retransmitted as follows:

- If there is no (N+2)th waveform request, then slave shall retransmit the saved waveform packets starting with the expected waveform sequence number.
- If there is (N+2)th waveform request, and if the “Delay sending new waveform” is true (See **TABLE 124: SLAVE PARAMETERS FROM HOST**) then slave shall retransmit the saved waveform packets starting with the expected waveform sequence number.
- Else if the “Delay sending new waveform” is false, then slave stops all the retransmission and transmission of the Nth waveform request and (N+1)th waveform request packets, slave shall transmit the (N+2)th waveform request packets.

[The master communicates the “expected waveform sequence number” using control byte 4. See **Fig. 17 (Packet Control Bytes)**.]

5.3.2 CHANNEL MANAGEMENT

An master and an slave can select any of ten MICS channels for communication (there are also ten lower MEDs channels and ten upper MEDs channels). The frequency range for these ten MICS channels is between 402 to 405 MHz. The frequency range for the lower MEDs channels is between 401 and 402 MHz. The frequency range for the upper MEDs channels is between 405 and 406 MHz. A communication channel is managed in distinct ways depending whether or not communication is established. Prior to establishing communication, each master and slave is responsible for its own channel management operations. The two primary channel management operations are:

- Channel Mapping
- Channel Recovery

A brief overview of channel management operations is as follows:

The master performs **Channel Mapping** operations. The slave obtains its initial channel via the wakeup packet. The slave does not transmit until it receives a valid packet from an master.

Note: According to FCC regulations, a channel may not be selected for communications if Channel Mapping operations have not been performed within the last five seconds.

Channel Recovery: is invoked when the loss of synchronous communications is detected either by the master or by the slave. Once synchronous communication loss is detected, a finite time is allowed to reestablish synchronous communication, during which Channel Mapping operations may be invoked by Channel Recovery. (For details, see **5.3.3 Channel Recovery**.)

5.3.2.1 CHANNEL MAPPING

Channel mapping is performed by the master’s physical layer to select a channel for communication.

Channel mapping should select a channel that does not contain adverse interference, and does not interfere with other users of the channel. This is accomplished by listening to each channel and selecting the quietest channel. The quietest channel is determined using RSSI (Received Signal Strength Indication).

Use of a particular channel must occur within 5 seconds of Listen Before Talk (mapping). This is an FCC requirement. Telemetry M satisfies this requirement by doing Channel Mapping “on demand”. Channel mapping will immediately precede the wakeup packet for the discover service, open service, and network service. Channel mapping also precedes the unspecified channel recovery internal service.

5.3.2.2 MAPPING SYSTEM PARTITIONING

Mapping is performed by the Data Link Layer in combination with the physical layer. The Data Link Layer interacts with the physical layer as follows:

1. Set the wakeup imminent duration (to allow channel mapping to complete shortly before the wakeup transmission begins)
2. Enable the wakeup imminent interrupt
3. Wait for the wakeup imminent interrupt
4. Set the channel frequency
5. Request single channel clear channel assessment (CCA) which returns an RSSI value for the channel
6. Retrieve the RSSI value upon the CCA interrupt
7. Repeat steps 4-6 until all channels are assessed or a channel’s RSSI value is less than the stop-on-threshold value
8. The quietest channel is selected (the channel with the lowest RSSI)

Note that steps (1) - (3) are only performed if synchronous wakeup is to be used. These steps are skipped if asynchronous wakeup is to be used.

Also note that it takes two clear channel assessments (CCAs) to characterize a channel. One assessment for each antenna is needed.

TLMM_M_357 - CH MAPPING - SELECT CHANNEL

Upon initiation of the channel mapping internal service, the master will perform the clear channel assessment on each channel/antenna pair (10 msec/pair) until all channels are assessed or a channel’s RSSI is below the stop-on-threshold value. The quietest channel is selected (the channel with the lowest RSSI).

Note that the listen before talk timer (LBT) is restarted each time a clear channel assessment is performed.

[Note that a channel’s RSSI value is set to the larger value of the RSSI on each antenna.]

[If ten channels are assessed it will take approximately 275 +/- 50 msec(1) (10 msec per channel/antenna pair).]

5.3.2.3 DIAGNOSTIC MAPPING

TLMM_M_358 - CH MAPPING - DIAGNOSTIC COLLECTION

The master will collect twenty milliseconds of data during diagnostic mapping. The RSSI value of each channel assessed will be returned in the external service response.

---

1. CCA of the MICS band takes approximately 275 msec. CCA of the MEDS band takes approximately 550 msec. CCA of both bands takes approximately 825 msec. Note that if HDR mode is enabled, assessment of each channel, antenna pair is lengthened by 5 msec (to allow the HDR synthesizer to stabilize). This means the CCA of the MICS band is lengthened from 275 msec to 375 msec

5.3.3 CHANNEL RECOVERY

Channel recovery is an internal service to the Data Link Layer. It is used to reestablish communication when a loss of synchronous communication is detected between the master and the slave. Loss of synchronous communication is typically caused by various types of environmental noise. The environmental noise may be transient or persistent.

The master and the slave will perform Channel Recovery operations in two distinct ways:

- Reestablish communication on the **same channel -** The channel on which the communication session was most recently established, or
- Reestablish communication on an **unspecified channel -** New channel selected by the Channel Mapping algorithm.

Following is a brief description of the two operations which are performed by the master and the slave for Channel Recovery:

Same channel recovery operation: The first approach is to recover the communication session on the same channel, with the recovery period defined by the “same channel recovery” timer. For the duration of the same channel recovery timer the master transmits Open request packets while the slave listens for an Open request packet. When a valid Open master packet is received, the slave transmits N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**). The communication session is considered reestablished when the master receives a valid Open response packet, and the same channel recovery timer is cancelled.

Unspecified channel recovery operation: The second approach is to recover the communication session on an unspecified channel, with the recovery period defined by the “unspecified channel recovery” timer. The master invokes the Channel Mapping algorithm to select a unspecified channel; meanwhile, the slave initiates asynchronous receptor wakeup. This allows the slave to listen to each channel (defined in the sniff list). Once a channel is selected, the master starts transmitting wakeup packets followed by native mode open request packets. If the slave hears a wakeup packet it will begin listening for an open packet. When a valid Open request packet is received, the slave transmits N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**). The communication session is considered reestablished when the master receives a valid Open response packet, and the unspecified channel recovery timer is cancelled.

5.3.3.1 REQUIREMENTS

The Channel Recovery algorithm is triggered when

- no valid preamble and packets are received for consecutive frames, or
- the master ID in the first master Data fragment packet, master ARQ packet, or master Diagnostic packet does not match the master ID in the last received Open request packet, or
- the slave ID in the first slave Data fragment packet or slave Diagnostic packet does not match the slave ID in the last received Open response packet, or
- the master receives a Data, Emergency, Open, or Remote Diagnostic service request when in the Inactive State.

This algorithm can be independently triggered by the master or the slave.

When the Channel Recovery algorithm is triggered, any External Service in progress is aborted and the associated un-transmitted data packets are discarded. The operations associated with the Channel Recovery algorithm are performed by using nominal data transfer mode in the Channel Recovery State.

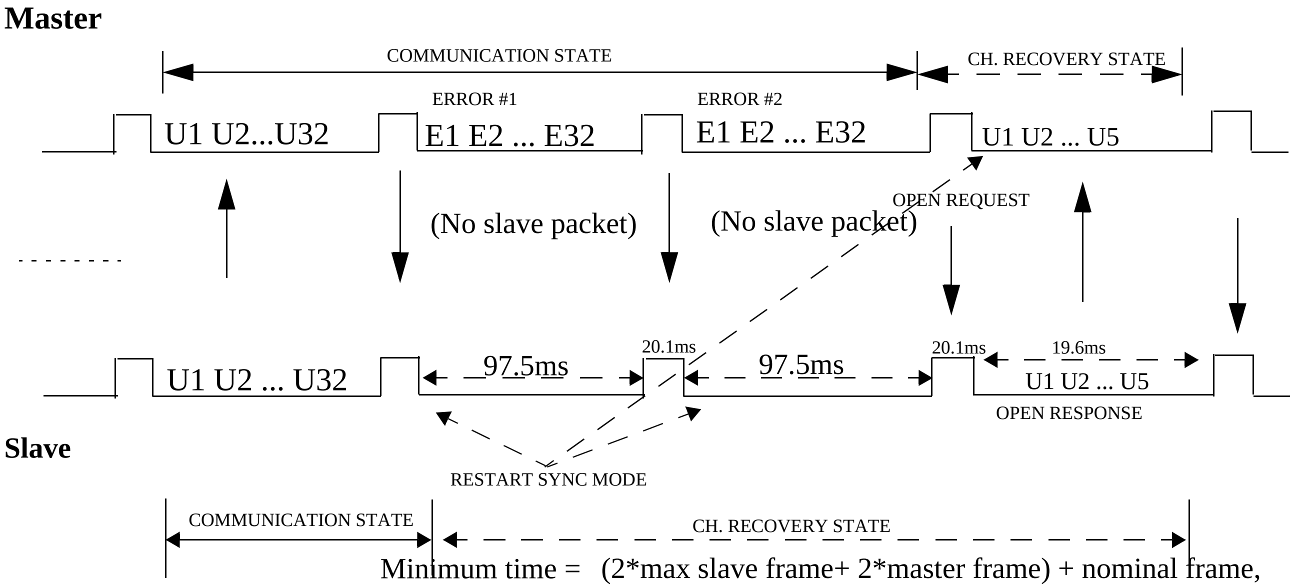
The Channel Recovery algorithm uses two local timers for its recovery operations. The names and default values of these timers are defined in **TABLE 79: CHANNEL RECOVERY TIMERS**.

TABLE 79: CHANNEL RECOVERY TIMERS

CHANNEL RECOVERY TIMERS	VALUE (MASTER)	VALUE (SLAVE)
Same **C**hannel **R**ecovery (SCR) timer[1]	440ms[2]	440ms
Unspecified **C**hannel **R**ecovery (UCR) timer	3 - 65535 sec[3]	3 - 65535 sec

1. This allows the master's and slave's Same Channel Recovery timers to overlap; see **Fig. 103 (Minimum Channel Recovery Time during Max Slave Throughput Transfer Mode)**.
2. This value comes from **Table 124 (SLAVE PARAMETERS FROM HOST)**.
3. This value comes from **TABLE 129: PARAMETERS FROM EEPROM**.

The example in **Fig. 103 (Minimum Channel Recovery Time during Max Slave Throughput Transfer Mode)** illustrates the Minimum Channel Recovery time during maximum slave throughput transfer mode.



Minimum time = (2*max slave frame+ 2*master frame) + nominal frame,
where,
(2*97.5 + 2*20.1) +19.6 = 254.8 ms
(4 ms turnaround time is being used)

This figure assumes master and slave baud rates of 14.8 and 100 Kbps respectively

Figure 103: Minimum Channel Recovery Time during Max Slave Throughput Transfer Mode

5.3.3.2 MASTER

When the Channel Recovery algorithm is triggered, the master first attempts to reestablish communication on the same channel. If this fails, then the master attempts to reestablish communication on an unspecified channel. In either case, the master transmits an Open request packet to reestablish synchronous communication. During unspecified channel recovery an open packet is proceeded by two or more wakeup packets.

The master triggers its Channel Recovery algorithm if no valid preamble and slave packets are received for at least two consecutive slave frames. When Channel Recovery is triggered, the master terminates any External Service in progress, switches the data transfer mode to a nominal mode, changes the telemetry state to the Channel Recovery State, and starts a SCR timer.

TLMC_M_120 - CH RECOVERY - TRIGGER CHANNEL RECOVERY

The Channel Recovery algorithm is triggered when any of the following conditions is true:

1. No valid slave packet is received for N consecutive frames, where N is determined using the parameter “number of consecutive frames without a valid receive packet threshold 1, 2 or 3” and the power inhibit mode. See **Table 123 (Master parameters from host)** and **Table 71 (POWER INHIBIT MODES)** respectively.
2. An Open service request is unsuccessful.
3. A Data, Emergency, Open, or Remote Diagnostic service request is received in the Inactive State (in which case, unspecified channel recovery is initiated).
4. The slave ID in the first data transaction packet or diagnostic transaction packet does not match the slave ID in the last received open packet.
5. A slave packet’s ACK bit is true but its reset ACK bit is false when both are expected to be true. This can occur when noise is received with a good message authentication code (MAC).
6. When the master/slave bit is set to “1”, it means “this” master has just received a native mode packet from another master. “This” master was expecting a packet from a slave.

[Here, ‘**data transaction**’ is used generically for Memory Data, WF Data, and Unrequested Data. The ID appears in the first fragment of the transaction.]

TLMC_M_121 - CH RECOVERY - START SAME CHANNEL RECOVERY

When the Channel Recovery algorithm is triggered (when the session began with an Open service request), then disable the security hardware; terminate the processing of any external service in progress; change to a nominal data transfer mode; inform the upper layer with a channel recovery started indication (see **Table 80 (SAME CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**; and start channel recovery by transmitting Open requests. Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (R_m) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (R_m) can be used for the duration of same channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

TABLE 80: SAME CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xA7 = Master Channel recovery indication 0x27 = Slave Channel recovery indication
Sub-type	N/A	1	The indication sub-type	0x01 = same
Size	N/A	2	The size of the info field	4
Info	Channel recovery indication status	1	status of the indication	0x04 = started
	Slave channel recovery reason	2	The reason slave same channel recovery was started.	See **Table 108 (Channel Recovery Reason)**(1)
	Master channel recovery reason	1	The reason master same channel recovery was started.	See **Table 105 (Master channel recovery reason)**(2)
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

1. This field is impertinent if the module is initialized as a master.
2. This field is impertinent if the module is initialized as a slave.

[Change to Channel Recovery State and start SCR timer, Inform the Upper Layer if an external service is aborted, inform the Upper Layer that Channel Recovery is started, Same channel recovery can’t be greater than 5 seconds.]

An SCR timer is used while reestablishing communication on the same channel. While the SCR timer is active, the master restarts the sync mode, transmits a valid Open request and waits for an slave response. If no valid response is received, then this cycle is repeated till the SCR timer expires.

TLMC_M_122 - CH RECOVERY - SAME CHANNEL RECOVERY OPEN MASTER PACKET

While in same channel recovery transmit an open request packet and wait for an open response. Repeat this process if a valid open response is not received. Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (R_m) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (R_m) can be used for the duration of same channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[During same channel recovery while SCR timer is active - Transmit an Open request, starting on an antenna and alternating on each retry, Restart the sync mode, Wait for an Open slave packet response. Repeat the above steps if a valid Open slave response is not received. **Note:** Open service is invoked locally. The actions performed for this Open service are the same as those described in **5.2 External Services**. The only exception is that upper layer is not notified of an Open request and the Open response.]

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

Synchronous communication is reestablished if an Open session response is received from a slave while the SCR timer is active. The SCR timer is cancelled and synchronous communication is resumed by transition to the Comm State.

TLMC_M_123 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL

If a valid slave Open packet is received while the SCR timer is active then transition to Comm State on successful response and send a channel recovery successful indication to the upper layer (see **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**. Additionally initialize the block cipher. See **Fig. 45 (Open - block cipher initialization)**for a secure/unsecure session initialization.

[Cancel the SCR timer; inform the Upper Layer that Channel Recovery is successful; initialize various parameters as done when Open service is processed prior to transitioning to the Comm State.]

TABLE 81: CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xA7 = Master Channel recovery indication 0x27 = Slave Channel recovery indication

TABLE 81: CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE)

Sub-type	N/A	1	The indication sub-type	0x01 = same 0x02 = unspecified
Size	N/A	2	The size of the info field	1
Info	Channel recovery indication status	1	status of the indication	0x01 = successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

If the SCR timer expires and a valid slave Open response is not received, then start Unspecified Channel Recovery. An Unspecified Channel Recovery (UCR) timer is used for reestablishing communication on a channel selected by the Channel Mapping internal service.

TLMC_M_124 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL

If a valid slave Open response is NOT received within the same channel recovery time then:

1. Start a UCR timer.
2. Select a unspecified channel via Channel Mapping internal service.
3. Start the Listen Before Talk timer.
4. Transmit wakeup packet(s) per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**.
5. Transmit Open request (on the left antenna)
6. Send an unspecified channel recovery started indication to the Upper Layer. See **Table 82 (UNSPECIFIED CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**.

TABLE 82: UNSPECIFIED CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xA7 = Master Channel recovery indication 0x27 = Slave Channel recovery indication
Sub-type	N/A	1	The indication sub-type	0x02 = unspecified
Size	N/A	2	The size of the info field	1
Info	Channel recovery indication status	1	status of the indication	0x04=started
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (R_m) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (R_m) can be used for the duration of unspecified channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[Note that the same channel duration cannot be greater than 5 seconds, otherwise the Channel Mapping algorithm will have to be initiated to reselect a channel (per the FCC).]

[Note, it is not required to have a new random number generated when entering the Unspecified Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Unspecified Channel Recovery starts during an unsecured session.]

TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION

The master will send wakeup packets as follows:

1. Transmit no wakeup packets in the polling (scanning on) case if the slave is still polling: ($time_{now} < time_{last\ native\ packet\ from\ slave}$ + slave recovery duration + remaining polling duration)
2. Transmit two wakeup packets in the polling (scanning off) case and the slave is still polling: ($time_{now} < time_{last\ native\ packet\ from\ slave}$ + slave recovery duration + remaining polling duration)
3. Transmit N wakeup packets otherwise; where N is chosen to be slightly longer than the slave's wakeup period. "N" is the parameter transmit wakeup burst length async.

[transmit wakeup burst length async is received from master parameters from host - see **Table 123 (Master parameters from host)**. Note that the antenna is toggled before the start of each wakeup burst.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup transmission is not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

While the UCR timer is active the master restarts the sync mode, transmits an Open request packet, and waits for an open response. If no response is received, then this cycle is repeated until the UCR timer expires.

TLMC_M_261 - CH RECOVERY - TOO MANY SAME CHANNEL RECOVERIES

If too many same channel recoveries are initiated in too short of a time, then unspecified channel recovery will be initiated. More specifically, if N same channel recoveries are initiated within T seconds then:

1. Start a UCR timer.
2. Disable the security hardware.
3. Select an unspecified channel via Channel Mapping internal service.
4. Start the listen before talk timer.
5. Transmit wakeup packet(s) per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**
6. Transmit open request (on the left antenna).
7. Send an unspecified channel recovery started indication to the Upper Layer. See **Table 82 (UNSPECIFIED CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (R_m) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (R_m) can be used for the duration of unspecified channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[The values of “N” and of “T” (above) come from **Table 129 (PARAMETERS FROM EEPROM)**. They are called “too many recoveries threshold” and “too many recoveries duration” respectively.]

[Note, it is not required to have a new random number generated when entering the Unspecified Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Unspecified Channel Recovery starts during an unsecured session.]

TLMC_M_125 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY

While the UCR timer is active, then:

1. Transmit an Open request packet, starting on one of the two antennas, and alternating the antenna on each retry.
2. Wait for a valid Open response packet.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (R_m) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (R_m) can be used for the duration of unspecified channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[Before Wait..., restart the sync mode.]

If there is no Open response, transmit Open on the other antenna. Continue until success, Listen Before Talk timeout, or UCR timeout.

[Note: Open service is invoked locally. The actions performed for this Open service are the same as those described in **5.2 External Services**. There are two exceptions. The first exception is that the upper layer is not notified of an Open request packet and the Open response packet. The second exception is that a “slave polling” timer is started (to interleave wakeup bursts with native mode open packets if the timer expires).]

[Note, it is not required to have a new random number generated when entering the Unspecified Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Unspecified Channel Recovery starts during an unsecured session.]

TLMC_M_126 - REMAP CHANNEL - UNSPECIFIED CHANNEL RECOVERY TIMER EXPIRATION

If no Open response at expiration of Listen Before Talk timeout:

1. Reselect best channel.
2. Restart Listen Before Talk timeout.
3. Transmit wakeup packet(s) per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**
4. Transmit an open request packet.
5. Wait for a valid Open response packet.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted.

Synchronous communication is reestablished on a different channel if an Open session response is received from an slave prior to the expiration of the UCR timer. The UCR timer is canceled and synchronous communication is resumed by transition to the Comm State.

TLMC_M_127 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL

If a valid Open response packet is received while the UCR timer is active then transition to Comm State and send a channel recovery successful indication to the upper layer (see **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**). See **Fig. 45 (Open - block cipher initialization)** for a secure/unsecure session initialization.

[Cancel the UCR timer; inform the Upper Layer that Channel Recovery is successful; initialize various parameters as when Open service is processed prior to transitioning to the Comm State. Note: Open service is invoked locally. The actions performed for this Open service are the same as described in **5.2 External Services**. The only exception is that the upper layer is not notified of an Open request packet and the Open response packet.]

If an Open response packet is not received prior to the expiration of the UCR timer, then pass an unsuccessful Channel Recovery indication to the upper layer, and transition to the Standby State.

TLMM_M_360 - CH RECOVERY - NATIVE TIMEOUT

The master will continue to send open packets for the native mode timeout duration see **Table 129 (PARAMETERS FROM EEPROM)**. If this timer expires it will send wakeup packets per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**.

[Note that the native mode timer is not used if the slave scans during polling. In this case a transition back to wakeup occurs upon an LBT timeout.]

TLMC_M_128 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL

A channel recovery failure indication will be sent to the upper layer if:

a. No open response packet is received prior to the expiration of the UCR timer, or

b. An open response packet with a status of “no BAN key” is received.

See **Table 83 (CHANNEL RECOVERY FAILED INDICATION (MASTER))**

[Transition to Inactive State, inform the Upper Layer of Channel Recovery failure, Note: Transition to Inactive State means that physical layer will be transferred to an idle state.]

.

TABLE 83: CHANNEL RECOVERY FAILED INDICATION (MASTER)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0xA7 - Master Channel recovery indication
Sub-type	N/A	1	The indication sub-type	0x02 - unspecified
Size	N/A	2	The size of the info field	1
Info	Channel recovery indication status	1	status of the indication	0x03 - timeout 0x09 - no BAN key

TABLE 83: CHANNEL RECOVERY FAILED INDICATION (MASTER)

CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

.

TABLE 84: CHANNEL RECOVERY FAILED INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x27 - Slave Channel recovery indication
Sub-type	N/A	1	The indication sub-type	0x02 - unspecified
Size	N/A	2	The size of the info field	2
Info	Channel recovery indication status	2	status of the indication	1st byte: 0x03 - timeout 2nd byte: 0x01 - without poll, or 2nd byte: 0x02 - with poll
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

5.3.3.3 SLAVE

The slave triggers its Channel Recovery algorithm if no valid preamble and master packets are received from the master for at least two frames. When Channel Recovery is triggered, the slave shall terminate any External Service in progress; switch the data transfer mode to a nominal mode; change the telemetry state to Channel Recovery State; start same channel recovery (SCR) timer; and inform the upper layer that Channel Recovery is triggered.

TLMC_S_129 - CH RECOVERY - TRIGGER CHANNEL RECOVERY

The Channel Recovery algorithm is triggered when

1. No valid master packet is received for N consecutive frames, where N is determined using the parameter "number of consecutive frames without a valid receive packet threshold 1, 2 or 3" and the power inhibit mode. See **Table 123 (Master parameters from host)** and **Table 71 (POWER INHIBIT MODES)** respectively.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:

0x0F0A : Bit3 - Received Master Packet Error Exceeded.]

2. The master ID in the master data transaction, master ARQ packet, or master Diagnostic packet does not match the master ID in the last received Open request packet; or

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:

0x0F0A :Bit4 -Received Master Type is ARQ and Master ID Mismatched,

0x0F0A: Bit5 - Received Master Type is Diagnostic and Master ID Mismatched, or

0x0F0A: Bit6 - Received Master Type is Memory Data and Master ID Mismatched.

Also Note that the master ID appears in data transaction's first fragment.]

3. No valid packet is received from the master after a secured open.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:

0x0F0A : Bit0 - First Received packet is Invalid Secured packet after open session, or

0x0F0B: Bit0 - No packet is received after open session.]

4. The master/slave bit is set to "0" it means "this" slave has just received a native mode packet from another slave. "This" slave was expecting a packet from a master.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:

0x0F0B : Bit5 - Received Slave Packet while in Communication State.]

5. Power inhibit is asserted and the power inhibit mode is idle immediate. Note that for this case the slave will not begin listening in native mode until power inhibit is deasserted.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:

0x0F0A : Bit7 -Power Inhibit triggered channel recovery.]

Once Channel recovery is triggered, the security hardware will be disabled.

The model ID and serial number information of the instrument with which the communication was successfully opened will be maintained until a different channel recovery fails.

TLMC_S_130 - CH RECOVERY - START SAME CHANNEL RECOVERY

When the Channel Recovery algorithm is triggered, transition to Channel Recovery State, start the SCR timer and send a channel recovery started indication to the upper layer. (see **Table 80 (SAME CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**).

Note that the channel recovery started indication will be sent to the upper layer only if bit "Send channel recovery indication to host" is set to "1". (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit "Send channel recovery indication to host")

[Terminate the processing of any external service in progress, stop slave transmission, and restart sync mode; change to nominal data transfer mode; inform the upper layer that Channel Recovery is started.]

Session will be reestablished with the master with which the session was opened. An SCR timer is used while reestablishing communication on the same channel. While the SCR timer is active, the slave listens for an Open request packet during the first cycle of the sync mode. If an Open request is not received then this cycle is repeated until the SCR timer expires.

TLMC_S_131 - CH RECOVERY - SAME CHANNEL RECOVERY INVOKE SCAN

While the SCR timer is active listen for a valid Open request packet from the master with which the session was opened.

[If any packet other than Open is received, then restart the sync mode and no slave response is transmitted.]

Synchronous communication is reestablished if an Open request packet is received from a master prior to the expiration of the SCR timer. The slave transmits N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**), cancels the SCR timer, and transitions to the Comm State.

TLMC_S_132 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL

While in Channel Recovery State, if the SCR timer is active and a valid Open request packet is received, then:

Transmit N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**). Note that a new random number (R_S) for the open response packet (see **Fig. 42 (Open Secure/Unsecure Response Packet)**) is generated when entering the same channel recovery and will be used for the entire (same and unspecified) channel recovery process. This is allowed because this random number (R_S) will not be used in transmitting valid information during channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

1. Initialize the block cipher. See **Fig. 45 (Open - block cipher initialization)** for a secure/unsecure session initialization.
2. Transition to the Comm State.
3. Send a channel recovery successful indication to the upper layer. See **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**. Note that the channel recovery successful indication will be sent to the upper layer only if bit “Send channel recovery indication to host” is set to “1”. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[When a valid open request packet is received followed by a diagnostic or data packet, cancel the SCR timer; inform the upper layer of successful Channel Recovery status; initialize various parameters as done when Open service is processed.]

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

If the same channel recovery timer expires and an Open request is not received, then start the Channel Recovery on a unspecified channel. A UCR timer is started, the sync mode is restarted, and the slave begins scanning.

TLMC_S_133 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL

While in Channel Recovery State, if the SCR timer expires and a valid Open request packet is NOT received, then start UCR timer and send an unspecified channel recovery started indication to the upper layer (see **Table 82 (UNSPECIFIED CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**).

Note that the unspecified channel recovery started indication will be sent to the upper layer only if bit “Send channel recovery indication to host” is set to “1”. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[Restart sync mode; report the Channel Recovery status to the application layer.]

When Open request packet is not received initiate asynchronous receptor wakeup.

TLMC_S_134 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY

While performing channel recovery, if the SCR timer expires and a valid open request packet is not received start the UCR timer. Additionally the slave will do the following:

1. Scanning off case
   a. Phase 1: listen for wakeup. The slave will continuously restart asynchronous receptor wakeup if it receives no valid wakeup
   b. Phase 2: listen for native mode open. The slave will listen for a native mode open packet if a valid wakeup is heard (and start a native mode timer). It will listen for an open packet until a valid open is received or a native mode timeout occurs (~200 msec). If a native mode timeout occurs, the slave will transition to phase 1.
2. Scanning on case
   a. Phase 1 - random channel listening - the slave will randomly select a channel and will restart the channel listening time and listen for a valid packet from the master. If no valid open packet has been received and the service duration timer is still active, the slave will switch to the phase 2 approach.
   b. Phase 2 - sequential channel listening - the slave will sequentially select the next channel and will restart the channel listening time and listen for a valid packet from the master. When every channel has been selected once the slave will switch to the Phase 1 approach.

During Channel Recovery operations, while scanning, the slave will not respond to any master packet other than an Open request packet from the master with which it was in communication.

TLMC_S_135 - CH RECOVERY - UCR TIMER ACTIVE NON OPEN MASTER PACKET

While UCR timer is active, if a master packet other than Open is received, the slave will not transmit a slave response.

Synchronous communication is reestablished if an Open session request is received from a master prior to the expiration of the UCR timer. The slave transmits N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**), cancels the UCR timer, and transitions to the Comm State.

TLMC_S_136 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL

While the UCR timer is active and a valid Open request packet from the same master is received, then:

Transmit N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**). Note that the random number (R_S) used in the open response packet (see **Fig. 43 (Open status byte (Response))**) was generated when entering the (same) channel recovery. This is allowed because this random number (R_S) has not been used in transmitting valid information.

1. Initialize various parameters as when an Open service is processed.
2. Initialize the block cipher. See **Fig. 45 (Open - block cipher initialization)** for a secure/unsecure session initialization.
3. Transition to the Comm State.
4. Send a channel recovery successful indication to the upper layer (see **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**). Note that the channel recovery successful indication will be sent to the upper layer only if bit “Send channel recovery indication to host” is set to “1”. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[When a valid open request packet is received followed by a diagnostic or data master packet, cancel the UCR timers; inform the upper layer of successful Channel Recovery status; various parameters like seq number are initialized prior to transitioning to the communication state.]

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

If an Open response is not received prior to the expiration of the unspecified Channel Recovery timer then transition to the Standby State.

TLMM_S_361 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL

While in Channel Recovery State if the UCR timer expires then transition to the Standby State and send a channel recovery failed indication to the upper layer. See **Table 84 (CHANNEL RECOVERY FAILED INDICATION (SLAVE))**.

Note that the channel recovery failed indication will be sent to the upper layer whether the bit “Send channel recovery indication to host” is set to “1” or not. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[Cancel the Scan listen timer; report the unsuccessful Channel Recovery status to the upper layer.]

[The indication to the Upper Layer will consist of Polling ON or Polling OFF based on the value of the Total Polling Duration Time in the open request packet (i.e., Zero meaning Polling OFF, otherwise Polling ON.]

5.3.4 DATA TRANSFER MODES

The purpose of the data transfer mode algorithm is to:

- Maximize the slave Memory Data throughput when higher Memory Data throughput is required, e.g., during interrogation
- Reduce the slave's battery consumption when Memory Data packets are not required.

The data transfer mode algorithm negotiates and allocates master and slave packet counts between the master and the slave. The use of this algorithm provides different master and slave data throughputs.

The transfer mode change request is initiated by the master only, in the Comm State. In order to change to a new transfer mode, the master puts the new transfer mode in the master packet. The slave compares the transfer mode in the master packet with its current transfer mode and if it is different then the slave changes to the new transfer mode and transmits the transfer mode information as part of its response packets.

The Data Link Layer provides three transfer modes:

- Nominal data transfer mode.
- Maximum slave throughput data transfer mode.
- Low Energy data transfer mode.

For definition of the three transfer mode, see **Table 85 (Data Transfer Mode)** .

TABLE 85: DATA TRANSFER MODE

TRANSFER MODE	VALUES
Low Energy	0
Nominal	1
Maximum Slave Throughput	2

Nominal data transfer mode: is used when the master packet type is not ARQ. During this mode, the master packet count is set to one, and the slave packet count is set to "N1" see parameter 'packet count for nominal mode' in **Table 123 (Master parameters from host)**, so each master frame contains one packet and every slave frame contains "N1" see parameter 'packet count for nominal mode' in **Table 123 (Master parameters from host)** packets.

Maximum slave throughput data transfer mode: is used when the master packet type is ARQ (the slave is transmitting Memory Data). During this mode, the master packet count is set to one, and the slave packet count is set to "N2" see parameter 'packet count for max throughput mode' in **Table 123 (Master parameters from host)**, so each master frame contains one packet and every slave frame contains "N2" see parameter 'packet count for max throughput mode' in **Table 123 (Master parameters from host)** packets.

Low Energy data transfer mode: is used only when the master packet type is filler. In this mode, the slave transmits only unrequested (waveform and/or unrequested) data packets. This mode is designed to save the slave's energy by limiting the transmitter and receiver's "ON" time. While in this mode the master uses positive/negative preamble correlation methods to manage the slave receiver ON times to conserve slave's battery energy, as shown in **Fig. 104 (Data Transfer Mode Examples)**.

Refer to **7.0 Appendices** for the data throughput information in each of these modes.

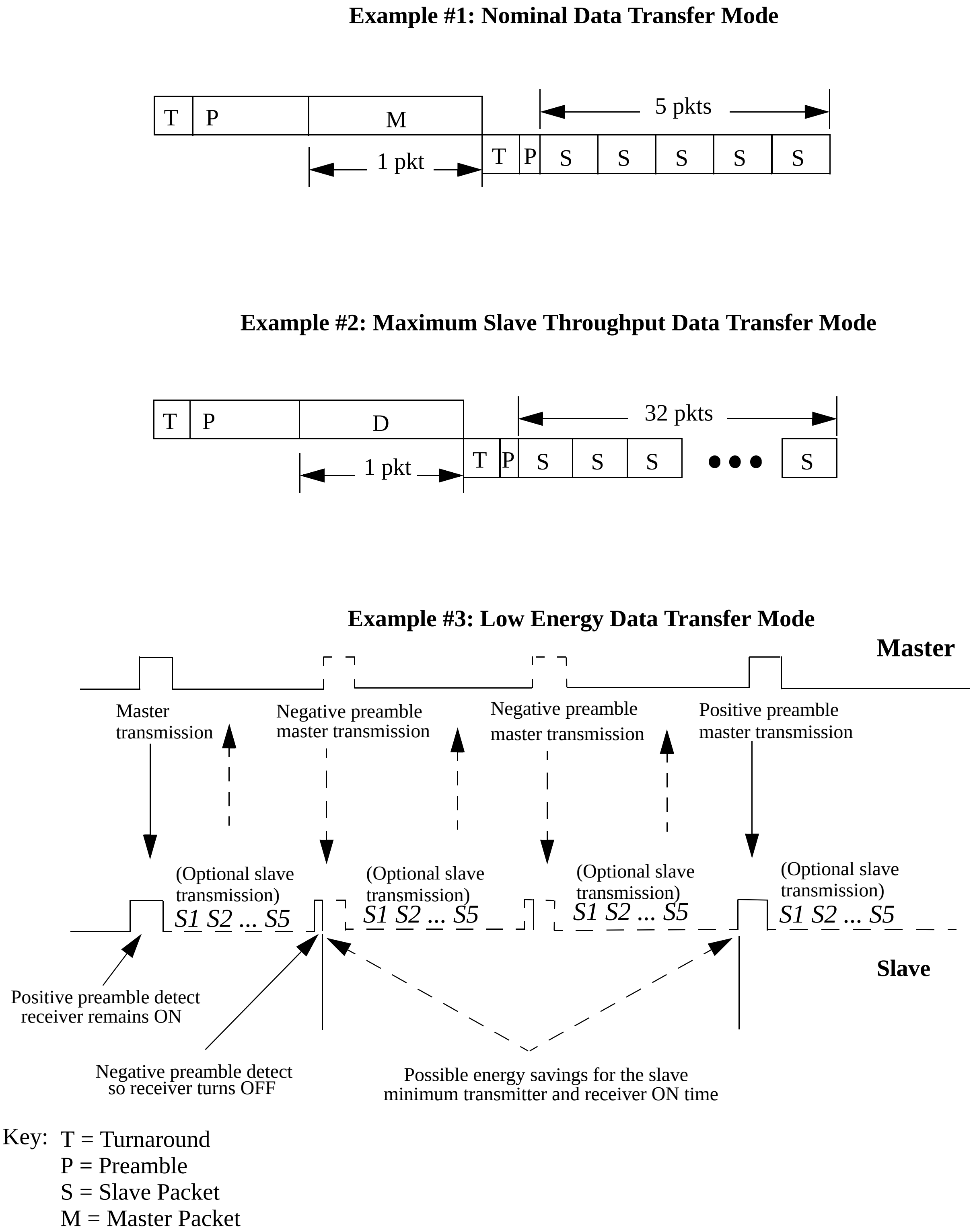


Figure 104: Data Transfer Mode Examples

5.3.4.1 REQUIREMENTS

A transition between data transfer modes is always initiated by the master. The change to the desired data transfer mode is transmitted by the master as part of the master packet.

5.3.4.2 MASTER

The change to the desired transfer mode is transmitted to the slave in the master packet.

TLMC_M_139 - DATA TRANSFER MODE - MODE TRANSITIONS

The data transfer mode will be selected according to **Table 86 (DATA TRANSFER MODE TRANSITIONS)**.

TABLE 86: DATA TRANSFER MODE TRANSITIONS

MASTER PACKET	CONDITION	DATA TRANSFER MODE
Diagnostic (filler)	N or more consecutive filler packets and reply timer inactive(1)	Low Energy
Diagnostic (filler)	N–1 or fewer consecutive filler packets or reply timer active	Nominal
Diagnostic (request)	—	Nominal
Data	Single-fragment master data transaction. (See footnote 1 below)	Max slave(2) throughput
Data	Multi-fragment master data transaction (not last fragment)	Nominal
Data	Multi-fragment master data transaction (last fragment) (See footnote 1 below)	Max slave throughput
ARQ	First fragment of final slave transaction not received yet (See footnote 1 below)	Max slave throughput
ARQ	First fragment of final slave transaction has been received	Nominal
Other (not listed above)	—	Nominal

1. Delay the entry into low energy mode if the reply timer is active, otherwise the slave data link layer will ignore a data service request from the slave host and memory read and write will fail.
2. Note: The data transfer mode is nominal if the max data throughput mode is not enabled (see figure 53 master ULH byte)

[The value of “N” above comes from **Table 129 (PARAMETERS FROM EEPROM)**. It is called “low energy threshold (consecutive filler master packets)”.]

TLMC_M_140 - DATA TRANSFER MODE - NOMINAL TO LOW ENERGY MODE TRANSITIONS.

When low energy mode is enabled and if the consecutive filler packet count is greater than or equal to the low energy threshold value, then data transfer mode transition is made from nominal to low energy.

[See **Table 129 (PARAMETERS FROM EEPROM)**.]

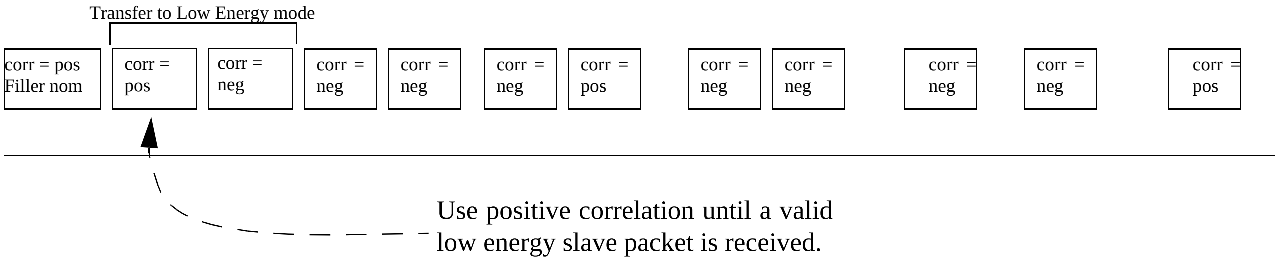
TLMC_M_141 - DATA TRANSFER MODE - NEGATIVE PREAMBLE RATIO IN LOW ENERGY MODE

The master will use a negative correlation preamble on N - 1 of N packets when the master and the slave are both in the low energy mode unless noted below. The Nth master packet will use a positive correlation .

Exception to the N-1 of N downlinks include the following:

- The parameter "frequent waveform expected" is true, or
- There is a new downlink sequence number for the go back N waveform acknowledgement and the parameter "frequent waveform expected" is false, or
- The ack bit is set to false in any uplink packet received with a good MAC and the parameter "frequent waveform expected" is false, or
- The uplink frame contains no packets with good MAC and the parameter "frequent waveform expected" is false

[Note: The “N” Correlate Negative Ratio value is defined in **Table 129 (PARAMETERS FROM EEPROM).**]



Use positive correlation until a valid low energy slave packet is received.

Note: Slave frames are not shown.

Figure 105: Example (N = Correlate_Negative_Ratio = 5)

TLMC_M_142 - DATA TRANSFER MODE - LOW ENERGY TO NOMINAL TRANSITION

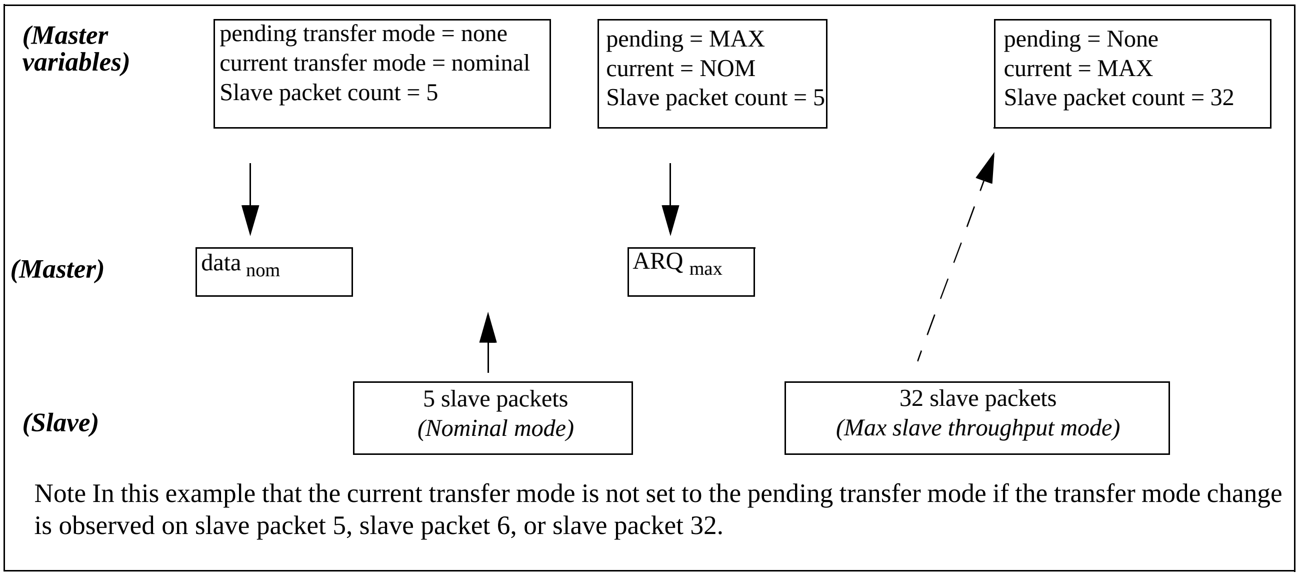
A transition from low energy to nominal mode will be made if the present master packet is a non-filler packet and the previous master packet was a filler packet in a low energy mode.

TLMC_M_143 - DATA TRANSFER MODE - TRANSITION TO NEW MODE ON MASTER PACKET

If the new data transfer mode packet is successfully received by the slave, then the master transitions to the new data transfer mode. Note the master will transition immediately when the nominal packet count is one packet.

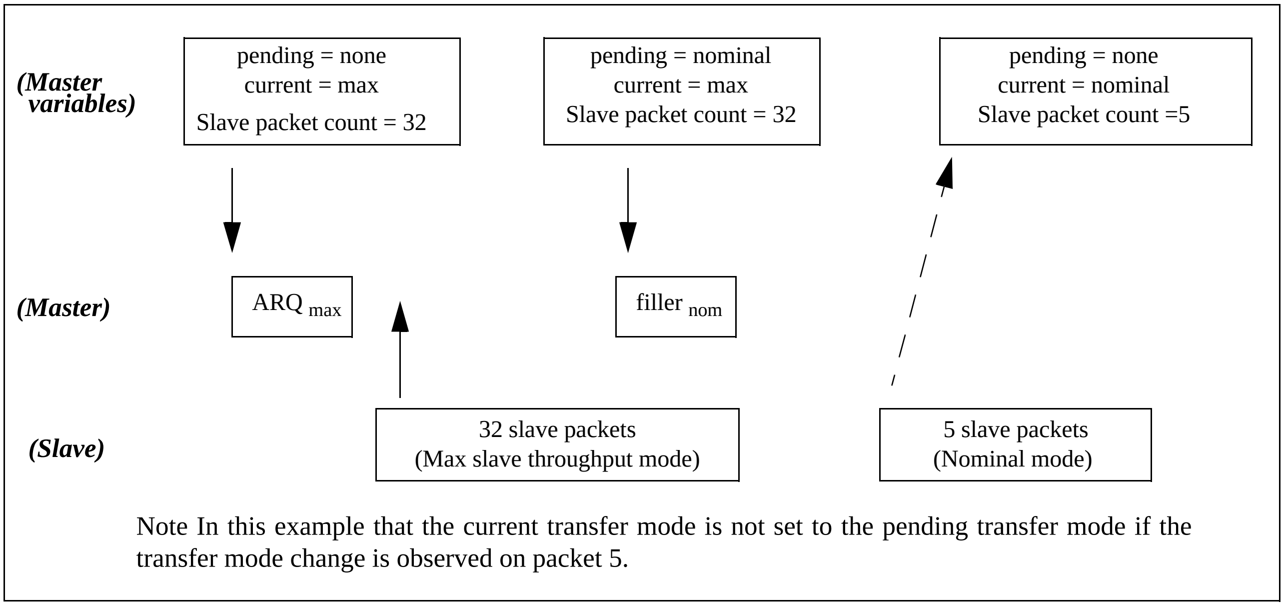
[Note that channel recovery will be triggered if the first N1-1 slave packets are corrupted upon nominal to max slave throughput or max slave throughput to nominal transitions. This occurs since one station has its slave packet count set to N1 (see parameter ‘packet count for nominal mode’ in **Table 123 (Master parameters from host)**) and the other has its packet count set to N2 (see parameter ‘packet count for max throughput mode’ in **Table 123 (Master parameters from host)**). Physical layer timing cannot be maintained in this scenario.]

[The current transfer mode is set to the pending transfer mode when the slave packet indicates the slave has changed.]



Note In this example that the current transfer mode is not set to the pending transfer mode if the transfer mode change is observed on slave packet 5, slave packet 6, or slave packet 32.

Figure 106: Nominal to Max Slave Throughput Mode Transition



Note In this example that the current transfer mode is not set to the pending transfer mode if the transfer mode change is observed on packet 5.

Figure 107: Max Slave Throughput to Nominal Mode Transition

5.3.4.3 SLAVE

The slave compares the transfer mode in the master packet with its current data transfer mode. If the current data transfer mode does not match the data transfer mode in the master packet then the slave will transition to the data transfer mode specified in the master packet. The slave's data transfer mode is transmitted as part of the slave packets.

TLMC_S_144 - DATA TRANSFER MODE - VALIDATE TRANSFER MODE AND RESPONSE

Upon receipt of a valid master packet, the slave will check the telemetry data transfer mode, and will perform as follows:

1. If the current data transfer mode matches the transfer mode in the master packet then the slave data transfer mode will not change.
2. If the current data transfer mode does not match the transfer mode of the master packet then the slave will transition to the data transfer mode as specified in the master packet.
3. The slave packet will contain the slave’s data transfer mode indication.
4. When in low energy mode slave will always uplink a filler packet when there is no data to be transmitted.

5.3.5 RESET

A master uses this service to reset Telemetry M communications to a known state. The following bullets briefly summarize this service:

- On the master side, the data link layer initiates the setting.
  - Data service - set if previous data service failed; false otherwise
  - Emergency service - unconditionally reset
- On the master, the Data Link Layer sets the Reset bit in the next master packet to the slave.
- On the slave, the Data Link Layer discards unacknowledged data packets and resets the Data Link Layer status counters and timers.
- On the slave, the Data Link Layer sets the Reset ACK bit in the next slave packet to the master.
- On the master, the Data Link Layer detects the Reset ACK in the slave packet, discards any unacknowledged data packets, and resets the Data Link Layer status counters and timers.

Fig. 108 (Reset Sequence Diagram) illustrates the Reset Service operations.

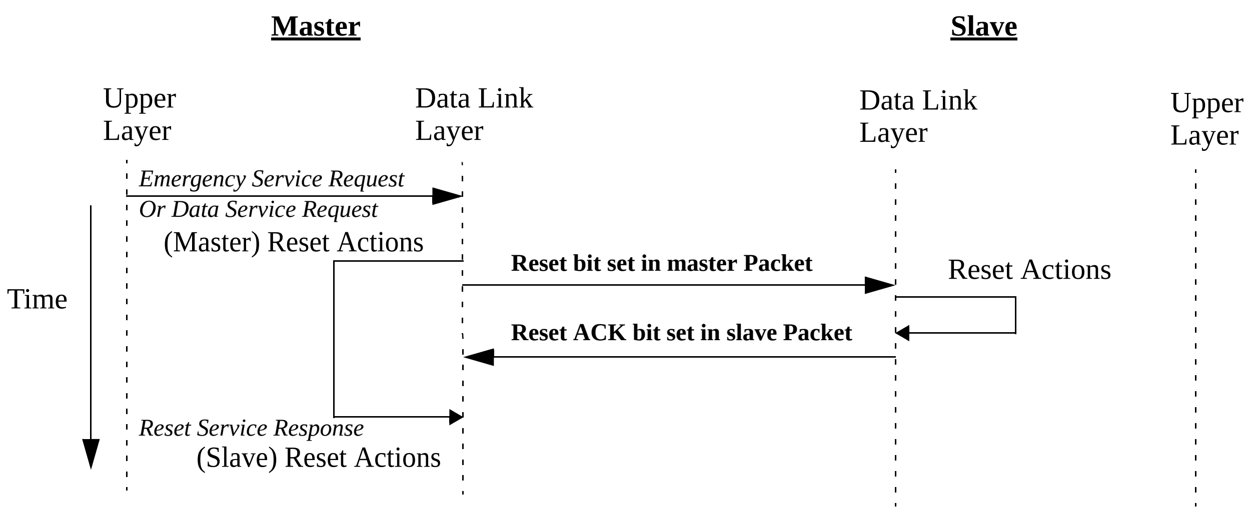


Figure 108: Reset Sequence Diagram

5.3.5.1 REQUIREMENTS

The following actions are performed for the Reset internal service:

- Reset local station parameters
- Reset applicable communication parameters.

TLMM_M_362 - RESET SERVICE - INVOKE SERVICE 1

The master will invoke the reset internal service upon receipt of an emergency service request.

TLMM_M_363 - RESET SERVICE - INVOKE SERVICE 2

The master will invoke the reset internal service upon receipt of a data service request when the most recent data or emergency service request failed.

TLMC_M_145 - RESET SERVICE - MASTER PACKET WITH RESET BIT

The master will do the following when the reset internal service is invoked:

1. Discard pending master data packet
2. Reset the master data packet sequence number
3. Set the reset bit to 1 in the master upper layer header byte. See **5.2.11.2**.
4. Set the Reset bit to 1 in the control byte of the next master packet.

The master then will wait for an slave packet with Reset ACK bit set in the Control Byte.

[An emergency service may result in discarded data packets. (Emergency interrupts a data service in progress. A data service will not result in discarded data packets since a data service cannot interrupt an “in-progress” service.)]

TLMC_M_146 - RESET SERVICE - MASTER RESET REQUEST RETRANSMISSION

If no valid slave packet with Reset ACK bit set is received from the remote station in the next slave frame, the Data Link Layer shall continue to retransmit the packet with the Reset bit set. The master shall continue this until a valid slave packet with Reset ACK bit set is received or the reply duration timeout occurs.

TLMC_S_147 - RESET SERVICE - RESET REQUEST VALIDATION

Upon receiving a valid master packet with the Reset bit set, the slave shall do the following:

1. Discard pending slave data packets.
2. Reset parameters for the ARQ algorithm, including the master data packet sequence number, the slave Memory Data sequence number, transaction number, fragment number; and the ARQ sliding window.
3. Set the Reset ACK bit in the slave packets.

Any data packets pending transmission or retransmission shall be discarded.

TLMC_M_148 - RESET SERVICE - OPERATION SUCCESSFUL

Upon receiving a valid slave packet with the Reset ACK bit set, the master shall do the following:

1. Discard pending data slave packets.
2. Reset parameters for the ARQ algorithm, including the slave Memory Data sequence number, transaction number, fragment number, and the ARQ sliding window.

Any data packets pending transmission or retransmission shall be discarded.

5.3.6 WIRELESS WAKEUP

Any node should be able to wake up any other node in the system. To achieve this, all nodes listen periodically for a wakeup transmission. The listening nodes are called receptors. A node wishing to communicate is called an instigator. The instigator is usually the master, but could be the slave. Once the instigator's wakeup burst is complete, the instigator begins transmitting native mode packets (e.g., discover, open, etc.). Once the receptor hears a valid wakeup, it begins listening and replies in native mode. See **Fig. 109 (Example of asynchronous wakeup (followed by native mode discover))**.

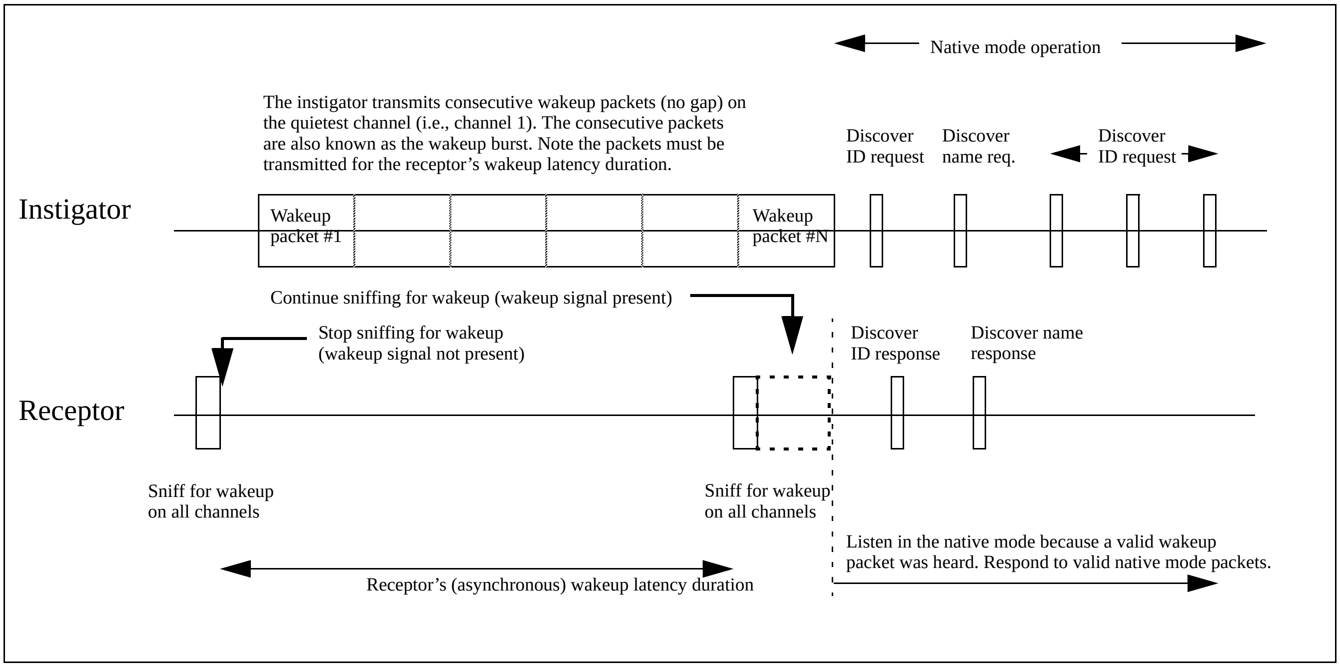


Figure 109: Example of asynchronous wakeup (followed by native mode discover)

It is desirable to have two types of wakeup mechanisms: synchronous and asynchronous.

- Asynchronous wakeup - used when an instigator node wants to wakeup one or more receptors. The instigator assumes he has no knowledge of the receptor's wakeup timing. The instigator's wakeup burst must be greater than the receptor's wakeup latency.
- Synchronous wakeup - used when an instigator node wants to wakeup a specific receptor(s). The instigator assumes he has knowledge of the receptor's timing. This allows the instigator to use a shorter wakeup burst.

Synchronous wakeup can save energy by utilizing a shorter wakeup burst(2). Synchronous wakeup should only be used when nodes stay in continuous close proximity and they exchange data frequently. Asynchronous wakeup should be used otherwise.

The instigator's asynchronous wakeup burst must be greater than or equal to the receptor's wakeup latency duration. This guarantees that the instigator can wake up the receptor (provided the instigator and receptor are within range of one another). The receptor's (asynchronous) wakeup latency duration is optimized to reduce the receptor's current drain, yet provide acceptable user responsiveness (see **Figure 109**).

2. Wakeup burst is defined as N consecutive wakeup packets.

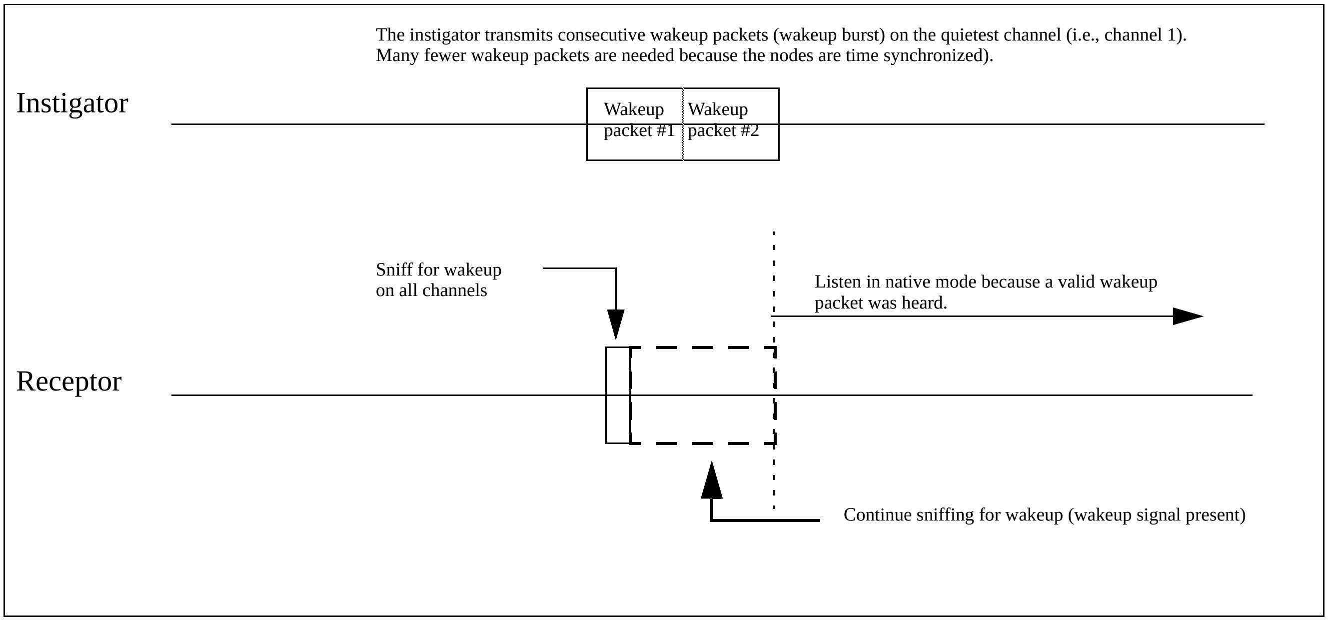


Figure 110: Synchronous wireless wakeup

In the synchronous wakeup mode, the instigator and receptor are synchronized. They each wake up at the same time. The wakeup packet tells the receptor which channel to use for native mode. This method saves (instigator) energy since its wakeup burst can be much shorter (see **Fig. 110 (Synchronous wireless wakeup)**).

A node acting as an instigator may wakeup one or more nodes. There are four types of wakeup commands:

- **Global broadcast wakeup -** typically used to wakeup all nodes within range. The instigator sends wakeup packet(s) with the wakeup type (in the header byte) to set to global. Additionally, it sets the station ID to the user supplied receptor ID and it it's instigator's ID (see **Figure 112**). All nodes of a certain model will wakeup if the user supplied receptor ID is (specified, wild card) for model and sub model respectively. Each receptor that receives a valid wakeup packet will “wakeup” (begin listening in the native mode).
- **Non-local broadcast wakeup -** used to wakeup all nodes not in any network. The instigator sends wakeup packet(s) with the wakeup type (in the header byte) set to non-local broadcast. Additionally, it sets the station ID to its own ID (see **Figure 112**). Each receptor that receives this packet with a good CRC will “wakeup” (begin listening in the native mode) if they are not in any network - it is for future use.
- **Local broadcast wakeup -** used to wakeup all nodes in a local network. The instigator sends the wakeup packet(s) with the wakeup type (in the header byte) set to local broadcast. Additionally, it sets the station ID to the network ID (see **Figure 112**). Each receptor in the network will compare his network table ID to the receptor bitmap. If the corresponding bit is set, the receptor will “wakeup” (begin listening in the native mode).
- **Unicast wakeup -** used to wakeup a specific receptor node The instigator sends the wakeup packet(s) with the wakeup type (in the header byte) set to unicast. Additionally it sets the station ID to the receptor's ID (see **Figure 112**). Each receptor that receives this packet with a good CRC will compare the wakeup station ID with its own ID. If they match, the receptor will “wakeup” (begin listening in the native mode).

5.3.6.1 MASTER (INSTIGATOR) SERVICE REQUEST

Wireless wakeup is an internal service utilized by the following external services: discover, open, polling, medical event, and network. Wireless wakeup is also utilized by the channel recovery internal service when the master knows the slave is no longer polling. During channel recovery, the master interleaves wireless wakeup with native mode open packets.

TABLE 87: INSTIGATOR WAKEUP VS. SERVICE

	GLOBAL	LOCAL	NON-LOCAL	UNICAST
Discover full external service (see **5.2.9.2**)(1)	X		X(2)	
Open external service (**5.2.10**)(3)				X
Security external service (see **5.2.19.1**)(4)				X
Network external service - data (see **5.2.18.4**)(5)		X		
Channel recovery internal service (see **5.3.3**)(6)				X

1. The discover wakeup packets will be transmitted asynchronously. The number of wakeup packets defined by “Transmit wakeup burst length async” in **Table 123 (Master parameters from host)** will be transmitted. If “Discover wakeup type” = 0, (see **Table 123 (Master parameters from host)**, The instigator will transmit Global wakeup packets.If “Discover wakeup type” = 1, The instigator will transmit Non-local wakeup packets.The discover full external service finds all nodes within range (of “this” master). The receptor only wakes up if the model (or model/sub-model) is wildcard OR if these values match its ID.
2. Shall be used to find all the nodes that have not been included in a network. For example: to initialize the network (build the network).
3. If “wakeup sync type” bit0 - “Open” (see **Table 129 (PARAMETERS FROM EEPROM)**) is set to “1”, the open wakeup packets will be transmitted synchronously. The number of wakeup packets defined by “txWakeUpBurstNumberSync” (See **A44088**, **Telemetry M RF Module EEPROM Specification**) is used. If “wakeup sync type” bit0 - “Open” is cleared “0”, the open wakeup packets will be transmitted asynchronously. The number of wakeup packets defined by “Transmit wakeup burst length async” in **Table 123 (Master parameters from host)** will be transmitted.
4. The security wakeup packets will be transmitted asynchronously. The number of wakeup packets defined by “Transmit wakeup burst length async” in **Table 123 (Master parameters from host)** will be transmitted.
5. If “wakeup sync type” bit1 - “Network” (see **Table 129 (PARAMETERS FROM EEPROM)**) is set to “1”, network wakeup packets will be transmitted synchronously. The number of wakeup packets defined by “txWakeUpBurstNumberSync” (See **A44088**, **Telemetry M RF Module EEPROM Specification**) is used. If “wakeup sync type” bit1 - “Network” is cleared “0”, network wakeup packets will be transmitted asynchronously. The number of wakeup packets defined by “Transmit wakeup burst length async” in **Table 123 (Master parameters from host)** will be transmitted.
6. During unspecified channel recovery, only two wakeup packets will be transmitted and will be transmitted asynchronously.

TABLE 88: INSTIGATOR WAKEUP INTERNAL SERVICE FIELD SOURCE

FIELD	SOURCE	SIZE	DESCRIPTION	VALUE
Wakeup type	Service request from host	1 byte	The wakeup type. Used to wakeup a specific receptor node, or a group of receptor nodes.	Global, non-local, local, or unicast.
Receptor ID	Service request from host	6 bytes	The station ID of the receptor node.	This parameter is only pertinent to the unicast wakeup type. It is not pertinent to the following wakeup types: global, non-local, and local.
Receptor bit map	Service request from host	1 byte	A bitmap telling each local node if it should respond.	See **Fig. 111 (Receptor bit map)** pertinent to local broadcast only
Channel	CCA	1 byte	The channel number within the specific band (for native mode operation).	1-10 MICS 11-30 MEDS
Transmit modulation for native mode	Configuration	1 byte	The modulation type of the native mode transmission.	see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
Receive modulation for native mode	Configuration	1 byte	The modulation of the native mode receive.	see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
Transmit data rate out of session	Configuration (initialization data from host)	1 byte	The default transmit data rate (or bit rate).	see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
Receive data rate out of session	Configuration (initialization data from host)	1 byte	The default receive data rate (or bit rate).	see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
User bytes	Service request from host	10 bytes	The user data bytes. These bytes are used for very small messages requiring minimal latency.	Each byte has the range 0 to 0xff.
Reed Solomon	Configuration	1 byte	Error detection/ correction circuitry.	Enabled/disabled
Wakeup security enable	Initialization value from host	1 byte	Privacy, authentication, etc.	Enabled/ disabled This parameter is only pertinent to the local broadcast wakeup type.(1)
Wakeup User Data Present	Service request from host	1 byte	Enabled when user data is exists in the User bytes	Enabled/disabled

1. Local broadcast is the only wakeup type that provides both the receptor and instigator IDs. Both IDs are needed to secure the wakeup user data.

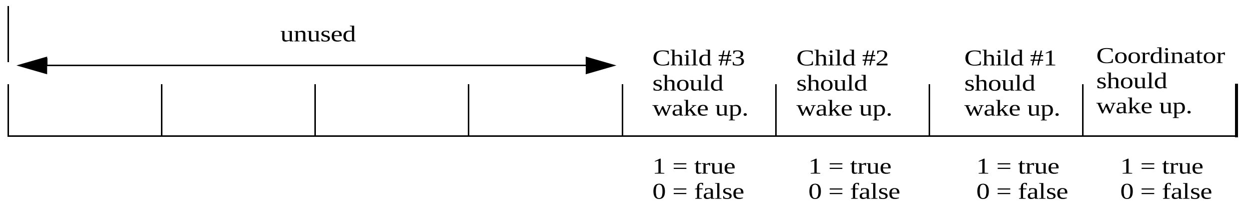


Figure 111: Receptor bit map

5.3.6.2 REQUIREMENTS

TLMM_M_364 - WIRELESS WAKEUP - INITIATE INSTIGATOR WAKEUP

A telemetry M instigator node, by initiating a external service: discover, open, security, or network shall invoke the wakeup internal service. Channel recovery shall also invoke the wakeup internal service. See **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**

[Information pertaining to wakeup, provided from the external service, will be transferred to the wakeup internal service. See**Table 87 (INSTIGATOR WAKEUP VS. SERVICE)**]

5.3.6.3 WAKEUP PACKETS

A wakeup packet consists of 22 bytes with 3 bytes of baseband hardware generated CRC (cyclic redundancy check) appended to the end.

- **Header Byte** - defines the control of the wakeup packet. The header byte defines items such as wakeup type (Global broadcast wakeup...) and user data security.
- **Sequence Bytes**- the next two bytes, in the wakeup packet, are reserved for the wakeup sequence number. For the instigator, the sequence number tracks how many wakeup packets are left to send. The sequence number bytes are filled in by the baseband hardware. For the receptor, the sequence number bytes are used to determine how many packets are left before the end of the wakeup burst.
- **Station Id and Receptor Bit Map Bytes**- the next 7 bytes are the station id and the receptor bit map. These bytes are used to determine the id of the node who is sending the wakeup packets and the ids of the nodes that should be receiving the wakeup packets The bytes are used differently depending on the type of wakeup packet being sent.
- **Configuration Bytes**- the two configuration bytes, in the wakeup packet, are used to setup the instigator's transmitting hardware and the receptor's receiving hardware to operate in a synchronized manner. Hardware settings such as modulation and data rate are saved in the configuration bytes.
- **User Data Bytes**- the user data bytes can be used to send small amounts of user defined data quickly, by not having to open a native mode session with the receptor. The user data can be sent unsecured or sent secured. Security reduces the number of user data bytes available, to the user, to four.

Many of the fields in the wakeup packet will be filled in by information provided by the instigator's internal wakeup service request.

5.3.6.4 REQUIREMENTS

Header	Wakeup seq #	Station ID	Receptor bit map	Configuration	User data	CRC
1 byte	2 bytes	6 Bytes	1 byte	2 bytes	10 bytes	3 bytes

Figure 112: Wakeup Packet

TLMM_MS_365 - WIRELESS WAKEUP - WAKEUP PACKET

The wakeup packet shall be defined as the following: see **Fig. 112 (Wakeup Packet)**

[For more information on how the individual wakeup packet bytes are defined see the following requirements:

TLMM_MS_366 - WIRELESS WAKEUP - WAKEUP HEADER BYTE,

TLMM_MS_367 - WIRELESS WAKEUP - WAKEUP SEQUENCE NUMBER BYTES,

TLMM_MS_368 - WIRELESS WAKEUP - WAKEUP STATION ID BYTES,

TLMM_MS_369 - WIRELESS WAKEUP - WAKEUP RECEPTOR BIT MAP BYTE,

TLMM_MS_370 - WIRELESS WAKEUP - WAKEUP CONFIGURATION BYTES,

TLMM_MS_371 - WIRELESS WAKEUP - WAKEUP USER DATA BYTES]

TLMM_MS_366 - WIRELESS WAKEUP -WAKEUP HEADER BYTE

The wakeup packet header byte shall have the following format:

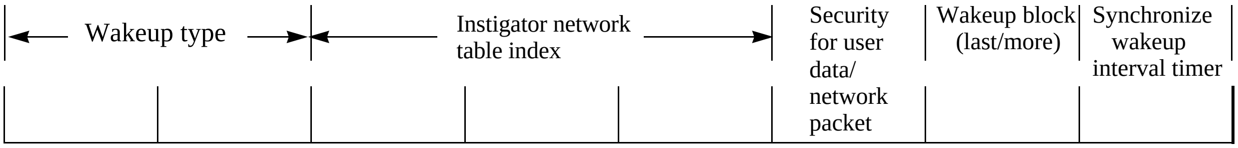


Figure 113: Wakeup header byte

1. Wakeup type - see **Table 89 (WAKEUP TYPES VALUES)**
2. Instigator network table index - see **Table 90 (WAKEUP NETWORK ID INDEX VALUES)**
3. Wakeup user data/network packet security enable - a value of one enables user data or network data security; a value of zero disables user data and network security [See **Table 91 (SECURE USER DATA/NETWORK PACKET)**
4. Last wakeup block - a value of one means this is the final block of wakeup packets; a value of zero means there will be more
5. Synchronize wakeup interval timer - a value of one means synchronize; the value is zero otherwise

[See **TLMM_MS_371 - WIRELESS WAKEUP -WAKEUP USER DATA BYTES** and **TLMM_M_372 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA INSTIGATOR**]

TABLE 89: WAKEUP TYPES VALUES

VALUE	TYPE
00	Unicast wakeup
01	Local broadcast wakeup
10	Non-local broadcast wakeup
11	Global broadcast wakeup

TABLE 90: WAKEUP NETWORK ID INDEX VALUES

VALUE	ID
000	Network Id (Coordinator)
001	Child 1
010	Child 2
011	Child 3

TABLE 91: SECURE USER DATA/NETWORK PACKET

User Data (see **Fig. 115 (Wakeup configuration bytes)**)	Security for User data/network packet (see **Fig. 113 (Wakeup header byte)**)	Note
0	0	No user data in the local wakeup packet; Network packet is not secured.
0	1	No user data in the local wakeup packet; Network packet is secured.
1	0	There is User data in the local wakeup packet; Neither user data nor network packet is secured.
1	1	There is User data in the local wakeup packet; Both user data and network packet are secured.

TLMM_MS_367 - WIRELESS WAKEUP -WAKEUP SEQUENCE NUMBER BYTES

The wakeup packet sequence bytes shall have the following format:

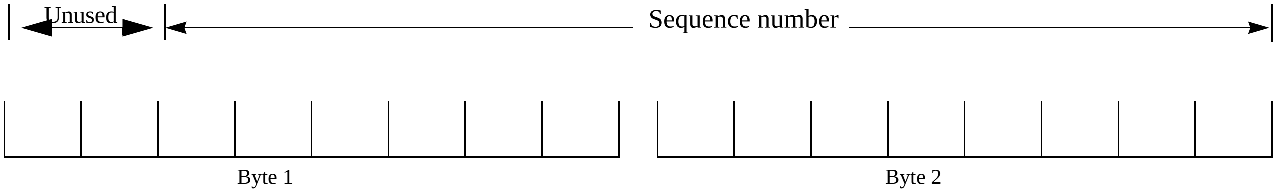


Figure 114: Wakeup sequence number bytes

1. Wakeup packet sequence number - The instigator data link layer shall write the baseband hardware’s starting sequence number and ending sequence number (where start > end)

[The baseband hardware shall fill in the sequence number, in the wakeup packet, send the wakeup packet, then decrements the fourteen bit sequence number and then shall send another wakeup packet until the end sequence number has been sent.]

TLMM_MS_368 - WIRELESS WAKEUP -WAKEUP STATION ID BYTES

The wakeup packet shall have six station id bytes having the following format:

1. station ID = network ID for the local broadcast wakeup type
2. station ID = receptor ID for the unicast wakeup type
3. station ID = for the non-local broadcast wakeup type, the three most significant bytes are the receptor model and sub-model which can be all wildcard (zero). The three least significant bytes are the instigator model and sub-model.
4. station ID = for the global broadcast wakeup type, the three most significant bytes are the receptor model and sub-model which can be all wildcard (zero). The three least significant bytes are the instigator model and sub-model.

TLMM_MS_369 - WIRELESS WAKEUP -WAKEUP RECEPTOR BIT MAP BYTE

The wakeup packet shall have a receptor map byte of the following format:

1. Each bit shall represent a node (child) in the network
2. A value of one in a given bit shall indicate the node (child) shall be woken up
3. A value of zero in a given bit shall indicate the node (child) shall not be woken up see **Fig. 111 (Receptor bit map)**

TLMM_MS_370 - WIRELESS WAKEUP -WAKEUP CONFIGURATION BYTES

The wakeup packet shall have two wakeup configuration bytes having the following format:

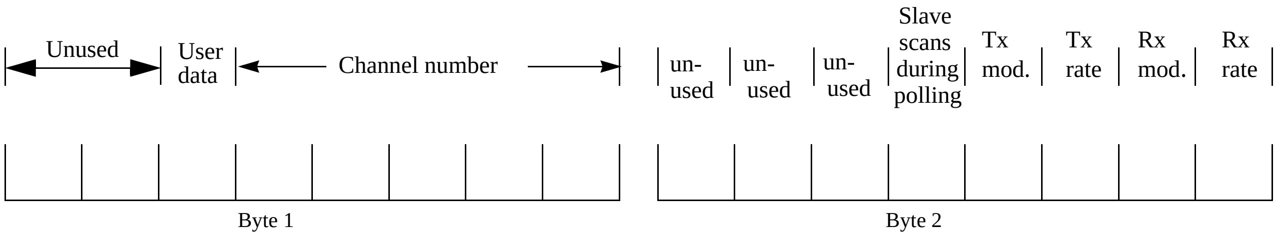


Figure 115: Wakeup configuration bytes

1. User data - a value of one indicates that user data exists; a value of zero indicates that no user data exists
2. Channel number - channel number of 1 to 30 to be used in native mode
3. Slave scans during polling - a value of one means the slave should scan during polling, a value of zero means the slave should alternate between native mode listening and consecutive asynchronous receptor wakeup
4. Instigator native mode transmit modulation - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
5. Instigator default transmit data rate - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**, used outside of the session.
6. Instigator native mode receive modulation - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
7. Instigator default receive data rate - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**,used outside of the session.

TABLE 92: TX/RX DEFAULT DATA RATE TABLE

	BAND	
TX/RX MODULATION	WIDE$^{(1)}$	NARROW$^{(2),(3)}$
DQPSK (LOW)	380 Kbps $(00)_b$	97 Kbps $(01)_b$
DBPSK (HIGH)	190 Kbps $(10)_b$	48 Kbps $(11)_b$

1. Wide is defined as the value 0
2. Narrow is defined as the value 1
3. MEDS is always narrow band. MICS can be wide band or narrow band.

TLMM_MS_371 - WIRELESS WAKEUP -WAKEUP USER DATA BYTES

The wakeup packet shall have 10 bytes of user data, these bytes will be provide as part of initialization data from the host. When wakeup user data security is enabled only 4 bytes shall be available for user data. The remaining 6 bytes shall be used for a 3 byte security counter and for a 3 byte security MAC.

[When security is enabled, the first 4 bytes are used for the user data bytes. Bytes 5,6 and 7 are used for the security counter. Bytes 8,9, and 10 are used for the security MAC.]

TLMM_M_372 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA INSTIGATOR

The instigator will perform the following to secure the wakeup user data bytes when the Wakeup security byte is enabled:

1. Generate an random number and use 3 bytes as the instigator's secure wakeup counter (CTR_i).
2. Calculate the AES_out using the block cipher using the instigator's secure wakeup counter (CTR_i), instigator's station ID (ID_i) the four user data bytes and the receptor bit map for the data input, and the BAN key (K_{BAN}) for the key input.

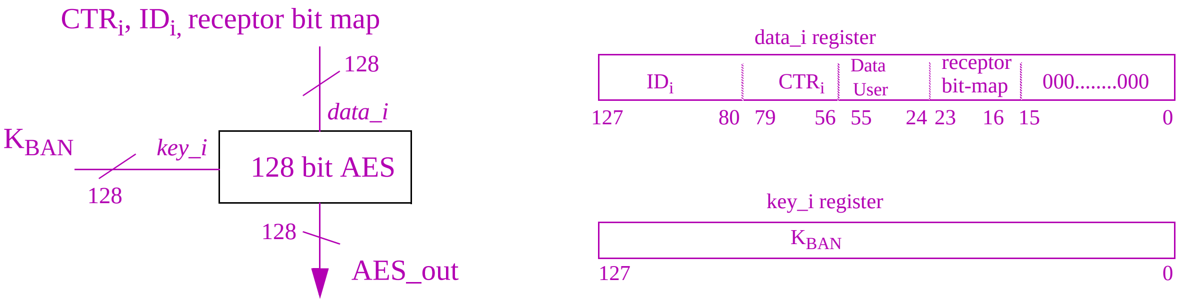


Figure 116: Calculate AES_out for wakeup security

3. Encrypt the 4 bytes of user data by calculating the key stream using the block cipher shown:

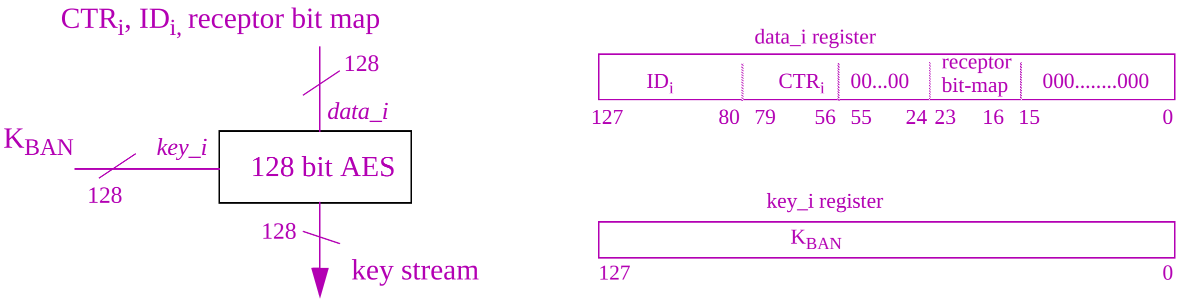


Figure 117: Calculate keystream to encrypt/ decrypt user data in a wakeup packet

4. Complete the user data encryption by performing an exclusive or of the user data with the key stream. created in step 3.

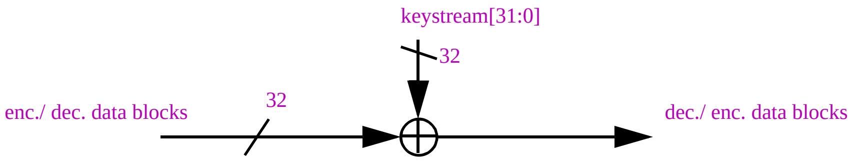


Figure 118: Encrypt/ decrypt the wakeup data

The 10 bytes of user data will contain the following bytes: four bytes encrypted user data, three byte of instigator secure wakeup counter (CTR_i), and three byte of MAC (AES_out[23:0].

5.3.6.5 WAKEUP TX (INSTIGATOR)

In order to have a successful synchronous wakeup sequence, the instigator must begin transmitting it's wakeup packets before the receptor's sniff interval begins, and must transmit enough wakeup packets to assure that at least one complete wakeup packet is sent after the receptor's sniff interval begins. The original synchronization should be done with an asynchronous transmission (restart wakeup interval upon the receipt of last packet, last block). Subsequent wakeup transmissions can use a much shorter synchronous wakeup burst. If one or more synchronous wakeup bursts fail, another asynchronous wakeup burst should be used. Finally the number of wakeup packets to use during a synchronous wakeup burst is dependent on the clock drift (how long since the last wakeup burst). The sync start time and synchronous wakeup burst length should be adjusted accordingly.

For asynchronous wakeup, the instigator must send a duration of wakeup packets that is at least slightly longer than the receptor's sniff interval. For long duration wakeup transmissions, the baseband hardware will break every 5 seconds to execute a required clear channel assessment (CCA).

5.3.6.6 REQUIREMENTS

TLMM_M_373 - WIRELESS WAKEUP -WAKEUP INTERVAL SYNC START TIME

To determine when the wakeup packets transmissions should start, the instigator shall set the wakeup interval sync start time.

[This defines the amount of time the wakeup transmission should start before the receptor's sniff interval starts. This is needed to compensate for clock drift between the instigator and receptor(s). The wakeup interval sync start time is also needed to account for the CCAs that will occur before the next sync interval.

To achieve a successful synchronous wakeup transmission, the instigator's sniff interval should match the receptor's sniff interval. The instigator's sniff interval is not in the EEPROM nor the parameters from host. This should be set with a local data service request or via the (address, value) pairs of the master initialize service.]

TLMM_M_374 - WIRELESS WAKEUP -WAKEUP INITIAL CCA

The instigator shall perform a clear channel assessment (CCA) before the start of wakeup packet transmission.

[The above requirement applies to both synchronous instigator wakeup and asynchronous receptor wakeup.]

If the last block wakeup transmissions take a long time (i.e., X second), master will have (5-X) seconds to transmit native mode packets before another CCA has to be started. This is fine if slave is not configured to scan during polling as the native mode timer is relative small (i.e., 500ms). If slave is configured to scan during polling, slave will need to scan on all the channels and therefore will have less chance hearing the native mode packets. By performing CCA after the wakeup transmissions, master will have more time to transmit native mode packets (i.e, Discover ID) and slave shall be able to hear the native mode packets.

TLMM_M_441 - WIRELESS WAKEUP - CCA AFTER WAKEUP

If instigator is configured to do “CCA before and after wakeup = 1” and “slave scan during polling =1” (see **Table 123 (Master parameters from host)**), the instigator shall perform a clear channel assessment (CCA) after the last block wakeup packets have been transmitted.

[The above requirement applies to both synchronous instigator wakeup and asynchronous receptor wakeup.But it does not apply to the network service]

TLMM_M_375 - WIRELESS WAKEUP -WAKEUP WITH CCA BREAKS

The instigator shall perform a clear channel assessment (CCA) every five seconds. This means that a wakeup burst duration longer than five seconds will be separated by a CCA. See **Fig. 119 (Wakeup with CCA breaks)**

[The above requirement applies to asynchronous receptor wakeup. It also applies to synchronous receptor wakeup, although synchronous receptor wakeup should rarely use a long wakeup burst (greater than 5 seconds). **Note** the data link layer will be notified, by the baseband hardware, when all of the wakeup packets have been sent. The data link layer will then take the next action of the discover service, open service, network service, medical event service, or channel recovery internal service (or more wakeup packets if necessary).]

.

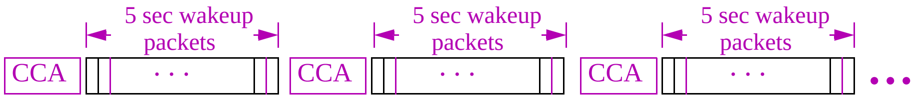


Figure 119: Wakeup with CCA breaks

5.3.6.7 SLAVE (RECEPTOR)

A node can configure itself to become a receptor by using the internal receptor wakeup service request. The node will then transition into a energy saving sleep state. If configured to synchronous sniff the wakeup baseband hardware will wakeup periodically to sniff for transmitted wakeup packets. If a valid wakeup packet is found, the baseband hardware will generate a interrupt to bring the node out of the sleep state.

5.3.6.8 REQUIREMENTS

The slave will sniff asynchronously in the following cases:

1. CHANNEL RECOVERY STATE if “slave scans during polling” is off and slave is in Unspecified Channel Recovery
2. ACQUIRE STATE if “slave scans during polling” is off and either one of the following is true:
    a) initiated by Patient Name Request or
    b) after native mode timeout

The slave will sniff synchronously in the following cases:

1. DISABLE STATE if either one of the following is true:
    a) In Disable CPU Off. or
    b) In Disable CPU On and a Security service duration or EBAN Key timer is active.

The slave will not sniff for wakeup packets in the following cases:

1. STANDBY STATE
2. COMM STATE
3. CHANNEL RECOVERY STATE if either one of the following is true:
   a) “slave scans during polling” is on or
   b) slave is in Same Channel Recovery.
4. ACQUIRE STATE if one of the following is true:
   a) “slave scans during polling” is on or
   b) the native mode timer has not expired.

TLMM_S_377 - WIRELESS WAKEUP - WAKEUP SNIFF FREQUENCY RANGE

The slave (receptor) shall be configurable to sniff channels in the following ways:

1. MICS only or
2. MEDS only or
3. Both MICS and MEDS

[Wakeup sniff list is defined in EEPROM, see wu_sniff_list in **A44088**, **Telemetry M RF Module EEPROM Specification** . The sniff list can be overwritten by slave host via initialization request, see **Table 124 (SLAVE PARAMETERS FROM HOST)**]

TLMM_S_378 - WIRELESS WAKEUP - SNIFF INTERVAL

Upon initialization the node will setup the sniff interval timer to periodically wakeup the baseband hardware to sniff for a wakeup packet.

[Wakeup sniff interval is provided by slave host via initialization request, see **Table 124 (SLAVE PARAMETERS FROM HOST)**]

TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP

If the wakeup disable bit in the initialization request is set to 1, the slave will not sniff for wakeup packets.

[The wakeup disable bit is provided by slave host via initialization request, see **Table 124 (SLAVE PARAMETERS FROM HOST)**.

If the wakeup disable bit is cleared, the slave hardware should sniff when the standby timer or the disable timer expires.

If wakeup disable bit is set, the slave hardware wakeup state machine is disabled and the module will not sniff for any wakeup packets.]

TLMM_S_379 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA RECEPTOR

The receptor will perform the following upon receipt of a wakeup packet containing secure user data (wakeup user data security bit set to enabled):

1. Use the received instigator’s secure wakeup counter (CTR_i contained in the wakeup packet).
2. Calculate the key stream using the block cipher. See **Fig. 117 (Calculate keystream to encrypt/ decrypt user data in a wakeup packet)**.
3. Decrypt the user data by performing an exclusive or of the user data with the key stream produced in bullet 3 above. See **Fig. 118 (Encrypt/ decrypt the wakeup data)**.
4. Calculate the AES_out using the block cipher using the instigator’s secure wakeup counter (CTR_i), instigator’s station ID (ID_i) and the four user data bytes, and the BAN key (K_{BAN}) for the key input. See **Fig. 116 (Calculate AES_out for wakeup security)**.
5. If the calculated MAC (AES_out[23:0]) is equal to the received user data (bytes 8, 9, and 10 of the user data),

   the user data is valid and wakeup indication will be sent to the host.(see **Table 93 (WAKEUP INDICATION (SLAVE))**);
   else,
   the user data is invalid and and wakeup indication with invalid MAC will be sent to the host.(see **Table 93 (WAKEUP INDICATION (SLAVE))**);

TLMM_S_380 - WIRELESS WAKEUP - WAKEUP COMPLETE INDICATION

A wakeup indication “Wakeup Complete”, containing the wakeup packet contents, shall be sent to the upper layer when the “last packet last block” wakeup packet is received (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[Note, user data may be present in the “Wakeup Complete” wakeup packet contents. Upon receiving a “Wakeup Complete”, the host will normally issue a initialize service request

Also note, an implicit wakeup indication (the same indication as described above) will be sent to the upper layer under the following conditions:

a. A wakeup packet (non last block) is heard, and

b. No subsequent wakeup packet (last block) is heard when performing asynchronous sniff.
This asynchronous sniff is designed to occur during the last block. The subsequent wakeup packet (last block) is less likely to be heard if the last wakeup block is short.]

TLMM_S_381 - WIRELESS WAKEUP - WAKEUP IN PROGRESS USER DATA AVAILABLE INDICATION

When valid user data is present, in the wakeup packets, a wakeup indication “Wakeup In Progress User Data Available” containing the wakeup packet contents shall be sent to the upper layer upon receipt of the first valid wakeup packet containing user data (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[When invalid user data is present, in the first valid wakeup packet, no wakeup indication shall be sent to the upper layer.]

TLMM_S_382 - WIRELESS WAKEUP - WAKEUP COMPLETE INDICATION ONLY

If the first valid wakeup packet, containing valid user data, is also the "last packet last block" wakeup packet; then only the "Wakeup Complete" indication shall be sent to the host.

If the first valid wakeup packet, containing invalid user data, is also the "last packet last block" wakeup packet; then only the "Wakeup Complete with Invalid user data" indication shall be sent to the host.

(see **Table 93 (WAKEUP INDICATION (SLAVE))**)

[Note that if the first wakeup packet is also the last wakeup packet and it has user data, two wakeup indications will be sent to the host. The first will be sent before reading EEPROM. The second will be sent after reading EEPROM. This is done in case the EEPROM read is delayed due to power inhibit. An initialization request while reading EEPROM will be rejected (if initialization sent after the first wakeup indication). Initialization after the second wakeup indication should be successful.]

When host receives the "Wakeup Complete with Invalid User Data" indication, host should send initialization request with valid Ban Key and Network Table to Mozart and then Mozart will update its Ban Key and Network Table accordingly. Then Mozart will be ready for the any other wakeup packet with user data once sniffing again.

TLMM_S_383 - WIRELESS WAKEUP - INDICATIONS

The received wakeup packet shall be assembled into an indication message and passed onto the host. The indication message shall have a format defined in **Table 93 (WAKEUP INDICATION (SLAVE))**.

TABLE 93: WAKEUP INDICATION (SLAVE)

FIELD	FIELD INFO	DESCRIPTION	VALUES
Type	NA	The type of indication	Wakeup indication = 0x2B
Sub-type	The type of packet received	The indication sub-type	0x01 = Wakeup Complete 0x02= Wakeup In Progress User Data Available 0x03 = Wakeup Complete with Invalid User Data
Size	The size of the wakeup packet	The size of the info field	0x16
Info	The wakeup packet data	Contains the bytes received in the latest wakeup packet	See wakeup packet format defined in **TLMM_MS_365 - WIRELESS WAKEUP - WAKEUP PACKET**

5.3.7 ANTENNA SELECTION

To improve reliability of data transfer in a fading signal environment, spatial diversity is implemented by the master. Spatial diversity is implemented with two antennas at the master, and with an antenna selection algorithm that selects the active antenna. If the signal level received by each antenna is independent (or nearly so), then when the signal level from the current antenna is inadequate to sustain reliable communications the second antenna will often have an adequate signal level. The antenna selection algorithm switches to the second antenna when packet errors occur during transmit or receive operations. Implementation details are specified in the paragraphs that follow.

The master will toggle the transmit antenna (vs. the previous transmit antenna) when the master detects that the slave is not receiving its packets correctly. The slave reports this by using the ACK bit (see control bytes). An exception occurs during the low energy mode. The slave is not expected to receive the master packet successfully when the master packet is sent using negative correlation preamble.

Toggle the transmit antenna (vs. the previous transmit antenna) if the first (N-1) uplink packets are received with MAC error.

TLMM_M_386 - ANTENNA SELECTION - TRANSMIT

The master will toggle the transmit antenna (vs. the previous transmit antenna) if any of the following conditions are met:

1. The previous master packet was received with a MAC error by the slave (provided the master packet was not sent with negative correlation preamble). The master knows the slave received its packet in error if the slave's ACK bit is false. Note that only the first N-1 packets are checked. In the event that N is equal to one, the single uplink packet is checked.
2. Each of the first (N-1) uplink packets are received with MAC error (or the single uplink packet has a MAC error in the event that N is equal to one.

[“N” can be either of the “slave xmit packet count for nominal mode” or “slave xmit packet count for max throughput mode” as determined by the preceding master packet’s transfer mode. See **Table 123 (Master parameters from host)** for the “packet count” parameters.]

5.3.8 SIGNAL STRENGTH INDICATOR

The signal strength indication can be read by the host at any time, this value is updated once per uplink frame. The frequency of update is approximately once per 10ms (uplink and downlink data rates both ~ 380Kbps). The signal strength indicator is located in dual port memory. The general purpose telemetry write 0 register (reg_hst.gp_tm_w0). This register is write only for Mozart and read only for the host, there are not read or write contention issue.

The signal strength indicator is displayed on the top left corner of the screen of the programmer.

The user desires an indication that communication will be reliable. A signal strength indicator is one way to accomplish this. One way to implement a signal strength indicator is to quantize the average signal.

TABLE 94: SIGNAL STRENGTH INDICATOR

Telemetry state	average signal (dB)	signal strength indicator
Communication	K6$^{(1)}$ <= aveSignal	7
Communication	K5$^{(1)}$ <= aveSignal < K6	6

TABLE 94: SIGNAL STRENGTH INDICATOR

Telemetry state	average signal (dB)	signal strength indicator
Communication	K4(1) <= aveSignal < K5	5
Communication	K3(1) <= aveSignal < K4	4
Communication	K2(1) <= aveSignal < K3	3
Communication	K1(1) <= aveSignal < K2	2
Communication	aveSignal < K1	1
not Communication(2)	-	0

1. See **Table 123 (Master parameters from host)**.
2. This includes standby, acquire, and channel recovery.

TLMC_M_156 - SIGNAL - STRENGTH INDICATOR

The master shall set the signal strength indicator to give feedback to the user about the link quality.

[signal strength indicator = min(SSI’, last SSI-1) if SSI’ < last SSI
= max(SSI’, last SSI+1) if SSI’ >= last SSI

where,
SSI’ = min(SSI’’,signal strength input limit) if no waveform packet in the slave frame
= SSI’’ otherwise
and
SSI’’ is determined from **Table 94 (SIGNAL STRENGTH INDICATOR)**

Note that “signal strength input limit” is found in **Table 123 (Master parameters from host)**. A value of “7” will disable the input limiting. Hosts without periodic waveform (or infrequent waveform relative to the “superframe interval”) should consider disabling input limiting. Hosts with frequent waveform (i.e., telemetry C Concerto) may set this value to “3”.]

The hardware measures the receive signal strength. It measures the signal during the first slave packet (after the preamble). Refer to **Fig. 120 (Signal to noise measurement system)**.

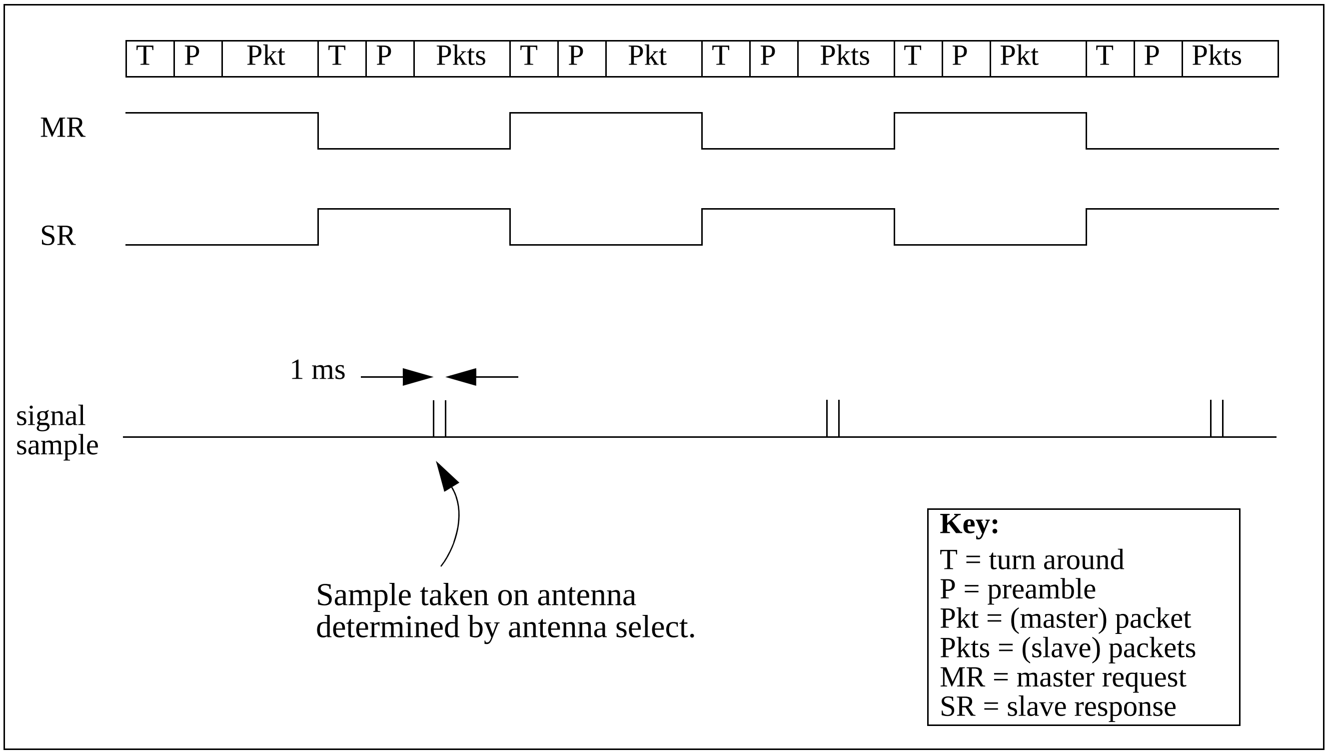


Figure 120: Signal to noise measurement system

The Data Link Layer will read the signal after the first slave packet is complete. It will not read the signal on any subsequent packets. These packets may be unused (analog hardware OFF to save energy). The slave will always utilize the first packet of the slave frame.

TLMM_M_387 - SIGNAL STRENGTH INDICATOR SETUP

The master will setup the hardware to take periodic signal measurements:

1. Enable power bar measurements
2. Set CCA dwell time to a value smaller than the duration of a slave packet (so the “signal” contains no portion of an inactive packet)

5.3.9 TUNE AND TRIM

The Mozart hardware can notify the data link layer of a phase lock loop (PLL) error. This error can occur if the Mozart module is not within the operating temperature range (it is too cold or too hot).

When a PLL lock error occurs, the data link layer should initiate a coarse trim and a fine trim on all 30 channels of the system, the data link layer initiated trim can either succeed or fail.

Upon trim success:

- The master will initiate channel assessment, transmit wakeup packets and send discover or open packets. Note that channel assessment is skipped if trim success occurs during communication (same channel recovery initiated). Also note that the wakeup packets are suppressed if same channel recovery is initiated, or if the master knows the slave is scanning.

- The slave will resume listening (for native mode packets or wakeup packets).

Upon trim failure:

- The master will go to standby and notify the host that downlinks to the slave have ceased.
- The slave will go to standby, notify the host that listening for master downlink has (momentarily) ceased. If the host does not issue a subsequent request, the slave will resume synchronous receptor wakeup.

5.3.9.1 REQUIREMENTS:

TLMM_MS_391 - PLL LOCK ERROR

Upon receiving the PLL lock error data link layer will:

1. Perform idle immediate and turn off wakeup,
2. Start the trim process by initiating a coarse trim and then a fine trim on all 30 channels.

[The trim process will stop if a trim failure occurs.]

TLMM_M_392 - MASTER TRIM SUCCESS

The master will perform the following if the data link layer initiated trim is successful:

  a. Perform clear channel assessment (see note below)
  b. Transmit wakeup (see note below)
  c. Transmit discover or open (discover if the discover service was interrupted by the PLL lock error, open otherwise)

Note that channel assessment is skipped if trim success occurs during communication (same channel recovery initiated). Also note that the wakeup packets are suppressed if same channel recovery is initiated, or if the master knows the slave is scanning (see **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**).

[The remote diagnostic service and the mapping diagnostic service is cancelled if interrupted by a PLL lock error (and a subsequent trim success). The service response status will be 0x26.]

TLMM_S_393 - SLAVE TRIM SUCCESS

The slave will perform the following if the data link layer initiated trim is successful:

Listen for Tel M native mode, listen for TelC packets, listen for asynchronous wakeup, or reconfigure for synchronous sniff (whichever was in progress when PLL lock error occurred).

[That is slave will resume the actions (if possible) before the PLL lock error is occurred:

1)Turn 2Volt off if slave is enabled by PLL lock error, or

2)Perform synchronous sniff if slave is sniffing during Disable CPU ON state when PLL lock error is occurred, or

3)Start same channel recovery if PLL lock error is occurred during communication state,or

4)Restart to listen for native mode in the same channel recovery if PLL lock error is occurred during same channel recovery state,or

5)Restart to listen (for native mode or asynchronous sniffing) in the unspecified channel recovery if PLL lock error is occurred during unspecified channel recovery state,or

6)Restart to listen for native mode or asynchronous sniffing in the acquire state if PLL lock error is occurred while polling in acquire state,or

7)Restart to uplink medical event if PLL lock error is occurred during uplinking medical events, or

8)Restart to listen for network packet if PLL lock error is occurred during network service.]

TLMM_M_394 - MASTER TRIM FAIL

The master will perform the following if the data link layer initiated trim fails:

a. Transition to the standby state (stop downlinking to the slave)
b. Send an external service response (failure) to the host if a service was in progress (see the requirement note below)
c. Send a telemetry ceased indication to the host

[Note that a discover cancelled indication is used for a trim failure during discover since a successful service response is sent when the service is initiated. All other service requests utilize a cancelled service response (failure code 0x1d).]

TLMM_S_395 - SLAVE TRIM FAIL

The slave will perform the following if the data link layer initiated trim fails:

a. Transition to the standby state if not in the disable state (momentarily stop listening to the master)
b. Send a telemetry ceased indication to the host if not in the disable state
c. Start the standby timer if not in the disable state. Note the disable timer is utilized in the disable state.

[Note that if the standby timer expires (or the disable timer expires) the slave will return to synchronous receptor wakeup operation. If the wakeup disable bit is set to 1 in the initialization request firmware does not set the wakeup enable bit bp_wakeup_en bit in REG_BP_WU_CTRL0 register. See requirement **TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP**]

TLMM_MS_396 - TELEMETRY CEASED INDICATION

The data link layer will send telemetry ceased indication to the host if the following conditions are true:

a. A PLL lock error occurs, and
b. The coarse trim or the fine trim procedure fails, and
c. The state is not disable

[This indication could be generated when PLL lock error is encountered during either the coarse or fine trim when not in the disable state. This could happen during the acquire, communication and channel recovery states.]

TABLE 95: TELEMETRY CEASED INDICATION (MASTER/SLAVE)

FIELD	FIELD INFO	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x35 - Telemetry ceased
Sub-type	N/A	1	The indication sub-type	n/a
Size	N/A	1	The size of the info field	1
Info	Telemetry Ceased indication status	1	Status of indication	0x01 - fine trim failed 0x02 - coarse trim failed
CRC	N/A	2	CRC of all the above field	This value is computed by data link layer and is used by upper layer to verify the integrity of the indication

TLMM_MS_397 - TRIM FAILURE INDICATION

The data link layer will send a trim failed indication to the host if the following conditions are true:

a. A PLL lock error occurs, and
b. The coarse trim or the fine trim procedure fails, and
c. The "trim failed indication to host" flag is set

[Trim failed indication flag is a configuration parameter - it can be set or cleared by the host]

TABLE 96: TRIM FAILURE INDICATION (MASTER/SLAVE)

FIELD	FIELD INFO	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x34 - Trim failure
Sub-type	N/A	1	The indication sub-type	n/a

TABLE 96: TRIM FAILURE INDICATION (MASTER/SLAVE)

Size	N/A	1	The size of the info field	3
Info	Trim failure indication status	1	Status of indication	0x0D - coarse trim failed 0x0E - fine trim failed
	Power inhibit status	1	Power inhibit status at the time of the trim failure	0 = power inhibit not asserted during trim 1 = power inhibit asserted during trim
	Firmware state	1	State of the firmware at the time of the trim failure	000 -0xff
CRC	N/A	2	CRC of all the above field	This value is computed by data link layer and is used by upper layer to verify the integrity of the indication

5.3.10 MEMORY TEST AND PARAMETER CORRUPTION

5.3.10.1 PARAMETER CORRUPTION

Soft Error can occur due to the upset of a bit due to ionizing radiation: thermal neutrons, energetic neutrons, and alpha particles. The 3V power domain (a.k.a., the "always on") is always powered up. There are two flavors of 3V registers: normal and radiation hardened (RH). Only RH 3V registers are essentially immune to ionizing radiation at the energy levels of concern.

As there is no parity check for always on registers from hardware, the following Parameters (in always on non-RH register area) shall be checked by firmware:

- those needed to perform WU sniffing or needed to boot the Tel M RFM CPU,
- those which are not normally refreshed from EEPROM and are not able to "heal" themselves over time.

There are certain parameters which are passed by the host to the Mozart module as part of the initialization service. Examples are BAN Key, Wakeup Falsing Enable and Wrap up interval, etc.

The BAN Key is used as part of the Security feature.

Wakeup falsing enable controls are used to optimize the wakeup algorithm for the hardware by lengthening or shortening the sniff and wrap up intervals, etc.

The wrap up interval is used to configure the refresh rate of non-volatile parameters and to accumulate the statistical information for the hardware to optimize the radio sensitivity for wakeup algorithm.

In the event that non-volatile parameters which do not have an EEPROM backup are corrupted (i.e., wakeup falsing enable), the host will be notified of parameter corruption so the parameters can be re-initialized.

5.3.10.2 REQUIREMENTS

TLMM_S_398 - PARAMETER CRC UPDATE

The CRC of all the Parameters in **Table 97 (PARAMETER CRC LIST)** will be calculated and stored

TABLE 97: PARAMETER CRC LIST

Address	#bytes	Parameter Name	Parameters could be Changed
0X040A-0X0429	32	Sniff list	when slave processes an Initialize Service Request, or when Slave is enabled by Wrap-up or High Current.
0X042B-0X0430	6	Coordinator ID	when slave processes an Initialize Service Request, or a Configure Network Request.
0X045C-0X045F	4	Noise floor estimation enables	when slave is booted
0X05B2-0X05CB	26	network table and its CRC	when slave processes an Initialize Service Request, or a Configure Network Request.
0X05CC - 0X05CE	3	Sniff Interval Original	when slave processes an Initialize Service Request.
0X05CF-0X5D0	2	Interval Wrap up Count Normal	
0X05D1	1	Wakeup Falsing Enable Bits	when Slave is booted, or when slave processes an Initialize Service Request.
0X05D2	1	Wakeup Falsing Cycle Bits	when slave is enabled by Wrap-up or High Current.
0X05D3	1	Restricted Channel Removed Actual	
0X05D4-0X05D9	6	Restricted Channels	
0X05DA-0X05E9	16	Ban Key	when slave processes an Initialize Service Request, or a Create Key or an Assert Key Security Request.
0X05EA	1	Security Flags	

[The CRC is stored at always on RH area 0x05A1-0x05A2.

“Slave Checks Always On CRC” parameter is located at always on RH area, 0x0005A0 (bit0).

The CRC is updated only when “Slave Checks always on CRC” parameter is set to TRUE.

“Slave Checks Always On CRC” parameter is cleared when slave is booted via reset by Hardware; “Slave Checks Always On CRC”is set to the same value as “slave shall calculate and check Always On parameters” (see **Table 124 (SLAVE PARAMETERS FROM HOST)** when slave processes an Initialize Service Request.]

TLMM_S_399 - PARAMETER CRC CHECK

The CRC of all the Parameters in **Table 97 (PARAMETER CRC LIST)** will be calculated and checked against to the stored CRC under the following cases:

1. On every boot (except any Resets), or

2. Before turn 2V off.

[Note the CRC will also be calculated and checked against the stored CRC before any of the parameters are changed.

The CRC is checked only when “Slave Checks always on CRC” parameter is set to TRUE.

Note the following:

1. The wrap up interrupt will not be processed until the disable timer expires if the interrupt occurs while the CPU is enabled. This negates any potential latency associated with this interrupt.

2. The wrap up interval is typically equal to the sniff interval times the wrap up count. However, the first wrap up interrupt will be one sniff interval shorter than all the remaining interrupts. For example if the sniff interval is two seconds and the wrap up count is five, the first wrap up interrupt will occur 8 seconds later. The remainder will occur every 10 seconds.

3. After initialization, the sniff interval is re-loaded but the wrap up count is not. The wrap up count is reloaded once the old one is complete.]

TLMM_S_400- PARAMETER CORRUPTION INDICATION

If the two CRCs compared in the PARAMETER CRC CHECK are not identical, a Parameter corruption indication will be sent to the Host.

TABLE 98: PARAMETER CORRUPTION INDICATION (SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x37 - Parameter Corruption Indication
Sub-type	N/A	1	The indication sub-type	0
Size	N/A	2	The size of the info field	1
Info	Parameter Indication status	1	Status of the indication	0x01 - successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

5.3.10.3 MEMORY TEST

Mozart firmware performs frequent memory tests to insure that memory integrity remains intact. This is accomplished by doing a CRC check on each structure once they have been copied into the RAM. These structures are defined in the **Table 122 (STRUCTURES IN EEPROM)**. If host receives the memory test failed indication this indicates that Mozart module’s RAM has been corrupted and its operation could be in question. It will be up to the host to take the appropriate action (i.e. reset the Mozart module or explant the device).

5.3.10.4 REQUIREMENTS

TLMM_MS_415 - MEMORY TEST - RAM STRUCTURES

A memory test will be conducted once every second on a RAM structure. If any of the CRC check fails then memory test indication will be generated to the host. If all the structures pass the CRC check firmware will restart the process.

[Note: CRC values are stored at a fixed location of 0x38C0. Every second firmware will check the CRC for one of the structures in the RAM in the following order: Ramware code area 1, area2, … area43, ramware enable and jump table, hardware1, hardware2, wakeup and firmware operation parameter structure**.** If any of the CRC check failed, the memory test failed indication will be sent to the host. If the check passed, the crc of the next structure will be checked on the next second. After the last parameter structure CRC is checked and passed, firmware will start from the first structure (i.e., ramware code area 1) CRC check again on the following second. If there is no ramware it will be skipped.]

TABLE 99: MEMORY TEST FAILED INDICATION (MASTER OR SLAVE)

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	The type of indication	0x38 - Ramware failed indication
Sub-type	N/A	1	The indication sub-type	0
Size	N/A	1	The size of the info field	1
Info	Parameter Indication status	1	ID of failed memory structure	See the requirement note
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the indication.

6.0 COMPLETION

This paragraph concludes this specification. This paragraph is followed by a set of appendices, including **Requirements Sorted by Tag**, **List Of Tables**, and others.

7.0 APPENDICES

This section contains supporting information for the Telemetry M protocol. This information consists of: Telemetry M Data Throughput Analysis; followed by miscellaneous sections that contain Diagnostics data information (local and implant), and mapping data information.

7.1 TELEMETRY M DATA THROUGHPUT ANALYSIS

This section provides data throughput analysis.

This section is divided into two main sections:

- Physical layer data rate requirements and other considerations
- A data throughput analysis for both master and slave messages

7.1.1 PHYSICAL LAYER REQUIREMENTS

Physical layer data rate requirements for Telemetry B, C and M are shown in Table 1:

TABLE 100: DATA RATE

	Telemetry B	Telemetry C	Telemetry M
Master data rate	**12.5 Kbps**	**14.844 Kbps**	**48, 97, 190, 380 Kbps**
Slave data rate	**87.5 Kbps**	**100 Kbps**	**48, 97, 190, 380 Kbps**

7.1.2 SYSTEM CONSIDERATIONS

In order to achieve the maximum data throughput with minimum complexity to the communication protocol the following items were considered:

- Typically the frequency of data transmitted in the slave direction is often and consists of Real time (Waveform packets), Unrequested (Supplemental Markers), and User requested (diagnostic/memory data). The overall data transmitted by the slave is more than the master.
- Typically the frequency of the master data transmission is low and the amount of data transmitted is less than the slave.
- A mechanism is needed to insure data integrity during a communication session between one master and one slave.
- Once the user has initiated the master request it should be transmitted within 2-3 seconds.
- At a minimum maintain the data throughput requirements or exceed the data through requirements of the existing telemetry systems.
- Keep the master and slave packet sizes fixed.

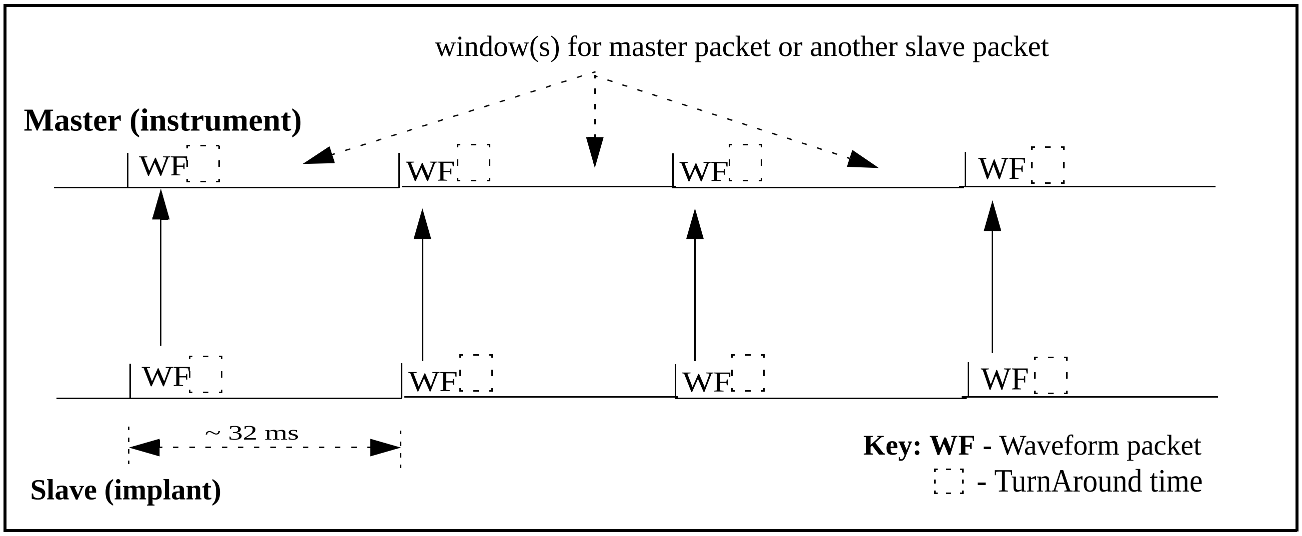
7.1.3 SUMMARY OF TELEMETRY B SYSTEM THROUGHPUT

In the telemetry B system, the slave (implant) transmits waveform packets at approximately every 32ms interval. After each waveform packet, the slave either receives a packet from the master or transmits one more slave packet. Once the waveform packet is transmitted, the slave listens for a master packet. If a master packet is detected then no slave data is transmitted. If no master packet is detected, the slave packet is transmitted. The timing of this sequence is shown below:

Master throughput is:

- A master frame is transmitted every 64ms
- Assuming the number of bytes per master frame is 29 bytes - excluding the (header, programmer id, and frame CRC)

The telemetry B master throughput *BmasterThroughput* is:



$$BmasterThroughput = \frac{NumberOfMasterBytesPerFrame}{64ms} = \frac{29bytes}{64ms} = 453((bytes)/s)$$

Diagnostic/Memory data slave throughput is:

- A slave frame can be transmitted at approximately every 32ms - a master acknowledgement is expected to be transmitted on the 8th frame.
- Assuming the number of bytes per slave frame is 230 bytes - excluding the (header, command code, implant status, and frame CRC)

The telemetry B slave throughput *BslaveThroughput* is:

$$BslaveThroughput = \frac{NumberOfSlaveBytesPerFrame}{32ms} = \frac{230bytes \times 7}{32ms \times 8} = 6289((bytes)/s)$$

7.1.4 SUMMARY OF TELEMETRY C SYSTEM THROUGHPUT

The data flow between the master and slave is as follows:

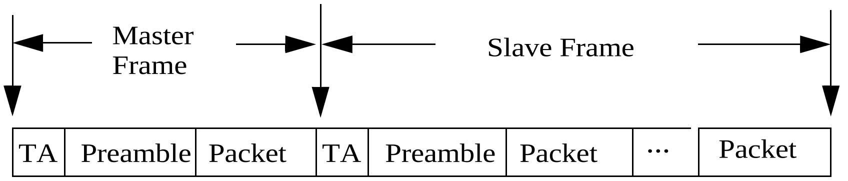


Figure 121: Telemetry C Frame

TA: Turnaround time
Note: No time scale

Master throughput is:

- A master frame is transmitted every 35.7ms (nominal mode)
- Assuming the number of bytes per master frame is 16 bytes - excluding the (header, and CRC)

Then, the highest telemetry C master throughput *CmasterThroughput* is:

$$\boxed{CmasterThroughput = \frac{NumberOfMasterBytesPerFrame}{35.7ms} = \frac{16bytes}{35.7ms} = 392((bytes)/s)}$$

Diagnostic/Memory data slave throughput is:

- A slave frame is transmitted every 115.6 ms (max throughput mode)
- Assuming the number of bytes per slave frame is 1024 bytes (32 packets * 32 bytes/ packet). This excludes the header and CRC.

The telemetry C slave throughput *CslaveThroughput* is:

$$\boxed{CslaveThroughput = \frac{NumberOfSlaveBytesPerFrame}{115.6ms} = \frac{1024bytes}{115.6ms} = 8858((bytes)/s)}$$

7.1.5 SUMMARY OF TELEMETRY M SYSTEM THROUGHPUT

The data flow between the master and slave is based as follows:

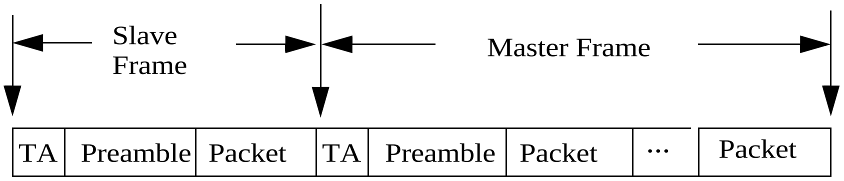


Figure 122: Telemetry M Frame

TA: Turnaround time
Note: No time scale

Assume that the master packet size (*Sm*) is controlled by the Reed Solomon code. A code of [63,51] results in a packet 63*6 bits long (378 bits = 47.25 bytes). Of the 378 total bits only 240 bits can be used for the packet content field (30 bytes). The rest is overhead for Reed Solomon, message authentication code and control bytes. Assume the master packet count is *Cm*.

Assume that the slave packet size (*Ss*) is controlled by the Reed Solomon code. A code of [63,51] results in a packet 63*6 bits long (378 bits = 47.25 bytes). Of the 378 total bits only 240 bits can be used for the packet content field (30 bytes). The rest is overhead for Reed Solomon, message authentication code and control bytes. Assume the slave packet count is *Cs*.

Finally, assume that 40 bits are used for the preamble.

7.1.5.1 MASTER FRAME TIMING

The master frame time *Tm* is:

$$\boxed{Tm = TurnaroudTime + PreambleTime + masterPktTime}$$

If master transmit data rate is Rm and only one master packet (Cm = 1) will be transmitted, then

$$\boxed{masterPktTime = \frac{378 \cdot Cmbits}{Rm((Kbits)/s)} = \frac{378bits}{Rm((Kbits)/s)}}$$

7.1.5.2 SLAVE FRAME TIMING

The slave frame time *Ts* is:

$$\boxed{Ts = TurnaroudTime + PreambleTime + slavePktTime}$$

If slave transmit data rate is Rs, then

$$\boxed{slavePktTime = \frac{(378 \bullet Cs)bits}{Rs((Kbits)/s)}}$$

Total Timing for master and slave frames

Then total time *Tt* for each master and slave frame is:

$$\boxed{Tt = Tm + Ts = 2(TurnaroudTime + Preamble) + \frac{378bits}{Rm((Kbits)/s)} + \frac{(378 \bullet Cs)bits}{Rs((Kbits)/s)}}$$

The TurnaroundTime and PreambleTime are different for different data rate. The TurnaroundTime depends on the turnaround count registers' value. See **TABLE 101: Telemetry M TurnAroundTime and Preamble Time** for default turnaround count which is selected by Mozart Programmer's Manual.

TABLE 101: TELEMETRY M TURNAROUNDTIME AND PREAMBLE TIME

(R_M, R_S) DATA RATE K_{bps}	TURNAROUND COUNT (default)	Turnaround Time (ms)	PREAMBLETIME (ms)
(48,48)	124	3.55	1.950
(97,97)	248	3.55	1.900
(190,190)	221	1.58	0.500
(380,380)	442	1.58	0.500

7.1.5.3 MASTER THROUGHPUT

The master throughput is:.

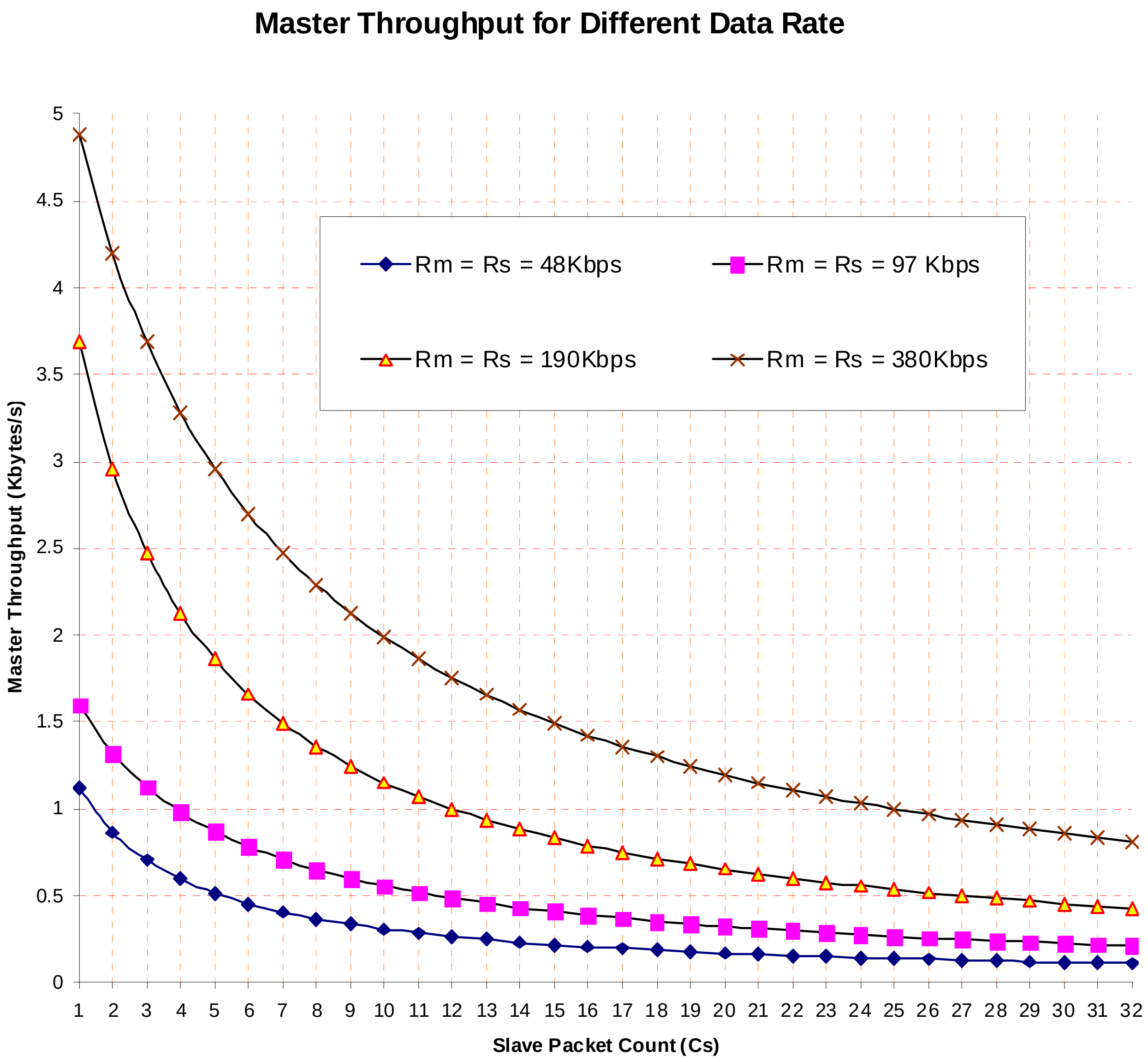
$$MasterThroughput = \frac{NnumberOfMasterContentBytesPerFrame}{Tt} = \frac{30}{Tt}((bytes)/(ms))$$

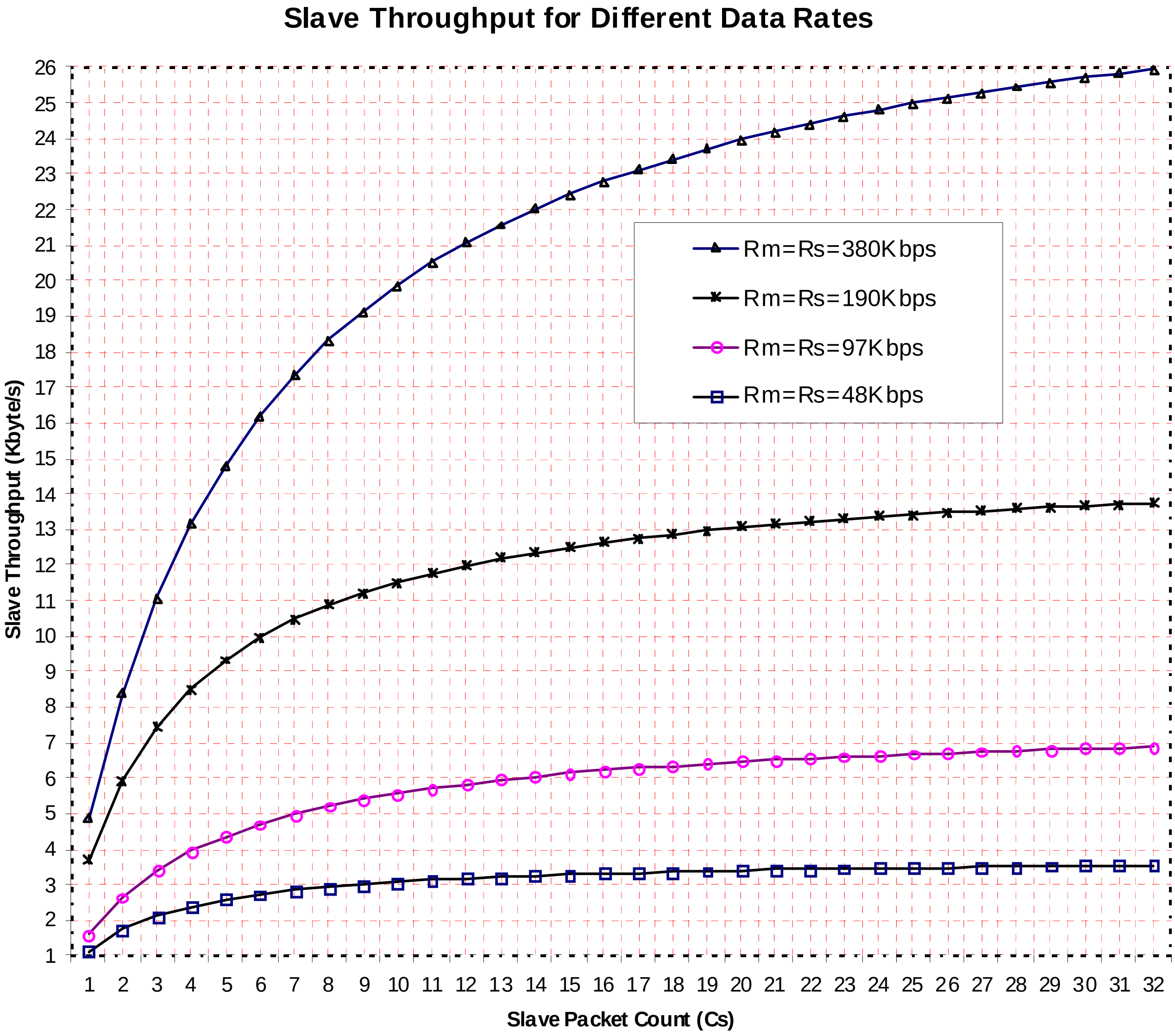
7.1.5.4 SLAVE THROUGHPUT

The slave throughput is:

$$SlaveThroughput = \frac{NumberOfSlaveContentBytesPerFrame}{Tt} = \frac{(30 \bullet Cs)}{Tt}((bytes)/(ms))$$

See the two figures below for master and slave throughput vs. different data rate and slave packet count Cs.


Master Throughput for Different Data Rate
Rm = Rs = 48Kbps
Rm = Rs = 97 Kbps
Rm = Rs = 190Kbps
Rm = Rs = 380Kbps
Master Throughput (Kbytes/s)
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Slave Packet Count (Cs)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32


Slave Throughput for Different Data Rates
Slave Throughput (Kbyte/s)
Slave Packet Count (Cs)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
Rm=Rs=380Kbps
Rm=Rs=190Kbps
Rm=Rs=97Kbps
Rm=Rs=48Kbps

7.1.5.5 THROUGHPUT SUMMARY

The table (below) summarizes the throughputs for each telemetry type:

TABLE 102: THROUGHPUT SUMMARY

TELEMETRY TYPE	DATA RATE (Rm, Rs)Kbps	MASTER THROUGHPUT (Kbytes/sec)	SLAVE THROUGHPUT(1) (Kbytes/sec)
B	(12.5, 87.5)	0.4	6.3
C	(14.8,100)	0.4	8.9
M	(48,48)	1.1(2)	3.5
	(97,97)	1.3(3)	6.9
	(190,190)	1.9(4)	13.8
	(380,380)	2.1(5)	26

1. Assumes the frame has 1 downlink packet and 32 uplink packets.
2. Assumes the frame has 1 downlink packet and 1 uplink packet.
3. Assumes frame has 1 downlink packet and 3 uplink packet.
4. Assumes frame has 1 downlink packet and 5 uplink packet.
5. Assumes frame has 1 downlink packet and 9 uplink packet.

The Telemetry M master throughput is for recommended nominal mode with only low energy mode enabled, and slave throughput is for recommended maximum slave transmit mode. See **TABLE 103: Telemetry M Recommend Slave Packet Count**. The recommended slave packet count (Cs) for nominal mode takes the low energy mode into count. This is because the maximum slave packet count (Cs) in the low energy mode is the same as the slave packet count (Cs) in the nominal mode and the active slave packet count (i.e., the actually transmitted slave packets) in the low energy mode can vary from 1 to the maximum slave packet count (Cs). In case there is only 1 slave packet is transmitted, the slave transmitter off time should be no more than 10ms in order to keep the channel from chosen by others.

TABLE 103: TELEMETRY M RECOMMEND SLAVE PACKET COUNT

(R_M, R_S) DATA RATE K_{BPS}	NOMINAL MODE			MAX. SLAVE TRANSMIT MODE
	ONLY LOW ENERGY MODE ENABLED	POWER INHIBIT MODE ENABLED	BOTH LOW ENERGY AND POWER INHIBIT MODE DISABLED	
(48,48)	1	NOT RECOMMENDED	CAN CHOOSE ANY VALUE FROM 1 TO 32	32
(97,97)	2			32
(190,190)	5	3		32
(380,380)	9	6		32

If power inhibit mode (1 to 3) is enabled during the nominal mode for slave, then the recommended slave packet count (Cs) for nominal mode will be even smaller. This is because the slave transmitter is off totally during power inhibit. For the data rate 48Kbps and 98Kbps both directions, even though slave packet count of 1 will exceed 10ms, therefore power inhibit mode is not recommended.

If neither low energy mode nor power inhibit mode is enabled, then the slave packet count for nominal mode can be any value from1 to 32. The consideration should be including the master throughput and the slave throughput as well as the amount of real time data to be transmitted etc.

7.2 MISCELLANEOUS

7.2.1 DIAGNOSTIC LOCAL DATA

TABLE 104: DIAGNOSTIC SERVICE RESPONSE — LOCAL DATA

BYTE OFFSET	DETAILED DESCRIPTION	HIGH LEVEL DESCRIPTION
1	Channel 1, Antenna 1 mapping value	Most recent mapping information(1)
2	Channel 1,Antenna 2 mapping value	
.	.	
.	.	
20	Channel 10, antenna 2 mapping value	
21	Telem state	Current master data link information
22	channel	
23	Channel recovery reason (see **TABLE 105: Master channel recovery reason**)	
24	Low energy mode (0 = off, non-zero = on)	
25-28	Master packets	
29-32	Receive packet timeouts	
33-36	Slave packets w/ good CRC	
37-40	Slave packets w/ CRC error	
41-44	Slave preamble timeouts	
45-48	# same channel rec init.	
49-52	# unspecified channel rec init.	
53-56	# channel rec success	
57-60	# channel rec failure	
61	Mozart FW version #	
62	Mozart FW version #	
63	Mozart FW version #	
64	Mozart FW version #	
65-68	Recovery to hibernate transitions	
69-72	Master bit error rate packets	
73-76	Slave bit error rate packet opportunities	
77-80	Slave bit error rate packets received	
81-84	Slave non bit error rate packets received	
85-88	packet CRC errors (after transmitting BER packet)	
89	PLL Lock Error count(2)	
90	Coarse Trim Failure count	
91	FineTrim Failure count.	
92	signal (instantaneous signal strength measurement)	
93-96	Slave bit errors	
97-100	Missed telemetry M Rx interrupts(3)	
101	Diagnostic Status Byte	
102-N	Slave bytes (diagnostic uplink contents)	

1. These values come from the most recent channel selection mapping or diagnostic mapping. Note that channel selection mapping is initiated by discover and open.

2. Number of trim successes = lock error count - coarse trim failure count - fine trim failure count
3. A missed telemetry M receive interrupt could occur if the CPU speed were too slow to support the rate of the receive interrupts.

TABLE 105: MASTER CHANNEL RECOVERY REASON

BIT	DESCRIPTION
0	Receive packet timeout occurred (digital mode only)
1	Received a packet from the slave with a mismatching ID (vs. ID at open)
2	Received a packet from the slave with inconsistency between the ACK bit and the reset ACK bit
3	N1 consecutive frames have occurred without receipt of a slave packet with a valid MAC
4	N2 consecutive frames have occurred without the receipt of a slave packet with a valid CRC nor valid preamble (telemetry C only)
5	Trim successful.
6	Bits 0-5 detail the explicit recovery reason. When this bit is set it means one of the above reasons occurred and additionally one or more Rx interrupts was missed (a receive interrupt to receive interrupt deadline was missed). This could occur if the host sets the CPU speed too low for the native mode data rate. This bit is not an explicit recovery reason.
7	Unused

7.2.2 DIAGNOSTIC IMPLANT DATA

TABLE 106: DIAGNOSTIC SERVICE RESPONSE — IMPLANT DATA

ADDRESS	DESCRIPTION	SIZE
0x0F00$^{(1)}$	Type from Host side(bit 7 bit 6 bit 5 bit 4 bit 3 bit2 bit1 bit0)$^{(2)}$	1byte
0x0F01	CPU fast speed	1byte
0x0F02	CPU slow speed	1byte
0x0F03	Received Data Rate Out of Session (Apply only to Telemetry M)	1byte
0x0F04	Transmit Data Rate Out of Session (Apply only to Telemetry M)	1byte
0x0F05- 0x0F08	Mozart Firmware Version Number	4bytes
0x0F09	Current Communication Channel Number	1byte
0x0F0A-0x0F0B	Channel Recovery Reason$^{(3)}$	2bytes
0x0F0C	Fault 1Flags(bit 7 bit 6 bit 5 bit 4 bit 3 bit2 bit1 bit0) $^{(4)}$	1byte
0x0F0D	Fault 2 Flags (bit 7 bit 6 bit 5 bit 4 bit 3 bit2 bit1 bit0)$^{(5)}$	1byte
0x0F0E	Max. Retransmission due to Good&Bad Downlink Counter	1byte
0x0F0F	Network Unexpected Event Counter	1bytes
0x0F10- 0x0F13	Total Received Master Packet Counter (Communication State)	4bytes
0x0F14- 0x0F17	Received Master Packet Bad Preamble Counter (Communication State)	4bytes
0x0F18- 0x0F1B	Received Master Packet Bad CRC Counter (Communication State)	4bytes
0x0F1C- 0x0F1F	Received Master Packet with Bad Preamble and Bad CRC Counter (Communication State)	4bytes
0x0F20- 0x0F23	Same Channel Recovery Started Counter	4bytes
0x0F24- 0x0F27	Same Channel Recovery Successful Counter	4bytes
0x0F28- 0x0F2B	Unspecified Channel Recovery Successful Counter	4bytes
0x0F2C- 0x0F2F	Total Slave Transmitted Counter (Communication State)$^{(6)}$	4bytes
0x0F30- 0x0F33	Slave Resend Waveform Packet Counter (not Low Energy Mode)	4bytes
0x0F34- 0x0F37	Slave New Waveform Packet Counter	4bytes
0x0F38- 0x0F3B	Slave Memory Resend Packet Counter due to Bad CRC Master Packet	4bytes
0x0F3C- 0x0F3F	Slave Memory Resend Packet Counter due to Good CRC Master Packet	4bytes
0x0F40- 0x0F43	Slave Supplemental Transaction Request Counter	4bytes
0x0F44- 0x0F47	Slave Supplemental Packet Counter	4bytes
0x0F48- 0x0F4B	Received Master Memory Transaction Request Counter	4bytes
0x0F4C- 0x0F4F	Received Master Memory Packet Counter	4bytes
0x0F50- 0x0F53	Slave Memory Transaction Request Counter	4bytes
0x0F54- 0x0F57	Slave Memory Packets Counter	4bytes
0x0F58- 0x0F5B	Received Master Bad CRC Counter (Non-Communication State)	4bytes
0x0F5C- 0x0F5F	Memory Resend Counter due to More Slave Space	4bytes

TABLE 106: DIAGNOSTIC SERVICE RESPONSE — IMPLANT DATA

ADDRESS	DESCRIPTION	SIZE
0x0F60- 0x0F63	Total Received Master Packets Counter (Non-Communication State)	4bytes
0x0F64- 0x0F67	Total Slave Transmitted Packets Counter (Non-Communication State)	4bytes
0x0F68- 0x0F6B	Received BER Master Bit Error Counter for Good & Bad CRC packets	4bytes
0x0F6C- 0x0F6F	Received BER Master Bit Error Counter for Good CRC packet	4bytes
0x0F70- 0x0F73	Received BER Master packets (good CRC) counter	4bytes
0x0F74- 0x0F77	BER Slave Transmitted Packets counter	4bytes
0x0F78- 0x0F7B	Slave Resend Waveform Packet Counter (Low Energy Mode)	4bytes
0x0F7C- 0x0F7F	Received Last Packet Last Block Wakeup Counter in Disable State	4bytes
0x0F80- 0x0F83	Received Last Packet Last Block Wakeup Counter in Acquire State	4bytes
0x0F84- 0x0F87	Received Last Packet Last Block Wakeup Counter in Recovery State	4bytes
0x0F88- 0x0F8B	Received Last Packet Last Block Wakeup Counter in Communication State	4bytes
0x0F8C- 0x0F95	Error Event Array	10bytes
0x0F96-0x0F9F	Current State Array when each Error Event occurs	10bytes
0x0FA0	The next Error Event and Current State Array Index	1bytes
0x0FA1-0x0F04	Power Inhibit Rising Counter in Communication State	4bytes
0x0FA5-0x0F08	Power Inhibit Rising Counter in Same Recovery State	4bytes
0x0FA9-0x0FAC	Power Inhibit Rising Counter in Different Recovery State	4bytes
0x0FAD-0x0FB0	Power Inhibit Rising Counter in Acquire State	4bytes
0x0FB1-0x0FB4	Power Inhibit Rising Counter in Standby State	4bytes
0x0FB5-0x0FB8	Power Inhibit Rising Counter in Disable On State	4bytes
0x0FB9-0x0FBC	Power Inhibit Rising Counter in Eeprom Access State	4bytes
0x0FBD-0x0FC0	Waveform request has transmitted counter	4bytes
0x0FC1-0x0FC4	Waveform request from slave host counter	4bytes
0x0FC5	PLL Lock Error count(7)	1 byte
0x0FC6	Coarse Trim Failure count	1 byte
0x0FC7	FineTrim Failure count.	1 byte
0x0FC8	First_scan_channel_number_when_unicase_rxed	1 byte
0x0FC9	slave received open type byte (see **Fig. 41 (Open type byte (Request)))**	1 byte
0x0FCA-0x0FD5	Unused Bytes	12 bytes

1. Default address for slave filler packet.
2. See **TABLE 107: Type From Host** for detail.
3. See **TABLE 108: Channel Recovery Reason** for detail.
4. See **TABLE 109: Fault 1 flags** for detail.
5. See **TABLE 110: Fault 2 Flags** for detail.

6. Consider Max. slave throughput mode, the number will be 0xFFFFFFFF in 179.59 days, the number will be 0xFFFFFF in 16.84hours if telemetry C is used.
7. Number of trim successes = lock error count - coarse trim failure count - fine trim failure count

TABLE 107: TYPE FROM HOST

0x0F00	bit0	**Telemetry Type (0- Telemetry C; 1- Telemetry M)**
	bit1	**Authorize To Talk Unsecurely (0 - Must be Secure; 1 - It’s OK to be unsecure)**
	bit2	**Number of Waveform Data Offset Bytes(0- 4bytes offset, 1- 2bytes offset) from the Service Request buffer 3 and Service Request buffer4**
	bit3- bit7	**unused**

TABLE 108: CHANNEL RECOVERY REASON

0x0F0A	bit0	**First Received packet is Invalid Secured packet after open session**
	bit1	**Received Open request while in communication state (For development only)**
	bit2	**Received Master Frame Error Exceeded (Tel C only)**
	bit3	**Received Master Packet Error Exceeded**
	bit4	**Received Master Type is ARQ and Master ID Mismatched**
	bit5	**Received Master Type is Diagnostic and Master ID Mismatched**
	bit6	**Received Master Type is Memory Data and Master ID Mismatched**
	bit7	**Power Inhibit triggered channel recovery**
0x0F0B	bit0	**No packet is received after open session**
	bit1	**Receive Interrupt Times out in Communication state(Apply only to Tel M, digital link, for development only)**
	bit2	**Received Master Data packet type in Low Energy Mode (For development only)**
	bit3	**Trim successful (For development only)**
	bit4	**Received Master Type is BER and Master ID Mismatched (For development only)**
	bit5	**Received Slave Packet while in Communication State**
	bit6-bit7	**unused**

TABLE 109: Fault 1 flags and **TABLE 110: Fault 2 Flags** are logged for firmware development only. None of the faults will generate a fault reset.

TABLE 109: FAULT 1 FLAGS

0x0F0C	bit0	**Hear more than 10 stations (Discover Name) during scan**
	bit1	**Memory Data master buffer over flow**
	bit2	**Master Transaction from other master station. Will trigger Channel Recovery.**
	bit3	**Too many unacknowledged memory packets, mem_unacked_buf overflow**
	bit4	**Local waveform buffer logic error (used in development)**
	bit5	**unused**
	bit6	**Telemetry M Transmit FW and HW Counters Mismatch (in Low Energy Mode)**
	bit7	**unused**

TABLE 110: FAULT 2 FLAGS

0xF0D	bit0	**Master Sequence Number Unexpected**
	bit1	**Error in mem_unacked_buf**
	bit2	**Memory packet size is zero**
	bit3	**the discover collision avoidance time is zero and the correspond master ID is still in discover collision avoidance control structure**
	bit4	**Local first Waveform Buffer Uncompleted Sent**
	bit5	**Local second Waveform Buffer Uncompleted Sent**
	bit6	**Stop Retransmit Waveform When Both Waveform Requests have been Received from Host**
	bit7	**Stop Retransmit and Transmission Waveform When new Waveform Request has been Received from Host**

7.2.3 MAPPING DATA

TABLE 111: MAPPING DIAGNOSTIC SERVICE RESPONSE - ATTACHED DATA

BYTE OFFSET	DETAILED DESCRIPTION	HIGH LEVEL DESCRIPTION
1	channel 1, antenna 1 mapping value	Current mapping information(1)
2	channel 1, antenna 2 mapping value	
.	.	
.	.	
20	channel 10, antenna 2 mapping value	

1. These values come from the just completed diagnostic mapping.

7.2.4 DISCOVER NAME SLAVE RESPONSE FORMAT

All the values in **TABLE 126: SLAVE POLLING REQUEST**will be uplinked in five discover name response packets. **Table 112 (DISCOVER NAME SLAVE RESPONSE FORMAT)** shows an example of the CRM devices.

TABLE 112: DISCOVER NAME SLAVE RESPONSE FORMAT

FIELD NAME[1]	REQUIRED/ OPTIONAL	BYTES
Station Serial Number	Required	20
Null termination	Required	01
Station Name	Required	32
Null termination	Required	01
Station Med/Vit	Required	02
Station status	Optional	02
Application-specific info	Optional	16
Tel M Patient Name	Required	30
Message CRC	Required	02

1. Same format as for Telemetry B, except the following are added to the format for Telemetry M: “Tel M Patient Name” and “Message CRC”.

7.3 PSEUDO RANDOM NUMBER GENERATION

The security feature requires pseudo-random numbers. This appendix describes the implementation of a NIST-recommended pseudo random number generator (PRNG) based on the ANSI X9.31 specification (appendix A.2.4). This realization of the ANSI X.9.31 PRNG specification uses the AES block cipher (FIPS PUB 197).

It is assumed that two 128 bit random number are “installed” into the telemetry M module during production. One of the random numbers is permanent and the other is updated every time a new pseudo-random number is generated (both reside in keep alive memory).

Generation of a 128 bit pseudo random number (PRN) using the AES block cipher, as specified in ANSI X9.31, takes pace as follows. First, assume that two 128 bit random numbers are available to the PRNG, K_R and R_0. K_R, the PRNG key, does not change and is always loaded into the *key_i* input of the block cipher. R_0 is the PRNG’s secret and random initialization vector which is updated with the new and current PRN (R_1, R_2,...) every time the PRNG generates a 128 bit pseudo random number. Both K_R and the current R need to be stored in keep alive memory as they are required for the generation of any new pseudo random numbers.

The generation of the (i+1)th PRN is a three step process. In step one, the AES block cipher is used to encrypt a nonce (number used once) using KR as the key input, as shown in **Fig. 123 (Step one in generating a 128 bit pseudo random number using the AES block cipher)**. The nonce, which could consist of the telemetry M module’s current real time clock value, is loaded into the 128 bit *data_i* input of the block cipher. If the real time clock is used as the nonce and the clock is longer than 128 bits, then the least significant 128 bits of the real time clock should be used as the nonce. If the real time clock is shorter than 128 bits, then zeros shall be used to pad the most significant bits of the nonce. The output of the block cipher (*data_o*) is an intermediate value called V1.

In step two, the 128 bit intermediate value V1 is XORed (modulo 2 added) with the current R (R_0 is the very first pseudo random number is to be generated, R_i otherwise). This 128 bit value, V1 XOR R_i, is then encrypted with the block cipher. As shown in **Fig. 124 (Step two in generating a 128 bit pseudo random number using the AES block cipher)**, this means that K_R is loaded into the *key_i* input of the block cipher and V1 XOR R_i is loaded into the *data_i* input of the block cipher. This output of the block cipher (*data_o*) is an intermediate value called V2.

In the third and final step, the 128 bit intermediate values V1 and V2 are XORed together and encrypted with the block cipher. As shown in **Fig. 125 (Step three in generating a 128 bit pseudo random number using the AES block cipher)**, this means that K_R is loaded into the *key_i* input of the block cipher and V1 XOR V2 is loaded into the *data_i* input of the block cipher. The output of the block cipher (*data_o*) contains the new PRN, R_{i+1}. If no additional pseudo random numbers are required then R_{i+1} is stored in memory for generation of future PRNs. The PRNG key, K_R, also needs to be stored in memory.

The PRNG algorithm is best summarized by using a functional description. If the block cipher is represented by the function F(*data_i*; *key_i*) = *data_o*, then the algorithm can be summarized as follows:

Step 1: V1 = F(nonce; K_R)

Step 2: V2 = F(V1 XOR R_i; K_R)

Step 3: R_{i+1} = F(V1 XOR V2; K_R)

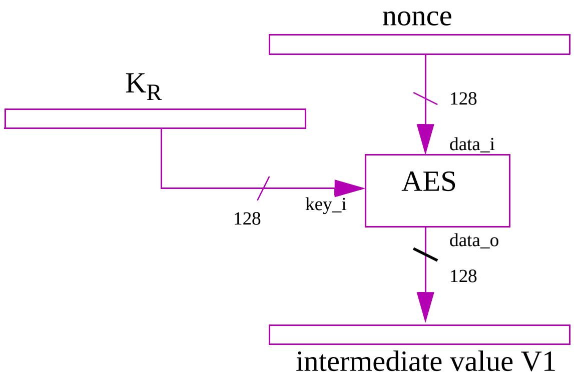


Figure 123: Step one in generating a 128 bit pseudo random number using the AES block cipher

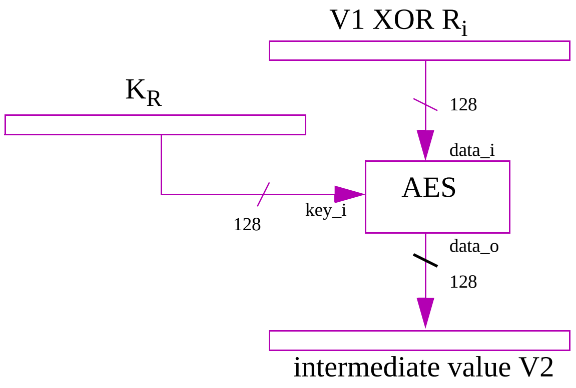


Figure 124: Step two in generating a 128 bit pseudo random number using the AES block cipher

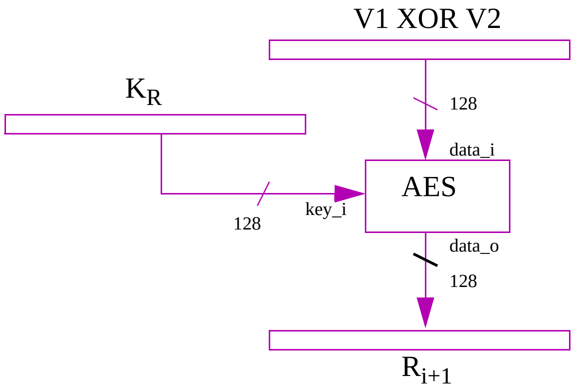


Figure 125: Step three in generating a 128 bit pseudo random number using the AES block cipher

7.4 MISC HOST INTERACTIONS

7.4.1 GENERAL NOTES:

The purpose of this section is to temporarily capture the interactions notes between host and Tel M firmware:

On April 17, 2008 a meeting was held between CSE and Neuro. We discussed the method the host would tell if the Mozart module was sniffing. The two choices were:

1. Issue host dummy write and initialize every 12 hours to redundantly "reconfigure" sniffing

2. Enable the power monitor bit every 12 hours (with interrupt enabled) for a duration greater than the sniff interval. If an interrupt occurs sniffing is enabled. If no interrupt occurs sniffing is not enabled. The host can then perform host dummy write and reconfigure sniffing.

The second method was agreed to by the CSE and Neuro teams. Since this algorithm could be used by multiple business units it is proposed that this is added to the protocol spec as text for the host (not a requirement). This way all business units can benefit from this knowledge.

7.4.2 DUAL PORT INTERFACE BUFFERS

TABLE 113: MOZART DUAL PORT REQUEST INTERFACE

DUAL PORT INTERFACE BUFFERS	MASTER	SLAVE
Request 0 buffer (512 bytes)	All commands Except Emergency	All Except Waveform and Data
Request 1 buffer (512 bytes)	Emergency	Uplink0
Request 2 buffer (512 bytes)	Unused	Uplink1
Waveform buffer 0 (256 bytes)	Unused	WF0
waveform buffer 1 (256 bytes)	Unused	WF1

TABLE 114: MOZART DUAL PORT INDICATION INTERFACE

INDICATION BUFFERS	MASTER	SLAVE
Indication 0 buffer (512 bytes)	Event indication (Notification)	Event indication (Notification)
Indication 1 buffer (512 bytes)	DATA	DATA 0 (DOWNLINK)
Indication 2 buffer (512 bytes)	UNREQUESTED DATA	DATA 1 (DOWNLINK)
Indication 3 buffer (256 bytes)	WAVEFORM	UNUSED
Indication 4 buffer (256 bytes)	UNUSED	WAKEUP

7.5 FIRMWARE INTERACTION BETWEEN HOST AND TELEMETRY

In order to send real time waveform data service requests independent of non-real time data service requests, the host slave application needs to utilize the waveform buffers (host interface request buffers 3 and 4) along with a few related host interface registers that are utilized to control these transactions. Keep in mind that when utilizing this alternate method of data transmission that no response is received from the slave data link layer as to the success or failure of the requested data transmission. This is because the channel is primarily designed for hardware DMA control blocks to act as the primary user of this channel in order to offload data transfer processing from the host firmware. Because the host interface hardware will not likely have the complexity to understand or react to a service response message, no such handshake occurs for these requests. Furthermore, this channel is designed to transfer real time data that is of no value if not delivered on time, so responding with a success or failure indication would add little or no benefit to the design.

The Tel M data link layer will retry a waveform request upon failure only until the next real-time waveform service request is issued. When a both a waveform and a memory write data service request are issued concurrently, the waveform data takes a higher priority and will be transmitted prior to the memory data request.

As stated previously, the waveform data request service is primarily designed for being exercised by host application hardware. However, the tel M data link layer provides the flexibility for host application software to also make these requests (however, hardware and firmware operation of this feature are mutually exclusive). To select between hardware and firmware control of this service, the host interface register bit tcds_enable of the register HST_MARKERS_TCDS. By setting the tcds_enable bit, it instructs the TM to service waveform data service requests in response to host interface writes to bits 3 (for buffer 3) and 4 (for buffer 4) of HST_IRQ2TEL. Alternatively, clearing the tcds_enable bit instructs the TM to service waveform data service requests in response to edge transitions of the TCDS input signal from the host hardware. When under hardware control, rising edges of TCDS means that data is available in buffer 4, while a falling edge indicates that data is available in buffer 3.

To specify the size of the payload contained in the waveform service request buffers, the slave host is required to specify the number of waveform bytes and the number of markers bytes/words that are contained in the buffer. The registers HST_WFCNT0 and HST_WFMKRCNT0 are used to specify the number of waveform bytes and markers contains in host interface buffer 3, while HST_WFCNT1 and HST_WFMKRCNT1 are used to specify the number of waveform bytes and markers contains in host interface buffer 4. Markers are located after the last waveform byte. When markers are present in the waveform buffer, bit marker0 (for host interface buffer 3) and bit marker1 (for host interface buffer 4) of the register HST_MARKERS_TCDS are used to indicate that markers are present as part of a waveform data service request.

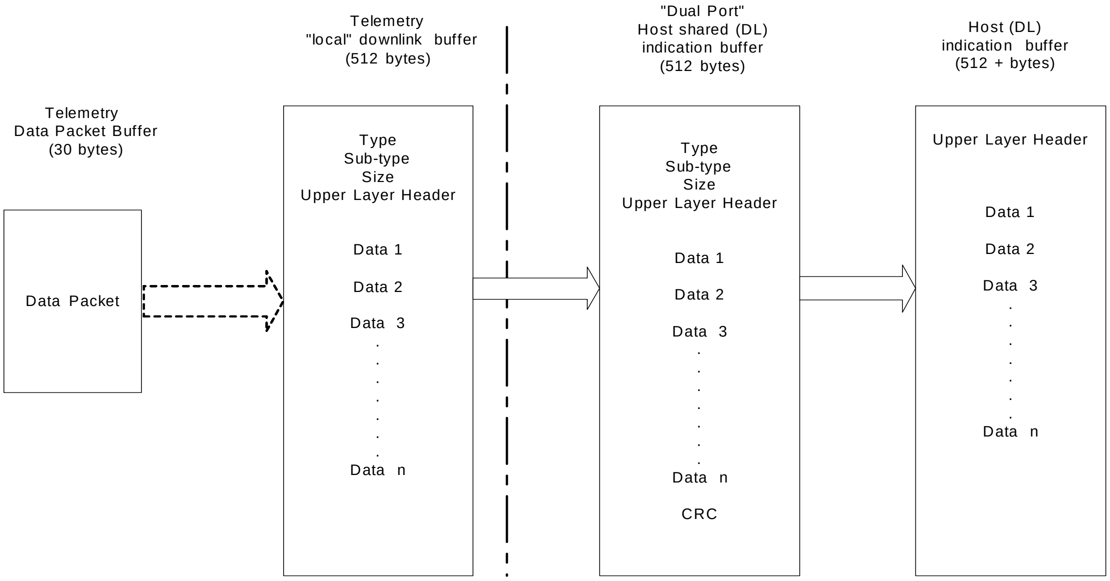
The description of the algorithms listed below provide the mechanism used to interact to received downlink, send memory and streaming uplink data.

7.5.1 INDICATION BUFFER USAGE (SLAVE)

Below is the mechanism used to receive the downlink data from external instrument and passing it to the host.

Indication Buffers

Downlink for implant & uplink for instrument - processing



The word downlink is being used in the description below - for instrument it represents uplink

Registers for Indication (DL)

hst_irq2host_0 - Interrupt register
# of bytes - third and forth bytes in shared (dl) buffer reflect the size of the indication data in the buffer
hst_irq2tel_1 - indication ack to telemetry

Algorithm

1. Each time a downlink packet is received
   a) Remove the telemetry specific header and CRC information
   b) Blockmove the data into the telemetry local_dl_buffer

2. When a final packet is received
   a) Blockmove the local dl buffer data into the shared dl buffer if downlink interrupt bit is low.
   b) Generate a indication interrupt to the HOST via **hst_irq2host_0.indication*(n)*** bit.
   c) HOST clears the (DL) Indication interrupt in the **hst_irq2host_0** register after moving the data to the HOST indication buffer". Hardware automatically generates **hst_irq2tel_1.indication*(n)*_ack.**

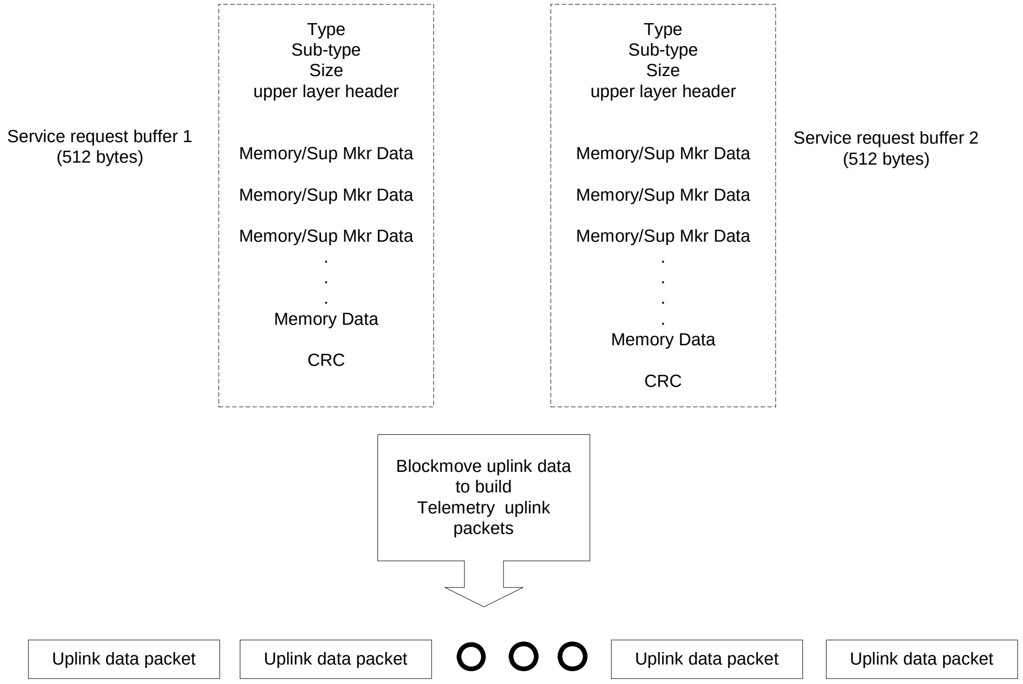
Note: The host use the upper layer header byte to combine the downlink transactions into the downlink message. Once the complete downlink message is in the host downlnk buffer, then host start to validate and start the processing of the dowlink message.

7.5.2 MEMORY/SUP. MARKER DATA BUFFER USAGE

Below is the mechanism used to received memory and supplemental marker data from the host and transmit.

Service request

Implant uplink, instrument downlink processing (Memory data or Supplemental Markers)



Uplink data packets ~ 30 bytes total
Each packets contains - header + data + MAC + RS

The word uplink is being used in the description below - for instrument it represents downlink

Registers

hst_irq2tel_0 - Interrupt register service_request_(n)

hst_irq2host_1 - service_response_(n) - NOTE - this should correspond to same as in hst_irq2tel_0.

Algorithm

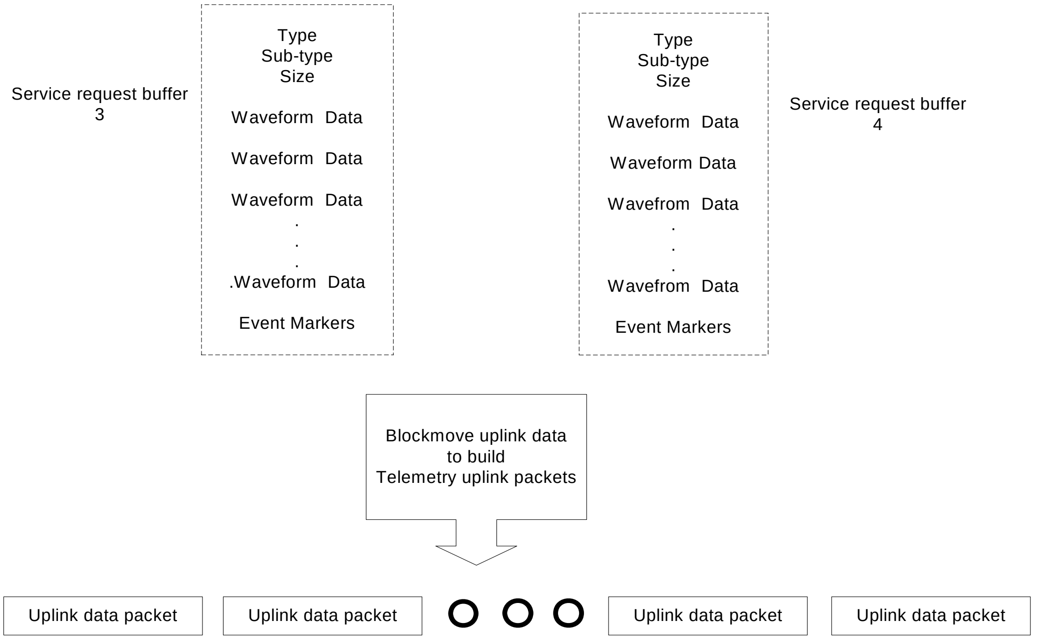
1. HOST checks the **service_request_*(n)*** register for empty buffer availability:
   a) Blockmove the data into the available buffer.
   b) Write the type, subtype and data size followed by CRC.
   c) Set the appropriate **service_request _*(n)*** in the **hst_irq2tel_0** register to generate interrupt toTelemetry.

2. When Telemetry C receives a memory data uplink request interrupt **local_hst_irq2tel_0** then
   a) FW clears the **hst_irq service_request_*(n)*** bit. (Stop HW from bothering FW).
   b) Build an uplink packet by blockmoving <= 30bytes of data in each packet.
   c) Transmit these packets.
   d) Once all the data is copied to the uplink packets (FW) clears appropriate **service_request_*(n)*** bit in **hst_irq2tel_0** register.
   e) Service response (uplink complete) interrupt is generated to the HOST:
      When (FW) clears the **service_request_*(n)*** bit in the **hst_irq2tel_0** register (HW) sets the **service_response_*(n)*** bit in the **hst_irq2host_1** register.

7.5.3 STREAMING DATA BUFFER USAGE

Below is mechanism used to received EGM/event marker data from the host to transmit.

Uplink Processing: Waveform data



Uplink data packets ~ 30 bytes total
Each packets contains - header + data + MAC+ RS

The word uplink is being used in the description below - for instrument it represents downlink

Registers

hst_irq2tel_0 - Interrupt register service_request_(n) - where n is 3 or 4.
hst_wfcnt0, hst_wfcnt1 - # of EGM uplink bytes in the service_request_(n) buffer
hst_markers_tcds - informs telemetry if markers are present in the buffer
hst_wfmkrcnt0, hst_wfmkrcnt1 - # of EGM & Event Markers uplink bytes in the service_request_(n) buffer

NOTE - For each buffer two registers are needed to determine the size (Opportunity for host improvement)

NOTE: if variable WF size is to be used - HW or FW from the host need to provide the size...

Algorithm

1. Host hardware fills the service request buffer with EGM.
    a) Host will put the marker in the available buffer (assuming HW is still doing EGM).
    b) Set the **service_request_*(n)*** bit in the **hst_irq2tel_0** to indicate the buffer written with data for uplink (done by hardware)
    c) In **hst_markers_tcds** register firmware sets bit0 or bit1 indicating if markers are in **service_request_(3) or service_request_(4)**

2. Upon receiving the service_request_(n) telemetry will do the following:
    a) FW will clear the **hst_irq** service_request_(n) (Keep HW from bothering FW).
    b) Check **hst_markers_tcds** register to use appropriate register to get the size of the buffer **hst_wfcnt0,hst_wfcnt1, hst_wfmkrcnt0, hstwfmkrcnt1**
    c) Blockmove the EGM/Marker data into the local buffers and clear appropriate **service_request_*(n)* in hst_irq2tel_0,** hardware will clear the hst_markers bits accordingly.
    Note for (c): Host can mask this interrupt since this is primarily used by the hardware

7.6 RAMWARE

The capability of be able to read and write memory on to the telemetry module allows the following:

- Ability to either download via telemetry a new set of functionality in the SRAM, or
- Ability to download firmware patches via telemetry to enhance or fix ROM issues.
- Ability for slave host to provide ramware via local data request.
- Ability for master host to provide ramware via local data request.

Typically ramware is generated by the master's host application software. Master host uses the upper layer header byte to indicate the intended recipient of the ramware (i.e. master or slave).

This allows the ramware to download to the telemetry module once a secure communication session is successfully opened (See **5.2.10 OPEN SESSION**). An external data service provides a mechanism through which master's host application downloads the ramware to the slave's telemetry module.

The slave host has an ability to provide ramware to the telemetry slave module prior to establishing the session using the local data request1 buffer.

Master's host uses two destination bits in the upper layer header byte (See **Figure 48: Master's ULH byte**and **Figure 49: Slave's ULH byte** for details). Master's host uses three types of record formats to download data to slave's telemetry module. These record types consist of Start of message followed by either Mozart module read or write or special function. The slave acknowledges the successful receipt of the transaction via Ready receive or Reject response.

TLMM_S_412 - RAMWARE RESTRICTIONS

The slave shall reject a RAMWARE request (data service request destined for slave Mozart) under the following conditions:

a. The session was started using Telemetry C, or

b. Telemetry M session is not secure, or

c. Telemetry M session was started with an EBAN key

[Ramware request from master host to slave mozart must be done using a secure telemetry M session started with a permanent BAN key. Telemetry C must use a master host to slave host data service request followed by a slave host to slave Mozart data request.

Additionally note that the master host should send the slave host a message to stop using service request buffer 1 before downlinking any ramware. The slave RFM uses the service request buffer when responding to a ramware downlink. If the slave host continues to use the buffer as well, unpredictable results will occur. The Slave Received Mozart Data Indication attempts to do the same thing, but a host to host downlink provides better protection. It prevents the race condition of the host sending a request on buffer 1 at the "same time" it is receiving a Slave Received Mozart Data Indication.]

TLMM_S_430 - MOZART DATA REQUEST INDICATION

The slave shall send Slave Received Mozart Data Indication (**TABLE 115: SLAVE RECEIVED MOZART DATA INDICATION**) to host when slave receives RAMWARE request (data service request destined for slave Mozart) from master during communication state.

When slave host receives the Slave Received Mozart Data Indication, host should stop using the data service request1 buffer and then acknowledge the indication. This is because slave Mozart will utilize the data service request1 buffer to send the Mozart data response to master once slave receives the indication acknowledge. Since there is no way for the slave host to know when the “ramware uplink” is complete, the slave host should discontinue using service request buffer 1 indefinitely (until the part is shutdown and re-enabled).

Finally since the slave host is prevented from using service request buffer 1, the master should no longer send data requests destined to the slave host after sending a ramware downlink. It is safe for the master to send normal (non-ramware) requests to the slave host once the session is closed and subsequently re-opened.

TABLE 115: SLAVE RECEIVED MOZART DATA INDICATION

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	Type of indication	0x39 - slave received mozart data indication
Sub-type	N/A	1		n/a
Size	N/A	2		1 bytes
Info	Slave Received Mozart Data Indication		Status of the indication	0x01 - successful
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

7.6.1 RAMWARE REQUESTS

There are three types of request messages - Mozart module memory read, Mozart module memory write and special function commands. Every message has a Start of Message record as the first two bytes and a message CRC as the last two bytes.

7.6.2 MOZART MODULE MEMORY READ AND WRITE

Master host to slave Mozart read and write commands must be performed after successful open session (communication state). Master host to master Mozart read and write commands must be performed before open session (master standby state). Slave host to slave Mozart read and write commands must be performed before open session (slave standby state). Each Mozart module message can contain a single read or a write request. The purpose of Mozart module memory read and write commands is to allow read and write access of the telemetry module’s local memory. These commands can also be used to download telemetry ramware patches into the module’s SRAM as necessary. And example of a Mozart module memory read or write message is shown below:

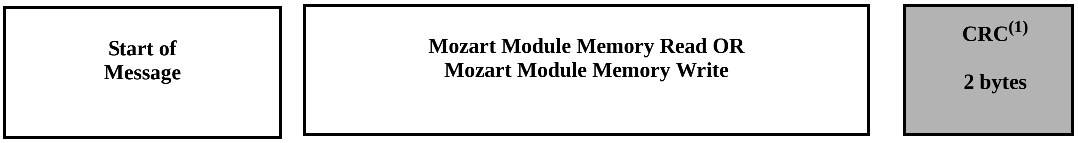


Figure 126: Mozart Module Memory Read or Write Message Format

(1) The CRC excludes the command code byte of the start of message record.

7.6.3 START OF MESSAGE

Each Mozart module memory read or write message will begin with a start of message record as follows:

Command Code (0xF0)	**Number records (0x02)**

Figure 127: Start of Message record format

The command code identifies this record is a start of message record. The second byte of the record defines the number of records in the message. The number of records will always be 2 for Mozart module memory read and write.

7.6.3.1 MOZART MODULE MEMORY READ

The master host or slave host will use the Mozart read command to retrieve the content of the Mozart master or Mozart slave's RAM. The maximum number of bytes to read is limited by the maximum transaction size (See **TABLE 113: Mozart dual port request interface**). Note that the master host can read the master Mozart memory or the slave Mozart memory. The slave host can only read the slave Mozart memory.

TLMM_MS_388 - MOZART MODULE MEMORY READ

The host will use the Mozart module memory read command to retrieve the content of telemetry RAM using the specified memory read record.

- The command code identifies the record as a Mozart module memory read record.
- The second, third, and fourth bytes of the record define the address of memory from which to start the read response.
- The fifth and sixth bytes define the number of memory bytes which are to be read. The number of memory bytes shall never be zero.

[The following are valid memory read addresses:

* 0x0000 - 0x3eff (fixed RAM - bank 0)
* 0x4000 - 0x7fff (non-fixed RAM - bank 1)
* 0x8000 - 0xbfff (non-fixed RAM - bank 2)

Other addresses are invalid.]

Command Code (0x00)	**Address (High)**	**Address (Med)**	**Address (Low)**	**#Memory Bytes (High)**	**#Memory Bytes (Low)**	**CRC** **2 bytes**

Figure 128: Mozart Module Memory Read Format

7.6.3.2 MOZART MODULE MEMORY WRITE

The host will use the Mozart module memory write command to write the content into the telemetry module’s memory using the memory write record. The memory write record can be used to write one or more data bytes. This memory write must be contiguous bytes from the starting address specified in the address field where message size is <= transaction size. (See **TABLE 113: Mozart dual port request interface**)

TLMM_MS_389 - MOZART MODULE MEMORY WRITE

The host will use the Mozart module memory write command to write contiguous bytes of the Mozart module.

- The command code identify the record as a Mozart memory write record
- The second, third, and fourth bytes of the record define the first memory location which is to be updated.
- The fifth and sixth bytes define the number of bytes to update. The number of bytes shall never be zero

[The following are valid memory write addresses:

* 0x0000 - 0x3eff (fixed RAM - bank 0)
* 0x4000 - 0x7fff (non-fixed RAM - bank 1)
* 0x8000 - 0xbfff (non-fixed RAM - bank 2)

Other addresses are invalid.]

Command Code (0x80)	**Address (High)**	**Address (Med)**	**Address (Low)**	**#Memory Bytes (High)**	**#Memory Bytes (Low)**	**Data (#bytes to be written)**	**CRC** **2 bytes**

Figure 129: Mozart Module Memory Write record format

7.6.3.3 SPECIAL FUNCTION COMMANDS

A message which contains a Special function command record (2 bytes), should begin with a Start of message as its first record. When the Mozart module receives a message with a special function command, it will execute that function (i.e., write EEPROM, or trigger/clear ramware). Refer to **TABLE 116: Special function commands** for the list of special command functions. Note that when the master host sends a special command to write EEPROM to the Mozart slave, the action will be deferred until the session is closed. This is done because concurrent telemetry and EEPROM writing cannot occur due to the large current demands of these functions.

Special function Command Code 0xA2	**Special function (see table below)**	**CRC 2 bytes**

Figure 130: Special function record format

TABLE 116: SPECIAL FUNCTION COMMANDS

VALUE	EEPROM STRUCTURES	RAM ADDRESS	OPERATION
0x01	P_HDW1	0X3600	write to eeprom(1)
0x02	P_HDW2	0X3670	write to eeprom(1)
0x03	P_WAKEUP	0x36E0	write to eeprom(1)
0x04	P_CONFIG	0x3750	write to eeprom(1)
0x05	P_RAMWARE	0x2300	write to eeprom(1)
0x06	TRIGGER RAMWARE	n/a	enable ramware
0x07	CLEAR RAMWARE	n/a	clear ramware(2)

1. The EEPROM block needs the CRC bytes swapped (whereas the application message level CRC does not). Additionally the polling duration (master initialize request) must be zero to write the slave EEPROM. Finally, the EEPROM is not written until a close session is performed (slave).
2. This command clears the RAMware from RAM. It does not clear it from EEPROM. If that is desired, a subsequent write ramware to EEPROM special command (value 0x05) must be used.

TLMM_MS_436 - MOZART MODULE WRITE TO EEPROM

When a valid write to EEPROM special function command is received from upper layer, Mozart shall write the appropriate structure to EEPROM. See **TABLE 116: Special function commands** (values 0x01 - 0x05).

TLMM_S_437 - MOZART MODULE WRITE TO EEPROM AFTER CLOSE WITHOUT POLLING INDICATION

When a valid write to EEPROM special function command is received from master Mozart Master, the slave Mozart shall write the appropriate structure to EEPROM after close with out polling indication is generated. (i.e., after close request is received and the polling duration is zero). See **TABLE 116: Special function commands** (values 0x01 - 0x05).

TLMM_MS_438 - MOZART MODULE ENABLE RAMWARE

When a valid enable ramware special function command is received from upper layer, Mozart shall enable ramware.

TLMM_MS_439 - MOZART MODULE CLEAR RAMWARE

When a valid clear ramware special function command is received from upper layer, Mozart shall clear the ramware areas.

7.6.4 RAMWARE RESPONSES

7.6.4.1 RESPONSE TO MOZART MEMORY READ

Response messages are constructed with one byte of command code which identifies the message, followed by a variable number of data bytes followed by two bytes of CRC on the data values. The message is generated by the Mozart module. An example is shown below.

Command Code (0x60)	**Data0**	**Data1**	**Data2**	**Data (n)**	**CRC** **2 bytes**

Figure 131: Mozart Module memory response format

7.6.4.2 RESPONSE TO MOZART MEMORY WRITE (RECEIVE READY)

The Mozart module will use a receive ready to signal that the processing of the memory write (or special command) is completed. An example is shown below.

Command Code (0x40)	**Status byte (0x00)**

Figure 132: Ready Receive message format

7.6.4.3 REJECT

The Mozart module will use a reject message to signal that the processing associated with the current Mozart request (memory read, memory write, or special command) has failed due to an interface error. This type of error is due to interface issues and can be used for system testing.

Command Code (0x48)	**Status byte**

Figure 133: Reject message format

TABLE 117: REJECT STATUS

VALUE	DESCRIPTION
0x01	CRC failure
0x02	Start of Message (F0,02) error
0x03	Unknown command
0x04	Record size too small (Slave only)
0x05	Record size too big (Slave only)
0x06	Message too short (Master only)
0x07	Invalid read record(1) (Master only)
0x08	Invalid write record(1) (Master only)
0x09	Run-time error(2)
0x0A	RAMWARE not supported in TelC (Master only)(3)
0x0B	RAMWARE not supported in unsecured session or secure session with EBAN key (Master only)(4)
0x0C	Record after the special command is unknown
0x0D	EEPROM write failed
0x0E	EEPROM write duration failed
0x0F	EEPROM write duration failed because Power Inhibit takes too long (slave only)

1. This status value is used by the master only. It occurs if the bytes to read (or write) are zero, too big, or any address is beyond 0xbfff (Note that fixed RAM bank 0 is at 0-0x3fff, non-fixed RAM bank 1 is at 0x4000-0x7fff and non-fixed RAM bank 2 is at 0x8000-0xbfff).
2. The format of the request message was correct, but the command failed due to one of the following: invalid CRC on RAM to copy to EEPROM.
3. This status can only occur when Master tries to send RAMWARE to slave Mozart while Telemetry C is running.
4. This status can only occur when Master tries to send RAMWARE to slave Mozart in unsecured session or secure session with EBAN key.

7.6.4.4 INTERFACE BETWEEN SLAVE HOST AND TELEMETRY MODULE

This capability allows the slave host to provide the ramware content to telemetry module. The host on the slave will use (data request1 buffer) to read, write, request to slave's telemetry module, as well as command telemetry module to write to EEPROM while telemetry module in the standby state. The telemetry module will be in the standby state after the initialization service request has been completed and no polling request has been issued by the slave host. The data service interface is used by the slave host to interface with the telemetry module. Slave's ULH byte should have a value of 0x30 see **Figure 49: Slave's ULH byte**. The data request transaction format for read, write, and special function command are the same as **Fig. 126 (Mozart Module Memory Read or Write Message Format)**. Slave's telemetry module will provide an appropriate data response upon completion. Following tables provide the interface responses read, write and special function commands:

.]

TABLE 118: SLAVE READ RESPONSE

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	Local ramware response	0x04 - data response
Sub-type	N/A	1	The service response sub-type	0 - memory data
Size	N/A	2	The size of the info field	n+6 bytes where n +1 = #bytes of data to read
Info	Slave local ramware		Status of the service request	byte1 - 0x01 (successful) byte2 - 0x30[1] byte3 - 0x60 byte4 - data0 byte5 - data1 .... byte(n+4) - data(n) byte(n+5) - crc high byte byte(n+6) - crc low byte and CRC for all data values (data0 through data(n))
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The ULH byte is 0x30 when the slave state is communication (response destined for master host). The ULH byte is 0x38 when the slave state is standby (response destined for slave host).

]

TABLE 119: SLAVE REJECT DATA SERVICE RESPONSE

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	Local ramware response	0x04 - data response
Sub-type	N/A	1	The service response sub-type	0 - memory data
Size	N/A	3	The size of the info field	4 bytes
Info	Slave local ramware		Status of the service request	byte1 - 0x01 (successful) byte2 - 0x30(1) byte3 - 0x48 byte4 - status byte For status byte information see **TABLE 117: Reject status**
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The ULH byte is 0x30 when the slave state is communication (response destined for master host). The ULH byte is 0x38 when the slave state is standby (response destined for slave host).

]

TABLE 120: SLAVE READY DATA SERVICE RESPONSE

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	Local ramware response	0x04 - data response
Sub-type	N/A	1	The service response sub-type	0 - memory data
Size	N/A	3	The size of the info field	4 bytes
Info	Slave local ramware		Status of the service request	byte1 - 0x01 (successful) byte2 - 0x30(1) byte3 - 0x40 byte4 - 0x00
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

1. The ULH byte is 0x30 when the slave state is communication (response destined for master host). The ULH byte is 0x38 when the slave state is standby (response destined for slave host).

]

TABLE 121: SLAVE EEPROM WRITE INDICATION

FIELD	INFO FIELD	# OF BYTES	DESCRIPTION	VALUES
Type	N/A	1	Type of indication	0x2F - eeprom write indication
Sub-type	N/A	1		n/a
Size	N/A	2		1 bytes
Info	Slave Eeprom Write Indication		Status of the indication	0x01 - successful 0x08 - failed 0x03 - timeout 0x12 - timeout because Power Inhibit takes too long
CRC	N/A	2	CRC of all the above fields. Two bytes.	This value is computed by the data link layer. It is used by the upper layer verify the integrity of the service response.

7.7 MASTER TELEMETRY C DIFFERENCES - LEM VS. MOZART

This section describes the master telemetry C differences (LEM vs. Mozart). All master telemetry C requirements are listed below (i.e., TLMC_E and TLMC_IE).

Master Telemetry C Requirements

TLMC_IE_001 - DOWNLINK PACKET CONTROL BYTES	IMPLEMENTED
TLMC_IE_002 - UPLINK PACKET CONTROL BYTES	IMPLEMENTED
TLMC_IE_003 - PACKET TYPES	IMPLEMENTED
TLMC_IE_004 - INSTRUMENT ID	IMPLEMENTED(A)
TLMC_IE_006 - EXTERNAL SERVICE REQUEST PROCESSING	IMPLEMENTED
TLMC_E_237 - SERVICE DURATION TOLERANCE	IMPLEMENTED(B)
TLMC_E_243 - UPLINK PACKET RESPONSE TIMEOUT	IMPLEMENTED
TLMC_E_010 - DISCOVER SERVICE REQUEST FROM C INTERFACE LAYER	IMPLEMENTED(C)
TLMC_E_011 - VALID IMPLANT ID WITH STOP ON MATCH	IMPLEMENTED(D)
TLMC_IE_012 - DISCOVER FULL - DISCOVER ID PACKET FORMAT	IMPLEMENTED
TLMC_E_013 - DISCOVER FULL - DOWNLINK DISCOVER ID REQUEST PACKET	IMPLEMENTED
TLMC_E_014 - DISCOVER FULL - RETRY DISCOVER ID REQUEST	IMPLEMENTED
TLMC_E_016 - DISCOVER FULL - LISTEN BEFORE TALK TIMER DURING ID	IMPLEMENTED(E)
TLMC_IE_017 - DISCOVER FULL - DISCOVER ID RESPONSE PACKET FORMAT	IMPLEMENTED
TLMC_IE_018 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET FORMAT	IMPLEMENTED
TLMC_E_019 - DISCOVER FULL - DOWNLINK DISCOVER NAME REQUEST PACKET	IMPLEMENTED(F)
TLMC_E_020 - DISCOVER FULL - RETRY DISCOVER NAME REQUEST	IMPLEMENTED
TLMC_IE_022 - DISCOVER FULL - DISCOVER NAME RESPONSE PACKET FORMATS	IMPLEMENTED
TLMC_E_024 - DISCOVER FULL - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(G),(H)
TLMC_E_025 - DISCOVER NAME - DOWNLINK REQUEST PACKET	NOT NEEDED(I)
TLMC_E_026 - DISCOVER NAME - RETRY DOWNLINK PACKET	NOT NEEDED
TLMC_E_027 - DISCOVER NAME -LISTEN BEFORE TALK TIMER DURING ID	NOT NEEDED
TLMC_E_028 - DISCOVER NAME - RESPONSE TO C INTERFACE LAYER	NOT NEEDED
TLMC_E_029 - DISCOVER STOP ON MATCH - DISCOVER ID RESPONSE MATCH INSTR ID	IMPLEMENTED
TLMC_E_030 - DISCOVER STOP ON MATCH - DISCOVER NAME REQUEST	IMPLEMENTED
TLMC_E_031 - DISCOVER STOP ON MATCH - MATCHING IMPLANT ID	NOT NEEDED(J)

A.Host interface difference. The master ID was set using the programmer ID ICCC request on the LEM. This is set via the master init request in Mozart.

B.Implementation difference. All timers had a resolution of 5 msec. On Mozart both discover and open have a resolution of one second. The tolerance for these request is +/- 1 second.

C.Host interface difference. The discover service duration is located in the master init request. The implant scan duration (now discover collision avoidance duration) is in the master init. The discover implant ID is still a parameter of discover on Mozart, but it does nothing.

D.Mozart does not have an explicit discover stop on match feature. However the discover full parameter called "discover slave ID" can be used to mimic the discover stop on match feature. On the LEM the "discover slave ID" cannot be set to all wildcards; on Mozart it can (i.e., discover full).

E.Implementation difference. LEM CCA took 223 to 934 msec. On Mozart it takes 275 msec.

F.Implementation difference. The discover retry is controlled by discover name retry timer (EEPROM) on Mozart. This timer has a resolution of 31 msec. It may not be possible to get exactly two retries.

G.Host interface difference. There are additional failed service responses on Mozart (see TLMC_M_024). The service response status 0x06 is not shown in this requirement. This response is pertinent to telemetry C. It means the service request is invalid in the current state (state is recovery; sending hibernate downlinks).

H.On the LEM the service response occurs at the end of the service. On Mozart the response is immediate. A timeout indication is used to communicate the service end. All ID/ name pairs were given at service end on the LEM. The ID/ name pairs are given as they are found on Mozart (discover indication). Note discover can be terminated via a discover stop service request as well (see req TLMM_M_274).

I.This feature was not brought forward since the host no longer needs it. The host won't given an ID without a name.

J.The Mozart discover implementation is terminated upon a service duration timeout or a discover stop service request.

TLMC_E_034 - DISCOVER STOP ON MATCH - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(G)
TLMC_E_040 - OPEN - C INTERFACE LAYER PARAMETERS	IMPLEMENTED(K)
TLMC_IE_041 - OPEN - DOWNLINK REQUEST PACKET FORMAT	IMPLEMENTED
TLMC_IE_042 - OPEN - UPLINK OPEN RESPONSE PACKET FORMAT	IMPLEMENTED
TLMC_E_043 - OPEN - DOWNLINK REQUEST	IMPLEMENTED(L)
TLMC_E_044 - OPEN - NO VALID UPLINK RESPONSE RETRY DOWNLINK REQUEST	IMPLEMENTED
TLMC_E_045 - OPEN - LISTEN BEFORE TALK TIMER OPERATIONS	IMPLEMENTED
TLMC_E_047 - OPEN - A VALID UPLINK RESPONSE	IMPLEMENTED
TLMC_E_048 - OPEN - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(M)
TLMC_E_215 - OPEN - HIBERNATE DOWNLINK REQUEST	IMPLEMENTED(N)
TLMC_E_049 - DATA - C INTERFACE LAYER TO DATA LINK LAYER PARAMETERS	IMPLEMENTED(O)
TLMC_IE_050 - DATA - DOWNLINK REQUEST PACKET FORMAT(S)	IMPLEMENTED
TLMC_E_051 - DATA - DOWNLINK OF DATA PACKET	IMPLEMENTED
TLMC_E_053 - DATA - PACKETIZATION OF THE DOWNLINK DATA	IMPLEMENTED
TLMC_E_054 - DATA - RETRANSMISSION OF DOWNLINK PACKET - NO ACK RECEIVED	IMPLEMENTED
TLMC_E_058 - DATA - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(P),(Q)
TLMC_E_063 - DATA - RECEIPT OF AN INVALID UPLINK PACKET	IMPLEMENTED
TLMC_E_064 - DATA - ASSEMBLING DATA PACKET INTO TRANSACTIONS	IMPLEMENTED(R)
TLMC_E_065 - DATA - RESPONSE BEFORE DATA INDICATION	NOT NEEDED(S)
TLMC_E_066 - DIAGNOSTIC SERVICE REQUEST FROM C INTERFACE LAYER	IMPLEMENTED(T)
TLMC_E_067 - REMOTE DIAGNOSTIC - DOWNLINK PACKET FORMAT	IMPLEMENTED
TLMC_E_071 - REMOTE DIAGNOSTIC - DOWNLINK REQUEST RETRANSMISSION	IMPLEMENTED
TLMC_E_072 - REMOTE DIAGNOSTIC - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(U)
TLMC_E_073 - LOCAL DIAGNOSTIC - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(V)
TLMC_E_074 - MAPPING DIAGNOSTIC - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(W)
TLMC_E_075 - DIAGNOSTIC FILLER - DOWNLINK FILLER PACKETS	IMPLEMENTED
TLMC_E_077 - EMERGENCY - C INTERFACE LAYER PARAMETERS	IMPLEMENTED(X)
TLMC_E_078 - EMERGENCY - DOWNLINK OF EMERGENCY REQUEST	IMPLEMENTED
TLMC_E_079 - EMERGENCY - DOWNLINK REQUEST NOT ACKNOWLEDGED	IMPLEMENTED
TLMC_E_080 - EMERGENCY - DOWNLINK REQUEST ACKNOWLEDGED	IMPLEMENTED
TLMC_E_081 - EMERGENCY - STATUS RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(Y),(Z)
TLMC_E_082 - RAW MODE - C INTERFACE LAYER DOWNLINK PARAMETERS	IMPLEMENTED(AA)
TLMC_E_083 - RAW MODE - DOWNLINK ATTACHED OPERATION	IMPLEMENTED

K.Host interface difference. The open service duration resides in the init request on Mozart. The open sub-type enumberations are as follows: 0 - open immediate; 1 - open hibernate.

L.Implementation difference. LEM CCA took 223 to 934 msec. On Mozart it takes 275 msec.

M.Host interface difference. There are additional failed service responses on Mozart (see TLMM_M_296). Note that the responses about BAN key do not apply.

N.The Mozart RFM does not send telem B device ID in parallel with telemetry C hibernate downlink packets.

O.Host interface change. On the LEM, the C interface layer managed the data service's reset parameter. In Mozart, the C interface layer has been removed. The data link layer now controls reset (see telem M req TLMM_M_362 and TLMM_M_363). Additionally the service duration has been replaced by the reply duration. On the LEM, the C interface layer controlled data service retry (failed due to recovery). On Mozart, this functionality has been moved into the data link layer (see TLMM_M_298, TLMM_M_299).

P.Host interface change. There are additional failed service responses on Mozart (see TLMC_M_058). The service response 0x29 is not used by telemetry C. Additionally the service response status 0x06 is not shown in this requirement. This response is pertinent to telemetry C. It means the service request is invalid in the current state (state is recovery; sending hibernate downlinks).

Q.On the LEM the service response occurs when the downlink transaction is fully acknowledged. On Mozart the service response occurs when the final uplink indication has been given to the master host.

R.Host interface difference. On the LEM, the data link layer sent a data indication to the C interface layer. The C interface layer then sent a downlink ICCC response to the host. On Mozart the data link layer sends a data indication directly to the host (see TLMM_M_297).

S.This requirement is not needed in Mozart since the host interface is different. Now a data service response comes after all indications.

TLMC_E_084 - RAW MODE - DOWNLINK ATTACHED UPLINK RESPONSE	IMPLEMENTED(AB)
TLMC_E_085 - RAW MODE - ECHO ON STATUS TO C INTERFACE LAYER	NOT NEEDED(AC)
TLMC_E_086 - RAW MODE - ECHO ON - DOWNLINK PACKET	NOT NEEDED
TLMC_E_087 - RAW MODE - ECHO ON - UPLINK PACKET	NOT NEEDED
TLMC_E_088 - RAW MODE - ECHO OFF STATUS TO C INTERFACE LAYER	NOT NEEDED
TLMC_E_089 - CLOSE - C INTERFACE LAYER DOWNLINK PARAMETERS	IMPLEMENTED(AD)
TLMC_IE_090 - CLOSE - DOWNLINK PACKET FORMAT	IMPLEMENTED
TLMC_IE_091 - CLOSE - UPLINK PACKET FORMAT	IMPLEMENTED
TLMC_E_092 - CLOSE - DOWNLINK REQUEST	IMPLEMENTED
TLMC_E_093 - CLOSE - RETRANSMISSION OF DOWNLINK REQUEST	IMPLEMENTED
TLMC_E_096 - CLOSE - RECEIPT OF A VALID RESPONSE	IMPLEMENTED
TLMC_E_097 - CLOSE - IMPLICITLY SUCCESSFUL	IMPLEMENTED
TLMC_E_098 - CLOSE - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(AE)
TLMC_E_099 - CLOSE - RECOVERY DURING CLOSE	IMPLEMENTED
TLMC_E_218 - MEDICAL EVENT - C INTERFACE PARAMETERS	IMPLEMENTED(AF)
TLMC_IE_219 - MEDICAL EVENT - UPLINK PACKET FORMAT	IMPLEMENTED
TLMC_E_220 - MEDICAL EVENT - MEDICAL EVENT REQUEST	IMPLEMENTED
TLMC_E_221 - MEDICAL EVENT - MEDICAL EVENT UPLINK	IMPLEMENTED
TLMC_E_222 - MEDICAL EVENT - INVALID MEDICAL EVENT UPLINK	IMPLEMENTED
TLMC_E_223 - MEDICAL EVENT - ANTENNA TIMEOUT	IMPLEMENTED
TLMC_E_224 - MEDICAL EVENT - LISTEN BEFORE TALK TIMEOUT	IMPLEMENTED
TLMC_E_228 - MEDICAL EVENT - RESPONSE TO C INTERFACE LAYER	IMPLEMENTED(AG)
TTLMC_E_100 - ARQ - STOP AND WAIT DOWNLINK ACKNOWLEDGMENT	IMPLEMENTED
TLMC_IE_101 - ARQ - DOWNLINK ARQ PACKET FORMAT	IMPLEMENTED
TLMC_E_102 - ARQ - UPDATING OF ARQ INFORMATION	IMPLEMENTED
TLMC_E_103 - ARQ - MEMORY DATA UPLINK PACKET ACKNOWLEDGMENT	IMPLEMENTED
TLMC_E_104 - ARQ - UPLINK ACK NOT RECEIVED	IMPLEMENTED
TLMC_E_105 - ARQ - INITIATE DOUBLE-UP WAVEFORM	IMPLEMENTED
TLMC_E_112 - CH MAPPING - COLLECT MAPPING DATA	IMPLEMENTED(AH)
TLMC_E_113 - CH MAPPING - PROCESS MAPPING DATA	IMPLEMENTED(AI)
TLMC_E_114 - CH MAPPING - CALCULATE AVERAGES	IMPLEMENTED
TLMC_E_115 - CH MAPPING - COLLECT MAPPING DATA	NOT NEEDED(AJ)
TLMC_E_116 - CH MAPPING - EVALUATE COLLECTED DATA	NOT NEEDED(AJ)
TLMC_E_117 - CH MAPPING - EVALUATION CONTROL	IMPLEMENTED(AK)

T.Host interface difference. The diagnostic service duration is not a parameter, it is obtained via the master init request (vs. a direct parameter of the diagnostic request on the LEM).

U.Host interface change. There are additional failed service responses on Mozart (see TLMC_M_072). The service response status 0x06 is not shown in this requirement. This response is pertinent to telemetry C. It means the service request is invalid in the current state (state is recovery; sending hibernate downlinks). Note, the info portion of the service response is defined in the telemetry M appendix (not the telemetry C appendix).

V.Host interface change. There are additional failed service responses on Mozart (see TLMC_M_073). Note, the info portion of the service response is defined in the telemetry M appendix (not the telemetry C appendix).

W.Host interface change. There are additional failed service responses on Mozart (see TLMC_M_074). The service response status 0x06 is not shown in this requirement. This response is pertinent to telemetry C. It means the service request is invalid in the current state (state is recovery; sending hibernate downlinks). Note, the info portion of the service response is defined in the telemetry M appendix (not the telemetry C appendix).

X.Host interface change. The service duration has been replaced by the reply duration (to be symmetric with the data service). This parameter resides in the master init reset on Mozart.

Y.Host interface difference. There are additional failed service responses on Mozart (see TLMC_M_081). The service response status 0x06 is not shown in this requirement. This response is pertinent to telemetry C. It means the service request is invalid in the current state (state is recovery; sending hibernate downlinks).

Z.On the LEM the data link sent an emergency service response when the emergency downlink was acknowledged. On Mozart, the emergency service response is not sent until the final uplink indication is given to the master host.

AA.Host interface difference. The raw mode service duration in the EEPROM (p_config) in Mozart. The raw mode types of "echo on" and "echo off" are not supported.

TLMC_E_118 - CH MAPPING - DIAGNOSTIC DATA COLLECTION	IMPLEMENTED
TLMC_E_119 - CH MAPPING - CALCULATE DIAGNOSTIC DATA AVERAGES	IMPLEMENTED(AL)
TLMC_E_120 - CH RECOVERY - TRIGGER CHANNEL RECOVERY	IMPLEMENTED(AM)
TLMC_E_121 - CH RECOVERY - START SAME CHANNEL RECOVERY	IMPLEMENTED
TLMC_E_122 - CH RECOVERY - SAME CHANNEL RECOVERY OPEN DOWNLINK	IMPLEMENTED
TLMC_E_123 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL	IMPLEMENTED
TLMC_E_124 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL	IMPLEMENTED
TLMC_E_261 - CH RECOVERY - TOO MANY SAME CHANNEL RECOVERIES	IMPLEMENTED
TLMC_E_125 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY	IMPLEMENTED(AN)
TLMC_E_126 - REMAP CHANNEL - UNSPECIFIED CHANNEL RECOVERY TIMER EXPIRATION	IMPLEMENTED(AO)
TLMC_E_127 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL	IMPLEMENTED
TLMC_E_128 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL	IMPLEMENTED
TLMC_E_139 - DATA TRANSFER MODE - MODE TRANSITIONS	IMPLEMENTED(AP)
TLMC_E_140 - DATA TRANSFER MODE - NOMINAL TO LOW ENERGY MODE TRANSITIONS	IMPLEMENTED
TLMC_E_141 - DATA TRANSFER MODE - NEGATIVE PREAMBLE RATIO IN LOW ENERGY MODE	IMPLEMENTED
TLMC_E_142 - DATA TRANSFER MODE - LOW ENERGY TO NOMINAL TRANSITION	IMPLEMENTED
TLMC_E_143 - DATA TRANSFER MODE - TRANSITION TO NEW MODE ON DOWNLINK	IMPLEMENTED
TLMC_E_145 - RESET SERVICE - DOWNLINK PACKET WITH RESET BIT	IMPLEMENTED(AQ)
TLMC_E_146 - RESET SERVICE - DOWNLINK RESET REQUEST RETRANSMISSION	IMPLEMENTED(AR)
TLMC_E_148 - RESET SERVICE - OPERATION SUCCESSFUL	IMPLEMENTED
TLMC_E_244 - RF HEAD LEDS	NOT NEEDED(AS)
TLMC_E_156 - SIGNAL - STRENGTH INDICATOR	IMPLEMENTED(AT)
TLMC_E_158 - CIL - REQUEST-RESPONSE-DEFINITION	NOT NEEDED(AU)
TLMC_E_159 - CIL - ID ALL REQUEST AND RESPONSE VIA DISCOVER FULL	NOT NEEDED(AV)
TLMC_E_161 - CIL - REQUEST ID WHEN DEVICE NAME NOT KNOWN	NOT NEEDED(AW)
TLMC_E_162 - CIL - REQUEST ID STRING DEVICE NAME KNOWN	NOT NEEDED
TLMC_E_164 - CIL - OPEN IMMEDIATE REQUEST	NOT NEEDED(AX)
TLMC_E_165 - CIL - INVALID ICCC OPEN IMMEDIATE REQUEST	NOT NEEDED
TLMC_E_166 - CIL - OPEN WITH DEVICE FOUND REQUEST	NOT NEEDED(AY)
TLMC_E_167 - CIL - INVALID OPEN WITH DEVICE FOUND REQUEST	NOT NEEDED
TLMC_E_217 - CIL - OPEN HIBERNATE REQUEST	NOT NEEDED(AZ)
TLMC_E_236 - CIL - OPEN STOP HIBERNATE REQUEST	NOT NEEDED
TLMC_E_168 - CIL - TARGET DEVICE ID	NOT NEEDED(BA)

AB.Host interface difference. There are additional failed service responses on Mozart (see TLMC_M_084). The service response status 0x06 is not shown in this requirement. This response is pertinent to telemetry C. It means the service request is invalid in the current state (state is recovery; sending hibernate downlinks).

AC.This feature was not ported since it took more CPU processing than available on Mozart. The protocol analyzer takes its place.

AD.Host interface difference. The close service duration resides in the master init request on Mozart.

AE.Host interface difference. There are additional failed service responses on Mozart (see TLMC_M_098).

AF.Host interface difference. The requirement TLMC_E_218 lists a master medical event service duration. This was never utilized on the LEM. It was not brought forward to Mozart.

AG.Host interface difference. There are additional failed service responses on Mozart (see TLMC_M_231). The service response status 0x06 is not shown in this requirement. This response is pertinent to telemetry C. It means the service request is invalid in the current state (state is recovery; sending hibernate downlinks).

AH.Implementation difference. Mozart does not collect preamble detected/ not detected during CCA.

AI.Implementation difference. Mozart CCA performs the screen phase only. It does not perform the evaluate phase.

AJ.This requirement is not needed due to a redesign of the CCA feature (no evaluate phase).

AK.Implementation difference. Mozart will select the quietest channel at the end of the screen phase. (see req M_357). Mozart noise not initialize the noise registers (the antenna select algorithm doesn't use them). Finally CCA cannot fail in Mozart. On the LEM it could if all channels detected a preamble.

AL.Implementation difference. The Mozart hardware computes the average CCA values.

AM.Implementation difference. Mozart has an additional recovery trigger (trim success). See req M_392.

AN.Implementation difference. Mozart does not send telem B device ID downlink in parallel with telem C open downlink during channel recovery.

TLMC_E_169 - CIL -VALIDATION OF TARGET DEVICE-ID	NOT NEEDED
TLMC_E_170 - CIL - CLOSE REQUEST	NOT NEEDED(BB)
TLMC_E_171 - CIL - DOWNLINK IN CLOSE REQUEST	NOT NEEDED
TLMC_E_172 - CIL - RESET SERIAL NUMBER AFTER CLOSE	NOT NEEDED
TLMC_E_173 - CIL - EMERGENCY DURING CLOSE REQUEST	PORTED(BC)
TLMC_E_174 - CIL - RAW MODE DOWNLINK (ATTACHED)	NOT NEEDED(BD)
TLMC_E_175 - CIL - REJECT RAW MODE DOWNLINK ATTACHED	PORTED(BE)
TLMC_E_176 - CIL - RAW MODE DOWNLINK (ECHO ON)	NOT NEEDED(BF)
TLMC_E_177 - CIL - RAW MODE DOWNLINK TO APPLICATION LAYER (ECHO ON)	NOT NEEDED
TLMC_E_178 - CIL - RAW MODE UPLINK DATA	NOT NEEDED
TLMC_E_179 - CIL - RAW MODE DOWNLINK (ECHO OFF)	NOT NEEDED
TLMC_E_180 - CIL - UPLINK DATA INDICATION TIMERS	NOT NEEDED(BG)
TLMC_E_181 - CIL - RESPONSE-APP-DATA-UPLINK	PORTED(BH)
TLMC_E_182 - CIL - FINAL ICCC DOWNLINK REQUEST	PORTED(BI)
TLMC_E_262 - CIL - RETRY DOWNLINK	PORTED(BJ)
TLMC_E_183 - CIL - MULTIPLE TRANSACTION DOWNLINK - NOT FINAL	PORTED(BK)
TLMC_E_184 - CIL - SERVICE DURATION TIMER	NOT NEEDED
TLMC_E_185 - CIL - EMERGENCY WHILE WAITING FOR DATA	PORTED(BL)
TLMC_E_186 - CIL - MESSAGE TAG IN THE DOWNLINK MESSAGE	HOST REQUIREMENT(BM)
TLMC_E_187 - CIL - UPLINK WITH NON-MATCHING TAG	HOST REQUIREMENT
TLMC_E_188 - CIL - EMERGENCY ICCC REQUEST DOWNLINK	NOT NEEDED(BN)
TLMC_E_189 - CIL - ICCC EMERGENCY REQUEST DOWNLINK	PORTED(BO)
TLMC_E_192 - CIL - SECOND ICCC EMERGENCY REQUEST WHILE PROCESSING FIRST	PORTED(BP)
TLMC_E_193 - CIL - UPLINK EMERGENCY DATA INDICATION TIMEOUT	PORTED(BQ)
TLMC_E_194 - CIL - RECOVERY-WHILE-WAITING-FOR-EMERGENCY	NOT NEEDED(BR)
TLMC_E_195 - CIL - EMERGENCY COMPLETE REQUEST	HOST REQUIREMENT(BS)
TLMC_E_196 - CIL - EMERGENCY COMPLETE RESPONSE	HOST REQUIREMENT
TLMC_E_197 - CIL - REMOTE STATISTICS ICCC REQUEST	NOT NEEDED(BT)
TLMC_E_198 - CIL - LOCAL STATISTICS ICCC REQUEST	NOT NEEDED(BU)
TLMC_E_199 - CIL - MAPPING STATISTICS ICCC REQUEST	NOT NEEDED(BV)
TLMC_E_210 - CIL - DISABLE DOWNLINK ICCC REQUEST	NOT NEEDED(BW)
TLMC_E_211 - CIL - ENABLE DOWNLINK ICCC REQUEST	NOT NEEDED(BX)
TLMC_E_245 - CIL - UPDATE DEVICE STATUS ICCC REQUEST	PORTED(BY)
TLMC_E_200 - CIL - TELEMETRY ON STATUS TO APPLICATION	NOT NEEDED(BZ)

AO.Implementation difference. CCA lasts 275 msec on Mozart. The LEM was variable: 223 to 934 msec.

AP.Host interface difference. The correlate negative ratio resides in the EEPROM (p_config) on Mozart.

AQ.Implementation difference. The master ULH reset bit is set if the data link layer detects conditions requiring reset (previous downlink failed or emergency).

AR.Implementation difference. The LEM continues to send reset until the service duration expires. Mozart continues to send reset until the reply duration expires. This is a very subtle difference.

AS.This requirement is not needed since the Mozart system has no RF head.

AT.Implementation difference. A new parameter called SSI limit is introduced on Mozart. To duplicate previous functionality, this parameter should be set to a value of "7" (no limiting). The parameter is intended for application which do not send periodic waveform. Also note that SSI is proportional to signal to noise ratio on the LEM. It is proportional to signal only on Mozart. Mozart does not measure the noise during the turn around time.

AU.This requirement is not needed since the C interface layer was removed when transitioning from LEM to Mozart.

AV.On the LEM the host used ID all to find all devices within range. On Mozart the host should use discover.

AW.The host used request ID string when it was given a list of ID/ name pairs where some had ID only (no name). This situation will not occur on Mozart. If the name is not obtained, Mozart does not give a discover indication to the host.

AX.On the LEM the host used open immediate ICCC request. On Mozart the host should use the open external service request. Additionally the LEM’s C interface layer had some parameter checking. This was moved to the data link layer on Mozart (i.e., model, sub-model, serial number may not contain wildcards).

AY.The host never used open device found on the LEM. This was confirmed in a 05-March-2010 email from John Farr.

TLMC_E_201 - CIL - TELEMETRY ON STATUS TO APPLICATION - CHANNEL RECOVERY. . . . . NOT NEEDED(CA)
TLMC_E_202 - CIL - TELEMETRY OFF MESSAGE APPLICATION. . . . . NOT NEEDED(CB)
TLMC_E_203 - CIL -TELEMETRY OFF MESSAGE DURING CHANNEL RECOVERY . . . . NOT NEEDED(CC)
TLMC_E_204 - CIL - CHART RECORDER DOWN ARROW - DOWNLINK STARTED . . . . NOT NEEDED(CD)
TLMC_E_205 - CIL - CHART RECORDER UP ARROW - UPLINK CONFIRMED . . . . NOT NEEDED(CE)
TLMC_E_206 - CIL - CONFIRM-DETECTED-STATE-CHANGE-2 . . . . NOT NEEDED
TLMC_E_207 - CIL - CONFIRM-DETECTED-STATE-CHANGE-3 . . . . NOT NEEDED
TLMC_IE_246 - CIL - PROGRAMMING HEAD PRESENCE OR ABSENCE STATUS . . . . PORTED(CF)
TLMC_E_247 - CIL - PROGRAMMING HEAD STATUS TO APPLICATION LAYER . . . . PORTED(CG)
TLMC_E_260 - CIL - MEDICAL EVENT UPLINK TO APPLICATION . . . . .NOT NEEDED
TLMC_E_235 - CIL - MEDICAL EVENT REQUEST AND RESPONSE . . . . NOT NEEDED(CH)
TLMC_E_248 - CIL - REACTION TO APPLICATION SYSTEM ERROR. . . . . NOT NEEDED(CI)

AZ.The host on the LEM used enable/ disable waveform ICCC request. The Mozart host should use open hibernate/ open normal external service request.

BA.The target ID was used for open device found. Since this feature wasn't used by the host this requirement was not brought forward into Mozart/

BB.The LEM host uses close ICCC request. The Mozart host should use close service request.

BC.This requirement was ported from the LEM's C interface layer to Mozart's data link layer. See status 0x03 in telem M requirement M_081.

BD.The LEM host uses ram mode ICCC request. Mozart host uses raw mode service request.

BE.This requirement was ported from the LEM's C interface layer to Mozart's data link layer. See status 0x27 in telem M requirement M_084.

BF.Raw mode echo was not ported on Mozart. This request took too much CPU processing. The protocol analyzer replaces this functionality.

BG.The LEM's concept indication timers was removed when transitioning to Mozart. The reply timer covers everything.

BH.This requirement was ported from the LEM's C interface layer. See telem M requirement M_297.

BI.This requirement (bullet 4) was ported from the LEM's C interface layer. See telem M requirement M_297. Note that Mozart does not strip the redundant command codes not the uplink CIH byte. The master host must perform that operation.

BJ.This requirement was ported (in spirit) from the LEM's C interface layer. See telem M requirement M_298.

BK.This requirement was ported (in spirit) from the LEM's C interface layer. See telem M requirement M_058. See note on non-final downlink transaction.

"TELEM C REQUIREMENTS IN TELEM M PROTOCOL DOCUMENT

TLMM_MS_428 - INITIALIZE - UPPER LAYER TO DATA LINK LAYER PARAMETERS
TLMM_MS_353 - SHUTDOWN - SERVICE REQUEST PARAMETERS
TLMM_MS_354 - SHUTDOWN - RESPONSE TO UPPER LAYER
TLMM_M_266 - MASTER INITIALIZE SERVICE REQUEST FROM UPPER LAYER
TLMM_M_267 - MASTER INITIALIZE SERVICE RESPONSE TO UPPER LAYER
TLMM_M_362 - RESET SERVICE - INVOKE SERVICE 1
TLMM_M_363 - RESET SERVICE - INVOKE SERVICE 2
TLMM_M_297 - DATA - WAIT FOR INDICATION
TLMM_M_298 - DATA - RECOVERY
TLMM_M_299 - DATA - REPLY TIMEOUT
TLMM_M_301 - EMERGENCY - WAIT FOR INDICATION
TLMM_M_302 - EMERGENCY - RECOVERY
TLMM_M_303 - EMERGENCY - REPLY TIMEOUT

BL.This requirement was "ported" from the LEM’s C interface layer in the sense that the data service stops when emergency is received. Note the LEM had a retry mechanism built in for failed data. This was just for failed data service requests (via recovery) not for failed requests via emergency.
BM.A system decision was made to push the message tagging to the host in Mozart.
BN.The LEM host uses emergency ICCC request. The Mozart host uses emergency service request.
BO.This requirement (bullet 4) was ported to Mozart. See telem M requirement M_301.
BP.Mozart will return "non-serial request" if a second emergency interrupts the first. Note the first emergency can be successful. See telem M requirement M_081 (service status 0x0f).
BQ.Mozart uses a reply timeout (vs. indication timeout). See telem M requirement M_301.
BR.On the LEM emergency is unsuccessful at recovery. On Mozart it is retried until the reply duration expires. This makes the feature symmetric with data.
BS.A host desiring this emergency complete functionality would need to add a layer between the host and Mozart. Rick Sanden has agreed this functionality will be added above the RFM (Mozart Host Interface or bridge component).
BT.The LEM uses remote statistics ICCC request. Mozart uses remote diagnostic service request.
BU.The LEM uses local statistics ICCC request. Mozart uses local diagnostic service request.
BV.The LEM uses mapping statistics ICCC request. Mozart uses mapping diagnostic service request.
BW.The host never used this feature on the LEM. It was not brought forward to Mozart.
BX.The host never used this feature on the LEM. It was not brought forward to Mozart.
BY.Mozart saves the device status (programming head detected during hibernate uplink) in a general purpose dual port register (tm_w1). The host can read this at any time.
BZ.The host can infer "telemetry on" from a successful open service response.
CA.The host can infer "telemetry on" from a successful recovery indication.

7.8 EEPROM CONTENT

The content of EEPROM consists of various structures, see **Table 122 (STRUCTURES IN EEPROM)**. For detail information regarding each structure, refer to Document **A44088**, **Telemetry M RF Module EEPROM Specification**.

Upon CPU enable, and if read EEPROM is successful, the parameters for each structure are copied from EEPROM into RAM. If read EEPROM is not successful, the parameters for each structure are copied from ROM into a RAM. Then the values in RAM will be copied to the corresponding registers.

The size for each of the four structures P_HDW1, P_HDW2, P_WAKEUP and P_CONFIG is fixed at 112bytes. As there are two bytes of CRCs for each 14bytes (block size) of data in EEPROM, each 112bytes of data will occupy 128bytes in EEPROM.

Similarly, the ramware control bytes is 14bytes in RAM and 16bytes in EEPROM; the maximum ramware code space reserved in RAM by firmware is 4816bytes and which occupies 5504bytes in EEPROM.

TABLE 122: STRUCTURES IN EEPROM

STRUCTURE NAMES		RAM ADDRESS	EEPROM ADDRESS
P_HDW1	Hardware Trim registers	0X3600-0x366F	0x00-0x7F
P_HDW2	Hardware Trim registers	0X3670-0x36DF	0x80-0x0FF
P_WAKEUP	Wakeup registers	0x36E0-0x374F	0x100-0x17F
P_CONFIG	Firmware specific parameters	0x3750-0x37BF	0x180-0x1FF
P_RAMWARE	Ramware control bytes	0x2300-0x230D	0x200-0x20F
P_RAMWARE_CODE	Ramware code	0x230E-0x35DD	0x280-0x17FF

CB.The host can infer "telemetry off" from a successful close response.

CC.The host can infer "telemetry off" from a recovery started indication.

CD.The host can infer downlink started from a "request first packet sent" indication.

CE.The host can infer a "confirm detect" from a successful service response (for data it should be a response where the request was the final transaction).

CF.This requirement was ported to Mozart. It results in a "device status" state change indication.

CG.This requirement was ported to Mozart. Mozart uses a device status indication. This indication has a type of 0xbb. Its sub-type is 0x01 if the slave is detecting the telemetry B programming head. The sub-type is zero otherwise. The size is 1 byte. The info[0] field is 0x01 (indication successful). Note that the LEM's device state change message communicated “device sending medical event uplink”. Mozart has a separate indication for medical event.

CH.The LEM uses a medical event ICCC request. Mozart uses a medical event service request.

CI.The Mozart RFM stays on if the host experiences a system error. On the LEM it will stay on for one minute and shut off.

7.9 MASTER CONFIGURATION PARAMETERS

Following is the sample of parameters provided by the host as part of initialization sample configuration values in **Table 123 (Master parameters from host)**can be adjusted to enhance the performance of the system. The resolution of each value depends on the value in the “LSB” column. For example, if “LSB” is “1seconds”, it means the timer is -1s to +1s. The [MIN, MAX] column indicates the firmware operation range. The byte order uses “big endian” convention.

For the actual values, see System Interface Specification.

TABLE 123: MASTER PARAMETERS FROM HOST

PARAMETER	BYTES	LSB	SAMPLE VALUE	[MIN, MAX](1)
Master Station ID(2)	6 bytes	n/a	non-zero value	[1, 0xFFFFFFFFFFFF]
Discover full service duration	2 bytes	1 second	120 seconds	[5, 0xFFFF]
Discover collision avoidance duration (for slave)	1 byte	1 second	30 seconds	[5, 0xFF]
Open service duration	1 byte	1 second	5 seconds	[1, 0xFF)]
Diagnostic service duration	1 byte	31.25 ms	1 second	[1, 0xFF)]
Close service duration	1 byte	31.25 ms	1 second	[1, 0xFF)]
Network service duration	1 byte	1 second	5 seconds	[1, 0xFF)]
Security service duration	1 byte	1 second	5 seconds	[1, 0xFF)]
Ephemeral BAN Key timer	2 bytes	1 second	360 seconds (6 minutes)	[5, 0xFFFF]
Requested slave unspecified channel recovery duration (UCR)	2 bytes	1 second	5 seconds	[5, 0xFFFF]
Requested slave normal polling duration	2 bytes	1 second	0 seconds(3)	[0, 0x12C]
Requested slave standby polling duration (Tel C only)	2 bytes	1 second	1800 sec. (30 min)	na
Transmit wakeup burst length async	2 bytes	1 packet	96 (It takes ~3 seconds for MICS wide band.) each packet represents - 31 ms (MICS wide band)(4) each packet represents - 93ms (MICS narrow band) each packet represents - 93ms (MEDS band)	[1,0x3FFF] 1-16383packets
Unused	3 bytes	n/a	n/a	n/a
CCA band(5)	1 byte	n/a	0x02- MICS	0x01 - MEDS, 0x02- MICS, 0x03- MICS & MEDS
Transmit Power Hold	1 bytes	n/a	0 - Change Transmit Power based on TelC, TelM, Wakeup on Mics or Meds Band	1 = Don’t Change Transmit Power (for manufacture testing) 0 = Change Transmit Power based on TelC, TelM, Wakeup on Mics or Meds Band

TABLE 123: MASTER PARAMETERS FROM HOST

PARAMETER	BYTES	LSB	SAMPLE VALUE	[MIN, MAX]$^{(1)}$
syncWakeupBurstType	1 byte	n/a	0	bit0: 1 - use sync wakeup transmit preceding open, 0 - use async wakeup transmit preceding open bit1: 1 - use sync wakeup transmit preceding network, 0 - use async wakeup transmit preceding network bits 2 -7: unused
Discover Wakeup Type	1 byte	n/a	0 - GLOBAL Wakeup	bit 0: 0 - GLOBAL Wakeup; 1 - NONLOCAL Wakeup; bits 1-7: Unused
Reply duration (for data & emergency)	1 byte	31.25 ms	3 seconds	[3,0xFF]
Security for wakeup user data/ network packet	1 byte	n/a	1 - secure	1 - secure, 0 - unsecure
Wakeup user data present	1 byte	n/a	0 - no data present	1 - user data present, 0 - no user data present
Wakeup user data	10 bytes	n/a	All zeroes	4 bytes for secure, 10 bytes for unsecure
Slave Model ID$^{(6)}$	2 bytes	n/a	non-zero value	[1,0xFFFF]
Master test feature	1 byte	n/a	0	bit0: 1 - raw mode service enabled bit1: 1 - RF test service enabled bits 2 -7: unused
RF/Digital link	1 byte	n/a	1 - RF$^{(7)}$	0 - digital, 1 - RF
Telemetry Scheme	1 byte	n/a	1 - TelM	0 - TelC, 1 - TelM
Transmit notification enabled (chart recorder triangles)	1 byte	n/a	1 - enable	0 - disable, 1 - enable
Signal strength indicator threshold K1	1 byte	1 count	48 counts (~24 dB)	na
Signal strength indicator threshold K2	1 byte	1 count	66 counts (~ 33 dB)	na
Signal strength indicator threshold K3	1 byte	1 count	84 counts (~42 dB)	na
Signal strength indicator threshold K4	1 byte	1 count	102 counts (~51 dB)	na
Signal strength indicator threshold K5	1 byte	1 count	120 counts (~60 dB)	na
Signal strength indicator threshold K6	1 byte	1 count	138 counts (~69 dB)	na
Master Xmit data rate for native mode	1 byte	See value	3-48Kbps	0-380Kbps, 1-97Kbps, 2-190Kbps, 3-48Kbps
Master Receive data rate for native mode	1 byte	See value	2-190Kbps	0-380Kbps, 1-97Kbps, 2-190Kbps, 3-48Kbps

TABLE 123: MASTER PARAMETERS FROM HOST

PARAMETER	BYTES	LSB	SAMPLE VALUE	[MIN, MAX](1)
TelM clock cpu fast speed	1 bytes	See value	0 = 22.8 Mhz	101 = 1.9MHz; 100 = 3.8MHz, 011 = 5.7 MHz, 010 = 7.6MHz, 001 = 11.4MHz, 000 = 22.8MHz(8)
TelM clock cpu slow speed	1 bytes	See value	0 = 22.8 Mhz	101 = 1.9MHz; 100 = 3.8MHz, 011 = 5.7 MHz, 010 = 7.6MHz, 001 = 11.4MHz, 000 = 22.8MHz(8)
HDR Mode	1 byte	n/a	0 - disable HDR mode	1 - enable HDR mode; 0 - disable HDR mode
Slave Xmit Packet count for nominal mode (For Tel M only)	1 byte	1 packet	non-zero value, default: 5 packets	See **Table 103 (Telemetry M Recommend Slave Packet Count)**
Slave Xmit Packet count for max throughput mode (For Tel M only)	1 byte	1 packet	non-zero value(9), default: 32 packets	
(10)Threshold1: # of consec frames with no valid receive packets (For Tel M only)	1 byte	1 frame	non-zero value, default:2 frames	[0x02,0xFF]
Threshold2: # of consec frames with no valid receive packets (For power inhibit mode in Tel M)	1 byte	1 frame	non-zero value, default:6 frames	[0x06,0xFF]
Threshold3: # of consec frames with no valid receive packets (For power inhibit mode in Tel M)	1 byte	1 frame	non-zero value, default:10 frames	[0x0A,0xFF]
Too many recoveries threshold	1 byte	n/a	5 recoveries(11)	[5, 10]
Too many recoveries duration	2 bytes	1 sec	5 seconds	[0x05,0xFFFF]
Delay new waveform detection(12)	1 byte	n/a	0 - no delay to waveform detection	0 - no delay to waveform detection 1 - delay waveform detection
Slave scans during polling	1 byte	n/a	0x01 - scan during polling	bit 0: 1 = scan during polling, 0 = alternating async sniff/ native during polling bit 1 - 7: unused
Signal strength indicator input limit(13)	1 byte	n/a	7	1..7
LoadBan Key NetworkTable	1 byte	n/a	0x00	bit 0: 0 = do not load Ban Key 1= load Ban Key - host should only set this bit if the request bankey local was successful. bit 1: 0= do not load Network Table bit 1: 1= load Network Table
Ban Key	16 bytes	n/a	all zeros	the saved Ban Key
Network Table	26 bytes	n/a	all zeros	the saved network table

TABLE 123: MASTER PARAMETERS FROM HOST

PARAMETER	BYTES	LSB	SAMPLE VALUE	[MIN, MAX][1]
CCA Before And After Wakeup Transmit	1 byte	n/a	0 = Perform CCA before wakeup transmitted	0 = Perform CCA before wakeup transmitted 1 = Perform CCA before and after-wakeup transmitted
Frequent Waveforms Expected (used in low energy mode only)	1 byte	n/a	0 = waveform(s) not expected in every uplink frame	0 = waveform(s) not expected in every uplink frame 1= waveform(s) expected in every uplink frame
AddressH, AddressM, AddressL, Value (Total of 80 addresses and values are supported)	4bytes	n/a	0x00, 0x00, 0x00, 0x00 (default) Up to 80 (address, value) pairs may be utilized. Zero in all three address bytes indicates the end	For development only.

1. The ranges are not explicitly enforced (by a failed service response) unless otherwise noted.
2. Must be non-zero otherwise initialize service request failure. Note: In Tel M all six bytes are utilized, TelC only utilizes the first three bytes (last three bytes are ignored).
3. When Requested slave normal polling duration = 0, slave won’t start polling after unspecified channel recovery fails or the session is closed. This should be OK for telemetry M because (1) If slave EBan key timer is running, then slave is sniffing in DisableON state. (2)If slave already be disabled (2v is off), slave is also sniffing. Master will send wakeup packets to wake up the slave right away.
4. If the CCA band is set to 3 - MICS and MEDS, and band control is set to wide, the length of each packet may be ~ 31ms or ~ 93ms depending on which channel, MICS or MEDS, is chosen for transmission.
5. The CCA band allows the user to select the band for wakeup transmit, native mode transmit, or medical event listening.
6. Config. slave model id must match open slave ID in open service request otherwise failure.
7. This value shall be zero (digital) for verification testing. The snf_abort_mask_test parameter in the P_Hardware1 EEPROM structure (see Document **A44088**, **Telemetry M RF Module EEPROM Specification**) should be set to 0x8F for a digital link, and to 0x00 for an RF Link.
8. The range is shown for bits 0 - 2. The hardware uses bits 3 and 4. These should be set to zero. Bits 5-7 are unused.
9. Slave xmit packet count for max throughput mode should be greater than or equal to slave xmit packet count for nominal mode.
10. These threshold is only pertinent to telemetry M. The telemetry C threshold is in EEPROM.
11. Note that this value should be set to zero (feature disabled) if the slave may perform periodic (idle immediate) power inhibit. The master cannot distinguish between pulse noise and power inhibit (idle immediate).
12. See the internal service (packet retransmission) for further definition of this parameter.
13. The use of this parameter is described in **5.3.8 Signal Strength Indicator**.

7.10 SLAVE CONFIGURATION TABLE

The sample configuration values in **Table 124 (SLAVE PARAMETERS FROM HOST)** can be adjusted to enhance the performance of the system. See the device-specific SPD (System Parameters Database) for the actual values. The byte order uses “big endian” convention.

TABLE 124: SLAVE PARAMETERS FROM HOST

PARAMETER	#BYTES	LSB	SAMPLE VALUE	[MIN, MAX]
Slave station ID	6 bytes		non-zero value	[1,0xFFFFFFFFFFFF]
Total Polling duration for Scheduled Transmission(Tel C only)(1)	2 bytes	1 second	30 seconds	[3, 0xFFFF]
Total Initial Polling duration(2)	2 bytes	1 second	10 seconds	[5, 0xFFFF]
Scan duration during Polling (Tel C only)	1 byte	1 second	3 seconds	[3,0xFF]
Sleep duration during Polling (Tel C only)	1 byte	1 second	0 seconds	[0,0xFF]
Wait for Open packet after transmitting (3) (Medical Event channel transmit time)	1 byte	5 ms	35 ms (TelC) 55 ms (Tel M)	[0x07,0xFF]
Medical Event Service Duration	1 byte	1 second	30 seconds	[3, 0xFF]
Security Service Duration	1 byte	1 second	3 seconds	[3, 0xFF]
Authorize to talk unsecurely	1 byte	n/a	1 - unsecure, 0 - secure	na
Wakeup Sniff Interval	3 bytes	~ 0.9ms	2048 (2 seconds)	[1,0x3FFFF] 0.97ms - 256s
Wrap Up Interval Number of Sniff Intervals between each Wrap Up (Mozart internal housekeeping routine).(4)	2 bytes	1 Sniff	21600 When the sniff interval is 2 seconds, this results in a wrap interrupt every 12 hours.	[0,0xFFFF] 0xff - 65535 sniff intervals
Sniff List and Band Control	36 bytes	n/a	See **Table 125 (SNIFF LIST FORMAT)**	
Falsing Enable (Typically set to zero for normal operation)	1 byte	n/a	0	Bit 0: 1 = Enable Peak Offender; 0 = Disable this feature Bit 1: 1 = Enable Lengthen Sniff Intvl; 0 = Disable this feature Bit 2:1 = Enable Incr. Phase 1 Coarse; 0 = Disable this feature Bit 3:1 = Enable Shorten Wrap Up; 0 = Disable this feature Bit 4: 1= Disable Wakeup; 0 Enable Wakeup Bit 5- Bit7: unused
Power Inhibit (This bit can be used by Telemetry C and Telemetry M although it's originally designed for power inhibit feature of Telemetry M)	1 byte	n/a	0x03	Bit 0: 1 = Enable WF in Recovery; 0 = Disable this feature Bit 1: 1 = Send channel recovery indication to host, 0 = Don’t send channel recovery indication to host (see footnote (5)) Bit 2: - 1 = Send Channel Recovery Imminent Indication to host; 0 = Don’t send Channel Recovery Imminent Indication to host Bit 3: - 1 = Send Channel Recovery Imminent Resume Indication to host; 0 = Don’t send Channel Recovery Imminent Resume Indication to host Bit 4: - Bit7: unused
Prog. head present status (Tel C only)	1 byte	n/a	0	0 - prog head not present, 1 - prog. head present

TABLE 124: SLAVE PARAMETERS FROM HOST

PARAMETER	#BYTES	LSB	SAMPLE VALUE	[MIN, MAX]
RF/Digital link	1 byte	n/a	1 - RF(6)	0 - digital, 1- RF
Telemetry scheme and waveform setup	1 byte	n/a	0x01	Bit 0: Telemetry scheme - 0 - TelC, 1 - TelM Bit 1: Waveform data offset 0 - Waveform data offset by four bytes 1 - Waveform data offset by two bytes Bit 2: Delay sending new waveform 0 – Send the (n+2)th waveform transaction immediately. When the (n+2)th waveform transaction is received from the host, stop retrying old waveform transactions. Stop the retry of the nth waveform transaction and cancel the (n+1)th waveform transaction. 1 – Delay sending the (n+2)th waveform transaction. When the (n+2)th waveform transaction is received from the host, continue to retry old waveform packets of the nth waveform transaction if needed and transmit the (n+1)th waveform transaction packets before transmitting packets from the (n+2)th waveform transaction. Bit 3: Scan Tel M Mics Band Only (see note (7)) Bit 4: Scan Tel M Meds Bands Only Bit 5: Scan Tel M Mics and Meds Bands Only Bit 6- Bit 7: unused
TelM clock cpu fast speed	1 byte	See value	0x03 - 5.7Mhz(8)	101 = 1.9MHz; 100 = 3.8MHz, 011 = 5.7 MHz, 010 = 7.6MHz, 001 = 11.4MHz, 000 = 22.8MHz(9)
TelM clock cpu slow speed	1 byte	See value	0x04 - 3.8Mhz	101 = 1.9MHz; 100 = 3.8MHz, 011 = 5.7 MHz, 010 = 7.6MHz, 001 = 11.4MHz, 000 = 22.8MHz(9)
Power reduction during security key transfer	1 byte	n/a	0 - no power reduction	0 - no power reduction 1 - use (Set 1) power reduction values from EEPROM (see structure p_hdw2) 2 - use (Set 2) power reduction values from EEPROM (see structure p_hdw2) 3 - use power reduction values from initialization
Unused	3 byte	n/a		
Slave Same channel recovery duration (should be the same as Master Same channel recovery duration, See **Table 129 (PARAMETERS FROM EEPROM)**)	1 byte	5 ms	0x58 - 440 ms	[0x24, 0xFF] 180ms - 1.275 s
Tel C version number	2bytes		'B' '0'	[0x0000, 0xFFFF]
Unused	1byte			
New Channel Recovery Trigger Threshold For Nominal mode	1 byte	1 packet	non-zero value default: 0x30	[0x04, 0xFF]
New Channel Recovery Trigger Threshold For Max. Transmit Mode	1 byte	1 packet	non-zero value default: 0x20	[0x04, 0xFF]
New Channel Recovery Imminent Threshold For Nominal mode	1 byte	1 packet	non-zero value default: 0x2E	[0x03, 0xFE]
New Channel Recovery Imminent Threshold For Max. Transmit Mode	1 byte	1 packet	non-zero value default: 0x1E	[0x03, 0xFE]

TABLE 124: SLAVE PARAMETERS FROM HOST

PARAMETER	#BYTES	LSB	SAMPLE VALUE	[MIN, MAX]
Transmit Power Hold	1 bytes	n/a	0 - Change Transmit Power based on TelC, TelM, Wakeup on Mics or Meds Band	1 = Don't Change Transmit Power (for manufacture testing) 0 = Change Transmit Power based on TelC, TelM, Wakeup on Mics or Meds Band
LoadBan Key Network-Table/Check Always On Parameters CRC	1 byte	n/a	0x00	bit 0: 0 = do not load Ban Key 1= load Ban Key host should only set this bit if the request bankey local was successful bit 1: 0= do not load Network Table bit 1: 1= load Network Table bit 2: 0 = slave does not calculate or check Always On parameters bit 2: 1= slave shall calculate and check Always On parameters
Ban Key	16bytes	n/a	all zeros	the saved Ban Bey
Network Table	26 bytes	n/a	all zeros	the saved network table
BP_PA_Coarse_PWR(10) (Power Amplifier Coarse Power Control Value)	1 byte	n/a	0x0F	Reduced Key Tx Power Set 3 [0x01, 0x0F] Bits 0-4 are used, Bits 5-7 are ignored.
BP_PA_FINE_PWR(10) (Power Amplifier Fine Power Control Value)	1 byte	n/a	0x0F	Reduced Key Tx Power Set 3 [0x01, 0x0F] Bits 0-4 are used, Bits 5-7 are ignored.
BP_PA_BIAS_BUF_TRIM(10) (Power Amplifier Buffer Bias Current Trimming value)	1 byte	n/a	0x07	Reduced Key Tx Power Set 3 [0x00,0x07] Bits 0-2 are used, Bits 3-7 are ignored.
BP_PA_BIAS_TRIM(10) (Enable PA Positive Feedback and PA output Bias Current Trimming value)	1 byte	n/a	0x07	Reduced Key Tx Power Set 3 [0x00,0x07] Bits 0-2 are used, Bits 3-7 are ignored.
BP_TX_AFILTI_DC_CTRL(10) (TX DAC Filter DC Control (I channel)	1 byte	n/a	0x3F	Reduced Key Tx Power Set 3 [0x00,0x3F] Bits 0-5 are used, Bits 6-7 are ignored.
BP_TX_AFILTQ_DC_CTRL(10) (TX DAC Filter DC Control (Q channel)	1 byte	n/a	0x3F	Reduced Key Tx Power Set 3 [0x00,0x3F] Bits 0-5 are used, Bits 6-7 are ignored.
BP_TX_DACI_TRIM(10) (TX DAC Gain Trimming values (I channel))	1 byte	n/a	0x1F	Reduced Key Tx Power Set 3 [0x00,0x1F] Bits 0-4 are used, Bits 5-7 are ignored.
BP_TX_DACQ_TRIM(10) (TX DAC Gain Trimming values (Q channel))	1 byte	n/a	0x1F	Reduced Key Tx Power Set 3 [0x00,0x1F] Bits 0-4 are used, Bits 5-7 are ignored.
Unused	58Bytes	n/a		
AddressH, AddressM, AddressL, Value (Total of 80 addresses and values are supported)	4bytes	n/a	0x00, 0x00, 0x00, 0x00 (default) Up to 80(address, value) pairs may be utilized. Zero in all three address bytes indicates the end	For development only.

1. Values for Total Polling duration for Scheduled Transmission and Master initial implant listen duration and (**Table 129 (PARAMETERS FROM EEPROM)** should be the same.

2. This parameter has a resolution of 5ms and a size of 1 byte if polling service request is obtained following a local wakeup packet.
3. “Wait for Open packet after transmitting” timer is started/re-started before medical event packet is sent on each channel. The timer is cancelled when a valid open request is received for Tel C. The timer is cancelled when the last open response packet has been transmitted for Tel M. Use 35ms or bigger for TelC. Use 55ms or bigger for TelM (48Kbps is the out-of-session default data rate for Tel M) to cover the time of transmission one medical event packet and exchanging an open request packet and open response packets.
4. Example: If Sniff Interval = 2 Sec, then for a Wrap Up Interval of 12 hours, set this field to 21600 = (60 Sec*60 Min*12 Hrs)/2 Sec
5. When the bit “send channel recovery indication to host” = 1, same channel recovery started indication, unspecified channel recovery started indication and channel recovery successful indication will be sent to the host.When the bit “send channel recovery indication to host” = 0, same channel recovery started indication, unspecified channel recovery started indication and channel recovery successful indication will not be sent to the host. Note the channel recovery failed indication will sent to the host whether the bit “send channel recovery indication to host” = 1 or not.
6. This value shall be zero (digital) for verification testing. The snf_abort_mask_test parameter in the P_Hardware1 EEPROM structure (see Document **A44088**, **Telemetry M RF Module EEPROM Specification**) should be set to 0x8F for a digital link, and to 0x00 for an RF Link.
7. If Telemetry scheme and waveform setup bit3-bit5 are all zero, MICS band will be used.
8. If the data rate is 190K for slave transmission, CPU fast speed = 5.7Mhz, CPU slow speed = 3.8Mhz.
9. The range is shown for bits 0 - 2. The hardware uses bits 3 and 4. These should be set to zero. Bits 5-7 are unused.
10. These values are only used if “power reduction during security key transfer” is set to a value of “3” (user supplied value). These values are not used otherwise.

TABLE 125: SNIFF LIST FORMAT

BYTE	VALUE	NOTES
1	Bit 0: 0 = leave sniff list unchanged 1 = load new sniff list Bits 1-7:Unused **Default = 0**	Bytes 2 through 34 only apply if bit 0 of this byte is equal to 1
2	Bit 0: 0 = Configure for Single Channel sniffs 1 = Configure for Triad sniffs Bits 1-7:Unused **Default = 0**	Configures the bp_triad_en bit of register REG_BP_WU_CTRL0. Refer to the Mozart Programmer’s Manual, 123970, for more information on Triad and Single channel sniffing.
3-34	Bits 0-4: center channel number Channels 1-10 = MICS Channels 11-20 = Lower MEDS Channels 21-30 = Upper MEDS **Note:** 0x00 - skip sniff, 0x1F - stop sniff (see footnote for more details) Bit-5: right channel enable (center channel + 1) Bit-6: center channel enable Bit-7: left channel enable (center channel – 1) **Default = 0**	Each of the bytes 3 through 34 are copied to Registers REG_BP_SNF_LIST_0 through REG_BP_SNF_LIST_31, respectively(1). Refer to the Mozart Programmer’s Manual, 123970, for more information on the Sniff List registers.
35	Bit 0: 0 = leave band ctrl unchanged 1 = load band ctrl Bits 1-7:Unused **Default = 0**	Byte 36 only applies if bit 0 of this byte is equal to 1

BYTE	VALUE	NOTES
36	Bit 0: 0 = Narrow band for MICS 1 = Wide band for MICS Bits 1-7:Unused **Default = 0**	Band Control applies to CCA, Sniff, and Transmit of Wakeup Packets. It also applies to MICS only.

1. For example, if single channel sniffs on channels 1-10 are desired to be in the list, the following entries must be contained in the table: 0x41, 0x42, 0x43, 0x44, 0x45, 0x46, 0x47, 0x48, 0x49, 0x4A, 0xFF. Note, 0xFF is used to indicate the end of a list. If triad sniffs on channels 11-30 are desired, sniffing 10 channels per sniff, the following entries could be used in the table: 0xEC, 0xEF, 0xF2, 0x54, 0x00, 0xF6, 0xF9, 0xFC, 0x5E, 0xFF. Note that 0x00 is used to indicate a stop or break between sniffs; the next sniff will continue at the following entry.

7.11 SLAVE POLLING REQUEST TABLE

The slave polling request size can vary from 1 byte to 137bytes. All the values will be uplinked in five discover name response packets. See **Table 126 (SLAVE POLLING REQUEST)**.

TABLE 126: SLAVE POLLING REQUEST

Parameter	#Bytes	VALUES
Polling Request	1-137bytes(1)	[0x00,0xFF]

1. 137 bytes maximum = (5 packets*30 bytes/packet) - 6 bytes master ID - 6 bytes slave ID - 1 byte proximity status

Table 112 (DISCOVER NAME SLAVE RESPONSE FORMAT) shows an example for the polling request info field.

7.12 SUMMARY FOR INDICATION AND REQUEST

Following is the summary of the indications which are generated by telemetry to the host See **Table 127 (INDICATION SUMMARY)**

TABLE 127: INDICATION SUMMARY

INDICATION	SLAVE	MASTER
Ready	0x20	0x20
Polling	0x21	**n/a**
Discover	0x22	0xA2
Open	0x23	**n/a**
Data	0x24	0xA4
Close	0x25	**n/a**
Shutdown	0x26	0x26
Channel Recovery	0x27	0xA7
Medical Event	0x28	0xA8
First req. pkt sent	**n/a**	0xAA
Wakeup	0x2B	**n/a**
Emergency	**n/a**	0xAC
Network	0x2D	0xAD
EE Write	0x2F	**n/a**
EEPROM read failed	0x30	0x30
Fault	0x33	0x33
Trim Fail	0x34	0x34
Telemetry Ceased	0x35	0x35
Proximity Security	0x36	**n/a**
Parameter Corruption	0x37	0x37
Memory Test Failed	0x38	0x38
Slave Received Mozart Data	0x39	**n/a**
Network complete	**n/a**	0xBA

Following is the summary of service requests which are generated by the host to telemetry See **Table 128 (SERVICE REQUEST SUMMARY)**

TABLE 128: SERVICE REQUEST SUMMARY

SERVICE REQUESTS	SLAVE	MASTER
Initialize	0x01	0x81
Discover	**n/a**	0x82
Open	**n/a**	0x83
Data	0x04	0x84

TABLE 128: SERVICE REQUEST SUMMARY

SERVICE REQUESTS	SLAVE	MASTER
Medical Event	0x0a[1]	0x85
Emergency	**n/a**	0x86
Close	**n/a**	0x87
Diagnostic	**n/a**	0x89
Polling	0x0A	**n/a**
Raw mode	**n/a**	0x8B
Network	0x0C	0x8C
Security	0x0D	0x8D
Power inhibit	0x0E	**n/a**
Shutdown	0x0F	0x8F

1. Medical Event Requests are processed via the Polling Service.

7.13 MASTER/SLAVE EEPROM PARAMETERS

The sample configuration values from EEPROM in **Table 129 (PARAMETERS FROM EEPROM)** can be adjusted to enhance the performance of the system. These values are contained in the P_CONFIG_STRUCTURE structure. See below for the suggested values.

TABLE 129: PARAMETERS FROM EEPROM

PARAMETER	BYTES	LSB	VALUE	[MIN, MAX]
Unused	2 bytes			
Master Raw mode service duration	1 byte	31.25 ms	0x20 - 1 second	[0x20,0xFF] 1s - 7.968s
Unused	1 byte			
Master Same channel recovery duration (should be the same as Slave Same channel recovery duration See **Table 124 (SLAVE PARAMETERS FROM HOST)**)	1 byte	5 ms	0x58 - 440 ms	[0x24, 0xFF] 180ms - 1.275 s
unused	1 byte			
Master unspecified channel recovery duration	2 bytes	1 second	`0x1C20 -` 7200 seconds (2 hours)	[0x05, 0xFFFF]
Master Low energy mode	1 byte	n/a	1 - enable (default), 0 - disable,	
Master Low energy threshold N (After N consecutive filler master packets, start low energy mode). Apply only if Master Low energy mode is enabled.	1 byte	1 packet	0x08 - 8packets	[0x08, 0xFF]
Master Low energy negative correlation ratio N (Use negative correlation of the preamble on N-1 of N transmits. One out of N will have positive correlation.) Apply only if Master Low energy mode is enabled.	1 byte	n/a	5 - 1 positive, 4 negative	[0x05, 0xFF]
Unused	4 bytes			
Native mode duration	1 byte	5 ms	0x64 - 500ms	[0x64,0xFF]
Wakeup Sync type	1 byte	n/a	Bit 0 - Open, 0 - Async, 1 - Sync Bit 1 - Network 0 - Async, 1 - Sync	0x00 - default
Number of consecutive receive frames without a valid packet (telem C)	1 byte	1 packet	0x04 - 4 packets	[1,0xFF]
Number of bad consecutive received frames (telem C)(1)	1 byte	1 frame	0x02 - 2 frames	[1,0xFF]
Unused	5 bytes			
Master Log2 SNR filter size(2)	1 byte	n/a	0x03 (SNR filter length = 8)	[1,7]
Master BER enable	1 byte	n/a	0x01 - enable, 0x00 - disable (default),	For development only
Master BER byte pattern	1 byte	n/a	0xAA - default	
Master initial implant listen duration (Tel C only) (3)	2 bytes	1 second	0x001E - 30 seconds	[5, 0xFFFF]

TABLE 129: PARAMETERS FROM EEPROM

PARAMETER	BYTES	LSB	VALUE	[MIN, MAX]
config_flags1	1byte	n/a	0x02	bit0: 1 - Bypass AES (for testing only); 0 - no Bypass AES bit1: 1- Slave scan during polling, 0- Slave sniffing during polling bit2 -bit 7: unused
wakeupConfigAO - long Sniff Time Meds Max	3 bytes		0,0,0	[0.0xFFFFFF]- 0 - 671s (40us/bit)
wakeupConfigAO - long Sniff Time Mics Max	3 bytes		0,0,0	[0.0xFFFFFF]- 0 - 671s (40us/bit)
wakeupConfigAO - long Sniff Time Meds Min	3 bytes		0,0,0	[0.0xFFFFFF]- 0 - 671s (40us/bit)
wakeupConfigAO - long Sniff Time Mics Min	3 bytes		0,0,0	[0.0xFFFFFF]- 0 - 671s (40us/bit)
wakeupConfigAO - phase1 Threshold Offset Mics	1 byte		0	[0,0xFF]
wakeupConfigAO - phase1 Threshold Offset Meds	1 byte		0	[0,0xFF]
wakeupConfigAO - Phase1 Threshold Offset Increase Amount	1 byte		Bit3 - bit 0: Meds Bit7 - bit 4: Mics	[0-0xF] [0-0xF]
wakeupConfigAO - phase2 High Aborts Mics Max	2 bytes		0,0	[0,0xFFFF] 0-65535 aborts
wakeupConfigAO - phase2 High Aborts Meds Max	2 bytes		0,0	[0,0xFFFF] 0-65535 aborts
wakeupConfigAO - phase2 High Aborts Mics Min	2 bytes		0,0	[0,0xFFFF] 0-65535 aborts
wakeupConfigAO - phase2 High Aborts Meds Min	2 bytes		0,0	[0,0xFFFF] 0-65535 aborts
wakeupConfigAO - phase1 Threshold Offset Decrease Amount	1 byte		Bit3 - bit 0: Meds Bit7 - bit 4: Mics	[0-0xF] [0-0xF]
wakeupConfigAO - restricted Channel Removed Max	1 byte		Bit3 - bit 0: Meds Bit7 - bit 4: Mics	[0-0xF] [0-0xF]
wakeupConfigAO - lengthen Sniff Interval Max	3 bytes		0,0,0	[0,0x3FFFF] 0-256s (.97ms/bit)
wakeupConfigAO - lengthen Sniff Interval Increase Amount	3 bytes		0,0,0	[0,0x3FFFF] 0-256s (.97ms/bit)
wakeupConfigAO - phase1 Threshold Offset Course Increase Amount Meds	1 byte		0	[0,0xFF]
wakeupConfigAO - phase1 Threshold Offset Course Increase Amount Mics	1 byte		0	[0,0xFF]
wakeupConfigAO - phase1 Threshold Offset Course Increase Max Meds	1 byte		0	[0,0xFF]

TABLE 129: PARAMETERS FROM EEPROM

PARAMETER	BYTES	LSB	VALUE	[MIN, MAX]
wakeupConfigAO - phase1 Threshold Offset Course Increase Max Mics	1 byte		0	[0,0xFF]
wakeupConfigAO - interval Wrapup Count Shorten	2 bytes		0,0	[0,0xFF]
wakeupConfigAO - restricted Channel Event Count Max	1 byte		0	[0,0xFF]
process Error Reset Enabled	1 byte		0x01 - enable (default); 0x00 - disable	na
process Error Reset Threshold	1 bytes		1	[1,10]
Send Trim Fail Indication To Host	1 byte		0x01 - enable (default); 0x00 - disable	na
Disable On Timer	1 byte	1 second	0x06	[0x01, 0xFF]
Standby Timer	1 byte	1 second	0x02	[0x02,0xFF]
Timer between Shutdown Indication and Turn off 2V	1 byte	5 ms	0x50 - 400ms	For Development Only
CCA control - restricted channel	1 byte	n/a	bit3-bit0: MICS channels restricted 0 - No MICS channels restricted 1 - MICS channel 1 restricted 2 - MICS channel 2 restricted ..... A- MICS channel 10 restricted B - F: no channels restricted Bit 4: restrict MEDS channels bordering MICS 1 - restricted ch. 20 & 21 0 - no restrictions Bit7-bit5: unused	na
Default Downlink Native Mode Data Rate (For Tel M Out of Session only)	1 byte	0-380Kbps, 1-97Kbps, 2-190Kbps, 3-48Kbps	3 - 48 Kbps (default);	[0,0x03]
Default Uplink Native Mode Data Rate (For Tel M Out of Session only)	1 byte	0-380Kbps, 1-97Kbps, 2-190Kbps, 3-48Kbps	3 - 48 Kbps (default);	[0,0x03]
Slave channel listen Duration (Tel M only)	1 byte	5ms	0x16 - 110ms	[0x16, 0xFF] 110ms - 1.275 s
CCA stop on threshold value	1 byte		0(4)	[0, 0xff]
Open uplink packet count (Tel M Only)	1 byte	1 packet	2(5)	[1, 5]
Unused	30 byte	n/a	n/a	

1. A bad received frame has a preamble timeout and no valid packets.
2. Must be <= 7 otherwise it is a fault.
3. Values for Master initial implant listen duration and Total Polling duration for Scheduled Transmission (**Table 124 (SLAVE PARAMETERS FROM HOST)**) should be the same.
4. All channels will be accessed when the "CCA stop on threshold value" is zero.
5. The time to scan all channels is shortened when the open uplink packet count is smaller (~1.7 seconds when the count is five and the non-communication rate is 48 Kbps; ~1.0 seconds when the count is two at 48 Kbps). This can speed up unspecified channel recovery. Additionally it shortens the minimum power inhibit de-assertion duration.

7.14 SUMMARY OF FIRMWARE FAULTS

TABLE 130: SUMMARY OF FIRMWARE FAULTS

FIRMWARE FAULT RESET CODE	DESCRIPTION
0x10	Emergency service request size too large for block copy (failed service response should have occurred). Possible memory leak.
0x11	Bad external service request when cancelling a service request (possible memory leak)
0x12	Unused
0x13	Unused
0x14	Discover invalid indication status (possible memory leak)
0x15	Invalid master indication type (possible memory leak)
0x16	Invalid master pending indication type (possible memory leak)
0x17	Invalid subtype for master network data service request (failed service response should have occurred). Possible memory leak.
0x18	Unused
0x19	Unused
0x1a	Attempting to send a master indication bigger than 512 bytes on buffer 1 (possible memory leak)
0x1b	Attempting to send a master indication bigger than 512 bytes on buffer 2 (possible memory leak)
0x1c	Attempting to send a master indication bigger than 512 bytes on buffer 3 (possible memory leak)
0x1d	Attempting to save a master service request bigger than 512 bytes (possible memory leak)
0x1e	Unused
0x1f	Unused
0x20	Indication buffer 1 “busy” when not expected (attempting to give discover name indication). Development error.
0x21	Indication buffer 1 “busy” when not expected (attempting to give network summary indication to the master host). Development error.
0x22 - 0x2f	Unused
0x30	Invalid state for firmware clear channel assessment algorithm. Possible memory leak.
0x31	Attempting to set master RF channel to an illegal value. Possible memory leak.
0x32	Invalid clear channel assessment band (not MICS, MEDS, or both). Bad parameter from host initialize requestor memory leak.
0x33-0x3f	Unused
0x40	Unused
0x41	Invalid receive packet type when checking station ID (packet should have been ignored further upstream). Possible memory leak.
0x42	Invalid receive packet with good MAC. This error is only enabled during development (not in final product since MAC is not foolproof; noise could be detected with good MAC). Detects development errors.

TABLE 130: SUMMARY OF FIRMWARE FAULTS

FIRMWARE FAULT RESET CODE	DESCRIPTION
0x43	Invalid master pending transfer mode. Possible memory leak.
0x46	Receive packet response timeout (telemetry M, RF). The hardware should unconditionally give a receive complete interrupt when next frame is receive.
0x47	Receive packet response timeout (telemetry C, beyond first synch cycle). The hardware should unconditionally generate a receive complete interrupt in this case.
0x48	Invalid master power inhibit mode. Possible memory leak.
0x49-0x4f	Unused
0x50	Master receive packet response timeout in the communication state. The hardware should unconditionally give a receive complete interrupt
0x51-0x5f	Unused
0x60	Bad master packet when determining uplink packet count. Possible memory leak.
0x61-0x7f	Unused
0x80	Illegal state machine event detected. Development error or memory leak.
0x81	Too many events for the state machine queue. Development error.
0x82	AES block busy for too long (waiting to start session key computation). Hardware interface error.
0x83	AES block busy for too long (waiting to complete session key computation). Hardware interface error.
0x84	AES block busy for too long (waiting to complete random number computation, step 1). Hardware interface error. Also see 0x88.
0x85	AES block busy for too long (waiting to start random number computation, step 2). Hardware interface error.
0x86	AES block busy for too long (waiting to complete random number computation). Hardware interface error.
0x87	Invalid channel in received wakeup packet. This fault will not occur in the field. It is enabled only during development.
0x88	AES block busy for too long (waiting to start random number computation, step 1). Hardware interface error.
0x89	CPU error bit set when waiting for AES computation to complete. Hardware interface error.
0x8a	Neither CPU error nor CPU done set when waiting for AES computation to complete. Hardware interface error.
0x8b	AES block busy for too long (waiting to start secure wakeup MAC computation). Hardware interface error.
0x8c	AES block busy for too long (waiting to start keystream computation to encrypt/ decrypt secure wakeup user data). Hardware interface error.
0x8d	AES block busy for too long (waiting to complete secure wakeup MAC computation). Hardware interface error.
0x8e	AES block busy for too long (waiting to start keystream computation to encrypt/ decrypt secure wakeup user data). Hardware interface error.

TABLE 130: SUMMARY OF FIRMWARE FAULTS

FIRMWARE FAULT RESET CODE	DESCRIPTION
0x8f-0x91	Unused
0x92	Memory test failure (RAMware).
0x93	Parameter corruption
0x94	Power inhibit asserted by master (telemetry C or telemetry M). This code is also used for power inhibit asserted by a slave configured for telemetry C.
0x95	Power inhibit deasserted by master (telemetry C or telemetry M). This code is also used for power inhibit deasserted by a slave configured for telemetry C.
0x96-0x97	Unused
0x98	RF channel out of range. Possible memory leak.
0x99-0xb0	Unused
0xb1	Indication buffer 1 “busy” when not expected (attempting to give network indication to the slave host). Development error.
0xb2-0xdf	Unused
0xe0	Illegal opcode.
0xe1	Software interrupt.
0xe2	Unused
0xe3	Trap when ramware is disabled.
0xe4	Indication0 queue overflow. Host not keeping up.
0xe5	Unexpected non-maskable interrupt.
0xe6	Indication4 queue overflow. Host not keeping up.
0xe7	Read burnin for EEPROM failed.
0xe8	Interlock condition (attempting to turn off two volt supply when wakeup not disabled by host)

Requirements Sorted by Tag

List Of Tables

List Of Figures

Extracted Requirements

TLMC_MS_001 - PACKET CONTROL BYTES

Fig. 17 (Packet Control Bytes) defines the fields within the control bytes of a master packet.

TLMC_MS_003 - PACKET TYPES

Table 4 (PACKET TYPES) defines the permissible types of Telemetry M packet.

TLMM_MS_406 - EEPROM - NON-VOLATILE REGISTERS

When the CPU is enabled and the EEPROM is read successfully, the Mozart module shall update the non-volatile always on registers. Upon a reset, the Mozart module shall update the volatile registers.

[The data read from the EEPROM is copied to a specific location into the RAM **TABLE 122: STRUCTURES IN EEPROM**]

TLMM_MS_407 - EEPROM - LOAD RAMWARE APPLICATION

The Mozart module will load the RAMWARE application (if one exists) into RAM if there are no unrecoverable error reading the RAMWARE section of the EEPROM. Additionally it will begin executing the RAMWARE application.

TLMM_S_408 - EEPROM - WAKEUP DURING EEPROM READ

When CPU is enabled via wakeup then wakeup indication will be generated to the host upon the successful read of the EEPROM. For wakeup indication see **TABLE 93: WAKEUP INDICATION (SLAVE)**

TLMM_MS_262- READY INDICATION - INDICATION TO UPPER LAYER

The Ready indication will be generated to the Upper Layer when Data link layer is enabled by Host, or via Reset and the EEPROM has been successfully read. See **Table 5 (READY INDICATION (MASTER OR SLAVE))**

TLMM_MS_409 - EEPROM - FAILURE INDICATION

When the CPU is enabled (host dummy write or wakeup) data will be read from the EEPROM. The Mozart module will send an EEPROM read failure indication to the host if there are unrecoverable errors encountered while reading any of the following:

a. p_hdw1 & p_hdw2 - mirror of non-volatile always on registers
b. p_wakeup - mirror of non-volatile always on registers, and mirror of volatile always on registers, and firmware operating parameters
c. p_config - firmware operating parameters
d. p_ramware_control and RAMWARE code

See **Table 6 (EEPROM READ FAILED INDICATION (MASTER OR SLAVE))**.

[ROM defaults will be used if (a) - (c) do not survive intact.

The content read from the EEPROM is listed in appendix section 7.7]

TLMM_MS_410 - POWER INHIBIT DURING EEPROM OPERATION

During the EEPROM operation if power inhibit is invoked then EEPROM operation will be suspended.

[During an EEPROM operation if power inhibit is invoked by the host data link layer will complete the on going EEPROM structure operation. The operation on the remaining structure will resume upon the termination of power inhibit.

A timeout indication will be generated to the Upper Layer if the total duration of the EEPROM read operation lasts longer than 10 seconds. Upon a successful read of the EEPROM transition will be made to DISABLE CPU ON state where disable ON timer will be started, power inhibit interrupts will be unmasked. Wakeup indication “Wakeup Complete”, containing the wake-up packet contents, shall be sent to the upper layer. (see **Table 93 (WAKEUP INDICATION (SLAVE))** if slave is waiting for the “last packet last block” wakeup packet (after the “first packet last block” wakeup packet has enabled the CPU).

EEPROM write operation will occur in Standby state. For detail about the EEPROM write operation, see **7.6.2 Mozart module Memory Read and Write**. A timeout indication (if EEPROM write command from Master) or response (if EEPROM write command from slave host) will be generated to the Upper Layer if the total duration of the EEPROM write operation lasts longer than Standby Timer because Power Inhibit takes too long (For “standby timer” see **Table 129 (PARAMETERS FROM EEPROM)**).

Power monitor bit shall only be set during EEPROM operation.]

TLMC_MS_004 - STATION ID

The ID is a 6-byte identifier which uniquely identifies each station. A valid ID shall have non-zero value.

TLMM_MS_426 - DEFAULT NATIVE MODE RATE

The master and slave shall transmit and receive native mode packets using a default rate before two way communication is established.

[See “default downlink native mode rate” and “default uplink native mode rate” in the p_config block of the EEPROM. Note that the open downlink packet will communicate the downlink and uplink rate for two way communication

Note that the slave’s default data rates come from EEPROM until a wakeup packet is received.]

TLMC_MS_006 - EXTERNAL SERVICE REQUEST PROCESSING

External service requests will be accepted in a serialized manner, with one exception: Emergency can interrupt any service in progress.

TLMC_M_237 - SERVICE DURATION TOLERANCE

All service requests are issued with a service duration. The tolerance for all service durations (measured at the application) is [-1 second, +1 second].

TLMM_MS_404 - FAULT INDICATION - INDICATION TO UPPER LAYER

The Fault indication will be generated to the Upper Layer when a fault reset occurs. See **Table 12 (FAULT INDICATION (MASTER OR SLAVE))**

TLMM_S_263 - SHUTDOWN INDICATION - INDICATION TO UPPER LAYER

When the standby timer, ephemeral BAN key timer, and security (key exchange authorization) timer have all expired, a shutdown indication will be generated to the upper layer, a transition will be made to the disable state. See **Table 13 (SHUTDOWN INDICATION (SLAVE, MASTER))**

[Then the slave will return to synchronous receptor wakeup operation. If the wakeup disable bit is set to 1 in the initialization request firmware does not set the wakeup enable bit bp_wakeup_en bit in REG_BP_WU_CTRL0 register. See requirement **TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP**.

Finally, the slave will wait for “Timer between Shutdown Indication and Turn off 2V” duration and then turn off the 2 volt supply. See **Table 129 (PARAMETERS FROM EEPROM)** for “Timer between Shutdown Indication and Turn off 2V”.]

TLMM_MS_413 - SHUTDOWN INDICATION - DISABLE TIMER EXPIRES

Upon the expiration of the disable ON timer shutdown indication will be generated to the upper layer and 2 volt supply will be turned off.

[A shutdown indication can occur when Mozart is enabled via reset, host dummy write or wakeup and the host does not send a subsequent initialize request.

Note that the 2 volt supply is not turned off until the “Timer between Shutdown Indication and Turn off 2V” (see **Table 129 (PARAMETERS FROM EEPROM)** expires after the expiration of the disable ON timer. In addition, during the “ Timer between Shutdown Indication and Turn off 2V” that follows the disable ON timer expiration, the mozart will not respond to wakeups.]

TLMM_MS_428 - INITIALIZE - UPPER LAYER TO DATA LINK LAYER PARAMETERS

When invoking the Initialize service request, the Upper Layer will provide information as specified in **Table 14 (INITIALIZE SERVICE REQUEST (MASTER OR SLAVE))**.

TLMM_MS_265 - INITIALIZE ADDRESS WITH VALUES

Upon receiving the initialization data, firmware will copy each value to the specified address up to, but not including, an address value of 0. Format for master see **Table 123 (Master parameters from host)** and for slave see **Table 124 (SLAVE PARAMETERS FROM HOST)**.

[Initialization data are address-data pairs contained in the parameters received from the host. This could be used to write any value to the specified address.]

TLMM_M_266 - MASTER INITIALIZE SERVICE REQUEST FROM UPPER LAYER

When invoking an Initialize service request, the Upper Layer in the master will provide information contained in the configuration file (see **Table 123 (Master parameters from host)**).

TLMM_M_267 - MASTER INITIALIZE SERVICE RESPONSE TO UPPER LAYER

An Initialize service response will be generated to the Upper Layer after the initialize service request is processed.

[The content of the response to the Upper Layer is specified in **Table 15 (INITIALIZE SERVICE RESPONSE (MASTER))**]

TLMC_S_008 - SLAVE INITIALIZE SERVICE REQUEST FROM UPPER LAYER

When invoking an initialize service request, the Upper Layer in the slave will provide information as specified in **Table 124 (SLAVE PARAMETERS FROM HOST)**

TLMC_S_009 - INITIALIZE SERVICE RESPONSE TO UPPER LAYER.

An initialize service response will be generated to the Upper layer after the initialize service request is processed, and a standby timer is started.

[The content of the response to the Upper Layer is specified in **Table 16 (INITIALIZE SERVICE RESPONSE (SLAVE))**.]

TLMM_S_241 - POLLING ENABLING CRITERIA

If the polling feature is enabled then it will be activated:

1. Pre-session – when a polling service request is received from the upper layer (i.e., valid wakeup packet received),
2. Post-session – when channel recovery is terminated unsuccessfully and the patient name service request is received from the upper layer.

3. Post-session – when a close request packet is received and the patient name service request is received from the upper layer.

The slave will use the polling duration in the open packet

[If session is not opened then default polling time value from the initialization data provided by the host will be used.]

TLMM_S_035 - POLLING - SERVICE REQUEST FROM UPPER LAYER

When invoking a Polling service request before a session has been established, use the Total Initial Polling Duration time provided by the host as part of initialization data when initialization service request is invoked.

[The polling timer can also be programmed as via Open master packet (see **5.2.10 OPEN SESSION**), this timer value specifies the duration of the polling period.]

TLMM_S_256 - POLLING - SERVICE RESPONSE TO UPPER LAYER

When poll request is received, the Data Link Layer will validate the Poll service request.

[The content of the response to the Upper Layer is specified in **Table 18 (POLLING SERVICE RESPONSE (SLAVE))**]

TLMM_S_432 - POLLING - HOST REQUIRES SECURE BUT LOCAL WAKEUP PACKET DOES NOT INCLUDE SECURITY

If slave’s “authorized to talk unsecurely” parameter is false as provided by the host, and a valid local wakeup packet with “security user data/network packet” set false is received, when polling request (upon Tel M) is received, slave will send an invalid polling response to upper layer.

[Host authorizes Tel M to talk secure/unsecure as part of initialization.]

TLMM_S_240 - POLLING INITIATION

Once the polling feature (scanning off) is activated Pre-session the slave will start the polling timer, start the native mode timer and start listening for a native mode packet. The native mode listen channel will come from the recently received wakeup packet.

Once the polling feature (scanning off) is activated Post-session the slave will start the polling timer and start listening for wakeup packets.

When the polling feature (scanning on) is activated (pre or post session) the slave will start the polling timer, start the channel listening timer and start listening for a native mode packet on a channel.

[For the polling (scanning off) case, if no wakeup packet has been received (i.e. Poll upon POR), the listen channel will be chosen randomly.

For the polling (scanning on, pre session) case, if Unicast wakeup packet has been received, the first listen channel will be the channel number indicated in the Unicast wakeup packet; if Global wakeup packet has been received, the first listen channel will be a randomly selected channel number.

For the polling (scanning on, post session) case, the first listen channel will be the channel number used in the last session.

Note that the slave’s polling mode (scanning on or scanning off) comes from EEPROM until a wakeup packet is received.]

TLMC_S_252 - POLLING - RESTART NATIVE MODE TIMER OR CHANNEL LISTEN TIMER

The slave will do the following upon the receipt of a valid discover ID, or discover name packet:

1. Restart the native mode timer for the polling (scanning off) case. Restart the channel listen timer for the polling (scanning on) case.
2. Restart the polling timer using the “last polling” value for the polling. “Last polling” is the latest polling duration via the initialize service or polling duration via downlink open packet.

TLMC_S_037 - POLLING - NATIVE MODE TIMEOUT/ CHANNEL LISTEN TIMEOUT

The slave will do the following upon a native mode timeout/ channel listening timeout for the scanning off/ scanning on cases respectively:

1. Begin listening for wakeup for the scanning off case. Utilize continual asynchronous receptor wakeup listens to do this. It will continue to listen for wakeup until a valid wakeup packet is received, or the polling timer expires.
2. Select a new channel for (native mode) listening for the scanning on case. Note that this is done using two phases:
    a. Phase 1 - random channel listening - the slave will randomly select a channel and will restart the channel listening time and listen for a valid packet from the master. If no valid open packet has been received and the service duration timer is still active, the slave will switch to the phase 2 approach.
    b. Phase 2 - sequential channel listening - the slave will sequentially select the next channel and will restart the channel listening time and listen for a valid packet from the master. When every channel has been selected once the slave will switch to the Phase 1 approach.

[Note that the slave will scan on the bands specified in **Table 124 (SLAVE PARAMETERS FROM HOST)**. See the parameter called “telemetry scheme and waveform setup”.]

TLMM_S_268 - POLLING - VALID WAKEUP PACKET

The slave will start the native mode timer and begin listening for a valid native mode packet if a valid wakeup packet is received during polling (scanning off).

TLMC_S_038 - POLLING - TERMINATION

The polling will terminate when Polling duration timer expires, or when a valid Open request is received. The slave shall discard all the master IDs stored during the polling operation. The slave will maintain the ID of the master with which communication is established.

TLMC_S_039 - POLLING INDICATION TO UPPER LAYER

A polling timeout event indication to the Upper Layer will be generated when the polling timer expires. See **TABLE 19: POLLING INDICATION (SLAVE)**.

TLMC_M_010 - DISCOVER SERVICE REQUEST FROM UPPER LAYER

When invoking a Discover service request, the Upper Layer will provide information as specified in **Table 20 (DISCOVER SERVICE REQUEST (MASTER))**.

TLMC_M_011 - VALID SLAVE ID

The valid slave IDs with wild card specification are defined in **Table 21 (VALID SLAVE ID WITH WILD CARD)**.

TLMC_M_024 - DISCOVER FULL - RESPONSE TO UPPER LAYER

A Discover service response to the Upper Layer will be generated when a Discover Full service request is received.

[The content of the response to the Upper Layer is specified in **Table 22 (DISCOVER SERVICE RESPONSE (MASTER))**.]

TLMC_MS_012 - DISCOVER FULL - DISCOVER ID PACKET FORMAT

The Discover ID request packet has the layout indicated in **Fig. 30 (Discover ID Request Packet)**.

TLMC_M_013 - DISCOVER FULL - DISCOVER ID REQUEST PACKET

When Discover service request is invoked by the Upper Layer, start a service duration timer and transmit a Discover ID request packet.

[A service duration timer will be started; select the least interfered channel (takes approximately 275 +/ 50 msec); start the Listen Before Talk (LBT) timer; perform instigator wireless wakeup, send a “first request packet sent” indication to the Upper Layer (see **Table 23**); start the native mode timer (if the slave does not scan during polling), transmit a discover ID request packet. Note the 'first request packet sent' indication is configurable. See the parameter called 'transmit notification enabled' in **Table 123 (Master parameters from host)**

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMM_M_269 - DISCOVER FULL - DISCOVER ID RESPONSE WINDOW

The master shall continue listening for more slave responses upon hearing any response (valid or not) provided the device ID response window is not complete.

TLMM_M_270 - DISCOVER FULL - RESTART NATIVE MODE TIMER

The master will restart the native mode timer whenever it receives a valid discover ID (or discover name) packet from the slave (if the slave does not scan during polling).

[Note that a channel is selected and instigator wakeup is performed when the native mode timer expires. Once instigator wakeup is complete, discover ID is transmitted again. See **Fig. 31 (Simplified master discover full state machine)**

Note that the native mode timer is not used when the slave scans during polling.]

TLMC_M_014 - DISCOVER FULL - RETRY DISCOVER ID REQUEST

If no valid Discover ID slave response packet is received within the discover ID response window, the master shall transmit another Discover ID request packet. The Discover ID retry will continue until:

1. A valid Discover ID response packet is received, or
2. The service duration timer expires, or
3. The native mode timer expires, or
4. The listen before talk timer expires.

[In order to retransmit the Discover ID request due to no slave response from a previous discover ID request data link layer will switch to a different antenna and will retransmit the discover ID request.]

TLMC_S_015 - DISCOVER FULL - DISCOVER ID RESPONSE

Upon receiving a valid Discover ID request packet the slave will perform as follows:

1. If the master packet contains an master ID which the slave received previously from a valid Discover Name request packet, then slave response will not be generated.
2. If the master packet contains a master ID which the slave has not received previously from a valid Discover Name request packet, the slave will transmit a Discover ID response packet in the first slave timeslot. It will transmit a second Discover

ID response packet in slave timeslot 2, 3,..., or 7. This timeslot will be randomly chosen (to reduce collisions). Note that the second response is not used if the slave scans during polling.

The slave will restart the native mode timer to allow the receipt of another master packet if the slave does not scan during polling.

TLMC_M_016 - DISCOVER FULL - LBT OR NATIVE MODE TIMEOUT

While the service duration timer is active and LBT timer expires or the native mode timer expires, the master will reselect the least interfered channel, toggle the antenna, perform instigator wireless wakeup, and transmit a discover ID packet.

[Note that the master will perform clear channel assessment before and after transmitting wakeup (if the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMC_MS_017 - DISCOVER FULL - DISCOVER ID RESPONSE PACKET FORMAT

The Discover ID response packet has the layout indicated in **Fig. 32 (Discover ID Response Packet)**.

TLMC_MS_018 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET FORMAT

The Discover Name request packet has the following layout indicated in **Fig. 33 (Discover Name Request Packet)**.

TLMC_M_019 - DISCOVER FULL - DISCOVER NAME REQUEST PACKET

After the receipt of a valid Discover ID response, the master will transmit a Discover Name request packet.

[The master will keep track of all the unique slave IDs, the discover name request will be transmitted on the same antenna as was used for Discover ID.

If multiple slave respond during the discover ID response window, the master will sequentially transmit a name request (and wait for a name response) to each slave.]

TLMC_M_020 - DISCOVER FULL - RETRY DISCOVER NAME REQUEST

If no valid slave Discover Name response is received, the master will retry the Discover Name request packet until the discover name retry timer expires. Note that this timer was started when the initial discover name downlink was sent. Once the timer expires, the master will continue the Discover service by transmitting a new Discover ID request packet.

[The retry attempt of the discover name request will use the same antenna as was used for the first discover name request. If no response is received from the retry attempts then discover ID will be transmitted using a different antenna.]

TLMC_S_021 - DISCOVER FULL - DISCOVER NAME RESPONSE

After the receipt of a valid master Discover Name request packet, the slave shall do the following:

1. Transmit the Discover Name response packet(s). see **Table 112 (DISCOVER NAME SLAVE RESPONSE FORMAT)** for details. The proximity status shall be the byte immediately following the slave ID see **Fig. 35 (Proximity status type byte)**.
2. The slave shall store the master station ID from the Discover Name packet for the collision avoidance duration indicated in the Discover Name request packet. The slave shall not transmit any Discover ID response for the subsequent Discover ID request from this master for the duration of the collision avoidance timer.
3. Send a discover name indication (sub-type of 0x01) to the host (see **Table 24 (DISCOVER INDICATION (SLAVE))**). This is an implicit security indication. The indication to the host will not be sent more than once a second.

[The slave shall update its current remaining Scan duration with that from the Discover Name request packet (discover ID collision avoidance duration) if the current remaining Scan duration is shorter. Also extend the listen time by 50ms if the remaining listen time is less than 50ms. The size of the collision avoidance buffer is ten, a circular buffer is used to keep track of number of masters issuing Discover full]

TLMM_S_271 - DISCOVER FULL - CANCEL COLLISION AVOIDANCE TIMER

The collision avoidance timer will be canceled upon receiving of an “authorize BAN key exchange service request or update proximity status service request”.

TLMC_MS_022 - DISCOVER FULL - DISCOVER NAME RESPONSE PACKET FORMATS

During the slave response of Discover Name the slave will use one or more of the packet formats as shown in **Fig. 34 (Discover Name Response Packet)**.

TLMM_M_272 - DISCOVER FULL - SUCCESSFUL NAME

The master will send a discover indication to the upper layer (with a status of success) if it successfully obtains the slave’s name. The format of the indication is shown below:

TLMC_S_023 - DISCOVER FULL - RESPONSE PARTITIONING

The slave will partition the Discover Name response packet(s) using **Table 26 (DISCOVER NAME RESPONSE FRAGMENTATION)**.

[Note that if a Discover Name (ensuing, not full) packet is needed, it must be the last packet of the transaction.]

TLMM_M_273 - DISCOVER FULL - TIMEOUT INDICATION

The master will send a discover indication (with a status of timeout) when the discover full service duration expires. See **Table 25 (DISCOVER INDICATION (MASTER))**.

TLMM_M_274 - DISCOVER FULL - STOP

The master will stop the discover full service and send a service response if it receives a stop discover full service request. Additionally it will send a discover indication (status of cancelled). See **Table 25 (DISCOVER INDICATION (MASTER))**.

TLMC_M_040 - OPEN - UPPER LAYER PARAMETERS

When invoking an Open service request, the Upper Layer will provide information as specified in **Table 27 (OPEN SERVICE REQUEST (MASTER))**.

TLMM_MS_281 - OPEN - REQUEST PACKET FORMAT

The open request packet has the format indicated in **Fig. 40 (Open Secure/Unsecure Request Packet)**

The “normal values” for the slave’s unspecified channel recovery (UCR) duration and slave’s polling duration will come from the initialize service request from the master host (**Table 123 (Master parameters from host)**).

TLMM_MS_282 - OPEN - RESPONSE PACKET FORMAT

The open response packet has the format indicated in **Fig. 42 (Open Secure/Unsecure Response Packet)**

[Nominal packet count and Max. throughput packet count in the open response packet is to support verification testing when Protocol Analyzer is used.]

TLMM_M_283 - OPEN - TRANSMIT MASTER PACKET

Upon receiving an open service request, the master will start a service duration timer and transmit the open request packet.

[Start the service duration timer; select the least interfered channel (takes approximately 275 +/- 50 msec); start the Listen Before Talk (LBT) timer; perform instigator wireless wakeup (including unsecured or secured user data if any exists); transmit the open request packet (including the 8 byte random number used to compute a session key); send a “first request packet sent” event indication to the Upper Layer (see **Table 23 (FIRST REQUEST PACKET SENT INDICATION (MASTER))**).

Note that instigator wireless wakeup is suppressed if the master knows the slave is scanning. Time now, time of last uplink, and remaining polling duration at the time of the last uplink are used to determine this.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup is not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

[The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**). Note that the master random number (Rm) can be re-used for the entire open secure service. When communication is established and subsequently lost, a new random number (Rm) must be generated during open and session recovery.]

TLMM_M_284 - OPEN - NO VALID SLAVE RESPONSE RETRY REQUEST

If no response is received, the master will resend the open request till the Service Duration timer is expired.

[If no valid open response packet is received, the master will perform the following:

1. Transmit an open request packet on the alternate antenna.

This process shall be continued until a valid open response packet is received, a Service Duration Timeout occurs, or a Listen Before Talk timer expires.]

TLMM_M_285 - OPEN - LISTEN BEFORE TALK TIMER OPERATIONS

If the listen before talk timer expires during an open service, the master will transition to unspecified channel recovery.

[As part of unspecified channel recovery following actions are performed: Re-select the least interfered channel for communication, Start the Listen Before Talk timer, toggle the antenna, perform instigator wireless wakeup, and transmit an open request. Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMC_S_286 - OPEN - RESPONSE TO A VALID OPEN REQUEST

Upon receiving a valid open request packet, during polling operation, the slave will fill up the transmit frame with N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**. Slave will update the unspecified channel recovery duration and polling duration using the values in the open request packet.

Upon completion of uplinking all the open response packets, the slave will terminate the polling operation; It will then transition to the Comm State. An open indication will be sent to the upper layer (see indication **Table 28 (OPEN INDICATION**

(SLAVE))). Additionally the slave will transition to the desired downlink rate and uplink rate specified in the open downlink packet. Finally the slave will reset (to zero) the waveform sequence number.

[Note that the open response packets will include the actual UCR duration and actual polling duration and an 8 byte slave random number (Rs).

Slave will restart the native mode time, channel listen time or the medical event transmit time if the timer (native mode time, channel listen time or the medical event transmit time) times out during the uplinking of the open response packets.]

TLMM_S_287 - OPEN - NO RESPONSE TO A VALID OPEN REQUEST

When slave's "authorized to talk unsecurely" parameter is false as provided by the host, and a valid open "unsecure" request is received then slave will not transmit any response packets.

[When this happens, slave will continue polling.]

[Host authorizes Tel M to open and/or unsecure session as part of initialization]

TLMM_S_288 - OPEN - RESPONSE TO A VALID OPEN UNSECURE REQUEST

When slave is authorized to open an "unsecure" session by the host, and a valid open "secure or unsecure" request is received then slave will accordingly transmit the response packets.

[Host authorizes Tel M to open secure and/or unsecure session as part of initialization]

TLMM_S_289 - OPEN - SESSION KEY COMPUTATION UPON VALID OPEN REQUEST

Upon the receipt of a valid open request the slave will compute a session key using the eight byte master random number (Rm), the eight byte slave random number (Rs) and one of the K values using the block cipher: the BAN key (KBAN), ephemeral BAN key (KEBAN), a fixed known token value (Ktoken). See **Fig. 44 (Computing a session key)**.

[The eight byte master random number (Rm), slave random number (Rs) and BAN Key (KBAN) or Ephemeral KEBAN will be loaded into the appropriate registers of the AES hardware block to generate the Session Key KSess. For an unsecure session - A fixed known token value (KToken) is loaded to the AES hardware block instead of the KBAN or KEBAN key(s)]

TLMM_S_290 - OPEN - RESPONSE SUCCESSFUL

Open successful status will be transmitted in the open response packet if:

1. Slave has the appropriate Ban Key as request by the master, or
2. Slave is authorized to communication unsecurely and open request is for unsecure open.

TLMM_S_292 -OPEN - ENABLE SECURITY BLOCK

Upon transmitting the open response the slave will enable the hardware's security block (security register to on). The slave will start the block cipher for message integrity, privacy, and freshness checking. This will be done by writing the session key to the block cipher key register. Additionally, the slave will initialize the block cipher nonce counter using the lower 5 bytes of the master random number (Rm). The most significant byte of the nonce counter is set to zero to allow a very long session because once the nonce reaches its max value it stops.

[Slave enables the hardware security block even if an unsecured session is opened - See **Fig. 45 (Open - block cipher initialization)**.]

[Note: An open indication is sent to the upper layer indicating that a session is opened; continue the Synchronous Mode operation and wait for a valid master packet. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).]

TLMM_S_293 - OPEN - UNSUCCESSFUL - MISSING KEY

The slave will send an unsuccessful open response if the following occurs:

1. The slave receives a valid open request and the open type is secured, and
2. The slave does not have the KBAN or KEBAN filled that was specified in the open type byte (See **Fig. 41 (Open type byte (Request))**

The slave will set the open response type byte to unsuccessful. See **Fig. 42 (Open Secure/Unsecure Response Packet)**.

TLMM_M_294 - OPEN - A VALID SLAVE RESPONSE

If a valid open response packet is received, the master will perform the following:

1. Return a successful open service response to the Upper Layer.
2. Transition to the Comm State.
3. Compute a session key using the eight byte master random number (Rm), the eight byte slave random number (Rs) and the BAN key (KBAN) using the block cipher. See **Fig. 44 (Computing a session key)**.
4. Transition to the desired downlink rate and uplink rate. Note that these originally came from **Table 123 (Master parameters from host)**.
5. Enable the hardware's security block (security register to on).
6. Start the block cipher for message integrity, privacy and freshness checking. This is done by writing the block cipher key register with the session key. Additionally, the nonce counter is initialized using the lower 5 bytes of the master random

number (Rm). The most significant byte of the nonce counter is set to zero to allow a very long session because once the nonce reaches its max value it stops. See **Fig. 45 (Open - block cipher initialization)**.

7. Clear the wakeup internal service user data buffer to prevent retransmission of stale user data.
8. Reset (to zero) the expected waveform sequence number.

[Note that the recently exchanged ephemeral BAN key will be used (to calculate a session key) if the ephemeral key is still valid. The ephemeral BAN key is valid until the ephemeral BAN key timer expires. Note: if open type is unsecure, Ktoken is used.]

TLMM_M_295 - OPEN - RESPONSE WITHOUT KBAN OR KEBAN

If a valid open secure response packets is received with unsuccessful status, then master will stop transmitting open requests.

[The master will transition to the standby state if the open uplink indicates the slave lacks the appropriate key.]

TLMM_M_296 - OPEN - RESPONSE TO UPPER LAYER

An open service response will be generated to the Upper Layer when the open service is completed.

[The content of the response to the Upper Layer is specified in **Table 29 (OPEN SERVICE RESPONSE (MASTER))**.]

TLMC_MS_049 - DATA - UPPER LAYER TO DATA LINK LAYER PARAMETERS

When invoking the Data service request, the master Upper Layer will provide information as specified in **Table 30 (DATA SERVICE REQUEST (MASTER OR SLAVE))**.

TLMC_MS_050 - DATA - MASTER REQUEST PACKET FORMAT(S)

The master data request has the format shown in **Fig. 50 (Master data packet formats)**

TLMC_M_051 - DATA - TRANSMISSION OF MASTER DATA PACKET

When the Data service request is invoked, the master will start the reply timer, format the user data into data fragment(s)/packet(s) and transmit the data packet(s).

[Initiate the Internal Service Reset if the reset parameter is True, Add the master ID in the first fragment, transmit the packet(s), send a first request packet sent event indication to the Upper Layer after sending the first data packet (see **Table 23**)]

TLMC_S_052 - DATA - ACKNOWLEDGEMENT OF MASTER DATA PACKET

The slave shall set the ACK bit in the slave packet(s) when it receives a master packet with good message authentication code (MAC). The ACK bit in the slave packet(s) will not be set if the master packet was received with a bad MAC.

[Note: A packet with a bad sequence number will also be ACKed and the data will be ignored.]

[See requirement **TLMC_S_055 - DATA - MASTER PACKETS WITH DUPLICATE SEQUENCE NUMBERS**.]

TLMC_M_053 - DATA - PACKETIZATION OF THE MASTER DATA

The master will fragment the user data into packets per **Table 31 (PACKETIZATION OF MASTER DATA)**.

[Note that if a data (ensuing, not full) packet is needed, it must be transmitted as the last packet of the transaction.]

[For example, if the size of the user data is 53 bytes, the transaction is fragmented as follows:

a. One data (first, full) packet - 30 bytes (6 bytes master ID, and 24 bytes of user data)

c. One data (ensuing, not full) packet - 29 bytes of user data]

TLMC_M_054 - DATA - RETRANSMISSION OF MASTER PACKET - NO ACK RECEIVED

After transmitting a data packet, if no slave packet with the ACK bit set is received, that master data packet shall be retransmitted. Only after an slave packet with the ACK bit set is received, the next data packet shall be transmitted.

[The master data service uses the Stop and Wait algorithm.]

TLMC_S_055 - DATA - MASTER PACKETS WITH DUPLICATE SEQUENCE NUMBERS

If the master data request packet has the same sequence number as the previous master data request packet, the slave will toss the data packet and retransmit unacknowledged Memory Data packets, if any exist.

TLMC_S_056 - DATA - VALIDATION OF MASTER DATA REQUEST

The slave shall process the master data packet if it is valid per **Table 10 (MASTER PACKET VALIDATION)**.

TLMC_S_057 - DATA - MASTER PACKETS INTO TRANSACTION

The master data packets are assembled into a transaction and are passed on to the slave’s upper layer (application).

[The data indication to the slave Upper Layer will be generated by the slave data link layer, after it successfully receives and assembles all the data packets of a given transaction. The content of the Indication to the Upper layer is defined in **Table 32 (DATA INDICATION (MASTER/SLAVE))**]

TLMC_M_058 - DATA - RESPONSE TO UPPER LAYER

When slave’s upper layer response is completely received or a reply timeout occurs then a data service response to the master upper layer will be generated. No slave upper layer response is expected when the master transaction is not final. In this case the data service response is generated when all the master data packets have been acknowledged by the slave.

[The content of the response to the Upper Layer is defined in **Table 33 (MASTER DATA SERVICE RESPONSE)**.]

TLMM_M_297 - DATA - WAIT FOR INDICATION

The master will send a data indication to the Upper Layer when the data transaction completes.

- For the (master) Memory Data service, when all the data packets are acknowledge by the slave (and the master transaction was the message's final transaction) the master will wait for slave data packet(s). It will re-assemble these packets into complete transactions (indications). It will forward these indications (in-order) to the Upper Layer. It will cancel the reply timer if it forwards the final slave indication to the Upper Layer;
- For the (slave) Unrequested Data (or Waveform) service, the master will reassemble all the slave data packets into a complete indication and forward it to the Upper Layer.

[The content of the Indication to the Upper layer is defined in **Table 32 (DATA INDICATION (MASTER/SLAVE))**.

If the reply timer expires while waiting for the final indication an (unsuccessful) data response will be sent to the Upper Layer.]

TLMM_M_298 - DATA - RECOVERY

The master will "suspend" the data service request if channel recovery is triggered. If channel recovery is successful (and the reply timer is still active), the master will retry the data service request.

TLMM_M_299 - DATA - REPLY TIMEOUT

The master will send an (unsuccessful) data service response to the upper layer if the reply timer expires during the data service request.

[The Upper Layer should consider the data "message" service unsuccessful if it receives a failed data service response.]

TLMM_M_300 - DATA - FIRST INDICATION

The master will set the first indication bit in the slave upper layer header byte before it forwards the first indication to the upper layer.

[Note: upper layer may receive multiple "first" data indications. This can occur during retry. The upper layer should restart the re-assembly of the response message each time it receives a "first" data indication.]

TLMC_S_059 - DATA - SLAVE UPPER LAYER PARAMETERS

When invoking the Data service request, the slave Upper Layer will provide information as specified in **Table 30 (DATA SERVICE REQUEST (MASTER OR SLAVE))**.

TLMC_S_058 - DATA - RESPONSE TO UPPER LAYER

When Data request is received, the Data Link Layer will validate the Data service request and will send the response to the upper layer.

[The content of the response to the Upper Layer is defined in **Table 34 (SLAVE DATA SERVICE RESPONSE)**.]

TLMC_S_060 - DATA - SLAVE TRANSMISSION OF DATA PACKETS

When the Data service request is invoked the data link layer will fragment the user data into slave packet(s) which then will be transmitted. The first fragment of the user data will contain the slave ID.

Note that if bit "Enable WF in Recovery" (see **Table 124 (SLAVE PARAMETERS FROM HOST)**) is set to "1", Data service requests (sub-type Waveform data) received during recovery state will be transmitted when the channel recovery is successful.

[The data link layer will fragment the user data and format them into slave packets, add slave ID in the first fragment, format the packet control byte according the Data type specified, transmit the packets via Synchronous Mode operation.]

TLMC_S_061 - DATA - SLAVE DATA PACKET FORMAT(S)

The slave will use one or more of the slave packet(s) formats shown in **Fig. 53 (Slave Data Packet(s) Format)**.

TLMC_S_062 - DATA - PACKETIZATION OF SLAVE DATA

The slave will packetize the user data into slave packets per **Table 35 (DATA PACKET RECEPTION VALIDATION)**.

[Note: If a data (ensuing, not full) packet is needed, it must be the last packet of the transaction.]

TLMC_M_063 - DATA - RECEIPT OF AN INVALID SLAVE PACKET

The master shall use the ARQ algorithm to acknowledge or request the retransmission of invalid or not received Memory Data packets See **Table 11 (SLAVE PACKET VALIDATION)**.

[A packet with an invalid sequence number is ignored.]

TLMC_M_066 - DIAGNOSTIC SERVICE REQUEST FROM UPPER LAYER

When the Upper Layer invokes the diagnostic service it will provide information as specified in **Table 36 (DIAGNOSTIC SERVICE REQUEST (MASTER))**.

TLMC_M_067 - REMOTE DIAGNOSTIC - MASTER PACKET FORMAT

When a remote diagnostic service request is invoked by the Upper Layer, the master will:

1. Start the service duration timer.
2. Transmit a diagnostic packet as defined by **Fig. 55 (Master diagnostic (request) packet)**. The diagnostic packet sub-type bit (in master control byte 2) should be set to one (request packet, not filler packet).

[Send a "first request packet sent" event indication to the Upper Layer (see **Table 23**).]

TLMC_S_068 - REMOTE DIAGNOSTIC - SLAVE PACKET FORMAT

The slave will respond to a valid Diagnostic request packet with one or more diagnostic response packet. The diagnostic packet sub-type bit (in the diagnostic status byte) should be set to one (response packet, not filler packet). The format of the slave Diagnostic packet is indicated in **Fig. 56 (Slave diagnostic packet)**.

TLMC_S_069 - REMOTE DIAGNOSTIC - STATUS BYTE FORMAT

Upon receipt of a diagnostic request packet, the slave will update the diagnostic status byte. If the diagnostic data address (in the request packet) is valid, the slave will set the request address bit to valid, and the diagnostic sub-type bit to response. If the diagnostic data address (in the request packet) is invalid, the slave will set the request address bit to invalid, and the diagnostic sub-type bit to filler. The format of the diagnostic status byte is shown in **Fig. 57 (Slave diagnostic status byte)**

TLMC_S_070 - REMOTE DIAGNOSTIC - SLAVE RESPONSE PACKET

The diagnostic response will be transmitted when there is available room in the frame, based on the slave packet priorities. The diagnostic response is cancelled if Channel Recovery is triggered.

The contents of this transaction are shown in **Table 106 (DIAGNOSTIC SERVICE RESPONSE — IMPLANT DATA)**.

A diagnostic response transaction will utilize these bytes as follows:

1. Diagnostic data byte 1 = content of diagnostic (request) address
2. Diagnostic data byte 2 = content of diagnostic (request) address+1
3. .
4. .
5. Diagnostic data byte n = content of diagnostic (request) address + (n-1)

[A diagnostic transaction shall be limited in size to (240 bytes - size of (slave station ID))]

TLMC_M_071 - REMOTE DIAGNOSTIC - MASTER REQUEST RETRANSMISSION

The master will continue to transmit the diagnostic request packet until it is ACKed, service duration timeout, channel recovery, or interrupted by emergency. Once ACKed, the master will wait for the slave diagnostic (response) transaction. The master will transmit filler packets while waiting for the slave transaction. Transmission of diagnostic filler packets continues until successful, service duration timeout, channel recovery, or interrupted by emergency.

TLMC_M_072 - REMOTE DIAGNOSTIC - RESPONSE TO UPPER LAYER

A Remote Diagnostic service response to the Upper Layer will be generated at the completion of the Remote Diagnostic service.

[The content of the response is shown in **Table 37 (REMOTE DIAGNOSTIC SERVICE RESPONSE (MASTER))**. Note that a service response status of 0x26 occurs if the service is interrupted by a PLL lock error (and a subsequent successful trim).]

TLMC_M_073 - LOCAL DIAGNOSTIC - RESPONSE TO UPPER LAYER

The local diagnostic service response to the Upper Layer will be generated immediately.

[The content of the response is shown in **Table 38 (LOCAL DIAGNOSTIC SERVICE RESPONSE (MASTER))**.]

TLMC_M_074 - MAPPING DIAGNOSTIC - RESPONSE TO UPPER LAYER

When the master's Upper Layer invokes the Mapping Diagnostic service, the Data Link Layer will initiate diagnostic mapping. When complete, the Data Link Layer responds back to the Upper Layer.

[The content of the response is shown in **Table 39 (MAPPING DIAGNOSTIC SERVICE RESPONSE (MASTER))**. Note that a service response status of 0x26 occurs if the service is interrupted by a PLL lock error (and a subsequent successful trim).]

TLMC_M_075 - DIAGNOSTIC FILLER - MASTER FILLER PACKETS

When the master does not have any other packets to transmit it will transmit diagnostic filler packet(s) as follows:

1. The diagnostic sub-type bit (in master control byte 2) will be cleared to a zero.
2. The filler packet will be transmitted to the slave during the absence of user data (and an ARQ packet).
3. The filler packet format will be as indicated in **Fig. 60 (Master diagnostic (filler) packet)**.

TLMC_S_076 - DIAGNOSTIC FILLER - SLAVE FILLER PACKETS

The slave will transmit filler transactions when there is no Data Packet to transmit.

See **Fig. 56 (Slave diagnostic packet)** for the packet format. The diagnostic sub-type bit (in the status byte) will be set to a zero (filler packet, not response packet). The contents of this transaction are shown in **Table 106 (DIAGNOSTIC SERVICE RESPONSE — IMPLANT DATA)**.

A diagnostic response transaction will utilize these bytes as follows:

1. Diagnostic data byte 1 = content of diagnostic (default) address
2. Diagnostic data byte 2 = content of diagnostic (default) address+1

.

.

n. Diagnostic data byte n = content of diagnostic (default) address + (n-1)

[The slave does not respond with a filler packet each time it receives a filler packet from the master. The diagnostic filler packets are not transmitted during the low energy transfer mode.]

TLMC_M_077 - EMERGENCY - UPPER LAYER PARAMETERS

When invoking the Emergency service request, the master Upper Layer will provide information as specified in **Table 40 (EMERGENCY SERVICE REQUEST (MASTER))**.

TLMC_M_078 - EMERGENCY - TRANSMISSION OF EMERGENCY REQUEST

When the Emergency service request is invoked, the master shall:

1. Stop the service request in progress (if there is one in progress)
2. Generate a “cancelled” response to the Upper Layer for the cancelled service (if any)
3. Start the reply duration timer
4. Format the user data into a data packet
5. Set the Reset bit to 1 in control byte1 of the data packet
6. Set the sequence number to 0 in control byte2 of the data packet
7. Transmit the packet to the slave

TLMC_M_079 - EMERGENCY - MASTER REQUEST NOT ACKNOWLEDGED

The master Emergency service request is considered not successful if the Emergency request is not acknowledged by the slave within the reply duration.

TLMC_M_081 - EMERGENCY - STATUS RESPONSE TO UPPER LAYER

An Emergency service response to the master’s Upper Layer will be generated. When the slave’s upper layer response is completely received or a reply timeout occurs.

[The content of the response to the Upper Layer is defined in **Table 41 (EMERGENCY SERVICE RESPONSE (MASTER))**.]

TLMM_M_301 - EMERGENCY - WAIT FOR INDICATION

The master will wait for slave packets when all emergency request packets are acknowledged by the slave (and the master transaction was the message’s final transaction). It will re-assemble these packets into complete transactions (indications). It will forward these indications (in-order) to the Upper Layer. It will cancel the reply timer if it forwards the final slave indication to the Upper Layer.

[The content of the Indication to the Upper layer is defined in **Table 42 (EMERGENCY INDICATION (MASTER))**.]

TLMM_M_302 - EMERGENCY - RECOVERY

The master will “suspend” the emergency service request - if channel recovery is triggered. If channel recovery is successful (and the reply timer is still active), the master will retry the emergency service request.

TLMM_M_303 - EMERGENCY - REPLY TIMEOUT

The master will send an (unsuccessful) emergency service response to the upper layer if the reply timer expires during the emergency service request.

[The Upper Layer should consider the data “message” service unsuccessful if it receives a failed emergency service response.]

TLMC_M_304 - EMERGENCY DURING CLOSE

The data link layer will send an unsuccessful emergency service response if the emergency service request is received while processing a close service request. The emergency service response status is Request invalid in current state (state is recovery; slave scanning).

TLMC_M_305 - EMERGENCY - SECOND EMERGENCY REQUEST WHILE PROCESSING FIRST

The master will do the following if it receives an Emergency service request (if an emergency service request is already in progress):

1. Send the (unsuccessful) Emergency service response to the upper layer.

[The response status will be “Emergency already in progress”.]

TLMC_M_082 - RAW MODE - UPPER LAYER RAW MODE PARAMETERS

When the Upper Layer invokes the Raw Mode service it will provide information as specified in **Table 43 (RAW MODE SERVICE REQUEST (MASTER))**.

TLMC_M_083 - RAW MODE - MASTER PACKET ATTACHED OPERATION

When a valid Raw Mode (Master Packet Attached) service is invoked, the master shall:

1. Start the service duration timer (comes from **Table 129 (PARAMETERS FROM EEPROM)**)
2. Transmit the raw packet
3. Send the Raw Mode service response to the Upper Layer (status of successful)

[Note that the service is complete after the master packet is sent. This service does not wait for positive acknowledgement from the slave.]

TLMC_M_084 - RAW MODE - MASTER PACKET ATTACHED RESPONSE

The Raw Mode (Master Packet Attached) service response to the Upper Layer will be generated at the completion of the service.

[The content of the response is shown in **Table 44 (RAW MODE SERVICE RESPONSE (MASTER))**.]

TLMC_M_089 - CLOSE - UPPER LAYER CLOSE PARAMETERS

When invoking a Close service request, the Upper Layer will provide information as specified in **Table 45 (CLOSE SERVICE REQUEST (MASTER))**.

TLMC_MS_090 - CLOSE - MASTER PACKET FORMAT

The Close Request packets have the layout indicated in **Fig. 64 (Master Close Request Packet)**.

TLMC_MS_091 - CLOSE - SLAVE PACKET FORMAT

The Close Response packets have the layout in **Fig. 65 (Slave Close Response Packet)**.

TLMC_M_092 - CLOSE - MASTER REQUEST

The actions performed upon receiving the Close service request when the state is Comm:

1. Start the service duration timer.
2. Transmit Close Request Packet and wait for the Close Response Packet.

[Send a “first request packet sent” event indication to the Upper Layer (see **Table 23**).]

TLMC_M_093 - CLOSE - RETRANSMISSION OF MASTER REQUEST

If no valid Close Response Packet is received from the slave, the Data Link Layer shall retransmit the Close Request packet. The master shall continue this until a valid Close Response packet is received or a Service Duration Timeout occurs or until recovery is triggered. Note that recovery is not initiated (if triggered).

TLMC_S_094 - CLOSE - RECEIPT OF A VALID CLOSE REQUEST

Upon receiving a valid master Close Request packet, the slave will transmit a close response packet and will stop the transmission of data packets.

[The security hardware will be disabled if slave starts to poll again upon polling request or sniff when standby timer times out.]

TLMC_S_095 - CLOSE - INDICATION TO SLAVE UPPER LAYER

The Close service indication will be generated to the Upper Layer when a successful Close response packet is transmitted.

[The indication to the Upper Layer may consist of Polling ON or Polling OFF based on the value of the Total Polling Duration Time in the open request (i.e., Zero meaning Polling OFF, otherwise Polling ON.]

TLMC_M_096 - CLOSE - RECEIPT OF A VALID RESPONSE

Upon receiving a valid Close Response Packet from the slave, or if a Close Service Duration Timeout occurs, the master will return a successful close service response to the Upper layer.

TLMC_M_097 - CLOSE - IMPLICITLY SUCCESSFUL

Upon receiving a valid close service request from the Upper Layer when the state is Channel Recovery or Standby or Inactive or Medical Event, the master will disable the security hardware and return a successful close service response to the Upper layer.

[Terminate physical layer communication, return a successful Close service response to the Upper Layer, transition to Standby State - Note: Close packet is not transmitted. The Close service is implicitly successful.]

TLMC_M_098 - CLOSE - RESPONSE TO UPPER LAYER

The security hardware will be disabled and a close service response will be generated after the close service is complete. The content of the response to the Upper Layer is specified in **Table 47 (CLOSE SERVICE RESPONSE (MASTER))**.

TLMC_M_099 - CLOSE - RECOVERY DURING CLOSE

If channel recovery is triggered during the close service, the service is implicitly successful. The master will disable the security hardware and return successful to the Upper layer.

[Terminate physical layer communication, return success to upper layer, transition to Standby.]

TLMC_S_414 - MEDICAL EVENT - SLAVE PACKET FORMAT

The slave Medical Event response packet has the format is indicated in **Fig. 67 (Medical Event Packet)**

TLMC_S_229 - MEDICAL EVENT - REQUEST

When medical event request is received, the Upper Layer will provide information as specified in **Table 48 (MEDICAL EVENT SERVICE REQUEST (SLAVE))**.

TLMC_S_307 - MEDICAL EVENT - CHANNEL TRANSMIT TIMOUT

The slave will restart native mode (transmit first) operation and re-send the Medical Event packet upon medical event channel transmit timeout. It will re-send this packet on a new channel.

[Slave will send Medical Event packet at a regular intervals as defined in **Table 124 (SLAVE PARAMETERS FROM HOST)**]

TLMC_S_416 - MEDICAL EVENT - INVALID OPEN REQUEST

The slave will restart native mode (transmit first) operation and re-send the Medical Event packet if it receives an Invalid Open packet.

If a valid Discover ID packet is received, Slave shall send a Discover Indication (i.e., Discover ID received while uplink medical events) to the host. See **Table 24 (DISCOVER INDICATION (SLAVE))**.

[An Invalid Open packet includes Open with mismatched slave ID, a packet other than Open, and packet with MAC error. Receipt of an invalid open packet causes the slave to transmit another medical event packet (and listen for an open packet).It will resend this packet on a new channel.]

TLMM_S_440 - MEDICAL EVENT - POLL UPON TELM REQUEST

If a poll upon Tel M request is received while uplink medical events, slave will stop uplink medical event, send polling response to host, and then slave shall start polling.

[See **TLMM_S_035 - POLLING - SERVICE REQUEST FROM UPPER LAYER**,
TLMM_S_256 - POLLING - SERVICE RESPONSE TO UPPER LAYER and
TLMM_S_241 - POLLING ENABLING CRITERIA for details]

TLMC_S_417 - MEDICAL EVENT - SERVICE DURATION

The slave will send Medical Event packets (and listen for open packets) until the Medical Event service duration timer expires.

TLMC_S_418 - MEDICAL EVENT - RESPONSE

When medical event request is received, the Data Link Layer will validate the medical event service request. A medical event service response will be sent to the host (see **TABLE 49: MEDICAL EVENT SERVICE RESPONSE (SLAVE)**). If the service status is successful, the slave will start the Medical Event Service Duration timer, and send a medical event packet.

TLMC_S_257 - MEDICAL EVENT - INDICATION TO UPPER LAYER

A medical event service will be stopped under the following conditions:

1. Medical event service duration timer expires medical event timeout indication will be generated to the upper layer **Table 50 (MEDICAL EVENT INDICATION (SLAVE))**.
2. A valid open packet is received an open successful indication will be generated to the upper layer.**Table 28 (OPEN INDICATION (SLAVE))**

TLMC_M_230 - MEDICAL EVENT - HOST PARAMETERS

When invoking a Medical Event service request, host will provide information as specified in **Table 51 (MEDICAL EVENT SERVICE REQUEST (Master))**

TLMC_M_231 - MEDICAL EVENT - RESPONSE TO HOST

A Medical Event response will be generated when the Medical Event service is initiated.

[The content of the response is shown in **Table 52 (MEDICAL EVENT SERVICE RESPONSE (Master))**.

TLMC_M_232 - MEDICAL EVENT - MEDICAL EVENT PACKET

If the master receives a valid medical event packet it will send a medical event indication to the host as specified in **Table 53 (MEDICAL EVENT INDICATION (Master))**

[The indication will be generated to the host no more frequently than one/second]

TLMC_S_258 - PATIENT NAME REQUEST

When Patient name request is invoked host will provide the patient name information.

[In the case of CRM this information resides in the device-specific system parameters database.]

TLMC_S_259 - PATIENT NAME SERVICE RESPONSE TO UPPER LAYER

When patient name request is received the Data Link Layer will validate the Patient Name service request (see **Table 54 (PATIENT NAME SERVICE REQUEST (SLAVE))**) and will generate an appropriate response (see **Table 55 (PATIENT NAME SERVICE RESPONSE (SLAVE))**).

TLMM_MS_309 - CONFIGURE NETWORK - UPPER LAYER PARAMETERS

When invoking a configure network service request, the Upper Layer shall provide information as specified in **Table 57 (CONFIGURE NETWORK SERVICE REQUEST)**.

TLMM_MS_310 - CONFIGURE NETWORK - RESPONSE TO UPPER LAYER

Upon the receipt of the network configuration table data link layer will generate a response to upper layer as specified in **Table 58 (CONFIGURE NETWORK SERVICE RESPONSE)**.

TLMM_M_311 - NETWORK DATA - UPPER LAYER PARAMETERS

When invoking a network data service request, the upper layer shall provide information as specified in **Table 59 (NETWORK DATA SERVICE REQUEST(MASTER))**.

TLMM_M_312 - NETWORK DATA - REQUEST AND RESPONSE PACKET FORMAT

The network data request and response packet shall have the format shown in **Fig. 73 (Network data Request packet)** and **Fig. 74 (Network data Response packet)**. In the request packet the source network table index shall be the index of the master.

Master shall echo the slave’s random numbers received in the network filler packets. In the response packet the source network table index shall be the index of a slave. The network ID shall be the station ID of the coordinator.

TLMM_S_313 - NETWORK DATA - FILLER PACKET FORMAT

The network filler packet shall be used by the slave to transmit its own random number to the master as well as respond to the network data request before the slave host responds to the network data request.

[The filler packet is to occupy the channel format of this packet is shown in **Fig. 79 (Network Filler Packet)**].

TLMM_M_314 - NETWORK DATA - TRANSMITTING REQUEST PACKETS

Upon receiving a network data service request from the upper layer master will:

1. Start the service duration timer.
2. Select the least interfered channel (takes 275 +/- 50 msec)
3. Start LBT timer.
4. Perform instigator wireless wakeup. See parameters in **Table 123 (Master parameters from host)** for Wakeup user data present, secure wakeup and user data.
5. Transmit the network data request packet.

TLMM_M_315 - NETWORK DATA - UPPER LAYER RESPONSE (MESSAGE LEVEL RESPONSE)

The network data request packet shall be transmitted until any one of the timers expires (the service duration timer or the LBT timer). When any of the timers expire a network complete indication will be sent to the host. See **Table 60 (NETWORK COMPLETE INDICATION (MASTER))**

[See **Fig. 70 (Network data (with message level acknowledgement) diagram)** for an example.

The upper layer sets “application reply requested” bit in the ULH byte to request upper layer (message level) response from the slave.

The antenna will be toggled each time a network packet is transmitted.]

TLMM_M_316 - NETWORK DATA - DATA LINK LAYER RESPONSE (PACKET LEVEL RESPONSE)

The network data request packet shall be transmitted until any one of the timers expires (the service duration or the LBT timer). When any of the timers expire a network complete indication will be sent to the host. See **Table 60 (NETWORK COMPLETE INDICATION (MASTER))**.

[See **Fig. 71 (Network data (with data link acknowledgement) diagram)** for an example.

The upper layer sets the “Expect response” field in the network data service request and clears the “application reply requested” bit in the ULH byte to request data link layer (packet level) response from the slave.

The data link layer will set the “Expect response” bit in control bytes of the network data request packet accordingly.

The antenna will be toggled each time a network packet is transmitted.]

TLMM_M_317 - NETWORK DATA - NO ACK

The network data request packet shall be transmitted until any one of the timer expires (the service duration or the LBT timer), if no response is required from the slave(s). When any of the timers expire a network complete indication will be sent to the host. See **Table 60 (NETWORK COMPLETE INDICATION (MASTER))**.

[The upper layer clears “Expect response” field in the network data service request and clears the “application reply requested” bit in the ULH byte to suppress any response from the slave(s).

The data link layer will clear the “Expect response” bit in the control bytes of the network data request packet accordingly.

The antenna will be toggled each time a network packet is transmitted.]

TLMM_M_318 - NETWORK DATA - SUCCESSFUL RESPONSE TO UPPER LAYER

Upon receiving a valid network data service request, the data link layer shall send a successful network data service response to the upper layer. See **Table 61 (NETWORK DATA SERVICE RESPONSE (MaSter))**

[The data link layer checks CRC and the parameters of the service request.]

TLMM_M_319 - NETWORK DATA - UNSUCCESSFUL RESPONSE TO UPPER LAYER

Upon receiving an invalid network data service request, the data link layer shall send an unsuccessful network data service response to the upper layer. **Table 61 (NETWORK DATA SERVICE RESPONSE (MaSter))**

[The data link layer checks CRC and the parameters of the service request.]

TLMM_M_320 - NETWORK DATA - INDICATION TO UPPER LAYER

A network summary indication shall be sent to the upper layer if any one of the timers expires (the service duration and the LBT timer). See **Table 62 (NETWORK SUMMARY INDICATION (MASTER))**

TLMM_M_321 - NETWORK DATA - RESPONSE COMPLETE AT MASTER

Upon receiving the response from all slaves a network summary indication will be sent to the host (see **Table 62 (NETWORK**

SUMMARY INDICATION (MASTER))). The final acknowledgement shall be re-sent until any one of the timers expires (the service duration timer or the LBT timer).

[This is done for the case that a slave does not receive the final acknowledgement and it is still listening in the native mode. Note, the slaves will not respond to the master’s final acknowledgement either it receives it or it does not receive it.]

TLMM_M_431 - NETWORK DATA - TRANSMIT SECURED NETWORK DATA PACKET

When the “security user data/network packet” bit is set in the local wake up packet, master should do the following before the transmission of every network data packet:

1. Start the block cipher for message integrity, privacy and freshness checking. The block cipher’s key input is set to AES_out[127:64] concatenated with KBAN[63:0]. The block cipher’s nonce counter is initialized using master random number (CTRi) See **Fig. 81 (Wakeup security - block cipher initialization (to secure native mode packets))**..
2. Enable the hardware’s security block (security register to on).

[For AES_out[127:64] and master Random number (CTRi), see **Fig. 116 (Calculate AES_out for wakeup security)**. In case there is no user data, a known pattern (0x00,0x00,0x00,0x00) will be used.]

[An idle immediate is performed before 1) and 2)]

[When the “security user data/network packet” bit is cleared, only idle immediate is performed]

TLMM_S_434 - NETWORK DATA - PROCESSING POLLING REQUEST

After enabled by a valid local wakeup, if a polling request (upon TEl M) is processed, slave shall start to poll, i.e., listen on the channel with the data rate provided by the local wakeup packet.

[Both the channel and transmit/receive data rate are from local wakeup packet.

“SlaveScansDuringPolling” bit does not apply in this case. There are no channel change in native mode and no async sniffing.

Slave will generate and save 3 bytes of random number. The 3bytes random number will be transmitted in the following Network Filler packets and also used to verify the Network Data Request packets]

TLMM_S_322 - NETWORK DATA - NO ACK OF INVALID REQUEST PACKET

No response packet shall be transmitted to the master if the invalid network data request packet is received.

[Then idle immediate is performed and the slave starts to listen in native mode again.]

TLMM_S_323 - NETWORK DATA - NO ACK OF REQUEST PACKET

No response packet shall be transmitted to the master if the received network data request packet has the “Expect response” bit set to zero in the control bytes.

[The master shall be the transmission-only device. The slave shall go to the disable state after processing the network data request packet.]

TLMM_S_324 - NETWORK DATA - ACK OF REQUEST PACKET

Upon receiving a valid network data request packet with the “expect response” bit set in control bytes or with the “application response requested” bit set in ULH, a network filler packet shall be transmitted with its own random number. If the random number in the received network data request packet matches to its own random number, the master’s bit in the ack bit map of the network filler response packet shall be set accordingly.

[Note: A network data response packet will be sent in lieu of a network filler packet (if a network service request occurs before the slave has had a chance to transmit a network filler packet).]

TLMM_S_325 - NETWORK DATA - SERVICE INDICATION TO THE UPPER LAYER

For Network Data Request packet “with no acknowledgement”, Network Data indication shall be sent to the upper layer upon receiving a valid network data request packet.

For Network Data Request packet “with message level acknowledgement” or “with data link acknowledgement”, upon receiving a valid network data request packet, slave shall also compare its own random number with the random number in the received Network Data Request packet. If they match, Network Data indication shall be sent to the upper layer.

[For Network Data Request packet “with no acknowledgement”, there is no random number comparison as slave does not transmit at all.

The content of the indication is defined **Table 63 (NETWORK Data INDICATION (SLAVE))**.

It is the host’s responsibility to check the network table CRC of the instigator with the network table CRC of its own. It is also host’s decision on what to do if the two network tables’ CRCs do not match. From a system's point of view it is probably best if the slave host ignores the network data from a therapy point of view. Additionally it is probably best if the slave attempts to respond to the instigator with some application failed status. The master can recognize the network table CRCs are mismatched and correct the slave table. If the slave host did not respond, the instigator wouldn't know if the tables were mismatched or if the instigator and slave were out of telemetry range.]

TLMM_S_326 -NETWORK DATA- ENABLE SECURITY BLOCK

1. Upon receipt of a poll service request or whenever network data response or network filler packet is transmitted, if the “security user data/network packet” bit is set in the local wakeup packet, the hardware’s security block will be enabled (security register to on).
2. At that time, the block cipher will be started for message integrity, privacy and freshness checking. The block cipher’s key input is set to AES_out[127:64] concatenated with KBAN[63:0]. The block cipher’s nonce counter is initialized using CTRi (see **Fig. 81 (Wakeup security - block cipher initialization (to secure native mode packets))**

[When the “security user data/network packet” bit is cleared, only idle immediate is performed]

TLMM_S_327 - NETWORK DATA - NETWORK DATA RESPONSE

Upon receiving a valid network data request from the upper layer, the data link layer shall transmit a network data response packet to the master. See **Table 64 (NETWORK DATA SERVICE REQUEST (SLAVE))**

The slave will retransmit the network data packet if it receives a network data packet with “this slave’s” ack bit not set.

[Note that a network data response packet will be sent in lieu of a network filler packet (if a network service request occurs before the slave has had a chance to transmit a network filler packet).]

TLMM_S_328 - NETWORK DATA - RESPONSE TIME SLOT

The network data filler packets or the network data response packets shall be transmitted to the sender in the designated time slot. The slave’s designated time slot shall be obtained using the equation below:

TLMM_S_329 - NETWORK DATA - SEQUENCE NUMBER

The slave shall ignore a duplicate network data request packet.

[A master shall transmit the same network data request packet multiple times to all designated slaves if it does not receive expected the response from any one of the slaves. A master will always transmit the network data request packet with a sequence number of zero. The slave will ignore duplicate network data request. The slave will only process one network data packet and send one Network data indication to host after it receives a valid wakeup packet.]

TLMM_S_330 - NETWORK DATA - INVALID RESPONSE TO INDICATION

The slave data link layer will send an immediate unsuccessful service response to the upper layer if the slave data link layer receives a network data service request when the master does not expect an application response (see **Fig. 48 (Master’s ULH byte)** - application reply requested bit). See **Table 65 (NETWORK DATA SERVICE RESPONSE (SLAVE))**.

[Note that the slave host uses a network data service request to send an application response to the master host. This requirement describes the case where the master host does not want an application response, yet the slave host is trying to provide one.]

TLMM_S_331 - NETWORK DATA - SERVICE SUCCESSFUL AT SLAVE

Once acknowledge is received from the master:

For packet level response slave will terminate polling,

For message lever response a successful service response will be sent to the upper layer and polling will be terminated. See **Table 65 (NETWORK DATA SERVICE RESPONSE (SLAVE))**.

[The slave shall stop native mode listening, cancel all timers and return to the standby state.]

TLMM_S_332 - NETWORK DATA - SERVICE UNSUCCESSFUL AT SLAVE

When the polling timer expires while a network data service request has been received but the service response has not been generated, an unsuccessful network service response will be generated to the upper layer. See **Table 65 (NETWORK DATA SERVICE RESPONSE (SLAVE))**.

Slave will also generate polling timeout indication to the upper layer.

[The slave shall stop native mode listening, cancel all timers and return to the standby state.]

TLMM_MS_403 - SECURITY - UPPER LAYER PARAMETERS

When invoking a Security service request, the Upper Layer will provide information as specified in **Table 66 (SECURITY SERVICE REQUEST (MASTER OR SLAVE))**. If the service type is correct but the sub-type is not equal to any of the values specifed for the master and slave respectively, the status of “invalid security sub-type” will be returned in the service response.

[The service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))** and **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.]

TLMM_MS_334 - SECURITY - CREATE BAN KEY/EPHEMERAL BAN KEY LOCAL

The master or slave will do the following upon the receipt of a create BAN key or ephemeral BAN key service request:

1. Create pseudo random number. See the appendix, section **7.3 Pseudo Random Number Generation**.
2. Save the pseudo random number as the new BAN key or ephemeral BAN key.
3. Send a successful create BAN key or ephemeral BAN key service response to the upper layer.
4. The EBAN key duration timer will also be started on the slave when an EBAN key is created on the slave due to this service request.

[The slave service response is generated per **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))** and the master service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))**]

TLMM_MS_335 - SECURITY - ASSERT BAN KEY/EPHEMERAL BAN KEY LOCAL

The master or slave will do the following upon the receipt of an Assert BAN key or ephemeral BAN key service request:

1. Save the BAN key or ephemeral BAN key contained in the service request as the new BAN key or ephemeral BAN key.
2. Send a successful assert BAN key or ephemeral BAN key service response to the upper layer.
3. The EBAN key duration timer will also be started on the slave when an EBAN key is asserted to the slave.

[The slave service response is generated per **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))** and the master service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))**]

TLMM_MS_336 - SECURITY - REQUEST BAN KEY OR EPHEMERAL BAN KEY LOCAL

The master or slave will do the following upon the receipt of a Request BAN key service request:

1. Read the BAN key or ephemeral BAN key from the memory;
2. Return the BAN key or ephemeral BAN key to its upper layer via a service response.

[The service response is generated per **Table 67 (SECURITY SERVICE RESPONSE (MASTER))** and **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**]

TLMM_MS_401 - SECURITY - MASTER/SLAVE REJECTS KEY REQUEST IF IT HAS NO BAN/EBAN KEY

The slave/master will send an unsuccessful request BAN/EBAN key local response if the slave/master has no corresponding permanent BAN key or EBAN key as requested.

[This will occur if the host has never created or asserted a permanent BAN key or EBAN key since the last reset]

TLMM_M_337 - SECURITY - PROXIMITY SECURITY REQUEST PACKET FORMAT

The proximity security request packet has the format indicated in **Fig. 86 (Proximity Security Request Packet)**

TLMM_S_338 - SECURITY - PROXIMITY RESPONSE PACKET FORMAT

The proximity security response packet has the format indicated in **Fig. 87 (Proximity Security Response Packet)**.

TLMM_M_339 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH REQUEST PACKET FORMAT

The request BAN/EBAN key via proximity switch request packet has the format indicated in **Fig. 88 (Request BAN/EBAN key via proximity switch request packet)**.

TLMM_S_340 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH RESPONSE PACKET FORMAT

The request BAN/EBAN key via proximity switch response packet has the format indicated in **Fig. 90 (Request BAN/EBAN key via proximity switch response packet)**.

[Note that if the remaining polling duration is greater than 255 seconds, this field will be set to 255 seconds. Also note that the “slave EBAN key exists” bit is only set when the EBAN key duration timer is running.]

TLMM_M_341 - SECURITY - TRANSMIT PROXIMITY SECURITY REQUEST PACKET

Upon receiving a request BAN key via proximity switch service request, the master will start a service duration timer and transmit the proximity switch security request packet.

[Start the service duration timer; select the least interfered channel (takes approximately 275 +/- 50 msec); start the Listen Before Talk (LBT) timer; perform instigator wireless wakeup, transmit the proximity switch security request packet, send a “first request packet sent” event indication to the Upper Layer (see **Table 23 (FIRST REQUEST PACKET SENT INDICATION (MASTER))**).

Note that instigator wireless wakeup is suppressed if the master knows the slave is scanning. Time now, time of last uplink, and remaining polling duration at the time of last uplink are used to determine this.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup if not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMM_M_342 - SECURITY - NO RESPONSE TO PROXIMITY SECURITY REQUEST

If no response is received, the master will resend the proximity switch security request until the Service Duration timer is expired.

[If no valid proximity switch security response packet is received, the master will re-transmit the proximity switch security request packet on the alternate antenna.

This process shall be continued until a valid proximity switch security response packet is received, a Service Duration Timeout occurs, or a Listen Before Talk timer expires.]

TLMM_M_343 - SECURITY - VALID RESPONSE TO PROXIMITY SECURITY REQUEST

If a valid proximity security response packet is received, the master will transmit a request BAN/EBAN key packet to the slave.

TLMM_M_344 - SECURITY - NO RESPONSE OR REJECTION TO REQUEST BAN/EBAN KEY VIA PROXIMITY SWITCH

If no response is received or if the BAN/EBAN key request is rejected (because the proximity switch is not yet open), the master will resend the request BAN/EBAN key packet until successful or until the Service Duration timer is expired.

[The master will re-transmit the request BAN/EBAN key packet on the alternate antenna if no valid BAN/EBAN key response packet is received. The master will re-transmit the request BAN/EBAN key packet on the same antenna if the BAN/EBAN key request is rejected (since the proximity switch is not yet open).

This process shall be continued until a valid BAN/EBAN key response packet is received, a Service Duration Timeout occurs, or a Listen Before Talk timer expires.]

TLMM_M_346 - SECURITY - REQUEST BAN/EBAN key VIA PROXIMITY SWITCH - RESPONSE TO UPPER LAYER

A request BAN/EBAN key via proximity switch service response will be generated to the Upper Layer when the service is completed.

[The content of the response to the Upper Layer is specified in **Table 67 (SECURITY SERVICE RESPONSE (MASTER))**.]

TLMM_M_347 - SECURITY - LBT TIMEOUT DURING REQUEST BAN/EBAN KEY EXCHANGE VIA PROXIMITY SWITCH SERVICE

If the listen before talk (LBT) timer expires during a request BAN/EBAN key via proximity switch service, the master will do the following:

1. Re-select the least interfered channel for communication,
2. Start the Listen Before Talk timer when a good channel is found,
3. Toggle the antenna,
4. Transmit a wakeup packet,
5. Transmit a proximity security request.

[Note that “transmit wakeup packet” is suppressed if the master knows the slave is scanning. Time now, time of last uplink, and remaining polling duration at the time of last uplink are used to determine this.

Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup is not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMM_S_348 - SECURITY - SLAVE RESPONSE TO A VALID PROXIMITY SWITCH SECURITY REQUEST PACKET

The slave will send a proximity switch security response packet upon receiving a valid proximity switch security request packet. Additionally the slave will send a proximity security indication to the slave’s upper layer (host). The indication to the upper layer (host) will not be generated more than once a second. See **Table 69 (PROXIMITY SECURITY INDICATION (SLAVE))**

[Upon receipt of the proximity security indication, the slave’s upper layer should do the following:
1. Power up the proximity switch circuitry
2. Determine if the user has “opened the proximity switch”. This may be polling a reed switch, etc. The proximity switch is short range. This ensures that the proximity switch is not opened anonymously.
3. Send an authorize (proximity switch) BAN/EBAN key delivery to the data link layer
4. Power down the proximity switch circuitry]

TLMM_S_349 - SECURITY - SLAVE REJECTS A VALID REQUEST BAN/EBAN KEY VIA PROXIMITY SWITCH REQUEST

The slave will send an (unsuccessful) request BAN/EBAN key (via proximity switch) response packet under the following conditions:

1. The slave receives a request BAN/EBAN key (via proximity switch) request packet, and
2. The slave is not authorized to deliver the BAN/EBAN key to the master. Specifically, the security (authorize key delivery) service duration timer is not active.

TLMM_S_350 - SECURITY - SLAVE ACCEPTS A VALID REQUEST BAN/EBAN key VIA PROXIMITY SWITCH REQUEST

The slave will generate a EBAN key, for EBAN key requested, and will send a (successful) request BAN/EBAN key (via proximity switch) response packet under the following conditions:

1. Attenuate the transmit power according to the settings in slave configuration table. Once the transmitting process of response is finished original transmission power should be restored.

2. The slave receives a request BAN/EBAN key (via proximity switch) request packet, and
3. The slave is authorized to deliver the BAN/EBAN key to the master (it has “recently” received an authorize BAN/EBAN key delivery service request from the upper layer (host)).

When delivering the BAN/EBAN key the transmit power should be attenuated according to the settings in slave configuration **Table 124 (SLAVE PARAMETERS FROM HOST)**, once the transmitting process of response is finished original transmission power should be restored. BAN/EBAN key delivery is authorized upon receipt of the authorize BAN/EBAN key delivery (via proximity switch) service request. A service request timer is started upon receipt of the service request. BAN/EBAN key delivery is no longer authorized when the service duration timer expires.

[The EBAN key is generated when slave is entering the Acquire state and delivered when sending the request (EBAN) key response packet.]

TLMM_S_351 - SECURITY - AUTHORIZATION TO GIVE BAN/EBAN KEY SERVICE RESPONSE

When authorize BAN/EBAN key delivery request is received then response will be generated See **Table 66 (SECURITY SERVICE REQUEST (MASTER OR SLAVE))**.

[The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.The security service duration will be started upon the receipt of a authorize key delivery service request. The security service duration timer will be re-started upon the receipt of an authorize key delivery service request if the security service duration timer is already running (from a previous update proximity status service request.) When the key delivery is authorized, the slave will set the “unidentified proximity transmitter near” bit and the “specific proximity transmitter near” bit. See **Fig. 35 (Proximity status type byte)** for the bit definition.

Also note that the EBAN key duration timer will be restarted on the slave upon the delivery of the EBAN Key to the master.]

TLMM_S_352 - SECURITY - USAGE OF EPHEMERAL BAN KEY

The use of ephemeral BAN key is limited by the ephemeral BAN key duration parameter. Once this timer expires a secure session may not be opened (via the ephemeral BAN key). If a secure session is opened before the timer expires, the timer is cancelled. The ephemeral BAN timer is restarted upon a close session or channel recovery timeout. If the timer expires, the ephemeral BAN key is no longer valid. Note that the timer is cancelled if a new session is successfully started using the ephemeral BAN key. The ephemeral BAN key timer also expires when the slave receives a Shut Down request from its host.

[The ephemeral BAN key timer value is received from Master as part of proximity switch request packet. The EBAN key duration timer can also be updated via assert EBAN duration service request]

TLMM_S_405 - SECURITY - ASSERT EBAN KEY DURATION

The EBAN key duration will be updated upon the receipt of an assert EBAN key duration service request. If the EBAN key duration timer is running, the new value will take effect immediately. A service response will also be sent to the upper layer.

[Note: The EBAN key duration should be greater than the channel recovery duration by the host. When the existing EBAN key duration timer is running and new values is provided via assert EBAN key duration service request the new value will take effect immediately - the expiration of the running timer will be based on the new value. Also note that because the EBAN key duration timer has a resolution of one second, it may take up to one second for it to expire if the newly asserted EBAN key duration implies the timer should expire immediately.

The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.]

TLMM_S_419 - SECURITY - UPDATE PROXIMITY STATUS

The slave will do the following upon receipt of an update proximity status service request:

1. Start the security service duration timer
2. Update the proximity status byte (set the “unidentified proximity transmitter near” bit). See **Fig. 35 (Proximity status type byte)**.
3. Send an update proximity service response to the upper layer

[The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.

Also note that the discover collision avoidance timer is cancelled upon the receipt of the update proximity status service request]

TLMM_S_420 - SECURITY - SERVICE DURATION TIMEOUT

The slave will do the following when the security service duration timer expires:

1. Clear the proximity status byte (clear the “unidentified proximity transmitter near” bit and the “specific proximity transmitter near” bit). See **Fig. 35 (Proximity status type byte)**.

[The content of the response to the Upper Layer is specified in **Table 68 (SECURITY SERVICE RESPONSE (SLAVE))**.

Also note that the security key delivery is no longer allowed once this timer expires.]

TLMM_M_422 - POWER INHIBIT - CHANNEL RECOVERY THRESHOLD

The master data link layer will set its (the master’s) channel recovery threshold (number of consecutive frames without a valid packet) using the received power inhibit bits (control byte 5, bits 5 and 6 from the slave). See **Table 71 (POWER INHIBIT**

MODES)

[Note that the original recovery thresholds come from the master initialization request. The power inhibit mode is just selecting threshold 1, threshold 2 or threshold 3.]

TLMM_S_264 - POWER INHIBIT - UPPER LAYER TO DATA LINK LAYER PARAMETERS

When invoking the power inhibit service request, the Upper Layer will provide information as specified in **TABLE 72: POWER INHIBIT SERVICE REQUEST (SLAVE)**.

TLMM_S_421 - POWER INHIBIT - TRANSMIT PACKET

The slave data link layer will do the following upon the receipt of a power inhibit service request:

1. Communicate the power inhibit sub-type to the master by setting the power inhibit bits on every transmit packet (see control byte 5)

[The slave transmit packet's control byte 5, bits 5 and 6 shall mirror the power inhibit service request sub-type.]

TLMM_S_427 - POWER INHIBIT SERVICE RESPONSE TO UPPER LAYER

A power inhibit service response will be generated to the Upper Layer after the power inhibit service request is complete.

[The content of the response to the Upper Layer is specified in **Table 73 (POWER INHIBIT SERVICE RESPONSE (SLAVE))**].

TLMM_S_423 - POWER INHIBIT IDLE IMMEDIATE - ASSERTED

If the power inhibit interrupt is asserted (while power inhibit mode is idle immediate),

1. While slave is in communication state, Slave will start same channel recovery immediately. (see **TLMC_S_130 - CH RECOVERY - START SAME CHANNEL RECOVERY**).
2. While slave is not in communication state. Slave will stop sniffing if sniffing is enabled or perform idle immediate if transmission or receiving is progress.

[For #1, note that the only difference from **TLMC_S_130 - CH RECOVERY - START SAME CHANNEL RECOVERY** is slave will not restart sync mode.]

TLMM_S_424 - POWER INHIBIT IDLE IMMEDIATE - DEASSERTED

If the power inhibit is deasserted (while power inhibit mode is Idle immediate),

1. While the same channel recovery time is active, slave will attempt to recover the communication session on the same channel.
2. While the unspecified channel recovery time is active, slave will attempt to recover the communication session following the unspecified channel recovery algorithm (see **5.3.3 Channel Recovery**).
3. While the polling duration is active, slave will restart to listen in native mode or sniffing. (see **5.2.8 Polling**).
4. While the disable ON timer is active, slave will restart the sync sniffing.
5. While the medical service duration is active, slave will restart to uplink medical event packet.(see **5.2.16 Medical Event**)

[For #1, the slave will restart sync mode when power inhibit is deasserted.]

TLMM_S_425 - POWER INHIBIT - RESPONSE

The slave data link layer will set its (the slave's) channel recovery threshold (number of consecutive frames without a valid packet) using the last power inhibit service request from the slave host. Additionally it will mirror the power inhibit mode on all its subsequent transmit packets.

[The radio will be disabled when power inhibit is asserted. This will cause the slave to receive packets with MAC error. In order to avoid channel recovery, the slave host will use power inhibit roll through (mode 1, 2, or 3). The recovery threshold (paired with the roll through mode) should be chosen to be slightly longer than the power inhibit duration. Once power inhibit is deasserted, the slave should receive packets without MAC error.]

TLMM_S_442 - POWER INHIBIT - WAKEUP COMPLETE INDICATION DURING SYNC SNIFF

If power inhibit interrupt is asserted while slave is synchronously sniffing and waiting for the "last packet last block" wakeup packet (after the "first packet last block" wakeup packet has already been received), a wakeup indication "Wakeup Complete", containing the wakeup packet contents, shall be sent to the upper layer. (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[After the "first packet last block" wakeup packet has already been received, Mozart hardware will turn the radio off and start a timer. When the timer expires, Mozart hardware will generate the "last packet last block" wakeup packet interrupt. If power inhibit is asserted, the timer will be turned off and no "last packet last block" wakeup packet interrupt will be generated by the hardware. Instead of waiting for the next synchronous sniffing window after power inhibit is deasserted, a wakeup indication "Wakeup Complete" is sent to the upper layer when the power inhibit is asserted.]

TLMM_S_447 - POWER INHIBIT - WAKEUP COMPLETE INDICATION DURING ASYNC SNIFF

If power inhibit interrupt is asserted while slave is asynchronously sniffing and waiting for the "last packet last block" wakeup packet (after the "first packet last block" wakeup packet has already been received), a wakeup indication "Wakeup Complete", containing the wakeup packet contents, shall be sent to the upper layer. (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[Slave will perform asynchronously sniffing during polling and unspecified channel recovery if Slave scans during polling = 0x00 (see **Table 123 (Master parameters from host)**).]

TLMM_S_448 - POWER INHIBIT - CHANNEL LISTEN TIMER FOR SCAN DURING POLLING

While slave is configured to do scan during polling, if power inhibit interrupt is asserted while slave is listening on one channel during polling or unspecified channel recovery, slave will stop the channel listen timer. Then when power inhibit interrupt is deasserted, slave will restart the channel listen timer and listen on the same channel.

[Slave will change the scan channel during polling or unspecified channel recovery only if the "channel listen timer" times out while power inhibit status is low (i.e., power inhibit has been deasserted or power inhibit has never been asserted.)]

TLMM_S_449 - POWER INHIBIT - MEDICAL EVENT TRANSMIT TIME

If power inhibit interrupt is asserted before the "medical event transmit time" timesout (while slave is uplinking medical event on one channel), slave will stop the medical event transmit timer. Then when power inhibit interrupt is deasserted, slave will uplink medical event on the same channel and restart "the medical event transmit time" .

[Slave will change the channel during uplinking medical event only if the "medical event transmit time" times out while power inhibit status is low (i.e., power inhibit has been deasserted or power inhibit has never been asserted.)]

TLMM_S_450 - POWER INHIBIT - OPEN RESPONSE PACKETS

If power inhibit interrupt is asserted while slave is uplinking open response packets,

- when slave is configured to do scan during polling - slave will perform idle immediate and stop the channel listen time. Then when power inhibit interrupt is deasserted, slave will listen on the same channel and restart the channel listen time;
- when slave is configured to do asynchronously sniffing/scan during polling - slave will perform idle immediate. Then when power inhibit interrupt is deasserted, slave will listen on the same channel.

[Note the native mode timer is not stopped and this is because native mode timer is much longer than power inhibited duration.]

- when slave is running the medical event service duration timer,slave will perform idle immediate and stop the medical event transmit time. Then when power inhibit interrupt is deasserted, slave will uplink medical event on the same channel and restart "the medical event transmit time"

TLMM_S_451- CHARGE NOISE - USE NEW CHANNEL RECOVERY THRESHOLD

When slave host sends the Power Inhibit Service Request with sub-type = 0x04 (see **Table 72 (POWER INHIBIT SERVICE REQUEST (SLAVE))**, then slave mozart will use the "New Channel Recovery Trigger Threshold For Nominal mode" and "New Channel Recovery Trigger Threshold For Max. Transmit Mode" from **Table 124 (SLAVE PARAMETERS FROM HOST)** to trigger the channel recovery.

[The slave transmit packet's control byte 5, bits 5 and 6 shall NOT mirror the power inhibit service request sub-type when the sub-type = 0x04. Bits 5 and 6 will contain the last power inhibit mode uplinked prior to mode 4.]

TLMM_S_453- CHARGE NOISE - SAME CHANNEL RECOVERY IMMINENT

If the power inhibit mode is 4 and configured (i.e., Send Channel Recovery Imminent Indication to host = 1, see Parameter "Power Inhibit" in **Table 124 (SLAVE PARAMETERS FROM HOST)**), then slave mozart shall send Same Channel Recovery Imminent Indication (**Table 74 (SAME CHANNEL RECOVERY IMMINENT INDICATION (SLAVE))**)to slave host when "New Channel Recovery Imminent Threshold For Nominal mode" or "New Channel Recovery Imminent Threshold For Max. Transmit Mode" in **Table 124 (SLAVE PARAMETERS FROM HOST)** is reached.

TLMM_S_454- CHARGE NOISE - SAME CHANNEL RECOVERY IMMINENT RESUMED

If the power inhibit mode is 4 and configured (i.e., Send Channel Recovery Imminent Resumed Indication to host = 1, see Parameter "Power Inhibit" in **Table 124 (SLAVE PARAMETERS FROM HOST)**), and when the first packet is received with good MAC after Same Channel Recovery Imminent Indication has been sent, then slave mozart shall send Same Channel Recovery Imminent Resumed Indication (**Table 75 (SAME CHANNEL RECOVERY IMMINENT RESUMED INDICATION (SLAVE))**) to slave host.

TLMM_MS_353 - SHUTDOWN - SERVICE REQUEST PARAMETERS

When invoking a shutdown service request, upper layer will information as specified in **Table 76 (SHUTDOWN SERVICE REQUEST)**.

TLMM_MS_354 - SHUTDOWN - RESPONSE TO UPPER LAYER

Upon the receipt of the shutdown service request, data link will perform the following actions:

1. Terminate any data transmission and/or receive operations,
2. Start a disable timer and send a shutdown response to host

[Cancel all the timers, Transition to a disable state, configure the module to be in a receptor mode (per initialization parameters). Terminate physical layer communication, transition to disable state, send a successful response. Note that before turning off the 2 volt supply, the slave will return to synchronous receptor wakeup operation. If the wakeup disable bit is set to 1 in the initialization request firmware does not set the wakeup enable bit bp_wakeup_en bit in REG_BP_WU_CTRL0 register. See requirement **TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP**]

TLMC_M_100 - ARQ - STOP AND WAIT MASTER ACKNOWLEDGMENT

After transmitting the Memory Data master packet, check for the ACK bit in the slave packet(s) and perform as follows:

1. If the ACK bit is set in at least one valid slave packet, then transmit the next Memory Data master packet (if the previous packet was not the final packet).
2. If the ACK bit is not set in any of the received slave packet(s) or no valid slave packets are received, then retransmit the previous master packet in the next frame.

TLMC_MS_101 - ARQ - MASTER ARQ PACKET FORMAT

A master ARQ Request packet has the format indicated in **Fig. 102 (Master ARQ Request Packet)**.

TLMC_M_102 - ARQ - UPDATING OF ARQ INFORMATION

The master will update the ARQ information if it receives a memory packet (if the master packet was not Discover, Open, Network, or Security).

[The state is Comm when the master packet is not Discover, Open, Network nor Security.]

TLMC_M_103 - ARQ - SLAVE MEMORY DATA PACKET ACKNOWLEDGMENT

For each slave Memory Data packet that is received:

1. Verify the sequence number of the received packet to ensure it is within the sliding window limits. A received packet with a sequence number outside the sliding window limits will be ignored.
2. Adjust the SR-ARQ information based on the received packet. Update the FOP field with the sequence number of the First Outstanding Packet required to be retransmitted. In the bitmap, clear the bit for the packet received.
3. Transmit the ARQ packet to the slave in the next master frame.

[Note that the last bit of the bitmap is never used (always 1).]

TLMC_M_104 - ARQ - SLAVE ACK NOT RECEIVED

When no Memory Data packet is received in the slave frame then perform the following:

1. If the previous master packet was an ARQ packet and it is not acknowledged (ACK bit not set) in the slave frame, then retransmit the ARQ packet in the next master frame.
2. The ARQ packet will stop when ARQ packet is acknowledged via ACK bit set in at least one slave packet.

TLMM_M_355 - ARQ - UPDATING THE EXPECTED WAVEFORM SEQUENCE NUMBER 1

When “delay new waveform detection” is false, the transaction number of the most recently received waveform packet will be used to set the expected waveform sequence number:

- If all the fragments (packets) from the transaction has been received, the expected waveform sequence number will be the received transaction’s (last waveform sequence number+1) modulus 256.
- If any of the fragments (packets) from the transaction have not been received, the expected waveform sequence number will be the waveform sequence number of the first missing fragment (packet).

[The “delay new waveform detection” parameter is contained in the master initialization table (see **Table 123 (Master parameters from host)**).

The master communicates the “expected waveform sequence number” using control byte 4. See **Fig. 17 (Packet Control Bytes)**.]

TLMM_M_429 - ARQ - UPDATING THE EXPECTED WAVEFORM SEQUENCE NUMBER 2

When “delay new waveform detection” is true, the expected waveform sequence number will set according to the table below: Note that the second most recently received waveform transaction is discarded when the new waveform transaction number is any value other than (N+1) % 8, where N is the transaction number of the second most recently received waveform transaction.

[For example, the master will continue to request retries from waveform transaction 1 until it receives a packet from waveform transaction 3 (actually any transaction other than transaction 2).

The “delay new waveform detection” parameter is contained in the master initialization table (see **Table 123 (Master parameters from host)**).

The master communicates the “expected waveform sequence number” using control byte 4. See **Fig. 17 (Packet Control Bytes)**.]

TLMC_S_106 - ARQ - STOP AND WAIT MASTER PACKETS

A master packet received with a valid message authentication code (MAC) will be acknowledged by setting an ACK bit in all the slave packets of the next slave frame; otherwise, the ACK bit will not be set in any slave packet(s) of the slave frame.

[The slave frame may include the retransmitted packets.]

TLMC_S_107 - ARQ - RECEIPT OF AN ARQ PACKET

Upon the receipt of an ARQ packet the slave will perform as follows:

1. Retransmit the unacknowledged Memory Data packets using the FOP and bitmap as follows:
   - FOP indicates the sequence number of the packet not received or may need to be retransmitted, followed by a bitmap status of the remaining packets.
   - A set bit “1” in the bitmap indicates the packet(s) is not received or may need to be retransmitted.
   - A cleared bit “0” in the bitmap indicates a packet was successfully received, therefore should not be retransmitted.
2. Adjust the lower and upper limits of the sliding windows based on the ARQ information.

TLMC_S_108 - ARQ - MASTER ARQ PACKET NOT RECEIVED

If the ARQ packet is not received then re-transmit the unacknowledged Memory Data packet(s) from the previous frame(s).

[If packet space is available in the slave frame.]

TLMC_S_109 - ARQ - IMPLICIT ACK OF VALID SLAVE MEMORY DATA

Once a slave packet is transmitted with a last fragment bit set, and the Memory Data request is received in the following master frame then the slave will perform as follows:

1. Memory data packets already transferred will be considered acknowledged.
2. The sliding window limits will be advanced accordingly. (New sequence #)

TLMC_S_110 - ARQ - MEMORY DATA ACKNOWLEDGEMENT NOT RECEIVED

If the slave does not receive acknowledgment for the transmitted Memory Data packets from the previous frame then the slave will retransmit the Memory Data packets from the previous frame if packet space is available and if the following apply:

1. No waveform packets to transmit.
2. No supplemental marker packets to transmit.

[For precedence rules, see **5.2.11** and **5.2.11.5 SLAVE DATA**.]

TLMM_S_356 - ARQ - RETRANSMIT WAVEFORM PACKET

The Nth transmitted waveform request packets shall be retransmitted as follows:

- If there is no (N+2)th waveform request, then slave shall retransmit the saved waveform packets starting with the expected waveform sequence number.
- If there is (N+2)th waveform request, and if the “Delay sending new waveform” is true (See **TABLE 124: SLAVE PARAMETERS FROM HOST**) then slave shall retransmit the saved waveform packets starting with the expected waveform sequence number.
- Else if the “Delay sending new waveform” is false, then slave stops all the retransmission and transmission of the Nth waveform request and (N+1)th waveform request packets, slave shall transmit the (N+2)th waveform request packets.

[The master communicates the “expected waveform sequence number” using control byte 4. See **Fig. 17 (Packet Control Bytes)**.]

TLMM_M_357 - CH MAPPING - SELECT CHANNEL

Upon initiation of the channel mapping internal service, the master will perform the clear channel assessment on each channel/antenna pair (10 msec/pair) until all channels are assessed or a channel’s RSSI is below the stop-on-threshold value. The quietest channel is selected (the channel with the lowest RSSI).

Note that the listen before talk timer (LBT) is restarted each time a clear channel assessment is performed.

[Note that a channel’s RSSI value is set to the larger value of the RSSI on each antenna.]

[If ten channels are assessed it will take approximately 275 +/- 50 msec (10 msec per channel/antenna pair).]

TLMM_M_358 - CH MAPPING - DIAGNOSTIC COLLECTION

The master will collect twenty milliseconds of data during diagnostic mapping. The RSSI value of each channel assessed will be returned in the external service response.

TLMC_M_120 - CH RECOVERY - TRIGGER CHANNEL RECOVERY

The Channel Recovery algorithm is triggered when any of the following conditions is true:

1. No valid slave packet is received for N consecutive frames, where N is determined using the parameter “number of consecutive frames without a valid receive packet threshold 1, 2 or 3” and the power inhibit mode. See **Table 123 (Master parameters from host)** and **Table 71 (POWER INHIBIT MODES)** respectively.
2. An Open service request is unsuccessful.
3. A Data, Emergency, Open, or Remote Diagnostic service request is received in the Inactive State (in which case, unspecified channel recovery is initiated).
4. The slave ID in the first data transaction packet or diagnostic transaction packet does not match the slave ID in the last received open packet.
5. A slave packet’s ACK bit is true but its reset ACK bit is false when both are expected to be true. This can occur when noise is received with a good message authentication code (MAC).
6. When the master/slave bit is set to “1”, it means “this” master has just received a native mode packet from another master.

"This" master was expecting a packet from a slave.

[Here, '**data transaction**' is used generically for Memory Data, WF Data, and Unrequested Data. The ID appears in the first fragment of the transaction.]

TLMC_M_121 - CH RECOVERY - START SAME CHANNEL RECOVERY

When the Channel Recovery algorithm is triggered (when the session began with an Open service request), then disable the security hardware; terminate the processing of any external service in progress; change to a nominal data transfer mode; inform the upper layer with a channel recovery started indication (see **Table 80 (SAME CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**; and start channel recovery by transmitting Open requests. Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (Rm) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (Rm) can be used for the duration of same channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

[Change to Channel Recovery State and start SCR timer, Inform the Upper Layer if an external service is aborted, inform the Upper Layer that Channel Recovery is started, Same channel recovery can't be greater than 5 seconds.]

TLMC_M_122 - CH RECOVERY - SAME CHANNEL RECOVERY OPEN MASTER PACKET

While in same channel recovery transmit an open request packet and wait for an open response. Repeat this process if a valid open response is not received. Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (Rm) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (Rm) can be used for the duration of same channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[During same channel recovery while SCR timer is active - Transmit an Open request, starting on an antenna and alternating on each retry, Restart the sync mode, Wait for an Open slave packet response. Repeat the above steps if a valid Open slave response is not received. Note: Open service is invoked locally. The actions performed for this Open service are the same as those described in **5.2 External Services**. The only exception is that upper layer is not notified of an Open request and the Open response.]

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

TLMC_M_123 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL

If a valid slave Open packet is received while the SCR timer is active then transition to Comm State on successful response and send a channel recovery successful indication to the upper layer (see **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**. Additionally initialize the block cipher. See **Fig. 45 (Open - block cipher initialization)**for a secure/unsecure session initialization.

[Cancel the SCR timer; inform the Upper Layer that Channel Recovery is successful; initialize various parameters as done when Open service is processed prior to transitioning to the Comm State.]

TLMC_M_124 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL

If a valid slave Open response is NOT received within the same channel recovery time then:

1. Start a UCR timer.
2. Select a unspecified channel via Channel Mapping internal service.
3. Start the Listen Before Talk timer.
4. Transmit wakeup packet(s) per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**.
5. Transmit Open request (on the left antenna)
6. Send an unspecified channel recovery started indication to the Upper Layer. See **Table 82 (UNSPECIFIED CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (Rm) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (Rm) can be used for the duration of unspecified channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

[Note that the same channel duration cannot be greater than 5 seconds, otherwise the Channel Mapping algorithm will have to be initiated to reselect a channel (per the FCC).]
[Note, it is not required to have a new random number generated when entering the Unspecified Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Unspecified Channel Recovery starts during an unsecured session.]

TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION

The master will send wakeup packets as follows:

1. Transmit no wakeup packets in the polling (scanning on) case if the slave is still polling: (timenow < timelast native packet from slave + slave recovery duration + remaining polling duration)
2. Transmit two wakeup packets in the polling (scanning off) case and the slave is still polling: (timenow < timelast native packet from slave + slave recovery duration + remaining polling duration)
3. Transmit N wakeup packets otherwise; where N is chosen to be slightly longer than the slave’s wakeup period. “N” is the parameter transmit wakeup burst length async.

[transmit wakeup burst length async is received from master parameters from host - see **Table 123 (Master parameters from host)**. Note that the antenna is toggled before the start of each wakeup burst.
Also note that the master will perform clear channel assessment before and after transmitting wakeup (if the wakeup transmission is not suppressed and the slave scans during polling). The master will restart the Listen Before Talk (LBT) timer after the second clear channel assessment. This minimizes the interaction between the time it takes the slave to scan all channels and the time before the master starts transmitting wakeup again.]

TLMC_M_261 - CH RECOVERY - TOO MANY SAME CHANNEL RECOVERIES

If too many same channel recoveries are initiated in too short of a time, then unspecified channel recovery will be initiated. More specifically, if N same channel recoveries are initiated within T seconds then:

1. Start a UCR timer.
2. Disable the security hardware.
3. Select an unspecified channel via Channel Mapping internal service.
4. Start the listen before talk timer.
5. Transmit wakeup packet(s) per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**
6. Transmit open request (on the left antenna).
7. Send an unspecified channel recovery started indication to the Upper Layer. See **Table 82 (UNSPECIFIED CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (Rm) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (Rm) can be used for the duration of unspecified channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).
[The values of “N” and of “T” (above) come from **Table 129 (PARAMETERS FROM EEPROM)**. They are called “too many recoveries threshold” and “too many recoveries duration” respectively.]
[Note, it is not required to have a new random number generated when entering the Unspecified Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Unspecified Channel Recovery starts during an unsecured session.]

TLMC_M_125 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY

While the UCR timer is active, then:

1. Transmit an Open request packet, starting on one of the two antennas, and alternating the antenna on each retry.
2. Wait for a valid Open response packet.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted. Note that if open secure packets are transmitted, a new random number (Rm) must be generated (see **Fig. 40 (Open Secure/Unsecure Request Packet)**). The same random number (Rm) can be used for the duration of unspecified channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).
[Before Wait..., restart the sync mode.]
If there is no Open response, transmit Open on the other antenna. Continue until success, Listen Before Talk timeout, or UCR timeout.

[Note: Open service is invoked locally. The actions performed for this Open service are the same as those described in **5.2 External Services**. There are two exceptions. The first exception is that the upper layer is not notified of an Open request packet and the Open response packet. The second exception is that a "slave polling" timer is started (to interleave wakeup bursts with native mode open packets if the timer expires).]

[Note, it is not required to have a new random number generated when entering the Unspecified Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Unspecified Channel Recovery starts during an unsecured session.]

TLMC_M_126 - REMAP CHANNEL - UNSPECIFIED CHANNEL RECOVERY TIMER EXPIRATION

If no Open response at expiration of Listen Before Talk timeout:

1. Reselect best channel.
2. Restart Listen Before Talk timeout.
3. Transmit wakeup packet(s) per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**
4. Transmit an open request packet.
5. Wait for a valid Open response packet.

Open (unsecure) packets are transmitted if the original open service was open unsecure; otherwise open secure packets are transmitted.

Synchronous communication is reestablished on a different channel if an Open session response is received from an slave prior to the expiration of the UCR timer. The UCR timer is canceled and synchronous communication is resumed by transition to the Comm State.

TLMC_M_127 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL

If a valid Open response packet is received while the UCR timer is active then transition to Comm State and send a channel recovery successful indication to the upper layer (see **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**). See **Fig. 45 (Open - block cipher initialization)** for a secure/unsecure session initialization.

[Cancel the UCR timer; inform the Upper Layer that Channel Recovery is successful; initialize various parameters as when Open service is processed prior to transitioning to the Comm State. Note: Open service is invoked locally. The actions performed for this Open service are the same as described in **5.2 External Services**. The only exception is that the upper layer is not notified of an Open request packet and the Open response packet.]

TLMM_M_360 - CH RECOVERY - NATIVE TIMEOUT

The master will continue to send open packets for the native mode timeout duration see **Table 129 (PARAMETERS FROM EEPROM)**. If this timer expires it will send wakeup packets per requirement **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**.

[Note that the native mode timer is not used if the slave scans during polling. In this case a transition back to wakeup occurs upon an LBT timeout.]

TLMC_M_128 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL

A channel recovery failure indication will be sent to the upper layer if:

a. No open response packet is received prior to the expiration of the UCR timer, or
b. An open response packet with a status of "no BAN key" is received.

See **Table 83 (CHANNEL RECOVERY FAILED INDICATION (MASTER))**

[Transition to Inactive State, inform the Upper Layer of Channel Recovery failure, Note: Transition to Inactive State means that physical layer will be transferred to an idle state.]

TLMC_S_129 - CH RECOVERY - TRIGGER CHANNEL RECOVERY

The Channel Recovery algorithm is triggered when

1. No valid master packet is received for N consecutive frames, where N is determined using the parameter "number of consecutive frames without a valid receive packet threshold 1, 2 or 3" and the power inhibit mode. See **Table 123 (Master parameters from host)** and **Table 71 (POWER INHIBIT MODES)** respectively.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:
0x0F0A : Bit3 - Received Master Packet Error Exceeded.]

2. The master ID in the master data transaction, master ARQ packet, or master Diagnostic packet does not match the master ID in the last received Open request packet; or

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:
0x0F0A :Bit4 -Received Master Type is ARQ and Master ID Mismatched,
0x0F0A: Bit5 - Received Master Type is Diagnostic and Master ID Mismatched, or
0x0F0A: Bit6 - Received Master Type is Memory Data and Master ID Mismatched.
Also Note that the master ID appears in data transaction's first fragment.]

3. No valid packet is received from the master after a secured open.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:
0x0F0A : Bit0 - First Received packet is Invalid Secured packet after open session, or
0x0F0B: Bit0 - No packet is received after open session.]

4. The master/slave bit is set to “0” it means “this” slave has just received a native mode packet from another slave. “This” slave was expecting a packet from a master.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:
0x0F0B : Bit5 - Received Slave Packet while in Communication State.]

5. Power inhibit is asserted and the power inhibit mode is idle immediate. Note that for this case the slave will not begin listening in native mode until power inhibit is deasserted.

[The following bit in **Table 108 (Channel Recovery Reason)** will be set:
0x0F0A : Bit7 -Power Inhibit triggered channel recovery.]

Once Channel recovery is triggered, the security hardware will be disabled.

The model ID and serial number information of the instrument with which the communication was successfully opened will be maintained until a different channel recovery fails.

TLMC_S_130 - CH RECOVERY - START SAME CHANNEL RECOVERY

When the Channel Recovery algorithm is triggered, transition to Channel Recovery State, start the SCR timer and send a channel recovery started indication to the upper layer. (see **Table 80 (SAME CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**).

Note that the channel recovery started indication will be sent to the upper layer only if bit “Send channel recovery indication to host” is set to “1”. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[Terminate the processing of any external service in progress, stop slave transmission, and restart sync mode; change to nominal data transfer mode; inform the upper layer that Channel Recovery is started.]

TLMC_S_131 - CH RECOVERY - SAME CHANNEL RECOVERY INVOKE SCAN

While the SCR timer is active listen for a valid Open request packet from the master with which the session was opened.

[If any packet other than Open is received, then restart the sync mode and no slave response is transmitted.]

TLMC_S_132 - CH RECOVERY - SAME CHANNEL RECOVERY SUCCESSFUL

While in Channel Recovery State, if the SCR timer is active and a valid Open request packet is received, then:

Transmit N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**). Note that a new random number (Rs) for the open response packet (see **Fig. 42 (Open Secure/Unsecure Response Packet)**) is generated when entering the same channel recovery and will be used for the entire (same and unspecified) channel recovery process. This is allowed because this random number (Rs) will not be used in transmitting valid information during channel recovery. The algorithm for generating a random number is described in the appendix (see section **7.3 Pseudo Random Number Generation**).

1. Initialize the block cipher. See **Fig. 45 (Open - block cipher initialization)** for a secure/unsecure session initialization.
2. Transition to the Comm State.
3. Send a channel recovery successful indication to the upper layer. See **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**. Note that the channel recovery successful indication will be sent to the upper layer only if bit “Send channel recovery indication to host” is set to “1”. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[When a valid open request packet is received followed by a diagnostic or data packet, cancel the SCR timer; inform the upper layer of successful Channel Recovery status; initialize various parameters as done when Open service is processed.]

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

TLMC_S_133 - CH RECOVERY - SAME CHANNEL RECOVERY UNSUCCESSFUL

While in Channel Recovery State, if the SCR timer expires and a valid Open request packet is NOT received, then start UCR timer and send an unspecified channel recovery started indication to the upper layer (see **Table 82 (UNSPECIFIED CHANNEL RECOVERY STARTED INDICATION (MASTER OR SLAVE))**).

Note that the unspecified channel recovery started indication will be sent to the upper layer only if bit “Send channel recovery indication to host” is set to “1”. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[Restart sync mode; report the Channel Recovery status to the application layer.]

TLMC_S_134 - CH RECOVERY - START UNSPECIFIED CHANNEL RECOVERY

While performing channel recovery, if the SCR timer expires and a valid open request packet is not received start the UCR timer. Additionally the slave will do the following:

1. Scanning off case
    a. Phase 1: listen for wakeup. The slave will continuously restart asynchronous receptor wakeup if it receives no valid wakeup
    b. Phase 2: listen for native mode open. The slave will listen for a native mode open packet if a valid wakeup is heard (and start a native mode timer). It will listen for an open packet until a valid open is received or a native mode timeout occurs (~200 msec). If a native mode timeout occurs, the slave will transition to phase 1.
2. Scanning on case
    a. Phase 1 - random channel listening - the slave will randomly select a channel and will restart the channel listening time and listen for a valid packet from the master. If no valid open packet has been received and the service duration timer is still active, the slave will switch to the phase 2 approach.
    b. Phase 2 - sequential channel listening - the slave will sequentially select the next channel and will restart the channel listening time and listen for a valid packet from the master. When every channel has been selected once the slave will switch to the Phase 1 approach.

TLMC_S_135 - CH RECOVERY - UCR TIMER ACTIVE NON OPEN MASTER PACKET

While UCR timer is active, if a master packet other than Open is received, the slave will not transmit a slave response.

TLMC_S_136 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY SUCCESSFUL

While the UCR timer is active and a valid Open request packet from the same master is received, then:

Transmit N open response packets (“N” = open uplink packet count in **Table 129 (PARAMETERS FROM EEPROM)**). Note that the random number (Rs) used in the open response packet (see **Fig. 43 (Open status byte (Response))**) was generated when entering the (same) channel recovery. This is allowed because this random number (Rs) has not been used in transmitting valid information.

1. Initialize various parameters as when an Open service is processed.
2. Initialize the block cipher. See **Fig. 45 (Open - block cipher initialization)** for a secure/unsecure session initialization.
3. Transition to the Comm State.
4. Send a channel recovery successful indication to the upper layer (see **Table 81 (CHANNEL RECOVERY SUCCESSFUL INDICATION (MASTER OR SLAVE))**). Note that the channel recovery successful indication will be sent to the upper layer only if bit “Send channel recovery indication to host” is set to “1”. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[When a valid open request packet is received followed by a diagnostic or data master packet, cancel the UCR timers; inform the upper layer of successful Channel Recovery status; various parameters like seq number are initialized prior to transitioning to the communication state.]

[Note, it is not required to have a new random number generated when entering the Channel Recovery during an unsecured session. However, for implementation simplicity, it is acceptable if a new random number is generated as well when the Channel Recovery starts during an unsecured session.]

TLMM_S_361 - CH RECOVERY - UNSPECIFIED CHANNEL RECOVERY UNSUCCESSFUL

While in Channel Recovery State if the UCR timer expires then transition to the Standby State and send a channel recovery failed indication to the upper layer. See **Table 84 (CHANNEL RECOVERY FAILED INDICATION (SLAVE))**.

Note that the channel recovery failed indication will be sent to the upper layer whether the bit “Send channel recovery indication to host” is set to “1” or not. (See **Table 124 (SLAVE PARAMETERS FROM HOST)** for bit “Send channel recovery indication to host”)

[Cancel the Scan listen timer; report the unsuccessful Channel Recovery status to the upper layer.]

[The indication to the Upper Layer will consist of Polling ON or Polling OFF based on the value of the Total Polling Duration Time in the open request packet (i.e., Zero meaning Polling OFF, otherwise Polling ON.]

TLMC_M_139 - DATA TRANSFER MODE - MODE TRANSITIONS

The data transfer mode will be selected according to **Table 86 (DATA TRANSFER MODE TRANSITIONS)**.

[The value of “N” above comes from **Table 129 (PARAMETERS FROM EEPROM)**. It is called “low energy threshold (consecutive filler master packets)”.]

TLMC_M_140 - DATA TRANSFER MODE - NOMINAL TO LOW ENERGY MODE TRANSITIONS.

When low energy mode is enabled and if the consecutive filler packet count is greater than or equal to the low energy threshold value, then data transfer mode transition is made from nominal to low energy.

[See **Table 129 (PARAMETERS FROM EEPROM)**.]

TLMC_M_141 - DATA TRANSFER MODE - NEGATIVE PREAMBLE RATIO IN LOW ENERGY MODE

The master will use a negative correlation preamble on N - 1 of N packets when the master and the slave are both in the low energy mode unless noted below. The Nth master packet will use a positive correlation .

Exception to the N-1 of N downlinks include the following:

- The parameter "frequent waveform expected" is true, or
- There is a new downlink sequence number for the go back N waveform acknowledgement and the parameter "frequent waveform expected" is false, or
- The ack bit is set to false in any uplink packet received with a good MAC and the parameter "frequent waveform expected" is false, or
- The uplink frame contains no packets with good MAC and the parameter "frequent waveform expected" is false

[Note: The “N” Correlate Negative Ratio value is defined in **Table 129 (PARAMETERS FROM EEPROM)**.]

TLMC_M_142 - DATA TRANSFER MODE - LOW ENERGY TO NOMINAL TRANSITION

A transition from low energy to nominal mode will be made if the present master packet is a non-filler packet and the previous master packet was a filler packet in a low energy mode.

TLMC_M_143 - DATA TRANSFER MODE - TRANSITION TO NEW MODE ON MASTER PACKET

If the new data transfer mode packet is successfully received by the slave, then the master transitions to the new data transfer mode. Note the master will transition immediately when the nominal packet count is one packet.

[Note that channel recovery will be triggered if the first N1-1 slave packets are corrupted upon nominal to max slave throughput or max slave throughput to nominal transitions. This occurs since one station has its slave packet count set to N1 (see parameter ‘packet count for nominal mode’ in **Table 123 (Master parameters from host)**) and the other has its packet count set to N2 (see parameter ‘packet count for max throughput mode’ in **Table 123 (Master parameters from host)**). Physical layer timing cannot be maintained in this scenario.]

[The current transfer mode is set to the pending transfer mode when the slave packet indicates the slave has changed.]

TLMC_S_144 - DATA TRANSFER MODE - VALIDATE TRANSFER MODE AND RESPONSE

Upon receipt of a valid master packet, the slave will check the telemetry data transfer mode, and will perform as follows:

1. If the current data transfer mode matches the transfer mode in the master packet then the slave data transfer mode will not change.
2. If the current data transfer mode does not match the transfer mode of the master packet then the slave will transition to the data transfer mode as specified in the master packet.
3. The slave packet will contain the slave’s data transfer mode indication.
4. When in low energy mode slave will always uplink a filler packet when there is no data to be transmitted.

TLMM_M_362 - RESET SERVICE - INVOKE SERVICE 1

The master will invoke the reset internal service upon receipt of an emergency service request.

TLMM_M_363 - RESET SERVICE - INVOKE SERVICE 2

The master will invoke the reset internal service upon receipt of a data service request when the most recent data or emergency service request failed.

TLMC_M_145 - RESET SERVICE - MASTER PACKET WITH RESET BIT

The master will do the following when the reset internal service is invoked:

1. Discard pending master data packet
2. Reset the master data packet sequence number
3. Set the reset bit to 1 in the master upper layer header byte. See **5.2.11.2**.
4. Set the Reset bit to 1 in the control byte of the next master packet.

The master then will wait for an slave packet with Reset ACK bit set in the Control Byte.

[An emergency service may result in discarded data packets. (Emergency interrupts a data service in progress. A data service will not result in discarded data packets since a data service cannot interrupt an “in-progress” service.)]

TLMC_M_146 - RESET SERVICE - MASTER RESET REQUEST RETRANSMISSION

If no valid slave packet with Reset ACK bit set is received from the remote station in the next slave frame, the Data Link Layer shall continue to retransmit the packet with the Reset bit set. The master shall continue this until a valid slave packet with Reset ACK bit set is received or the reply duration timeout occurs.

TLMC_S_147 - RESET SERVICE - RESET REQUEST VALIDATION

Upon receiving a valid master packet with the Reset bit set, the slave shall do the following:

1. Discard pending slave data packets.
2. Reset parameters for the ARQ algorithm, including the master data packet sequence number, the slave Memory Data sequence number, transaction number, fragment number; and the ARQ sliding window.

3. Set the Reset ACK bit in the slave packets.

TLMC_M_148 - RESET SERVICE - OPERATION SUCCESSFUL

Upon receiving a valid slave packet with the Reset ACK bit set, the master shall do the following:

1. Discard pending data slave packets.
2. Reset parameters for the ARQ algorithm, including the slave Memory Data sequence number, transaction number, fragment number, and the ARQ sliding window.

TLMM_M_364 - WIRELESS WAKEUP - INITIATE INSTIGATOR WAKEUP

A telemetry M instigator node, by initiating a external service: discover, open, security, or network shall invoke the wakeup internal service. Channel recovery shall also invoke the wakeup internal service. See **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**

[Information pertaining to wakeup, provided from the external service, will be transferred to the wakeup internal service. See**Table 87 (INSTIGATOR WAKEUP VS. SERVICE)**]

TLMM_MS_365 - WIRELESS WAKEUP - WAKEUP PACKET

The wakeup packet shall be defined as the following: see **Fig. 112 (Wakeup Packet)**

[For more information on how the individual wakeup packet bytes are defined see the following requirements:
TLMM_MS_366 - WIRELESS WAKEUP - WAKEUP HEADER BYTE,
TLMM_MS_367 - WIRELESS WAKEUP - WAKEUP SEQUENCE NUMBER BYTES,
TLMM_MS_368 - WIRELESS WAKEUP - WAKEUP STATION ID BYTES,
TLMM_MS_369 - WIRELESS WAKEUP - WAKEUP RECEPTOR BIT MAP BYTE,
TLMM_MS_370 - WIRELESS WAKEUP - WAKEUP CONFIGURATION BYTES,
TLMM_MS_371 - WIRELESS WAKEUP - WAKEUP USER DATA BYTES]

TLMM_MS_366 - WIRELESS WAKEUP -WAKEUP HEADER BYTE

The wakeup packet header byte shall have the following format:

1. Wakeup type - see **Table 89 (WAKEUP TYPES VALUES)**
2. Instigator network table index - see **Table 90 (WAKEUP NETWORK ID INDEX VALUES)**
3. Wakeup user data/network packet security enable - a value of one enables user data or network data security; a value of zero disables user data and network security [See **Table 91 (SECURE USER DATA/NETWORK PACKET)**
4. Last wakeup block - a value of one means this is the final block of wakeup packets; a value of zero means there will be more
5. Synchronize wakeup interval timer - a value of one means synchronize; the value is zero otherwise

[See **TLMM_MS_371 - WIRELESS WAKEUP -WAKEUP USER DATA BYTES** and **TLMM_M_372 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA INSTIGATOR**]

TLMM_MS_367 - WIRELESS WAKEUP -WAKEUP SEQUENCE NUMBER BYTES

The wakeup packet sequence bytes shall have the following format:

1. Wakeup packet sequence number - The instigator data link layer shall write the baseband hardware’s starting sequence number and ending sequence number (where start > end)

[The baseband hardware shall fill in the sequence number, in the wakeup packet, send the wakeup packet, then decrements the fourteen bit sequence number and then shall send another wakeup packet until the end sequence number has been sent.]

TLMM_MS_368 - WIRELESS WAKEUP -WAKEUP STATION ID BYTES

The wakeup packet shall have six station id bytes having the following format:

1. station ID = network ID for the local broadcast wakeup type
2. station ID = receptor ID for the unicast wakeup type
3. station ID = for the non-local broadcast wakeup type, the three most significant bytes are the receptor model and sub-model which can be all wildcard (zero). The three least significant bytes are the instigator model and sub-model.
4. station ID = for the global broadcast wakeup type, the three most significant bytes are the receptor model and sub-model which can be all wildcard (zero). The three least significant bytes are the instigator model and sub-model.

TLMM_MS_369 - WIRELESS WAKEUP -WAKEUP RECEPTOR BIT MAP BYTE

The wakeup packet shall have a receptor map byte of the following format:

1. Each bit shall represent a node (child) in the network
2. A value of one in a given bit shall indicate the node (child) shall be woken up
3. A value of zero in a given bit shall indicate the node (child) shall not be woken up see **Fig. 111 (Receptor bit map)**

TLMM_MS_370 - WIRELESS WAKEUP -WAKEUP CONFIGURATION BYTES

The wakeup packet shall have two wakeup configuration bytes having the following format:

1. User data - a value of one indicates that user data exists; a value of zero indicates that no user data exists
2. Channel number - channel number of 1 to 30 to be used in native mode

3. Slave scans during polling - a value of one means the slave should scan during polling, a value of zero means the slave should alternate between native mode listening and consecutive asynchronous receptor wakeup
4. Instigator native mode transmit modulation - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
5. Instigator default transmit data rate - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**, used outside of the session.
6. Instigator native mode receive modulation - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**
7. Instigator default receive data rate - see **Table 92 (TX/RX DEFAULT DATA RATE TABLE)**,used outside of the session.

TLMM_MS_371 - WIRELESS WAKEUP -WAKEUP USER DATA BYTES

The wakeup packet shall have 10 bytes of user data, these bytes will be provide as part of initialization data from the host. When wakeup user data security is enabled only 4 bytes shall be available for user data. The remaining 6 bytes shall be used for a 3 byte security counter and for a 3 byte security MAC.

[When security is enabled, the first 4 bytes are used for the user data bytes. Bytes 5,6 and 7 are used for the security counter. Bytes 8,9, and 10 are used for the security MAC.]

TLMM_M_372 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA INSTIGATOR

The instigator will perform the following to secure the wakeup user data bytes when the Wakeup security byte is enabled:

1. Generate an random number and use 3 bytes as the instigator’s secure wakeup counter (CTRi).
2. Calculate the AES_out using the block cipher using the instigator’s secure wakeup counter (CTRi), instigator’s station ID (IDi) the four user data bytes and the receptor bit map for the data input, and the BAN key (KBAN) for the key input.
3. Encrypt the 4 bytes of user data by calculating the key stream using the block cipher shown:
4. Complete the user data encryption by performing an exclusive or of the user data with the key stream. created in step 3.

The 10 bytes of user data will contain the following bytes: four bytes encrypted user data, three byte of instigator secure wakeup counter (CTRi), and three byte of MAC (AES_out[23:0].

TLMM_M_373 - WIRELESS WAKEUP -WAKEUP INTERVAL SYNC START TIME

To determine when the wakeup packets transmissions should start, the instigator shall set the wakeup interval sync start time.

[This defines the amount of time the wakeup transmission should start before the receptor’s sniff interval starts. This is needed to compensate for clock drift between the instigator and receptor(s). The wakeup interval sync start time is also needed to account for the CCAs that will occur before the next sync interval.

To achieve a successful synchronous wakeup transmission, the instigator’s sniff interval should match the receptor’s sniff interval. The instigator’s sniff interval is not in the EEPROM nor the parameters from host. This should be set with a local data service request or via the (address, value) pairs of the master initialize service.]

TLMM_M_374 - WIRELESS WAKEUP -WAKEUP INITIAL CCA

The instigator shall perform a clear channel assessment (CCA) before the start of wakeup packet transmission.

[The above requirement applies to both synchronous instigator wakeup and asynchronous receptor wakeup.]

TLMM_M_441 - WIRELESS WAKEUP - CCA AFTER WAKEUP

If instigator is configured to do “CCA before and after wakeup = 1” and “slave scan during polling =1” (see **Table 123 (Master parameters from host)**), the instigator shall perform a clear channel assessment (CCA) after the last block wakeup packets have been transmitted.

[The above requirement applies to both synchronous instigator wakeup and asynchronous receptor wakeup.But it does not apply to the network service]

TLMM_M_375 - WIRELESS WAKEUP -WAKEUP WITH CCA BREAKS

The instigator shall perform a clear channel assessment (CCA) every five seconds. This means that a wakeup burst duration longer than five seconds will be separated by a CCA. See **Fig. 119 (Wakeup with CCA breaks)**

[The above requirement applies to asynchronous receptor wakeup. It also applies to synchronous receptor wakeup, although synchronous receptor wakeup should rarely use a long wakeup burst (greater than 5 seconds). Note the data link layer will be notified, by the baseband hardware, when all of the wakeup packets have been sent. The data link layer will then take the next action of the discover service, open service, network service, medical event service, or channel recovery internal service (or more wakeup packets if necessary).]

.

TLMM_S_377 - WIRELESS WAKEUP - WAKEUP SNIFF FREQUENCY RANGE

The slave (receptor) shall be configurable to sniff channels in the following ways:

1. MICS only or
2. MEDS only or
3. Both MICS and MEDS

[Wakeup sniff list is defined in EEPROM, see wu_sniff_list in **A44088**, **Telemetry M RF Module EEPROM Specification** .

The sniff list can be overwritten by slave host via initialization request, see **Table 124 (SLAVE PARAMETERS FROM HOST)**]

TLMM_S_378 - WIRELESS WAKEUP - SNIFF INTERVAL

Upon initialization the node will setup the sniff interval timer to periodically wakeup the baseband hardware to sniff for a wake-up packet.

[Wakeup sniff interval is provided by slave host via initialization request, see **Table 124 (SLAVE PARAMETERS FROM HOST)**]

TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP

If the wakeup disable bit in the initialization request is set to 1, the slave will not sniff for wakeup packets.

[The wakeup disable bit is provided by slave host via initialization request, see **Table 124 (SLAVE PARAMETERS FROM HOST)**.

If the wakeup disable bit is cleared, the slave hardware should sniff when the standby timer or the disable timer expires.

If wakeup disable bit is set, the slave hardware wakeup state machine is disabled and the module will not sniff for any wakeup packets.]

TLMM_S_379 - WIRELESS WAKEUP- SECURE WAKEUP USER DATA RECEPTOR

The receptor will perform the following upon receipt of a wakeup packet containing secure user data (wakeup user data security bit set to enabled):

1. Use the received instigator's secure wakeup counter (CTRi contained in the wakeup packet).
2. Calculate the key stream using the block cipher. See **Fig. 117 (Calculate keystream to encrypt/ decrypt user data in a wakeup packet)**.
3. Decrypt the user data by performing an exclusive or of the user data with the key stream produced in bullet 3 above. See **Fig. 118 (Encrypt/ decrypt the wakeup data)**.
4. Calculate the AES_out using the block cipher using the instigator's secure wakeup counter (CTRi), instigator's station ID (IDi) and the four user data bytes, and the BAN key (KBAN) for the key input. See **Fig. 116 (Calculate AES_out for wakeup security)**.
5. If the calculated MAC (AES_out[23:0]) is equal to the received user data (bytes 8, 9, and 10 of the user data),
   the user data is valid and wakeup indication will be sent to the host.(see **Table 93 (WAKEUP INDICATION (SLAVE))**);
   else,
   the user data is invalid and and wakeup indication with invalid MAC will be sent to the host.(see **Table 93 (WAKEUP INDICATION (SLAVE))**);

TLMM_S_380 - WIRELESS WAKEUP - WAKEUP COMPLETE INDICATION

A wakeup indication "Wakeup Complete", containing the wakeup packet contents, shall be sent to the upper layer when the "last packet last block" wakeup packet is received (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[Note, user data may be present in the "Wakeup Complete" wakeup packet contents. Upon receiving a "Wakeup Complete", the host will normally issue a initialize service request

Also note, an implicit wakeup indication (the same indication as described above) will be sent to the upper layer under the following conditions:

a. A wakeup packet (non last block) is heard, and

b. No subsequent wakeup packet (last block) is heard when performing asynchronous sniff. This asynchronous sniff is designed to occur during the last block. The subsequent wakeup packet (last block) is less likely to be heard if the last wakeup block is short.]

TLMM_S_381 - WIRELESS WAKEUP - WAKEUP IN PROGRESS USER DATA AVAILABLE INDICATION

When valid user data is present, in the wakeup packets, a wakeup indication "Wakeup In Progress User Data Available" containing the wakeup packet contents shall be sent to the upper layer upon receipt of the first valid wakeup packet containing user data (see **Table 93 (WAKEUP INDICATION (SLAVE))**).

[When invalid user data is present, in the first valid wakeup packet, no wakeup indication shall be sent to the upper layer.]

TLMM_S_382 - WIRELESS WAKEUP - WAKEUP COMPLETE INDICATION ONLY

If the first valid wakeup packet, containing valid user data, is also the "last packet last block" wakeup packet; then only the "Wakeup Complete" indication shall be sent to the host.

If the first valid wakeup packet, containing invalid user data, is also the "last packet last block" wakeup packet; then only the "Wakeup Complete with Invalid user data" indication shall be sent to the host.

(see **Table 93 (WAKEUP INDICATION (SLAVE))**)

[Note that if the first wakeup packet is also the last wakeup packet and it has user data, two wakeup indications will be sent to the host. The first will be sent before reading EEPROM. The second will be sent after reading EEPROM. This is done in case the EEPROM read is delayed due to power inhibit. An initialization request while reading EEPROM will be rejected (if initialization sent after the first wakeup indication). Initialization after the second wakeup indication should be successful.]

TLMM_S_383 - WIRELESS WAKEUP - INDICATIONS

The received wakeup packet shall be assembled into an indication message and passed onto the host. The indication message shall have a format defined in **Table 93 (WAKEUP INDICATION (SLAVE))**.

TLMM_M_386 - ANTENNA SELECTION - TRANSMIT

The master will toggle the transmit antenna (vs. the previous transmit antenna) if any of the following conditions are met:

1. The previous master packet was received with a MAC error by the slave (provided the master packet was not sent with negative correlation preamble). The master knows the slave received its packet in error if the slave's ACK bit is false. Note that only the first N-1 packets are checked. In the event that N is equal to one, the single uplink packet is checked.
2. Each of the first (N-1) uplink packets are received with MAC error (or the single uplink packet has a MAC error in the event that N is equal to one.

[“N” can be either of the “slave xmit packet count for nominal mode” or “slave xmit packet count for max throughput mode” as determined by the preceding master packet’s transfer mode. See **Table 123 (Master parameters from host)** for the “packet count” parameters.]

TLMC_M_156 - SIGNAL - STRENGTH INDICATOR

The master shall set the signal strength indicator to give feedback to the user about the link quality.

[signal strength indicator = min(SSI’, last SSI-1) if SSI’ < last SSI
= max(SSI’, last SSI+1) if SSI’ >= last SSI

where,
SSI’ = min(SSI’’,signal strength input limit) if no waveform packet in the slave frame
= SSI’’ otherwise
and
SSI’’ is determined from **Table 94 (SIGNAL STRENGTH INDICATOR)**

Note that “signal strength input limit” is found in **Table 123 (Master parameters from host)**. A value of “7” will disable the input limiting. Hosts without periodic waveform (or infrequent waveform relative to the “superframe interval”) should consider disabling input limiting. Hosts with frequent waveform (i.e., telemetry C Concerto) may set this value to “3”.]

TLMM_M_387 - SIGNAL STRENGTH INDICATOR SETUP

The master will setup the hardware to take periodic signal measurements:

1. Enable power bar measurements
2. Set CCA dwell time to a value smaller than the duration of a slave packet (so the “signal” contains no portion of an inactive packet)

TLMM_MS_391 - PLL LOCK ERROR

Upon receiving the PLL lock error data link layer will:

1. Perform idle immediate and turn off wakeup,
2. Start the trim process by initiating a coarse trim and then a fine trim on all 30 channels.

[The trim process will stop if a trim failure occurs.]

TLMM_M_392 - MASTER TRIM SUCCESS

The master will perform the following if the data link layer initiated trim is successful:

  a. Perform clear channel assessment (see note below)
  b. Transmit wakeup (see note below)
  c. Transmit discover or open (discover if the discover service was interrupted by the PLL lock error, open otherwise)

Note that channel assessment is skipped if trim success occurs during communication (same channel recovery initiated). Also note that the wakeup packets are suppressed if same channel recovery is initiated, or if the master knows the slave is scanning (see **TLMM_M_359 - CH RECOVERY - WAKEUP TRANSMISSION**).

[The remote diagnostic service and the mapping diagnostic service is cancelled if interrupted by a PLL lock error (and a subsequent trim success). The service response status will be 0x26.]

TLMM_S_393 - SLAVE TRIM SUCCESS

The slave will perform the following if the data link layer initiated trim is successful:

Listen for Tel M native mode, listen for TelC packets, listen for asynchronous wakeup, or reconfigure for synchronous sniff (whichever was in progress when PLL lock error occurred).

[That is slave will resume the actions (if possible) before the PLL lock error is occurred:
1)Turn 2Volt off if slave is enabled by PLL lock error, or
2)Perform synchronous sniff if slave is sniffing during Disable CPU ON state when PLL lock error is occurred, or

3)Start same channel recovery if PLL lock error is occurred during communication state,or
4)Restart to listen for native mode in the same channel recovery if PLL lock error is occurred during same channel recovery state,or
5)Restart to listen (for native mode or asynchronous sniffing) in the unspecified channel recovery if PLL lock error is occurred during unspecified channel recovery state,or
6)Restart to listen for native mode or asynchronous sniffing in the acquire state if PLL lock error is occurred while polling in acquire state,or
7)Restart to uplink medical event if PLL lock error is occurred during uplinking medical events, or
8)Restart to listen for network packet if PLL lock error is occurred during network service.]

TLMM_M_394 - MASTER TRIM FAIL

The master will perform the following if the data link layer initiated trim fails:

a. Transition to the standby state (stop downlinking to the slave)
b. Send an external service response (failure) to the host if a service was in progress (see the requirement note below)
c. Send a telemetry ceased indication to the host

[Note that a discover cancelled indication is used for a trim failure during discover since a successful service response is sent when the service is initiated. All other service requests utilize a cancelled service response (failure code 0x1d).]

TLMM_S_395 - SLAVE TRIM FAIL

The slave will perform the following if the data link layer initiated trim fails:

a. Transition to the standby state if not in the disable state (momentarily stop listening to the master)
b. Send a telemetry ceased indication to the host if not in the disable state
c. Start the standby timer if not in the disable state. Note the disable timer is utilized in the disable state.

[Note that if the standby timer expires (or the disable timer expires) the slave will return to synchronous receptor wakeup operation. If the wakeup disable bit is set to 1 in the initialization request firmware does not set the wakeup enable bit bp_wakeup_en bit in REG_BP_WU_CTRL0 register. See requirement **TLMM_S_402 - WIRELESS WAKEUP - DISABLE WAKEUP**]

TLMM_MS_396 - TELEMETRY CEASED INDICATION

The data link layer will send telemetry ceased indication to the host if the following conditions are true:

a. A PLL lock error occurs, and
b. The coarse trim or the fine trim procedure fails, and
c. The state is not disable

[This indication could be generated when PLL lock error is encountered during either the coarse or fine trim when not in the disable state. This could happen during the acquire, communication and channel recovery states.]

TLMM_MS_397 - TRIM FAILURE INDICATION

The data link layer will send a trim failed indication to the host if the following conditions are true:

a. A PLL lock error occurs, and
b. The coarse trim or the fine trim procedure fails, and
c. The "trim failed indication to host" flag is set

[Trim failed indication flag is a configuration parameter - it can be set or cleared by the host]

TLMM_S_398 - PARAMETER CRC UPDATE

The CRC of all the Parameters in **Table 97 (PARAMETER CRC LIST)** will be calculated and stored
[The CRC is stored at always on RH area 0x05A1-0x05A2.
“Slave Checks Always On CRC” parameter is located at always on RH area, 0x0005A0 (bit0).
The CRC is updated only when “Slave Checks always on CRC” parameter is set to TRUE.
“Slave Checks Always On CRC” parameter is cleared when slave is booted via reset by Hardware; “Slave Checks Always On CRC”is set to the same value as “slave shall calculate and check Always On parameters” (see **Table 124 (SLAVE PARAMETERS FROM HOST)** when slave processes an Initialize Service Request.]

TLMM_S_399 - PARAMETER CRC CHECK

The CRC of all the Parameters in **Table 97 (PARAMETER CRC LIST)** will be calculated and checked against to the stored CRC under the following cases:
1. On every boot (except any Resets), or
2. Before turn 2V off.
[Note the CRC will also be calculated and checked against the stored CRC before any of the parameters are changed.
The CRC is checked only when “Slave Checks always on CRC” parameter is set to TRUE.
Note the following:

1. The wrap up interrupt will not be processed until the disable timer expires if the interrupt occurs while the CPU is enabled. This negates any potential latency associated with this interrupt.
2. The wrap up interval is typically equal to the sniff interval times the wrap up count. However, the first wrap up interrupt will be one sniff interval shorter than all the remaining interrupts. For example if the sniff interval is two seconds and the wrap up count is five, the first wrap up interrupt will occur 8 seconds later. The remainder will occur every 10 seconds.
3. After initialization, the sniff interval is re-loaded but the wrap up count is not. The wrap up count is reloaded once the old one is complete.]

TLMM_S_400- PARAMETER CORRUPTION INDICATION

If the two CRCs compared in the PARAMETER CRC CHECK are not identical, a Parameter corruption indication will be sent to the Host.

TLMM_MS_415 - MEMORY TEST - RAM STRUCTURES

A memory test will be conducted once every second on a RAM structure. If any of the CRC check fails then memory test indication will be generated to the host. If all the structures pass the CRC check firmware will restart the process.

[Note: CRC values are stored at a fixed location of 0x38C0. Every second firmware will check the CRC for one of the structures in the RAM in the following order: Ramware code area 1, area2, … area43, ramware enable and jump table, hardware1, hardware2, wakeup and firmware operation parameter structure. If any of the CRC check failed, the memory test failed indication will be sent to the host. If the check passed, the crc of the next structure will be checked on the next second. After the last parameter structure CRC is checked and passed, firmware will start from the first structure (i.e., ramware code area 1) CRC check again on the following second. If there is no ramware it will be skipped.]

TLMM_S_412 - RAMWARE RESTRICTIONS

The slave shall reject a RAMWARE request (data service request destined for slave Mozart) under the following conditions:

a. The session was started using Telemetry C, or
b. Telemetry M session is not secure, or
c. Telemetry M session was started with an EBAN key

[Ramware request from master host to slave mozart must be done using a secure telemetry M session started with a permanent BAN key. Telemetry C must use a master host to slave host data service request followed by a slave host to slave Mozart data request.

Additionally note that the master host should send the slave host a message to stop using service request buffer 1 before downlinking any ramware. The slave RFM uses the service request buffer when responding to a ramware downlink. If the slave host continues to use the buffer as well, unpredictable results will occur. The Slave Received Mozart Data Indication attempts to do the same thing, but a host to host downlink provides better protection. It prevents the race condition of the host sending a request on buffer 1 at the “same time” it is receiving a Slave Received Mozart Data Indication.]

TLMM_S_430 - MOZART DATA REQUEST INDICATION

The slave shall send Slave Received Mozart Data Indication (**TABLE 115: SLAVE RECEIVED MOZART DATA INDICATION**) to host when slave receives RAMWARE request (data service request destined for slave Mozart) from master during communication state.

TLMM_MS_388 - MOZART MODULE MEMORY READ

The host will use the Mozart module memory read command to retrieve the content of telemetry RAM using the specified memory read record.

- The command code identifies the record as a Mozart module memory read record.
- The second, third, and fourth bytes of the record define the address of memory from which to start the read response.
- The fifth and sixth bytes define the number of memory bytes which are to be read. The number of memory bytes shall never be zero.

[The following are valid memory read addresses:
* 0x0000 - 0x3eff (fixed RAM - bank 0)
* 0x4000 - 0x7fff (non-fixed RAM - bank 1)
* 0x8000 - 0xbfff (non-fixed RAM - bank 2)
Other addresses are invalid.]

TLMM_MS_389 - MOZART MODULE MEMORY WRITE

The host will use the Mozart module memory write command to write contiguous bytes of the Mozart module.

- The command code identify the record as a Mozart memory write record
- The second, third, and fourth bytes of the record define the first memory location which is to be updated.
- The fifth and sixth bytes define the number of bytes to update. The number of bytes shall never be zero

[The following are valid memory write addresses:

* 0x0000 - 0x3eff (fixed RAM - bank 0)
* 0x4000 - 0x7fff (non-fixed RAM - bank 1)
* 0x8000 - 0xbfff (non-fixed RAM - bank 2)
Other addresses are invalid.]

TLMM_MS_436 - MOZART MODULE WRITE TO EEPROM

When a valid write to EEPROM special function command is received from upper layer, Mozart shall write the appropriate structure to EEPROM. See **TABLE 116: Special function commands** (values 0x01 - 0x05).

TLMM_S_437 - MOZART MODULE WRITE TO EEPROM AFTER CLOSE WITHOUT POLLING INDICATION

When a valid write to EEPROM special function command is received from master Mozart Master, the slave Mozart shall write the appropriate structure to EEPROM after close with out polling indication is generated. (i.e., after close request is received and the polling duration is zero). See **TABLE 116: Special function commands** (values 0x01 - 0x05).

TLMM_MS_438 - MOZART MODULE ENABLE RAMWARE

When a valid enable ramware special function command is received from upper layer, Mozart shall enable ramware.

TLMM_MS_439 - MOZART MODULE CLEAR RAMWARE

When a valid clear ramware special function command is received from upper layer, Mozart shall clear the ramware areas.