PIN Pad PP190
Programmer's Manual

Personal ID Number Pad

Revision 0

2014-02-06

FEDERAL COMMUNICATIONS COMMISSION STATEMENT

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

NOTE

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

You are cautioned that any change or modifications to the equipment not expressly approve by the party responsible for compliance could void your authority to operate such equipment.

NOTICE

The issuer of this manual has made every effort to provide accurate information. The issuer will not be held liable for any technical and editorial omission or errors made herein; nor for incidental consequential damages resulting from the furnishing, performance or use of this material. This document contains proprietary information that is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced, or translated without the prior written consent of the issuer. The information provided in this manual is subject to change without notice.

第十二條 經型式認證合格之低功率射頻電機，非經許可，公司、商號或使用者均不得擅自變更頻率、 大功率或變更原設計之特性及功能。

第十四條 低功率射頻電機之使用不得影響飛航安全及干擾合法通信；經發現有干擾現象時，應立即停用，並改善至無干擾時方得繼續使用。

前項合法通信，指依電信法規定作業之無線電通信。低功率射頻電機須忍受合法通信或工業、科學及醫療用電波輻射性電機設備之干擾。

AGENCY APPROVED

- FCC class B
- CE class B

WARRANTY

This product is served under one-year warranty to the original purchaser. Within the warranty period, merchandise found to be defective would be repaired or replaced. This warranty applies to the products only under the normal use of the original purchaser, and in no circumstances covers incidental or consequential damages through consumers' misuse or modification of the products.

Document History

Document Version	Apply to FW version	Change
0A	190BL00A 190SC00A 190PM00A	First SQA
0B	190BL00A 190SC00B 190PM00A	1. First SQA debug.
0C	190BL00A 190SC00C 190PM00A	1. Second SQA debug. 2. Add new command ”01”、”16”. 3. Add new command “94”、”96”. 4. Add new function “self diagnostic”.
0D	190BL00A 190SC00D 190PM00A	1. Third SQA debug 2. Add new function “Remote key injection” (R00 ~ R02)
0E	190BL00A 190SC00E 190PM00A	1. Fourth SQA debug 2. Add new function RTC(Real time clock) and new command “P18”
0F	190BL00A 190SC00F 190PM00A	1. Fifth SQA debug a. Modify command 02. b. Modify command 08. c. Modify command 72. d. Modify command Z2 to response <EOT> at end of transmit. e. Modify command Z60. f. Modify command 70 (PIN Entry Request with DUKPT) to check exist of DUKPT key early.

Table of Contents

Section 2 PP190 Setup & Diagnostic Menu

➢ Start Up Self Test

PP190 will perform a series of self-tests during start up, which include:

- Internal firmware checksum: PP190 will verify the internal firmware checksum to ensure the integrity of the firmware program. If firmware checksum error, PP190 will show following prompt and reject further commands:

```
** A L E R T **
ROM CHKSUM FAILD
```

- Security Memory Integrity: PP190 will verify secret personalization information written in the Battery Powered Key (BPK) register of the CPU. If BPK verification failed (possibly by security breach or internal battery exhausted,) PP190 will show following prompt and reject further commands:

```
** A L E R T **
PED WAS TAMPERED
```

➢ Call up Diagnostic Menu

Press function key **[CLR] + [3]** (quickly press '3' after [CLR] released) of PP190 will call up diagnostic menu when PP190 in idle state. The default 2 passwords for diagnostic menu are "87806799" (both passwords)

DISPLAY	ACTION
(Idle prompt)	Power on. Press [CLR]+[3]
Password 1?	Input first setup password and press [ENTER]
Password 2?	Input second setup password and press [ENTER]
HW Tests **Display Info** **Set LCD Backlight** **Set Keypad Beep** **Update Password**	Use left button [-] and right button [-] to scroll up and down. [ENTER] to execute.

➢ Diagnostic Menu 1: HW Tests

DISPLAY	ACTION
Display Test	Display two pages of test pattern: First page is turn on all pixels to check if there are any dot

	damage. Press any key or wait 10 sec to continue. Second page shows PP190 character sets. Press any key or wait 5 sec to leave.
Keypad Test	PP190 will echo user's input key at line 2. Press [CAN] to leave this test.

➢ Diagnostic Menu 2: Display Info

DISPLAY	ACTION
Show SerialNum	Display current serial number. Refer to message 06.
Show Version	Display current firmware version.

➢ Diagnostic Menu 3: Set LCD Backlight

DISPLAY	ACTION
Light Always ON **Light Auto OFF**	First item will set LCD backlight always on. This setting is the same with message 1J with parameter 1. Second item will set PP190 enable its backlight in following situation: a. Any key is pressed. b. PIN entry command is working c. Selecting Menu. And backlight will automatically turn off after 3 seconds of above operation ends.

➢ Diagnostic Menu 4: Set Keypad Beep

DISPLAY	ACTION
Beep ON	Key press with beep.
Beep OFF	Key press without beep.

➢ Diagnostic Menu 5: Update Password

DISPLAY	ACTION

Update Password1	PP190 will show following message: **NEW PASSWD** ******** **CONFIRM PASSWD** ******** User should press 1st password, press [ENTER] to enter 2nd password, then press [ENTER] to finish input. If two passwords mismatch the password will not be changed. Password must have 4 characters at least, with maximum 8 characters.
Update Password2	PP190 will show following message: **NEW PASSWD** ******** **CONFIRM PASSWD** ******** (Usage is the same with password 1.)

➢ About USB virtual COM port (only applied on USB version)

PP190 USB version will identify itself as a virtual COM port for Windows 2000/XP device enumeration. When Windows requests PP190's device driver, please provide a directory name which contains PP190 USB driver, and answer "proceed anyway" when prompted with driver certification questions.

The baud rate of PP190 virtual COM port is determined by the application program. When AP calls Windows API to open COM port, PP190 and Windows virtual COM port driver will adjust its baud rate according to the parameters sent to API function.

Section 3 Message format

This chapter details the format of messages exchanged between the host and PIN Pad.

➢ Notation Conventions

The following conventions are used to make the description of messages more readable:

Control Codes

Control codes (non-displayable codes) are represented by two to three capital letters enclosed in angled brackets "<>". This PIN Pad uses 12 control codes in total. Their actual code, when referenced, is represented by two hex digits enclosed in angled brackets, <0F> for example. The following table lists their usage and value in hex codes.

CODE	HEX VALUE	USAGE
STX	02	Denotes the beginning of a message frame
ETX	03	Denotes the ending of a message frame
EOT	04	Indicates communication session terminated
ACK	06	Acknowledge of message received
SI	0F	Denotes the beginning of a message frame
SO	0E	Denotes the ending of a message frame
NAK	15	Indicates invalid message received
SUB	1A	Message parameter follows
FS	1C	Field separator
GS	1D	Message ID follows

[LRC]

Each message frame transmitted is followed by an LRC byte to detect communication error. This byte should be regarded as part of the message frame but comes after the ending delimiter character. [LRC] is used to represents this LRC byte in describing message frames.

LRC is calculated as an XORed value of every byte after start code in the message frame except itself, that means from the next byte of <STX> or <SI> through the <ETX> or <SO> byte.

[item]

A descriptive item name enclosed in bracket denotes a string or data byte that has no fixed value.

➢ Message frame summary

Data exchanged between PIN Pad and host computer are grouped into "message frames". Each message frame has one of the two frame formats listed below:

- `<STX>[message ID][data]<ETX>[LRC]`
- `<SI>[message ID][data]<SO>[LRC]`

Each type of message has a unique value in its message ID field. In the following texts, we reference a message type by its message ID value, e.g. "message 70".

Message type

Messages exchanged between the Signature PIN Pad and the HOST can be divided into two categories. One is for administration and maintenance, in general administrative messages have <SI> packet header and will return message to HOST by the same message ID.

The other is for various transactions, in general transaction messages have <STX> packet header, and comes in pair. Even number message packets sends command and data to Signature PIN pad, then odd number message packets returns the result.

Time-out

Whenever the PIN Pad sends a message, a response (<ACK> character for acknowledgement or <NAK> character if LRC error occurred) from host is expected. If the PIN Pad does not receive a response within 5 seconds, it will retransmit the last packet. If PIN pad does not receive <ACK> or <NAK> after two retransmit attempts, it will send <EOT> character and this communication session will be terminated.

Transmission Error

The PIN Pad expects the host computer to send a NAK when the host decides that an invalid frame is received. When the PIN Pad receives a NAK, it will retransmit its last message. If the message retransmitted is invalid again, then a NAK should be sent by host to request for another try. The PIN Pad will keep on retransmitting until an <ACK> or <EOT> is received.

Packet Error

When PIN pad received a good transmission but invalid packet (wrong message id) it will ignore the packet. If the packet has acceptable message id but have wrong format. PIN pad will send <EOT> as error message. When in PIN entry functions it will send more detail error code.

Section 4 Administration and maintenance messages

➢ Message 01 Self Test

Format: **<SI>01[test item]<SO>[LRC]**

Message length: Fixed 7 bytes.

Usage: Field maintenance users can issue message 01 to do interactive PIN Pad self-tests. Test results will be displayed on PIN Pad.

'**04**' PIN Pad will detect specific pattern of key presses as a "correct" pattern, which is "[F1] [MENU] [F2] 1 2 3 4 5 6 7 8 9 [CAN] 0 [ENTER] [CLEAR]". Key press pattern other than above will be treated as fail. Issue message 72 will interrupt this test, too.

'**05**': PIN pad will display 2 test pages: First one is a full screen of black dots to check for damaged dot. Press [ENTER] or wait 10 seconds to display page 2, which is some characters. Press [ENTER] again to end this test.

'**06**': PIN pad will display its serial number on the LCD display.

'**07**': PIN pad will execute a communication test, see next page for message flow.

Message element:

Field	Length	Value and description
<SI>	1	<0F>
01	2	Message ID
[Test item]	2	01 02 03 04 Keypad test 05 Display test 06 Check serial number 07 Communication test
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow: (for test item 04 through 06)

HOST	Direction	PIN Pad
Message 01	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs) (Execute self test)

	←	<EOT> (when test done)

Message flow: (for test 07)

HOST	Direction	PIN Pad
Message 01	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	09 Request Packet
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
09 Response Packet	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	09 Response Packet
<ACK> (Good LRC) <NAK> (Bad LRC)	→	
	←	<EOT> (when test done)

➢ Message 02 Load Master Key

Format: **<SI>02[Key ID][Key value] <FS>[Usage][Mode]<SO>[LRC]**
(with clear text key)
<SI>02[Key ID][Key value (ANSI TR31 format)]<SO>[LRC]
(with encrypted key)

Message length: Variable (38 to 94 bytes).

Usage: Load Master Keys into PP190.
PP190 can store 16 master keys; each has a one digit ID. Master keys are divided into three groups of different functions. Refer to **Appendix A: Key management** for key usage and ID definition.

PP190 implements multiple security measures to conform Payment Card Industry (PCI) security requirement. In order to load clear text master keys, two authorized people with their password are required. Otherwise the user must issue message 02 with encrypted key value (ANSI TR31 format). See next entity "**Symmetric Keys Loading Authentication**" for detailed information.

Note: 1. The **[key value]** field's format is ASCII string with range '0'-'9', 'A'-'V', which represents a hexadecimal byte in two characters, i.e. "1F" represents 0x1F.
2. PP190 requires key loading key (master key #F) to be TDES.
3. Pass key loading authentication and then load new clear text master key will erase all other master keys, to prevent malicious key substitution. For more information refer to "**Symmetric Keys Loading Authentication**" at page 24.

Message element:

Request fame (HOST to PP190)

Field	Length	Value and description
<SI>	1	<0F>
02	2	Message ID
[Key ID]	1	'0' to '9', 'A' to 'F' (A is not used)
[Key value]	Var.	Hexadecimal string for key value. Clear text format: 32 bytes for double length, 48 bytes for triple length. TR31 format: 56 bytes for single length, 72 bytes for double length, 88 bytes for triple length.
<FS>	1	Field separator. (Optional, only available in clear text format frame if following [Usage] and [Mode] exists)
[Usage]	2	Optional: ANSI TR-31 key usage for clear text frame. Available value are: "K0" for key encryption. (id 0 ~ 9, B ~ F) "P0" for PIN encryption. (id 0 ~ 9) "M3" for MAC calculation. (id B ~ E) If omitted, default value is "K0"
[Mode]	1	Optional: ANSI TR-31 key mode for clear text frame. Available value are: 'D' for decryption only. (K0 keys) 'E' for encryption only (P0 / D0 keys) 'G' for MAC generation only (M3 keys) 'V' for MAC verification only (M3 keys) If omitted, default value is 'D'.
<SO>	1	<0E>
[LRC]	1	Checksum

Request fame – Error message (HOST to PP190)

Field	Length	Value and description
<SI>	1	<0F>
02	2	Message ID
?	1	
[Err msg]	1	'1': KLK does not exist. '2': Key value duplicated with other existing key. '3': Internal fail: fail to allocate memory '4': Internal fail: fail to read key structure '7': Fail to decrypt key value. 'A': TR31 format error. 'B': Insecure key inject. (New key is longer than the key used to protect it.) 'C': Fail to verify MAC value. 'E': Key usage incompatible with key ID.
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 02 (request frame)	→	
	←	<ACK> /<NAK>/<EOT>
	←	Processing request. If format error, send <EOT> and end. Message 02 (echo of request frame).
Verify echo frame. If verify ok, send <ACK>. If packet LRC error, send <NAK>. If host want to cancel key loading procedure, send <EOT>.	→	
	←	Save key value and send <EOT>

Example:

Clear Text

Master key to be loaded: 1919191919191919 5B5B5B5B5B5B5B5B

The Key ID you want to load: 0

The resulting 02 message : <SI>02019191919191919195B5B5B5B5B5B5B5B<SO>[LRC]

Encrypted (ANSI TR-31 2005 Key Variant Binding Method)

Key encrypting key (Mkey #F): 1919191919191919 5B5B5B5B5B5B5B5B

Master key to be loaded (K0): AA55AA55AA55AA55 3434343434343434

Key Block Header (KBH): (ASCII) A0072K0TD00N0000

TDES CBC encrypted key value: 7D2D21FC9ECD3EEC BB0A2615BD8F0560 5722120BDFF2CCAC

Left 4 bytes of MAC value: 319C3198

The Key ID you want to load: 0

The resulting 02 message:

<SI>020A0072K0TD00N00007D2D21FC9ECD3EECBB0A2615BD8F05605722120BDFF2CCAC319C3918 <SO>[LRC]

Encrypted (ANSI TR-31 2010 Key Derivation Binding Method)

Key condition: Load a double length PIN encryption key to key position #1

Key block protection key (KBPK): 1919191919191919 5B5B5B5B5B5B5B5B

PIN encryption key to be loaded: AA55AA55AA55AA55 3434343434343434

Padded key data: 0080 AA55AA55AA55AA55 3434343434343434 1C2965473CE2

Key Block Header (KBH): (ASCII) B0080P0TE00N0000

Derived Key block encryption key (KBEK): DB7F2A99D5647A7D D3EDFE3DA7CF5B21

Derived Key block MAC key (KBMK): 87EE6C0795954446 A34A0BB5F305BCE1

(See **Appendix A** for detail derive process)

CMAC of (KBH + Padded key data), using KBMK: EA391E5834C1AA0C

(See **Appendix A** for detail CMAC algorithm)

Use CMAC as IV to do TDES CBC encryption on padded key data, using KBEK:

Encrypted key data: 3C4F5024C59C182F 7165BC870FCB7F63 456AAE07DB736C32

The resulting 02 message:

<0F>021B0080P0TE00N0000 3C4F5024C59C182F 7165BC870FCB7F63 456AAE07DB736C32 EA391E5834C1AA0C<0E>

➢ Symmetric Keys Loading Authentication

In order to make PP190 accept clear text key loading frame, the key loading authentication must be processed.

[Enter key loading authentication menu]

Press [CLR]+[2] on the keypad of PP190, then PP190 will show key injection authentication login screen as following:

```
ENTER PASSWORD 1:
```

(*Default password will be sent to authentic owner separately*)

The first authorized person come to enter 1st password on keypad and press [ENTER].

Then PP190 will prompt to enter 2nd password if 1st password is correct. If 2nd password is correct, too, PP190 will enter key loading mode and show following menu:

```
KEY INJECT MODE
Update Password1
Update Password2
Inject MKEY/IPEK
```

Use [F1] and [F4] key to navigate light bar to "Inject MKEY/IPEK", then press [ENTER]. Then user is free to load clear text master key by message 02, or load DUKPT initial key by message 90 and 94.

[Timing constraint and message constraint of Key Inject Mode]

According to PCI security requirement, PIN pad cannot stay in Key Inject Mode forever. Thus when PP190 entered Key Inject Mode, its internal timer will start to countdown, and its operating system will monitor specific message packets. If any one of following criteria is matched, PP190 will exit Key Inject Mode and reject message 02(clear text form) and 90, 94 command:

1. When PIN pad idled for 60seconds, it will exit Key Inject Mode. (Each time 02 / 90 / 94 / 08 / 96 is succeeded, the 60 seconds counter will reset to 60 again.)
2. When PIN pad has been in Key Inject Mode for 15 minutes. It will unconditionally exit Key Inject Mode.
3. When PIN pad receives messages other than 02 / 90 / 94 / 08 / 86, it will exit Key Inject Mode.
4. When user pressed CAN key on keypad, it will exit key inject mode.

[Master key substitution protection]

When user entered Key Inject Mode, PIN pad operating system will set up a new "Key Injecting Session". **The first injected clear text master key in a new session will erase all other master keys.** The other master keys loaded in the same session will not erase any other master key.

DUKPT key set 0 and set 1 will not erase each other.

Example flow to load master keys with security:

In the following example we assume a bank receives a new PP190 and wants to initialize it before deploy. And want to update some master keys after its deployed. We also assume the master key to be loaded is position 0 and position F; their values are already stored in a Tamper Resistant Security Module (TRSM) in a secure way.

1. The bank must generate two passwords, and make two authorized people to keep them separately.
2. Authorized people must enter KEY INJECT AUTH menu and change password 1 and password 2.
3. After password changed, connect PIN pad to TRSM, enter KEY INJECT AUTH menu again and choose Inject MKEY/IPEK function.
4. Operate TRSM to load master key #F and master key #0.

   After step 4 finishes, user can issue other commands to PIN pad (such as message 08 to select key #0 as active master key) or turn it off and deploy it.

5. To load or update master keys at field site, user should issue encrypted command 02.

➢ Message 04 Check Master Key

Format: **<SI>04[key ID][Key Info Query]<SO>[LRC]**

Message length: Variable (6 or 7) bytes.

Usage: Host sends this message to PIN Pad for checking if the master key with an ID of [key ID] has been loaded or not. Message 04 should be used before loading any master key. Message 04 can be also used to query key information (key usage/mode/algorithm) if the designated key is not empty.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
04	2	Message ID
[key ID]	1	Master key ID (0~9, A~G)
[Key Info Query]	1	<Option>, 1: query key information
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
04	2	Message ID
[response code]	1	0 Master key not loaded F Master key loaded
[Key usage]	2	<Option, if key info query filed is set> "K0": Key encrypting key. (Master key for PIN / MAC / Data key) "P0": PIN key "D0": Data key "M1": MAC key for MAC algorithm 1 "M3": MAC key for MAC algorithm 3
<FS>	1	<Option, if key info query filed is set> <1C>, filed separator
[Mode]	2	<Option, if key info query filed is set> "E": Encryption use "D": Decryption use
<FS>	1	<Option, if key info query filed is set> <1C>, filed separator
[Algorithm]	2	<Option, if key info query filed is set>

		"T": Triple DES "D": Single DES
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 04 (request)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Check requested memory location Message 04 (response)
<ACK> (Good echo) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
	←	<EOT>

➢ Message 05 Load Serial Number

Format: **<SI>05[string]<SO>[LRC]**

Message length: Variable, maximum length is 21 bytes

Usage: Load the PIN Pad with the serial number given in the message frame. PIN Pad will send the whole message frame back to host as a confirmation of good reception. Host should then send an <ACK> to confirm or <EOT> to cancel this serial number loading process if the LRC is good but serial number echoed is incorrect. Follow the standard <NAK> process if an invalid LRC is detected.

Message element:

Field	Length	Value and description
<SI>	1	<0F>
05	2	Message ID
[string]	0..16	Alphanumeric string (0~9, A~Z, a~z)
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 05	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 05 (echo frame) or <EOT> indicate error.
<ACK> (Good echo) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
	←	(Stores serial number) <EOT>

➢ Message 06 Get Serial Number

Format: **<SI>06<SO>[LRC]**

<SI>06[string]<SO>[LRC]

Message length: Fixed 5 bytes for requesting message, variable for response message (max 21 bytes.)

Usage: This message is used to get serial number of the PIN Pad. PIN Pad will send the serial number previously loaded or string of 16 '0's as the serial number if it has not been loaded. Serial number will be displayed on LCD, too.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
06	2	Message ID
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
06	2	Message ID
[string]	0..16	String for serial number
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 06 (request)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 06 (response frame) or <EOT> if read error
<ACK> (Good echo) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
	←	<EOT>

➢ Message 07 Test DES Implementation

Format: **<SI>07[master key][clear text][cipher text]<SO>[LRC]**

Message length: Fixed 53 bytes.

Usage: This message is used to validate DES implementation of PIN Pad. Testing result will be shown on the PIN Pad display and return response code for remote diagnostic.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
07	2	Message ID
[Master key]	16	Master Key used of encoding (hexadecimal string)
[Clear text]	16	Clear text for encoding (hexadecimal string)
[Cipher text]	16	Known ciphered text (hexadecimal string)
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
07	2	Message ID
[response code]	1	0: Test Success F: Test Failed.
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 07 (request)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 07 (response)
<ACK>/<NAK>/ <EOT>		
	←	<EOT>

➢ Message 08 Select Master Key

Format: **<SI>08[KeyID]<SO>[LRC]**

Message length: Fixed 6 bytes.

Usage: This message is used to select one of the 10 possible PIN encrypting master keys previously loaded using message 02. The selected master key will be used in the following transactions.

Note: **Check master key existence before change**:
This message does not respond for checking master key existence. You may choose an empty master key without notice.
TDES capability: If selected master key is a double length key (32 characters when loaded with message 02), PP190 will treat all session keys (in MK/SK message 70, Z60, Z62) as EDE encrypted by this master key. (See Appendix A)
Confirm key existence before issue 08: message 08 is not responsible for check if [KeyID] has a valid master key, use message 04 before 08.

Message element:

Field	Length	Value and description
<SI>	1	<0F>
08	2	Message ID
[KeyID]	1	0~9, one of Master key id.
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 08	→	
	←	<ACK>/<NAK>/<EOT>
	←	[Success] <SI>080<SO> [Fail] <SI>08[errCode]<SO> <EOT>

Error Message:

Error Code	Meaning
'1'	Key index > 9

➢ Message 09 Communication Test

Format: **<SI>09<SO>[LRC]**

<SI>09<SUB>PROCESSING<SO>[LRC]

Message length: Fixed 5 bytes for requesting message, fixed 16 bytes for response message.

Usage: This message is used to test communication link between HOST and the PIN Pad. Both HOST and PIN Pad can initiate communication test. The initiating party should send the requesting message; the other party should response with the response message that should be ACKed if received correctly. After verifying that the response message is correctly, the initiating party should send back the same response message and the receiving party should acknowledge this message. Testing results are shown on the PIN Pad display.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
09	2	Message ID
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
09	2	Message ID
<SUB>	1	<1A>
[Test string]	10	ASCII string "PROCESSING"
<SO>	1	<0E>
[LRC]	1	Checksum

Result frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
09	2	Message ID
[response code]	1	0: Test Success F: Test Failed.
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 09 (request)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 09 (response frame)
<ACK> (Good echo) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
Message 09 (response)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC)
	←	Message 09 (result frame)
<ACK> (Good echo) <NAK> (Bad LRC) (<EOT> after 3 NAKs)		
	←	<EOT>

➢ Message 11 PIN Pad Device Connection Test

Format: **<SI>11<SO>[LRC]**

Message length: Fixed 5 bytes.

Usage: This message is used to ensure that the PIN Pad is attached to the HOST working normally. PIN Pad will response an ACK (or NAK if LRC incorrect) within one second.

Message element:

Field	Length	Value and description
<SI>	1	<0F>
11	2	Message ID
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 11	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)

➢ Message 13 Adjust COM1 Baud Rate (RS-232 version only)

Format: **<SI>13[baud code][mode]<SO>[LRC]**

Message length: Variable, 6 bytes.

Usage: This message will change the working baud rate and transmit mode of PP190 for later operations. The setting is kept in the battery-powered memory, which will not be erased until security is breached or the battery exhausted. Baud rate will be changed after message flow ends.

Note: If [mode] parameter is not specified, the default transmit mode is N, 8, 1.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
13	2	Message ID
[baud code]	1	ASCII character '1' = 1200bps '2' = 2400bps '3' = 4800bps '4' = 9600bps '5' = 19200bps '6' = 38400bps '7' = 57600bps '8' = 115200bps
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
13	2	Message ID
[status]	1	ASCII character '0' for success '1' for parameter error
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 13 (request)	→	

	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 13 (response)
<ACK>/<NAK>/<EOT>	→	
	←	<EOT>
		(Change working baud rate and save setting)

Message flow:

HOST	Direction	PIN Pad
Message 14 (Request frame)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 14 (Response frame)
<ACK>/<NAK>/<EOT>	→	
	←	<EOT>

➢ Message 16 Remote self-test request

Format: **<SI>16<SO>[LRC]**

Message length: Fixed 5 bytes.

Usage: This message is used to ensure that the PP190 attached to the HOST is working normally. PP190 will response an ACK (or NAK if LRC incorrect) within one second. If multiple tests failed, response code will concatenate such as "<SI>1625<SO>".

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
16	2	Message ID
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
16	2	Message ID
[Response]	1 .. 3	0 – Healthy 2 – System Core checksum fail 5 – Master keys CRC error
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 16 (Request frame)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 16 (Response frame)
<ACK>/<NAK> /<EOT>	→	
	←	<EOT>

➢ Message 17 Request random number

Format: **<SI>17<SO>[LRC]**

Message length: Fixed 5 bytes.

Usage: This message is used to request PIN Pad to generate an 8bytes random number block. This random number is generated by hardware TRNG that is certified with sufficient security.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
17	2	Message ID
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
17	2	Message ID
[RndBlk]	16	Random number block generated by PP190. Format: hexadecimal string.
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 17 (Request frame)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 17 (Response frame)
<ACK>/<NAK> /<EOT>	→	
	←	<EOT>

➢ Message 18 Get/Set PIN pad system time

Format: **<SI>18<SO>[LRC] (Request frame to get system time)**
<SI>18[YYYY][MM][DD][HH][MM][SS]<SO>[LRC] (Request frame to set system time)

Message length: Fixed 5 bytes or 19 bytes.

Usage: This message is used to set real world time in PP190 and for EMV level 2 transaction log.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
18	2	Message ID
[YYYY]	4	(optional, only set time need) AD year, i.e. "2006"
[MM]	2	(optional, only set time need) Month, "01"~"12"
[DD]	2	(optional, only set time need) Day of month, "01"~"31"
[HH]	2	(optional, only set time need) Hour, "00"~"23"
[MM]	2	(optional, only set time need) Minute, "00"~"59"
[SS]	2	(optional, only set time need) Second, "00"~"59"
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
18	2	Message ID
[status]	1	0: Success F: Failed.
[YYYY]	4	(optional, only get time will response) AD year, i.e. "2006"
[MM]	2	(optional, only get time will response) Month, "01"~"12"
[DD]	2	(optional, only get time will response) Day of month, "01"~"31"
[HH]	2	(optional, only get time will response) Hour, "00"~"23"
[MM]	2	(optional, only get time will response) Minute, "00"~"59"
[SS]	2	(optional, only get time will response) Second, "00"~"59"
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 18 request frame	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 18 Response Frame
<ACK> /<NAK> /<EOT>	→	
	←	Processing and send <EOT>

➢ Message 19 Query Firmware Version

Format: **<SI>19[part]<SO>[LRC] (request frame)**

<SI>19.[Version].[SubVer].[Chksum] <SO>[LRC] (response frame)

Message length: Fixed 6 bytes (request frame) / 82 bytes (response frame).

Usage: This message is used to query PP190 firmware version number and firmware check sum value.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
19	2	Message ID
[part]	1	Firmware Part number 1: System Core 2: Prompt Message
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
19	2	Message ID
.	1	<2E>, field separator
[Version]	8.	Firmware version (ASCII string)
.	1	<2E>, field separator
[SubVer]	2	Firmware sub version ('0'~'9')
.	1	<2E>, field separator
[chksum]	64	Firmware checksum ('0'~'9, 'A'~'F'')
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 19 (Request frame)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 19 (Response frame)
<ACK>/<NAK>/<EOT>	→	
	←	<EOT>

➢ Message 1J Turn ON/OFF LCD Backlight

Format: **<SI>1J[option]<SO>[LRC]**

Message length: Fixed 6 bytes.

Usage: This message can control the global backlight ON or OFF for the LCD of PP190 with backlight option. By default, PP190 will turn on its LCD backlight when it receives PIN entry or clear text entry message such as 70 or Z52, and turn it off when those functions exits. With message "1J1", the PP190 will keep LCD backlight turned ON until "1J0" is issued.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
1J	2	Message ID
[option]	1	ASCII character '0': Turn off LCD backlight '1': Turn on LCD backlight
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
1J	2	Message ID
[status]	1	ASCII character '0': Turn off LCD backlight '1': Turn on LCD backlight
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 1J	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 1J (Response frame)
<ACK>/ <NAK>/ <EOT>	→	
	←	<EOT>
		LCD backlight turned ON/OFF

➢ Message 1M Setup Keypad Beeper

Format: **<SI>1M[option]<SO>[LRC]**

Message length: Fixed 6 bytes.

Usage: This message is used to turn on or turn off beeper when the keypad is pressing.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<SI>	1	<0F>
1M	2	Message ID
[option]	1	ASCII character '0': Disable keypad beeper. '1': Enable keypad beeper.
<SO>	1	<0E>
[LRC]	1	Checksum

Response frame (PIN Pad to HOST)

Field	Length	Value and description
<SI>	1	<0F>
1M	2	Message ID
[status]	1	ASCII character '0': Keypad beeper disabled. '1': Keypad beeper enabled.
<SO>	1	<0E>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 1M	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 1M (Response frame)
<ACK>/ <NAK>/ <EOT>	→	
	←	<EOT>

Section 6 Online transaction messages with Master/Session Keys (MK/SK)

➢ Message 70 PIN entry request (MK/SK)

Format: **<STX>70.[Account]<FS>[session key][Amount]**
<FS>[timeout]<ETX>[LRC]

Message length: Variable 36 to 51 bytes (max. 67 bytes for TDES session key).

Usage: Display prompt and accept customer PIN input. The following prompt will be displayed:

```
"Total Amount"
 "$xxx.xx"
 "Enter PIN"
 "Push "ENTER""
```

The PIN Pad will then wait till the PIN entered and [ENTER] key is pressed. After ENTER key is pressed, the string "PIN PAD" and "PROCESSING" will be displayed until the CLEAR key is pressed. During this period, the PIN Pad will not process any message other than the CANCEL message (message 72).

NOTE: **Aborting transaction:** Press CLEAR button to reset the PIN input and CAN (cancel) button to abort the transaction.

PIN length: According to ANSI X9.8 standard, the length of PIN should between 4 to 12 digits. If user inputs less than 4 digits and press ENTER, PP190 will beep for error and continue to wait for user's input. When user inputs 13th character, PIN pad will beep for error, conserves PIN character 1st to 12th, and wait for ENTER.

This message has DES Time Throttle: See Appendix A for details.

Master key must be selected before transaction: PP190 will warn and refuse message 70 if message 08 was not issued before.

Triple DES capability: Following table shows the logic of PP190 when processing single-length and double-length MK/SK. (TDES in EDE order, see Appendix A).

Session Key: If the selected key is with usage "P0", the session key should be all zeros.

Session key / Master key	Double length	Single length
Double length	PP190 TDES decrypts L-key and R-key of [session key] value, using active master key. PIN blocks are TDES encrypted by decrypted session key.	PP190 TDES decrypts [session key] value, using active master key. PIN blocks are DES encrypted by decrypted session key.

Single length	PP190 DES decrypts L-key and R-key of [session key] value, using active master key. PIN blocks are TDES encrypted by session key.	PP190 DES decrypts [session key] value, using active master key. PIN blocks are DES encrypted by session key.

Message element:

Field	Length	Value and description
<STX>	1	<02>
70	2	Message ID
.	1	<2E>, delimiter
[Account]	8..19	Card account number
<FS>	1	<1C>, field separator
[session key]	16 or 32	Working key encrypted using selected master key. 32-characters session key produces TDES encrypted PIN block with EDE order. Format: hexadecimal string. This filed should be all zeros if the selected key is with usage "P0"
[Amount]	4..8	Amount of goods to be displayed on PIN Pad.
<FS>	1	(optional) <1C>, field separator
[timeout]	1	(optional) ASCII character from '1' to '9' which is the timeout value in the unit of 30 seconds. Default = 9x30 = 270 seconds.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 70	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs) Prompt user to enter PIN.
	←	Message 71 or <EOT> when input timed out or user pressed [CAN]
<ACK> (Good echo) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
		Display "PIN PAD PROCESSING"

➢ Message 71 Encrypted PIN Block Response

Format: **<STX>71.<fkey flag><PIN length>01[PIN][LRC] (PIN block frame)**
<STX>71[error code]<ETX>[LRC] (Error code frame)

Message length: Fixed 27 bytes for PIN block frame, 6 bytes for error code frame.

Usage: Send the entered PIN to HOST in encrypted format.

Message element:

Field	Length	Value and description
<STX>	1	<02>
71	2	Message ID
.	1	<2E> delimiter
[Fkey flag]	1	Always '0' (This field is kept to retain old model compatibility.)
[PIN length]	2	00, 04..12 length of PIN entered
01	2	01 format of PIN block, always 01
[PIN]	16	Encrypted PIN blocks Format: hexadecimal string.
<ETX>	1	<03>
[LRC]	1	Checksum

Message 71 (Error message)

Field	Length	Value and description
<STX>	1	<02>
71	2	Message ID
[Error code]	1	Code to indicate error (see next page)
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 70/Z60/Z62	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 71 or <EOT> when input tined out or user pressed [CAN]
<ACK> (Good echo) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
		Display processing prompt

Error codes:

Code	Meaning
'0'	Null Account input field.
'1'	Key value error. (Active master key not exist, or session key value conflicted with the usage of active master key, or session key length longer than active master key)
'2'	Account number shorter than 8 digits.
'3'	Account number longer than 19 digits.
'4'	Account number have character other than '0'-'9'.
'5'	Working key format error.
'6'	Timeout value error
'7'	No more DES operation within 60 min. (see Appendix A)
'8'	From 70, Amount string format error. From Z62, PIN count, Accept Null PIN flag, and Prompt string format error.
'A'	Currently selected master key over range (Master key slot A to F will cause this error message because they are supposed to do authentication and MAC, not for PIN entry)
'B'	Flash memory read/write error
'C'	Memory buffer allocation error
'E'	Data length error in a field.
'G'	Specified file not found or authentication error.
'H'	Receive command 72.

'I'	Cancel key is press.
'J'	PIN entry timeout.

➢ Message 72 PIN Entry Cancel

Format: **<STX>72<ETX>[LRC]**

Message length: Fixed 5 bytes.

Usage: Cancel current transaction and return the PIN Pad to IDLE state, used to interrupt command in process. If PP190 receives message 72 while processing user input such as swipe card or enter PIN, It will send <EOT> to acknowledge that operation is canceled.

Message element:

Field	Length	Value and description
<STX>	1	<02>
72	2	Message ID
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 72	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	<EOT> Optional. If PIN pad is waiting for user's PIN input.

➢ Message Z0 Move Display Cursor

Format: **<STX>Z0[XX][YY]<ETX>[LRC]**

Message length: Fixed 9 bytes.

Usage: Move the display cursor. Z0 message is enabled when PIN pad receives first Z2 message. **Under Z2-authenticated mode, Z0 message is also disabled.**

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z0	2	Message ID
[XX]	2	X-coordinate, 01 ~ Max. Characters per line
[YY]	2	Y-coordinate, 01 ~ Max. line
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z2	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Display string.
Message Z0	→	
	←	<ACK> / <NAK> / <EOT>
		PIN pad moves cursor
Message Z2 (without clear screen)	→	
	←	<ACK> / <NAK> / <EOT>
		Display 2nd string from the coordinate specified by Z0.

➢ Message Z1 Reset State

Format: **<STX>Z1<ETX>[LRC]**

Message length: Fixed 5 bytes.

Usage: Force the PIN Pad to enter IDLE state.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z1	2	Message ID
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z1	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)

➢ Message Z2 Display String

Format: **<STX>Z2<SUB>[string]<ETX>[LRC] (Request frame, normal)**
<STX>Z2<GS>[PromptID]<SUB><ETX>[LRC]
(Request frame, authenticated)
<STX>Z2<RS>[PromptID]<SUB><ETX>[LRC]
(Request frame, authenticated for PIN entry)
<STX>Z2[status]<ETX>[LRC]
(Response frame, authenticated)

Message length: Variable, at least 6 bytes.

Usage: PIN Pad to show the indicated prompt string on its display, until [CAN] key is pressed. If the first character of message is <GS> (0x1D) or <RS> (0x1E), PIN pad will treat following message string as ID number, and search its predefined message table for corresponding message string, then display the string on the screen.

Note: 1. Two Z2 message with authenticated prompt ID can be issued in serial to form a longer sentence, or used in combination with normal string which contains only digits.
2. Z2 message with PIN entry prompt will force user issue every message with <SUB>, which implies the PIN entry message can't be concatenated.
3. PIN pad will temporarily turn off timer display for the first Z2 message it received. After Z42, Z50, Z60 are performed, [CAN] key is pressed, or any other message received and processed, PIN pad will turn on the timer display.

Message element:

Z2 request frame (normal mode)

Field	Length	Value and description
<STX>	1	<02>
Z2	2	Message ID
<SUB>	1	<1A> (optional) When <SUB> exists, PIN pad will clear screen contents and hide pop window before display string.
[string]	0 .. 32	ASCII string to be displayed
<ETX>	1	<03>
[LRC]	1	Checksum

Z2 request frame (authenticated mode with fixed prompt)

Field	Length	Value and description
<STX>	1	<02>
Z2	2	Message ID
<GS>	1	<1D>, mark of authenticated frame with fixed prompt.
Prompt ID	3	Prompt ID that corresponds to fixed prompt provided by PIN pad. Decimal string: 001 ~ 999.
<SUB>	1	<1A> (optional) When <SUB> exists, PIN pad will clear screen contents and hide pop window before display string
<ETX>	1	<03>
[LRC]	1	Checksum

Z2 request frame (PIN entry mode with fixed prompt)

Field	Length	Value and description
<STX>	1	<02>
Z2	2	Message ID
<RS>	1	<1E>, mark of PIN entry frame with fixed prompt.
Prompt ID	3	Prompt ID that corresponds to fixed PIN entry prompt provided by PIN pad. Decimal string: 001 ~ 999.
<SUB>	1	<1A> PIN pad will clear clear screen contents and hide pop window before display string
<ETX>	1	<03>
[LRC]	1	Checksum

Z2 response frame (authenticated mode)

Field	Length	Value and description
<STX>	1	<02>
Z2	2	Message ID
[status]	1	'0': OK '1': Prompt ID not supported.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

Normal frame

HOST	Direction	PIN Pad
Message Z2	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Display string

Authenticated frame

HOST	Direction	PIN Pad
Message Z2	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z2 (response frame)
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
	←	Display string <EOT> (if received <ACK>)

➢ Message Z3 Display Line Prompts

Format: **<STX>Z3[count]<SUB>[prompt1]<FS>[prompt2..7]<ETX>[LRC]**

(Request frame, normal)

<STX>Z3<GS>[PromptID1]<FS>[PromptID2..7]<SUB><ETX>[LRC]

(Request frame, authenticated)

<STX>Z3<RS>[PromptID1]<FS>[PromptID2..7]<ETX>[LRC]

(Request frame, authenticated for PIN entry)

<STX>Z3[status] <ETX>[LRC]

(Response frame, authenticated)

Message length: Variable 8 to 124 bytes.

Usage: The PIN Pad will display the received prompt strings (up to 7 lines of prompt). If the length of prompt exceeds the maximum characters per line, this prompt will be truncated.

Message element:

Z3 request frame (normal mode)

Field	Length	Value and description
<STX>	1	<02>
Z3	2	Message ID
[Count]	1	Number of prompts to be displayed
<SUB>	1	<1A> (optional) When <SUB> exists, PIN pad will clear clear screen contents and hide pop window before display string
[Prompt1]	var	First string to be displayed, max length is one line (20 or 40 characters depend on font size).
<FS>	1	<1C>, field separator
[Prompt2..7]	var	Remaining strings to be displayed. Note. <FS> is required between messages
<ETX>	1	<03>
[LRC]	1	Checksum

Z3 request frame (authenticated mode or PIN entry mode)

Field	Length	Value and description
<STX>	1	<02>
Z3	2	Message ID
<GS> or <RS>	1	<1D> for authenticated mode <1E> for PIN entry mode (In these mode, PIN Pad will clear screen contents and hide pop window before showing prompts.)
[Prompt ID1]	3	Prompt ID that corresponds to fixed prompt provided by PIN pad. Decimal string: 001 ~ 999.
<FS>	1	<1C>, field separator
[Prompt ID2..7]	3	Prompt ID that corresponds to fixed prompt provided by PIN pad. Decimal string: 001 ~ 999. Note. <FS> is required between prompt ID.
<ETX>	1	<03>
[LRC]	1	Checksum

Z3 response frame (authenticated mode)

Field	Length	Value and description
<STX>	1	<02>
Z3	2	Message ID
[status]	1	'0': OK '1': Prompt ID not supported.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

Normal frame

HOST	Direction	PIN Pad
Message Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Display prompts as required

Authenticated frame

HOST	Direction	PIN Pad
Message Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z3 (response frame)
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
	←	Display string <EOT> (if received <ACK>)

➢ Z2 / Z3 Authenticated mode with fixed prompt

To enable message Z42 and Z50, user has to issue Z2 / Z3 message with a prompt ID supported by PIN pad (See ***Appendix C***). These prompts are verified during Payment Card Industry (PCI) Security Conformance tests to make sure users will not expose sensitive information (such as PIN) accidentally.

For security reason, to issue authenticated frame of Z2 / Z3 at the first time, the <SUB> flag is mandatory. After Z2 authenticated mode entered, PIN pad will accept two kind of Z2 packet:

1. Z2 packet in normal mode, without <SUB> flag, and contains only digits (0~9)
2. Z2 packet in authenticated mode, without <SUB> flag.

For example, issue Z2<GS>005<SUB> and Z2<GS>016 will show "PLEASE ENTER DRIVER LICENSE" on the screen.

➢ Z2 / Z3 PIN entry mode with fixed prompt

To enable Z60, user has to issue Z2 / Z3 message with a prompt ID supported by PIN pad, dedicated for PIN entry (See ***Appendix D***). These prompts are verified during Payment Card Industry (PCI) Security Conformance tests to make sure users will not misunderstand PIN entry request as other non-sensitive data. Also message Z62's prompt1 and prompt2 will be checked to see if they are listed in this prompt table. If not, PIN pad will reject Z62.

Any other messages other than Z2, Z3, Z42, Z50, and Z60 or any unsuccessful Z2 / Z3 messages (wrong prompt ID, format error, Z2 message includes non decimal characters) will make PIN pad to leave Z2 / Z3 authenticated mode to avoid attack.

➢ Message Z2 Display String with Authentication Code

Format: **<STX>Z2<FS>[KeyID][MAC][Mode][string]<SUB><ETX>[LRC]**
(Request frame)
<STX>Z2[status]<ETX>[LRC] (Response frame)

Message length: Variable.

Usage: This command allows acquirer to show free message on screen as prompt for clear text entry (Z42, Z50) and PIN entry (Z60). PP190 will verify MAC value by the following rule:

***** Collect [Mode] character, [string] (exclude white space, punctuation marks and digits), and <SUB> character (if exist), as byte array, padding with ASCII '0' (0x30) to the multiple of 8.

***** Use the key specified by [KeyID] and ISO-9797-1 Algorithm 3 to generate message authentication code for above data.

***** Compare the leftmost 4 bytes of MAC value and the one written in the Z2 command. If MAC value matches, PP190 will display the [string] written in Z2 command.

Note: 1. If Z2 (string with MAC) is used in combination with Z2 (fixed prompt), their mode character (GS / RS) must be the same; Otherwise PIN pad will reject secondary Z2.

2. PIN pad will temporarily turn off timer display for the first Z2 message it received. After Z42, Z50, Z60 are performed, [CAN] key is pressed, or any other message received and processed, PIN pad will turn on the timer display.

3. If continuous Z2 (string with MAC) without <SUB> cause the string out of LCD display's range, some character will be cut.

Message element :

Z2 with MAC, request frame

Field	Length	Value and description
<STX>	1	<02>
Z2	2	Message ID
<FS>	1	<1C>, field separator.
[KeyID]	1	'B' ~ 'E', key to verify MAC value. The specified key must have usage 'M3' and mode 'V'.
[MAC]	8	Message authentication code of following message (including <SUB> if exist).
[Mode]	1	<GS> (0x1D) for Non-PIN entry. <RS> (0x1E) for PIN entry.
[string]	0 .. 32	ASCII string to be displayed
<SUB>	1	<1A> (optional)

		When <SUB> exists, PIN pad will clear screen contents and hide pop window before display string, and also reset entry mode.
<ETX>	1	<03>
[LRC]	1	Checksum

Z2 with MAC, response frame

Field	Length	Value and description
<STX>	1	<02>
Z2	2	Message ID
[status]	1	'0': OK '1': MAC key ID error (out of 'B'~'E'). '2': MAC key attribute error. '3': MAC value error. '4': Packet format error.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z2	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z2 (response frame)
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
	←	Display string <EOT> (if received <ACK>)

➢ Message Z3 Display Line Prompts with Authentication Code

Format: **<STX>Z3<FS>[KeyID][MAC][count][Mode][prompt1]<FS>**
[prompt2..7]<SUB><ETX>[LRC] (Request frame)
<STX>Z3[status]<ETX>[LRC] (Response frame)

Message length: Variable.

Usage: This command allows acquirer to show free message on screen as prompt for clear text entry (Z42, Z50) and PIN entry (Z60). PP190 will verify MAC value by the following rule:

***** Collect [Mode] character, [prompt_n] (exclude white space, punctuation marks and digits), and <SUB> character (if exist), as byte array, padding with ASCII '0' (0x30) to the multiple of 8.

***** Use the key specified by [KeyID] and ISO-9797-1 Algorithm 3 to generate message authentication code for above data.

***** Compare the leftmost 4 bytes of MAC value and the one written in the Z2 command. If MAC value matches, PP190 will display the [string] written in Z2 command.

Note: 1. If Z3 (string with MAC) is used in combination with Z3 (fixed prompt), their mode character (GS / RS) must be the same; Otherwise PIN pad will reject secondary Z3.

2. PIN pad will temporarily turn off timer display for the first Z2 message it received. After Z42, Z50, Z60 are performed, [CAN] key is pressed, or any other message received and processed, PIN pad will turn on the timer display.

Message element:

Z2 with MAC, request frame

Field	Length	Value and description
<STX>	1	<02>
Z3	2	Message ID
<FS>	1	<1C>, field separator.
[KeyID]	1	'B' ~ 'E', key to verify MAC value. The specified key must have usage 'M3' and mode 'V'.
[MAC]	8	Message authentication code of following message (including <SUB> if exist).
[count]	1	'1' ~ '7', number of following prompts.
[Mode]	1	<GS> (0x1D) for Non-PIN entry. <RS> (0x1E) for PIN entry.
[prompt1]	Var.	First string to be displayed, max length is one line (20 or 40 characters depend on

		font size).
<FS>	1	<1C>, field separator
[prompt N]	Var.	Second to end string to be displayed. Each prompt is separated by <FS>.
<SUB>	1	<1A> (optional) When <SUB> exists, PIN pad will clear clear screen contents and hide pop window before display string
<ETX>	1	<03>
[LRC]	1	Checksum

Z3 with MAC, response frame

Field	Length	Value and description
<STX>	1	<02>
Z2	2	Message ID
[status]	1	'0': OK '1': MAC key ID error (out of 'B'~'E'). '2': MAC key attribute error. '3': MAC value error. '4': Packet format error.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z3 (response frame)
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
	←	Display string <EOT> (if received <ACK>)

➢ Example to use Z2 / Z3 with Authencation Code.

[Example 1]

1. Use message 02 (ANSI TR31 frame) to load following key to position 'B':
   BCDE90123456789ABCDE90123456789A, Usage = M3, Mode = V.
2. Assume we want to clear screen and display following string for PIN entry: "AMOUNT 123456.78 ENTER YOUR PIN".
3. The data for MAC generation ('A' to 'Z', 'a' to 'z' and ISO8859-15 high page character from 0xBC to 0xFF, padded with ASCII 0):
   <RS>AMOUNTENTERYOURPIN<SUB>0000
   → 1E414D4F554E54454E544552594F555250494E1A30303030
   The white spaces and digits are not counted into MAC, this feature enables acquirer to issue PIN entry prompts with different amount, but keep the same MAC value.
4. Use the key specified in the step 1 to calculate ISO9797-1 algorithm 3 MAC.
   The result is: C51401D727D761E2.
   Take leftmost 4 bytes as MAC value: C51401D7.
5. Send <02>Z2<1C>BC51401D7<1E>AMOUNT 123456.78 ENTER YOUR PIN<1A><03> to PIN Pad, Then message Z60 can be issued to request PIN entry.
6. Send <02>Z2<1C>BC51401D7 <1A>AMOUNT 123.45 ENTER YOUR PIN<03> to PIN Pad to see the same MAC applies to different amounts.

[Example 2]

1. Use message 02 (ANSI TR31 frame) to load following key to position 'B':
   6AC292FAA1315B4D8234B3A3D7D5933A, Usage = M3, Mode = V.
2. Assume we want to clear screen and display for non-PIN entry: "MESSAGE ONE 1.0" and "MESSAGE TWO 2.0".
3. The data for MAC generation (padded with ASCII 0):
   <GS>MESSAGEONE<FS>MESSAGETWO<SUB>0
   → 1D4D4553534147454F4E451C4D45535341474554574F1A30
4. Use the key specified in the step 1 to calculate ISO9797-1 algorithm 3 MAC.
   Take leftmost 4 bytes as MAC value: 22C0BAD9.
5. Send <02>Z3<1C>B22C0BAD92<1D>MESSAGE ONE 1.0<1C>MESSAGE TWO 2.0<1A><03> to PIN pad.

➢ Message Z42 Read Key Code

Format: **<STX>Z42[timeout]<ETX>[LRC]**

Message length: Variable 6 to 9 bytes.

Usage: Once PP190 receives this command, it begins polling functional key array until timeout. If PP190 received Z2 / Z3 authenticated frame before Z42, it will return any key pressed by user by ASCII key codes via message Z43. Else it will return only function key codes (F1, F2, F3, F4, CAN, CLR, ENTER), and reject numerical key (0 to 9). Multiple key press or combined key press will be discarded.

Abort input: Issue message 72 to abort the operation.

Note: **Z2/Z3 required:** Because Z42 will not show any message to prompt user operation, Z2 or Z3 should be issued before this command, or PIN pad will send <EOT> and stop.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z42	3	Message ID
[timeout]	1 to 3	ASCII character from 1 to 255, for example "10" means 10 seconds timeout.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PP190
Message Z2 or Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Show prompt message
Message Z42	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z43
<ACK>/<NAK>/<EOT>	→	

➢ Message Z43 Read Key Code Response

Format: **<STX>Z43[Keycode]<ETX>[LRC]**

Message length: Fixed 7 bytes.

Usage: This is the response frame of Z42.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z43	3	Message ID
[keycode]	1	'0' to '9' ASCII 'A' to 'C' denotes 3 function keys. 'A' = [F1] 'B' = [F2] 'C' = [F3] 'D' = [F4] '*' = [CAN] '#' = [ENTER] '/' = [CLR] '?' means time out.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

(Please refer to message Z42)

➢ Message Z50 String Entry Request

Format: **<STX>Z50[echo flag][timeout][max entry]<ETX>[LRC]**

Message length: Variable 10 to 12 bytes.

Usage: Request user to input string on keypad.

Then PP190 will wait for keypad input and store ASCII data into internal buffer. To input English character on the keypad, press [F2] key to rotate the last character. For example, press [1], [F2], [F2] will input a 'Z' character into PP190. The maximum length of internal buffer is 32 characters.

User can use [CLR] to clear input buffer and input again, or [CAN] to cancel input.

Press '0' and press [F2] will transform '0' into period or white space.

Abort input: Issue message 72 to abort the operation.

Note: A Z2 or Z3 message with authenticated frame must be issued before Z50. Otherwise PIN pad will refuse to execute.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z50	3	Message ID
[echo flag]	1	'0': echo input as '*' '1': echo input as is '2': do not echo
[timeout]	3	ASCII character from 1 to 255 to set the timeout for each keypress, for example "010" means 10 seconds timeout after the last keypress.
[max entry]	1 or 2	(optional) Maximum entry count. Range from 00 to 32 (or 0 to 32)
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PP190
Message Z2 or Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Show prompt message
Message Z50	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z51 (or <EOT> when input cancelled)
<ACK>/<NAK>/<EOT>	→	

➢ Message Z51 String Entry Response

Format: **<STX>Z51[string]<ETX>[LRC]**

Message length: Variable, maximum 55 bytes.

Usage: This is the response frame of Z50 and ZG.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z51	3	Message ID
[string]	1..32	User inputted string. '?' means time out. '!' means file not found or authentication error. (For Message ZG error response) '%' means prompt ID not supported. (For Message ZG error response)
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

(Please refer to message Z50)

➢ Message Z60 PIN entry request with external prompt (MK/SK)

Format: **<STX>Z60.[account]<FS>[session key]<FS>[timeout]<ETX>[LRC]**

Message length: Variable 32 to 43 bytes (max. 59 bytes for TDES session key).

Usage: Request the PIN Pad to accept customer PIN entry and encrypt it using the account number and working key sent along in this message. The encrypted PIN block should be retrieved via message 71.

Note: **Z2/Z3 (PIN entry mode) required:** Message Z2 or Z3 (PIN entry mode) should be issued before this command, or PIN pad will send <EOT> and stop.

Aborting Transaction: Please refer to message 70.

PIN length: Please refer to message 70.

Master key must be selected before transaction: Please refer to message 70.

Triple DES capability: Please refer to message 70.

Session Key: If the selected key is with usage "P0", the session key should be all zeros.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z60	3	Message ID
.	1	<2E>, delimiter
[Account]	8 .. 19	Account number
<FS>	1	<1C>, Field separator
[Session key]	16 or 32	Session key encrypted with selected master key. 32-characters session key produces TDES encrypted PIN block with EDE order. Format: hexadecimal string. This filed should be all zeros if the selected key is with usage "P0"
<FS>	1	(Optional) <1C>, Field separator
[timeout]	1	(Optional) ASCII character from '1' to '9' which is the timeout value in the unit of 30 seconds. Default = 9x30 = 270 seconds.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z2 or Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Show prompt message
Message Z60	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 71 (after customer PIN entered), or <EOT> when input timed out or user pressed [CAN]
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message Z62 PIN entry request with customized prompt (MK/SK)

Format: **<STX>Z62.[account]<FS>[session key][minPIN][maxPIN] [null flag][prompt1]<FS>[prompt2]<FS>[proc prompt]<FS> [timeout]<ETX>[LRC]**

Message length: Variable 39 to 100 bytes (max. 116 bytes for TDES session key).

Usage: Request the PIN Pad to display the prompt message in this data frame, accept customer PIN entry and encrypt it using the account number and working key sent along in this message. Display the [proc prompt] when the PIN has been entered. The encrypted PIN block should be retrieved via message 71.

NOTE: **Aborting Transaction:** Please refer to message 70.

PIN length: Although Z62 allow programmer to specify the maximum and minimum PIN length, but it is not allowed to set the value of [maxPIN] and [minPIN] to exceed ANSI x9.8 specification except allow null PIN.

Master key must be selected before transaction: Please refer to message 70.

Triple DES capability: Please refer to message 70.

Session Key: If the selected key is with usage "P0", the session key should be all zeros.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z62	3	Message ID
.	1	<2E>, delimiter
[account]	8 .. 19	Account number
<FS>	1	<1C>, field separator
[session key]	16 or 32	Session key encrypted with selected master key. 32-characters session key produces TDES encrypted PIN block with EDE order. Format: hexadecimal string. This filed should be all zeros if the selected key is with usage "P0"
[minPIN]	2	00, 04 .. 12 minimum PIN length. ('00' only available when [null flag] set to 'Y').
[maxPIN]	2	00, 04 .. 12 maximum PIN length. ('00' only available when [null flag] set to 'Y').
[null flag]	1	Y Null PIN allowed N Null PIN not allowed
[prompt1]	1 .. 16	Prompt displayed before any key is pressed, alternate with prompt2
<FS>	1	<1C>, field separator
[prompt2]	1…16	Prompt displayed before any key is pressed, alternate with prompt1
<FS>	1	<1C>, field separator
[proc prompt]	1…16	Prompt displayed after PIN is entered
<FS>	1	(optional) <1C>, field separator
[timeout]	1	(optional) ASCII character from '1' to '9' which is the timeout value in the unit of 30 seconds. Default = 9x30 = 270 seconds.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z62	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		(Display [prompt1] and [prompt2] wait for user enter PIN)
	←	Message 71 (after customer PIN entered)
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	(Display [proc prompt])

➢ Message Z64 Query Key Check Value (KCV)

Format: **<STX>Z64[KeyId]<ETX>[LRC]**

Message length: Fixed 7 bytes.

Usage: This message will export the KCV of specified master key.

KCV is calculated as following:

1. Use [KeyID] specified key as encryption key.
2. Use "0000000000000000" (8 bytes zero) as data.
3. If the encrypt key is single length (8 bytes), use DES algorithm to encrypt the data, else, use TDES algorithm to encrypt the data.
4. Take leftmost 3 bytes as KCV, output KCV as message Z65.

Example: TDES key "0123456789ABCDEF FEDCBA9876543210" will have KCV as "08D7B4".

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z64	3	Message ID
[KeyID]	1	'0' ~ '9', 'A' ~ 'G', The ID of master keys
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z64	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z65
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message Z65 Key Check Value Response

Format: **<STX>Z65[KeyId][KCV]<ETX>[LRC]**

Message length: Variable. 13 bytes for KCV, or 8 bytes for error code.

Usage: This message is the response of Z64.

If [KeyID] specified in Z64 is pointing to a valid master key, the KCV will be sent.

Otherwise a question mark '?' will be sent.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z65	3	Message ID
[KeyID]	1	'0' ~ '9', 'A' ~ 'G', The ID of master keys
[KCV]	6 or 1	Success: 6 characters KCV. Fail: '?'.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

Refer to message Z64.

➢ Message Z66 Message Authentication Code (MAC) Request

Format: **<STX>Z66[PktType][SeqNo][KeyId] <FS> [SessionKey] <FS> [SecKeyId] <FS> [Message] <ETX> [LRC]**

Message length: Variable 14 to 270 bytes.

Usage: This message is used to generate MAC codes according to algorithm specified in ANSI X9.19 (ISO 9797-1). User can send ASCII strings or hexadecimal strings to PP190 by Z66 message to generate its MAC. User can also separate a long message into multiple Z66 messages with increasing sequence number to generate a MAC.

NOTE: **Message Length:** Onetime message can be up to 224 characters (equal to 112bytes when send as hexadecimal string because 2 characters represents 1 bytes). Multiple messages can have sequence number from 00 to 99, thus the maximum capacity of Z66 message is 22400 characters (or 11200 bytes in binary mode).

Multiple messages: When using multiple messages, [KeyId] and [SessionKey] and [SecKeyId] must be the same. [Message] must be the multiple of 8 characters (or 16 characters in binary mode). Or PP190 will generate a wrong MAC.

MAC algorithm: PP190 generate TDES MAC according to ISO9797-1 algorithm 3. (Padding with 0. Initial vector = 0. Refer to Appendix A point 10 for detail algoritgm.)

Session Key: The value of session key relates to the usage of specified master keys.

Usage of 1st Key ID	Usage of 2nd Key ID	Value of session key	MAC Key
"K0"	N/A	Non-zero	Session key
"M3" (mode G)	N/A	Zero	Master key specified by [KeyId]. If specified key is mode 'V', this is for MAC verification and cannot used to generate MAC for Z66 command.
"M1" (mode G)	"M1"	Zero	Master key specified by [KeyId] as left key, and master key specified by [SecKeyId] as right key.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z66	3	Message ID
[PktType]	1	'4' = ASCII last or only packet. '5' = ASCII first or middle of multiple packets. '6' = Binary last or only packet. '7' = Binary first or middle of multiple packets.
[SeqNo]	2	'00' to '99', for onetime only packet, set to 00.
[KeyId]	1	(Optional) Master key to use, range = 'B' to 'E'. If this filed is blank, the MAC master key will be the selected key 0 ~ 9.
<FS>	1	<1C>, field separator
[SessionKey]	32	Session key will be decrypted by: Master key pointed by [KeyId]. Format: hexadecimal string. This filed should be all zeros if the selected key is with usage "M1" or "M3"
<FS>	1	<1C>, field separator
[SecKeyId]	1	(Optional) Refer to note of Z66 usage. If first [KeyId] points to key with "K0" or "M3" usage, this field should be omitted.
<FS>	1	<1C>, field separator
[Message]	1-224	ASCII string or Hexadecimal string to be MACed.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: (Onetime only packets)

HOST	Direction	PIN Pad
Message Z66 (type 4,6)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z67 (with MAC)
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

Message flow: (Multiple packets)

HOST	Direction	PIN Pad
Message Z66 (Seq'00' and type 5,7)	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message Z67 (with status code '1')
<ACK> / <NAK> / <EOT>	→	
Message Z66 (Seq'01'--'98', type 5,7)	→	
	←	<ACK> / <NAK> / <EOT>
	←	Message Z67 (with status code '1')
……	……..	……..
Message Z66 (Sequence# larger than last packet, type 4,6)	→	
	←	<ACK> / <NAK> / <EOT>
	←	Message Z67 (with MAC)
<ACK> / <NAK> / <EOT>	→	

➢ Message Z67 Message Authentication Code (MAC) Response

Format: **<STX>Z67[status][MAC]<ETX>[LRC]**

Message length: Fixed 7 (status only) or 23 (with MAC) bytes.

Usage: PP190 generated MAC calculation response. It contains status codes or MAC.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z67	3	Message ID
[status]	1	'0'=Success, MAC follows '1'=Ready for next Z66 packet, user in multiple messages. '2'=Sequence numbers out of order '3'=Master key specified in [KeyId] not exist, or range unacceptable (id 0 to A), or usage not "K0", "M1", "M3. '4'=Master key specified in [SecKeyId] unreasonable or not exist. The [SecKeyId] only exists if [KeyId] points to a "M1" master key, and the [SecKeyId] itself should have "M1" usage. '5'=[Message] length have error (too long, zero length, or not even number in binary mode) '6'=[PkyType] flag has invalid value '7'=[Message] contents error (i.e. characters larger than 'F' in binary mode) '8'=[SessionKey] invaild '9'=MAC master key length should not be 8 'A'=Session key is incompatible to the usage of specified master key. (If MK's usage is "M1" or "M3", SK should contains all zero, if MK's usage is "K0", SK should not be zero.)
[MAC]	16	Calculated MAC.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

(Please refer to message Z66)

➢ Message Z7 Turn ON/OFF CANCEL Message Display

Format: **<STX>Z7[option]<ETX>[LRC]**

Message length: Fixed 6 bytes.

Usage: When a CANCEL message received or a CANCEL key pressed to cancel a current transaction, the PIN Pad will display a "CANCEL REQUESTED" message. This could be turned ON or OFF using message Z7.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z7	2	Message ID
[option]	1	0 CANCEL REQUESTED displayed 1 CANCEL REQUESTED not displayed
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z7	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		CANCEL REQUEST prompt turned ON/OFF

➢ Message Z8 Set Idle Prompt

Format: **<STX>Z8[prompt]<ETX>[LRC]**

Message length: Variable 6 to 21 bytes.

Usage: The PIN Pad will display an idle prompt when it is in IDLE state. HOST can change this idle prompt via message Z8. If the prompt field is filled with a null string, then the PIN Pad will use the default prompt afterwards.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z8	2	Message ID
[Prompt]	1 .. 16	Idle prompt to be used
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z8	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Displays idle prompt

Section 7 Online transaction messages with Derived Unique Key per Transaction (DUKPT)

The following messages are designed for Derived Unique Key Per Transaction (DUKPT) key management scheme described in ANSI X9.24-1992 and 2002 (Triple-DES DUKPT).

Note that some of the messages have the same IDs as those in MK/SK scheme, but with different message format.

[TDES Capability]

If PP190 receives double length key in message 90/94 (Load Initial Key), the following DUKPT operation will be done in TDES mode. PIN block will be TDES encrypted by derived key in EDE order.

[Secondary DUKPT Key Set]

PP190 provides 2nd key set of DUKPT operation for scalability. For example, customer can inject a DES initial key into key set 0 and a TDES initial key into key set 1, using key set 0 to process traditional DES transactions at first. When host systems ready to shift to TDES transaction, simply issue key set selection command (96) to make PP190 switch to key set 1 without recall all PP190 to inject new initial keys.

The following messages fall into this category:

60 Pre-Authorization PIN Entry Request
62 Pre-Authorization Amount Authorization Request
63 Pre-Authorization Amount Authorization Response
70 PIN entry request
78 PIN entry request via GUI
71 Encrypted PIN block response
72 PIN entry cancel
Z60 PIN entry request with external prompt (DUKPT)
Z62 PIN entry request with customized prompt
76 PIN Entry Test Request
90 Load First Initial Key Request
91 Load Initial Key Response
94 Load Second Initial Key Request
96 Select Active Key Set

➢ Message 60 Pre-authorization PIN Entry Request

Format: **<STX>60[account] <ETX>[LRC]**

Message length: Variable 13 to 24 bytes.

Usage: PIN pad will wait till the PIN entered and ENTER key is pressed. After PIN is entered, message 71 with PIN block will be sent as response. The HOST must transmit message 62 to ask for confirmation on transaction amount.

Note: **Z2/Z3 (PIN entry mode) required:** Message Z2 or Z3 (PIN entry mode) should be issued before this command, or PIN pad will send <EOT> and stop.
Aborting Transaction: Please refer to message 70(DUKPT).
PIN length: Please refer to message 70(DUKPT).

Message element:

Field	Length	Value and description
<STX>	1	<02>
60	2	Message ID
[Account]	8..19	Primary account number
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z2 or Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Show Prompt Messages
Message 60	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	(User enter PIN and press ENTER) Message 71
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
		Display "PIN PAD PROCESSING" until CLEAR pressed or another message received.
Message 62	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	(User confirm the amount) Message 63
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message 62 Pre-authorization Amount Authorization Request

Format: **<STX>62[DC Ind][amount]<ETX>[LRC]**

Message length: Variable 10 to 14 bytes.

Usage: Display prompt and accept customer PIN input. The following prompt will be displayed:

"Total Amount $xxx.xx"

"Enter – Confirm"

"Cancel – Decline"

xxx.x is the content of Amount field, with length between 4 to 8 positions. The PIN Pad will then wait till either CAN or ENTER key is pressed. If ENTER key is pressed, the PIN PAD will response with positive confirmation. If CAN is pressed, the PIN PAD will response a negative confirmation. During this period, the PIN Pad will not process any message other than the message 72(cancel transaction).

Message element:

Field	Length	Value and description
<STX>	1	<02>
62	2	Message ID
[DC Ind]	1	D/C: Debit/Credit Indicator
[amount]	4..8	Amount of goods to be displayed on PIN Pad.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

Please reference message 60.

➢ Message 70 PIN Entry Request (DUKPT)

Format: **<STX>70[account]<FS>[DC Ind][amount]<FS>[timeout]<ETX>[LRC]**

Message length: Variable 21 to 36 bytes.

Usage: Display prompt and accept customer PIN input. The following prompt will be displayed:

"Total Amount"

"$xxx.xx"

"Enter PIN"

"Push "ENTER""

xxx.x is the content of Amount field, with length between 4 to 8 positions. The PIN Pad will then wait till the PIN entered and [ENTER] key is pressed. After ENTER key is pressed, the string "PIN PAD" and "PROCESSING" will be displayed until the CLEAR key is pressed. During this period, the PIN Pad will not process any message other than the CANCEL message (message 72).

NOTE: **Aborting transaction:** Press CLEAR button to reset the PIN input and CAN (cancel) button to abort the transaction.

PIN length: According to ANSI X9.8 standard, the length of PIN should between 4 to 12 digits. If user inputs less than 4 digits and press ENTER, PIN pad will beep for error and continue to wait for user's input. When user inputs 13th character, PIN pad will beep for error, conserves PIN character 1st to 12th, and wait for ENTER.

Triple DES capability: If preloaded initial key is double length key, PP190 will produce TDES encrypted PIN block (EDE order).

Message element:

Field	Length	Value and description
<STX>	1	<02>
70	2	Message ID
[Account]	8..19	Primary account number
<FS>	1	<1C>, field separator
[DC Ind]	1	D/C: Debit/Credit Indicator
[Amount]	4..8	Amount of goods to be displayed on PIN Pad.
<FS>	1	(optional) <1C>, field separator
[timeout]	1	(optional) ASCII character from '1' to '9' which is the timeout value in the unit of 30 seconds. Default = 9x30 = 270 seconds.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 70	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 71 or <EOT> when [CAN] pressed or input timed out.
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
		Display "PIN PAD PROCESSING" until CLEAR pressed or another message received.

➢ Message 71 Encrypted PIN Block Response

Format: **<STX>71<fkey flag>[Key Serial#][PIN][LRC] (PIN block frame)**
<STX>71[error code]<ETX>[LRC] (Error code frame)

Message length: Variable 32 to 42 bytes.

Usage: Send the entered PIN to HOST in encrypted format.

Message element:

Field	Length	Value and description
<STX>	1	<02>
71	2	Message ID
[fkey flag]	1	Always '0' (This field is kept to retain old model compatibility.)
[Key Serial#]	10..20	Key Serial number used in encrypting PIN. Included only when PIN is entered. Format: hexadecimal string.
[PIN]	16	Encrypted PIN block Format: hexadecimal string.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

Please refer to message70 (DUKPT).

Error codes:

Code	Meaning
'0'	Null Account input field.
'2'	Account number shorter than 8 digits.
'3'	Account number longer than 19 digits.
'4'	Account number have character other than '0'-'9'.
'5'	[D/C ind] field not exist or format error.
'6'	Timeout value error.
'8'	Amount string format error.
'A'	No DUKPT key injected.
'B'	Flash read/write error.
'C'	Memory buffer allocation error.
'F'	DUKPT operation limit (1 million) reached, program stop.
'G'	Specified file not found or authentication error.

'H'	Receive command 72.
'I'	Cancel key is press.
'J'	PIN entry timeout.

➢ Message 72 PIN Entry Cancel

Format: **<STX>72<ETX>[LRC]**

Message length: Fixed 5 bytes.

Usage: Cancel current transaction and return the PIN Pad to IDLE state, used to interrupt command in process. If PIN Pad receives message 72 while processing user input such as signing, swipe card, enter PIN or key-in data, It will respond with <EOT> to acknowledge that operation is canceled.

Message element:

Field	Length	Value and description
<STX>	1	<02>
72	2	Message ID
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 72	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	<EOT> Optional. If PIN pad is waiting for user's input.

➢ Message Z60 PIN entry request with external prompt (DUKPT)

Format: **<STX>Z60.[account]<FS>[timeout]<ETX>[LRC]**

Message length: Variable 15 to 28 bytes.

Usage: Request the PIN Pad to accept customer PIN entry and encrypt it using the account number and working key sent along in this message. The encrypted PIN block should be retrieved via message 71.

Note: **Z2/Z3 (PIN entry mode) required:** Message Z2 or Z3 (PIN entry mode) should be issued before this command, or PIN pad will send <EOT> and stop.

Aborting Transaction: Please refer to message 70.

PIN length: Please refer to message 70.

Triple DES capability: Please refer to message 70.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z60	3	Message ID
.	1	<2E>, delimiter
[Account]	8 .. 19	Account number
<FS>	1	(Optional) <1C>, Field separator
[timeout]	1	(Optional) ASCII character from '1' to '9' which is the timeout value in the unit of 30 seconds. Default = 9x30 = 270 seconds.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z2 or Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Show prompt message
Message Z60	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 71 (after customer PIN entered), or <EOT> when input timed out or user pressed [CAN]
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message Z62 PIN entry request with customized prompt (DUKPT)

Format: **<STX>Z62.[account]<FS>[minPIN][maxPIN][null flag]**
[prompt1]<FS>[prompt2]<FS>[proc prompt]<FS>[timeout]<ETX>[LRC]

Message length: Variable 39 to 100 bytes.

Usage: Request the PIN Pad to display the prompt message in this data frame, accept customer PIN entry and encrypt it using the account number and working key sent along in this message. Display the [proc prompt] when the PIN has been entered. The encrypted PIN block should be retrieved via message 71.

NOTE: **Aborting Transaction:** Please refer to message 70.

PIN length: Although Z62 allow programmer to specify the maximum and minimum PIN length, but it is not allowed to set the value of [maxPIN] and [minPIN] to exceed ANSI x9.8 specification except allow null PIN.

Triple DES capability: Please refer to message 70.

Message element:

Field	Length	Value and description
<STX>	1	<02>
Z62	3	Message ID
.	1	<2E>, delimiter
[account]	8 .. 19	Account number
<FS>	1	<1C>, field separator
[minPIN]	2	00, 04 .. 12 minimum PIN length. ('00' only available when [null flag] set to 'Y').
[maxPIN]	2	00, 04 .. 12 maximum PIN length. ('00' only available when [null flag] set to 'Y').
[null flag]	1	Y Null PIN allowed N Null PIN not allowed
[prompt1]	1 .. 16	Prompt displayed before any key is pressed, alternate with prompt2
<FS>	1	<1C>, field separator
[prompt2]	1…16	Prompt displayed before any key is pressed, alternate with prompt1
<FS>	1	<1C>, field separator
[proc prompt]	1…16	Prompt displayed after PIN is entered
<FS>	1	(optional) <1C>, field separator
[timeout]	1	(optional) ASCII character from '1' to '9' which is the timeout value in the unit of 30 seconds. Default = 9x30 = 270 seconds.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message Z2 or Z3	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		Show prompt message
Message Z62	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
		(Display [prompt1] and [prompt2] wait for user enter PIN)
	←	Message 71 (after customer PIN entered)
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	(Display [proc prompt])

➢ Message 76 PIN Entry Test Request

Format: **<STX>76[account]<FS>[DC Ind][amount]<ETX>[LRC]**

Message length: Variable 19 to 34 bytes.

Usage: This message is designed to do DUKPT continuous PIN entry test. PP190 will send message71 assuming a PIN of '1234'.

Message element:

Field	Length	Value and description
<STX>	1	<02>
76	2	Message ID
[Account]	8..19	Primary account number
<FS>	1	<1C>, field separator
[DC Ind]	1	D/C: Debit/Credit Indicator
[Amount]	4..8	Amount of goods to be displayed on PIN Pad.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: This message is identical to message70 except that a PIN of '1234' is used instead of getting keypad input.

➢ Message 7A KSN output format

Format: **<STX>7A[KSN_format] <ETX>[LRC]**

Message length: Fixed 6 bytes.

Usage: This message will set the key serial number (KSN) format of message 71 (DUKPT frame). Format 0 is the original mode (strip leading 'F' of KSN) which is compatible of PP690, PP790SE and PP795, Format 1 is full mode (output full 20 characters of KSN).

Message element:

Field	Length	Value and description
<STX>	1	<02>
7A	2	Message ID
[KSN_format]	1	'0': message 71 output KSN without leading 'F' '1': message 71 output KSN with leading 'F'.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 7A	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)

➢ Message 90 Load First Initial Key Request

Format: **<STX>90[IPEK][KSN]<ETX>[LRC] (Clear Text)**
<STX>90[TR-31 Key Block]<ETX>[LRC] (Encrypted)

Message length: Fixed 41 or 57 bytes for clear text format, 93 or 109 bytes for TR-31 format.

Usage: Load first set of DUKPT initial key and serial number key to PP190. Consequent keys will be generated using provided data.
If 32-characters (double length) initial key being loaded, PP190 will do key generation, PIN entry, and other DUKPT operations in TDES manner.

PP190 implements multiple security measures to conform Payment Card Industry (PCI) security requirement. In order to load clear text IPEK key, two authorized people with their password are required. Otherwise the user must issue message 90 with encrypted key value (ANSI TR31 format). See "**Symmetric Keys Loading Authentication**" for detailed information.

Note: VISA required key serial number format are as follows:
4'F' characters, a 6-digit keyset identifier, 5-digit device ID, followed by a '0',
i.e. "FF FF kk kk kk dd dd d0 00 00"

Message element:

(Clear text format)

Field	Length	Value and description
<STX>	1	<02>
90	2	Message ID
[IPEK]	16 or 32	Initial PIN encryption key. 32-characters Initial key will make PP190 act in TDES DUKPT mode. Format: hexadecimal string.
[KSN]	20	Key serial number used in generating consequent keys. Format: hexadecimal string.
<ETX>	1	<03>
[LRC]	1	Checksum

(Encrypted format)

Field	Length	Value and description
<STX>	1	<02>
90	2	Message ID
[TR-31 Key Block]	88 or 104	TR-31 key block with optional header block that contains KSN. See Appendix A for detail.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 90	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 91
<ACK>/<NAK>/<EOT>	→	

Example:

Clear Text

IPEK key to be loaded: ABCDEF0123456789FEDCBA9876543210

KSN: FFFF9876543210E00000

The resulting 90 message :

<STX>90ABCDEF0123456789FEDCBA9876543210FFFF9876543210E00000<ETX>[LRC]

TR-31 Key Block

Key Block Protecting Key: AA55AA55AA55AA55 3434343434343434

IPEK key to be loaded: ABCDEF0123456789 FEDCBA9876543210

KSN: FFFF9876543210E00000

Key Block Header: B0104B1TX00N0100 KS18FFFF9876543210E00000

Padded IPEK: 0080 ABCDEF0123456789 FEDCBA9876543210 30111D18CC4C

Derived KBEK: 3C50E1B7962F2171DC8643F1D923ABF7

Derived KBMK: 46FBEEB64EAE26A650952DA4F6DD8325

CMAC of (KBH + Padded key data), using KBMK: 93C3D5EBC6C407E4

Use CMAC as IV to do TDES CBC encryption on padded key data, using KBEK:

Encrypted key data: EC86E6E3B24544F97C629FB0E0586A0285D35BA78E9B13FB

Result: <02>90B0104B1TX00N0100KS18FFFF9876543210E00000EC86E6E3B24544F9
7C629FB0E0586A0285D35BA78E9B13FB93C3D5EBC6C407E4<03>

➢ Message 91 Load Initial Key Response

Format: **<STX>91[Status]<ETX>[LRC]**

Message length: Variable (max 7 bytes.)

Usage: Confirmation of the initial key loading. PP190 will also show a message "IPEK n loaded" (n = 1 or 2) to confirm the success loading of initial key of set 1 and set 2 visually.

Message element:

Field	Length	Value and description
<STX>	1	<02>
91	2	Message ID
[Status]	1..2	'0' if successful '1' + [Error Code] if process failed.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: Please reference message90.

Error codes:

Code	Meaning
'1'	Processing message 90 without authentication, process authentication at first
'2'	IPEK and KSN format error: not hexadecimal character.
'3'	Data length error.
'4'	Internal memory allocation error.
'5'	Cannot read internal flash memory.
'7'	Cannot write new IPEK into flash memory.
'F'	DUKPT 1 million limit reached or flash memory write cycle has been exhausted.

➢ Message 94 Load Second Initial Key Request

Format: **<STX>94[IPEK][KSN][Key MAC]<ETX>[LRC]**

Message length: Fixed 41 bytes (57 bytes for TDES initial key).

Usage: Load second set of DUKPT initial key and serial number key to PP190. Consequent keys will be generated using provided data.

If 32-characters (double length) initial key being loaded, PP190 will do key generation, PIN entry, and other DUKPT operations in TDES manner.

PP190 will reject message 94 if it has not yet get the authentication. See **Symmetric Keys Loading Authentication** section for detail.

Note: VISA required key serial number format are as follows:

4'F' characters, a 6-digit keyset identifier, 5-digit device ID, followed by a '0',

i.e. "FF FF kk kk kk dd dd d0"

Message element:

Field	Length	Value and description
<STX>	1	<02>
94	2	Message ID
[IPEK]	16 or 32	Initial PIN encryption key. 32-characters Initial key will make PP190 act in TDES DUKPT mode. Format: hexadecimal string.
[KSN]	20	Key serial number used in generating consequent keys. Format: hexadecimal string.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 94	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message 91
<ACK>/<NAK>/<EOT>	→	

➢ Message 96 Select Active Key Set

Format: **<STX>96[keyset]<ETX>[LRC]**

Message length: Fixed 6 bytes.

Usage: Select active key set for following transactions. This parameter is kept in flash memory and lasts until next 96 message or DUKPT life cycle ends.

Message element:

Request frame (HOST to PIN Pad)

Field	Length	Value and description
<STX>	1	<02>
96	2	Message ID
[keyset]	1	ASCII character '0' = First key set '1' = Second key set
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message 96 request frame	→	
	←	<ACK>/<NAK>/<EOT>

Section 8 Remote key injection method

➢ Message R00 Load Vender Public Key

Format: <STX>R00[Last pkt][Pkt no.]<FS>[Exp len][Exp data]<FS>[Mod data]<ETX>[LRC]

Message length: Variable

Usage: This message is used to load vender public key to PP190. If vender public key is already exists in PP190, PP190 will return error, in this circumstance, user should use R01 command to update vender public key.

Message element:

Request fame (HOST to PP190)

Field	Length	Value and description
<STX>	1	<02>
R00	3	Message ID
[Last pkt]	1	'0': packet is not last. '1': packet is last.
[Pkt no.]	1	Packet sequence number. Range:'1'~'9'
<FS>	1	(optional, only first packet need) <1C>, Field separator
[Exp len]	1	(optional, only first packet need) Exponent length, value from '1' to '8'.
[Exp data]	1~8	(optional, only first packet need) Exponent data, hexadecimal string for exponent data.
<FS>	1	(optional, only first packet need) <1C>, Field separator
[Mod data]	Var. (Max. 256-byte per transmit)	(optional) Hexadecimal string for Modulus data to be loaded, and the modulus total length must be 512 bytes.
<ETX>	1	<03>
[LRC]	1	Checksum

Note: Modulus data must be 256 bytes (2048 bits), and it should convert to hexadecimal string for transmission, so modulus data is 512 bytes hexadecimal string in transmission. Because hardware restriction, host could send R00 command with several packet:

1. First packet contain:
   [Last pkt][Pkt no.]<FS>[Exp len][Exp data]<FS>a part of [Mod data].
2. The rest of packets contains:
   [Last pkt][Pkt no.] and the rest of [Mod data].
3. PP190 will send R00 response (R00F if success) after host send all command packet ([Last pkt] = '1').

Note: If the received packet's time interval is over 1 minute, PP190 will return timeout error.

Reply fame (PP190 to HOST)

Field	Length	Value and description
<STX>	1	<02>
R00	3	Message ID
[Status]	1	Status byte: '0': Packet received success, and wait for next packet. '1': Error occurred, abort transmission. 'F': Load vendor public key successfully.
[ErrCode]	1	(optional, if [Status] = 1) '1' = Command format error. '2' = Memory allocate fail. '3' = Vendor public key has loaded. '4' = received modulus length is not equal to user assigned length. '5' = Flash write fail. '6' = Timeout.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message R00	→	
	←	<ACK> <EOT>
	←	Message R00 ([Status] = '0')
<ACK> <EOT>	→	
Message R00 (while modulus data not end)	→	
	←	<ACK> <EOT>
	←	Message R00 ([Status] = '0')
……	……..	……
Message R00(with [Last pkt] = '1')		
		<ACK> <EOT>

	←	Send message R00 (reply frame).
<ACK> <EOT>	→	

➢ Message R01 Update RSA Key

Format:<STX>R01[Last pkt][Pkt no.][Key type][Data type]<FS>[Sig data]<ETX>[LRC] (1st message)
<STX>R01[Last pkt][Pkt no.][Key type][Data type]<FS>[Exp data]<ETX>[LRC] (2nd message)
<STX>R01[Last pkt][Pkt no.][Key type][Data type]<FS>[Mod data]<ETX>[LRC] (3rd message)

Message length: Variable.

Usage: This message is used to update PP190 Remote Key Injection RSA key (include Vender Public Key、Server Public Key、PP190 Private Key and PP190 Public Key).

Description:

Steps for update RSA key.

1. Calculate SHA256 hash value of new RSA key(32-byte), and following RSA Cryptography Standard to generate signature packet, reference to PKCS#1 v2.2:
   a. Generate an hexadecimal string ***PS*** with value 0xff, ***PS*** length equals to Sig msg Len - ***T*** Len - 3.
   b. Since PP190 only use SHA256 to generate hash value, the DigestInfo value of ***T*** (***D***)should be:
      (0x) 30 31 30 0d 06 09 60 86 48 01 65 03 04 02 01 05 00 04 20
      and ***T*** should be:
      D || Hash value of new RSA key(32-byte).
   c. Sig msg = 0x00 || 0x01 || ***PS*** || 0x00 || ***T***.
2. Generate RSA digital signature by Vender private key.
3. Issue 1st message R01 to PIN pad.
4. PIN pad will use Vender public key to authenticate the message format (the length of digital signature should equal to RSA modulus stored in PIN pad) and issues 1st message R01 with the authenticate result.
5. Issue 2nd and 3rd message R01 to PIN pad.
6. PIN pad will calculate the hash value of the [Exponent] and [Modulus] data from 2nd and 3rd message R01 and compare the hash value that decrypt from 1st message R01.

Message element:

1st:

Request fame (HOST to PP190)

Field	Length	Value and description
<STX>	1	<02>
R01	3	Message ID
[Last pkt]	1	'0': packet is not last. '1': packet is last.

[Pkt no.]	1	Packet sequence number. Range:'1'~'9'
[Key type]	1	(optional, only first packet need) '1': Vender public key '2': Server public key '3': PP190 public key '4': PP190 private key
[Data type]	1	(optional, only first packet need) Value: '1' (Signature data).
<FS>	1	(optional, only first packet need) <1C>, Field separator
[Sig data]	Var. (Max. 256-byte per transmit)	(optional) Hexadecimal string for Signature data of RSA key, and the total signature length must be 512 bytes.
<ETX>	1	<03>
[LRC]	1	Checksum

2nd:

Request fame (HOST to PP190)

Field	Length	Value and description
<STX>	1	<02>
R01	3	Message ID
[Last pkt]	1	'0': packet is not last. '1': packet is last.
[Pkt no.]	1	Packet sequence number. Range:'1'~'9'
[Key type]	1	(optional, only first packet need) '1': Vender public key '2': Server public key '3': PP190 public key '4': PP190 private key
[Data type]	1	(optional, only first packet need) Value: '2' (Exponent data).
<FS>	1	(optional, only first packet need) <1C>, Field separator
[Exp data]	Var. (Max. 256-byte per transmit)	(optional) Hexadecimal string for exponent data. (1~8 bytes for public key and

		for private key must be 512 bytes.)
<ETX>	1	<03>
[LRC]	1	Checksum

3rd:

Request fame (HOST to PP190)

Field	Length	Value and description
<STX>	1	<02>
R01	3	Message ID
[Last pkt]	1	'0': packet is not last. '1': packet is last.
[Pkt no.]	1	Packet sequence number. Range:'1'~'9'
[Key type]	1	(optional, only first packet need) '1': Vender public key '2': Server public key '3': PP190 public key '4': PP190 private key
[Data type]	1	(optional, only first packet need) Value: '3' (Modulus data).
<FS>	1	(optional, only first packet need) <1C>, Field separator
Modulus data	Var. (Max. 256-byte per transmit)	(optional) Hexadecimal string for Modulus data to be loaded, and the total signature length must be 512 bytes..
<ETX>	1	<03>
[LRC]	1	Checksum

Note:

1st message:

Signature data must be 256 bytes, and it should convert to hexadecimal string for transmission, so signature data is 512 bytes hexadecimal string in transmission. Because hardware restriction, host could send R01 command with several packets:

1. First packet contain:
   [Last pkt][Pkt no.][Key type][Data type]<FS>a part of [Sig data].
2. The rest of packets contains:
   [Last pkt][Pkt no.] and the rest of [Sig data].
3. PP190 will send load success response (R010 if success) after host send all signature data packet ([Last pkt] = '1').

2nd message:

Exponent data could be 1~8 bytes (public key) or 512 bytes (PP190 private key) hexadecimal string, host could send R01 command as following:

1. First packet contain:

   If [Key type] = 4 (PP190 private key):

   [Last pkt][Pkt no.][Key type][Data type]<FS>a part of [Exp data].

   The rest of packets contains:

   [Last pkt][Pkt no.] and the rest of [Exp data].

   Else

   [Last pkt][Pkt no.][Key type][Data type]<FS>[Exp data]

2. PP190 will send load success response (R010) after host send all exponent data packet ([Last pkt] = '1').

3rd message:

Modulus data must be 512 bytes hexadecimal string, host could send R01 command with several packet:

1. First packet contain:

   [Last pkt][Pkt no.][Key type][Data type]<FS>a part of [Mod data].

2. The rest of packets contains:

   [Last pkt][Pkt no.] and the rest of [Mod data].

3. PP190 will send load success response (R01F if success) after host send all modulus data packet ([Last pkt] = '1').

Reply fame (PP190 to HOST)

Field	Length	Value and description
<STX>	1	<02>
R01	3	Message ID
[Status]	1	Status byte: '0': Packet received success, and wait for next packet. '1': Error occurred, abort transmission. 'F': Load vendor public key successfully.
[ErrCode]	1	(optional, if [Status] = '1') ASCII character. '1' = SHA engine is busy '2' = Vendor public key not loaded '3' = Memory allocate fail '4' = Command format error '5' = Last state error

		'6' = Packet number error '7' = Key type error '8' = Data type error '9' = Range of key length error 'A' = Received modulus length is not equal to user assigned length. 'B' = Signature original data format error 'C' = Hash compare error 'D' = Flash write error 'E' = Flash erase error 'H' = AES engneer error 'I' = Timeout
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
1st Message R01	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')
<ACK> <EOT>	→	
1st Message R01 (while data not end)	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')
……	……..	……
	←	Message R01 ([Status] = '0')
1st Message R01 (with [Last pkt] = '1')	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')
2nd Message R01	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')

<ACK> <EOT>	→	
2nd Message R01 (while data not end)	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')
……	……..	……
	←	Message R01 ([Status] = '0')
2nd Message R01 (with [Last pkt] = '1')	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')
3rd Message R01	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')
<ACK> <EOT>	→	
3rd Message R01 (while data not end)	→	
	←	<ACK> <EOT>
	←	Message R01 ([Status] = '0')
……	……..	……
	←	Message R01 ([Status] = '0')
3rd Message R01 (with [Last pkt] = '1')	→	
	←	<ACK> <EOT>
	←	Send message R01 (reply frame).
<ACK> <EOT>	→	

➢ Message R02 Remote Key Injection

Format: <STX>R02[Last pkt][Pkt no.][Data type]<FS>[Sig data]<ETX>[LRC] (1st message)
<STX>R02[Last pkt][Pkt no.][Data type]<FS>[Enc data]<ETX>[LRC] (2nd message)

Message length: Variable

Usage: This message is used to load Terminal Master Key(TMK) to PP190.
(Terminal Master Key length should be 16-byte or 24-byte)

Description:

Steps for Remote Key Injection:

1. Send R02 packet, and PP190 will reply R02 with **Sig[PP_{pub}]$Vendor_{priv}$** to host, authenticate the Signature of PP190 Public Key, then use this public key to encrypt the Key Encryption Key (**MK or DUKPT**) and use server's private key to sign the hash value of Key Encryption Key (**MK or DUKPT**).
2. Calculate SHA256 hash value of Terminal Master Key (32-byte), and following RSA Cryptography Standard to generate signature packet, reference to PKCS#1 v2.2:
   a. Generate an hexadecimal string ***PS*** with value 0xff, ***PS*** length equals to Sig msg Len - ***T*** Len - 3.
   b. Since PP190 only use SHA256 to generate hash value, the DigestInfo value of ***T*** (***D***)should be:
      (0x) 30 31 30 0d 06 09 60 86 48 01 65 03 04 02 01 05 00 04 20
      and ***T*** should be:
      D || Hash value of new KEK key(32-byte).
   c. Sig msg = 0x00 || 0x01 || ***PS*** || 0x00 || ***T***.
3. Generate RSA digital signature by Server private key:
   Sig data = RSA(Sig msg)
4. Issue 1st message R02 to PIN pad.
5. PIN pad will use Server public key to authenticate the message format (the length of digital signature should equal to RSA modulus stored in PIN pad) and issues 1st message R02 with the authenticate result.
6. Use Terminal Master Key to generate a PKCS#1 encryption format, reference to PKCS#1 v2.2:
   a. Generate an hexadecimal string ***PS*** with random generated hex characters (***PS*** could not have any 0x00), ***PS*** length equals to Enc data Len - ***KEK*** Len - 3.
   b. Enc msg = 0x00 || 0x02 || ***PS*** || 0x00 || ***KEK***.
7. Generate Enc data by PP190 public key, which should be pre-load to host before R02 command:
   Enc data = RSA(Enc msg)
8. Issue 2nd message R02 to PIN pad.
9. PIN pad will use PP190 private key to decrypt data from 2nd message R02, and calculate the hash value of the TMK, then compare the hash value that decrypt from 1st message R02.

Message element:

1st:

Request fame (HOST to PP190)

Field	Length	Value and description
<STX>	1	<02>
R02	3	Message ID
[Last pkt]	1	'0': packet is not last. '1': packet is last.
[Pkt no.]	1	Packet sequence number. Range:'1'~'9'
[Data type]	1	(optional, only first packet need) Value: '1'(Signature data).
<FS>	1	(optional, only first packet need) <1C>, Field separator
[Sig data]	Var. (Max. 256-byte per transmit)	(optional) Hexadecimal string for Signature data. [Signature data]: Hash value of TMK with PKCS#1 format signed by Server private key. The signature total length must be 512 bytes.
<ETX>	1	<03>
[LRC]	1	Checksum

2nd:

Request fame (HOST to PP190)

Field	Length	Value and description
<STX>	1	<02>
R02	3	Message ID
[Last pkt]	1	'0': packet is not last. '1': packet is last.
[Pkt no.]	1	Packet sequence number. Range:'1'~'9'
[Data type]	1	(optional, only first packet need) Value: '2' (Encrypted KEK data).
<FS>	1	(optional, only first packet need) <1C>, Field separator
[Enc data]	Var. (Max. 256-byte	(optional) Hexadecimal string for encrypted data.

	per transmit)	[Encrypt data]: Encryption value of TMK (by PP190 public key). The encrypted total length must be 512 bytes.
<ETX>	1	<03>
[LRC]	1	Checksum

Note:

1st message:

Signature data must be 256 bytes, and it should convert to hexadecimal string for transmission, so signature data is 512 bytes Hex string in transmission.

Because hardware restriction, host could send R02 command with several packet:

1. First packet contain:
   [Last pkt][Pkt no.][Data type]<FS>a part of [Sig data].
2. The rest of packets contains:
   [Last pkt][Pkt no.] and the rest of [Sig data].
3. PP190 will send load success response (R020 if success) after host send all signature data packet ([Last pkt] = '1').

2nd message:

Encrypted data must be 512 bytes hexadecimal string, host could send R02 command with several packet:

1. First packet contain:
   [Last pkt][Pkt no.][Data type]<FS>a part of [Enc data].
2. The rest of packets contains:
   [Last pkt][Pkt no.] and the rest of [Enc data].
3. PP190 will send load success response (R02F if success) after host send all encrypted data packet ([last pkt] = '1').

Reply fame (PP190 to HOST)

Field	Length	Value and description
<STX>	1	<02>
R02	3	Message ID
[Sig data]	512 bytes	(Only first R02 packet gets response with [Sig data].) Hexadecimal string for **Sig[PP$_{pub}$]Vendor$_{priv.}$** Signature of PP190 public key.
[Status]	1	Status byte: '0': Packet received success, and wait for next packet. '1': Error occurred, abort transmission. 'F': Load vendor public key successfully.
[ErrCode]	1	(optional, if [Status] = 1)

		'1' = SHA engine is busy '2' = KEK length error '3' = Memory allocate fail '4' = Command format error '5' = Last state error '6' = Packet number error '7' = Verify encrypted format error '8' = Data type error '9' = Range of key length error 'A' = Received modulus length is not equal to user assigned length. 'B' = Signature original data format error 'C' = PP190 public key is not loaded 'D' = Server public key is not loaded 'E' = PP190 private key is not loaded 'H' = KEK load fail. 'I' = AES engineer fail. 'J' = Hash compare error 'K' = Time out. 'L' = PP190 private key verify fail. 'M' = PP190 vendor public key is not loaded.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Send R02 packet	→	
	←	<ACK> <EOT>
	←	Message R02 with [Sig data]
<ACK> <EOT>	→	
1st Message R02	→	
	←	<ACK> <EOT>
	←	Message R02 ([Status] = '0')
<ACK> <EOT>	→	
1st Message R02 (while data not end)	→	

	←	<ACK> <EOT>
	←	Message R02 ([Status] = '0')
……	……..	……
	←	Message R02 ([Status] = '0')
1st Message R02 (with [Last pkt] = '1')		
	←	<ACK> <EOT>
	←	Message R02 ([Status] = '0')
2nd Message R02	→	
	←	<ACK> <EOT>
	←	Message R02 ([Status] = '0')
<ACK> <EOT>	→	
2nd Message R02 (while data not end)	→	
	←	<ACK> <EOT>
	←	Message R02 ([Status] = '0')
……	……..	……
	←	Message R02 ([Status] = '0')
2nd Message R02 (with [Last pkt]='1')	→	
	←	Send message R02 (reply frame).
<ACK> <EOT>	→	

Section 9 EMV Level 2 transaction messages

EMV Level2 transaction messages are divided into 2 groups. One is EMV-configuration data operation messages (T01, T03, T05, T07, T09 and T0B) and the other one is EMV-transaction messages (T11, T13, T15, T17, T19, T1C, T21, T25, T27 and T29).

The EMV transaction messages issuing sequence is control by PIN pad, an invalid sequence will terminate EMV transaction. At the beginning of EMV transaction, user has to issue messages T11 to make PIN pad negotiate with card and generate a candidate list of EMV-application that supported by PIN pad and card both, and then select a highest priority one automatically or selected by user (according to the terminal configuration data installed in PIN pad), finally return the EMV-application name. Message T15 is used for terminal-side to transmit transaction information such as amount and then PIN pad do a complete transaction with card if the transaction needs not to be authorized online. Terminal can issue message T1D to transmit additional transaction data into PIN pad for EMV transaction, such as online response data, magnetic stripe card track data. Message T17 is applied if the transaction needs to be authorized online, terminal-side will transmit necessary information via this message to PIN pad to continue the rest steps of transaction. If the response from host contains issuer script (see EMV Book), terminal-side applies message T19 to input these scripts into PIN pad and PIN pad will issue these scripts at appropriate time to card. Message T1C is used to terminate an EMV transaction. Finally, message T21 is used for terminal-side to get the transaction information through EMV transaction.

Terminal can apply Txx messages to handle a complete EMV transaction except that the transaction must be changed to magnetic stripe card processing. According to EMV rule, if terminal fails to read IC card, the transaction could be change to magnetic stripe card transaction. Because of different types of magnetic stripe card, the magnetic stripe card processing should be taken by terminal. Terminal could issue message Q1 provided by PIN pad to make user swipe his card and then issue message 70 to complete a magnetic stripe card transaction. In this situation, terminal will get response of T11 message that indicates an failed IC card read, terminal should then issue message T1D, T15 and T17 to PIN pad for batch data capture. The flow chart for changing to magnetic stripe card processing could be referred in "Overall EMV level 2 transaction flow reference" section.

➢ Message T51 Terminal Configuration Setup

Format: **<STX>T51[Pkt No.][Total Pkts]<SUB>[DO]<ETX>[LRC]**

Message length: Variable.

Usage: Host can use this command to send **terminal configuration data** to PIN pad, this command can be sent many times. PIN pad will save those data inside and apply those data when do the transaction. PIN pad will send the message T52 (Terminal Configuration Setup Response) to host.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T51	3	Message ID
Pkt No.	1	Decimal. Packet sequence number (1 ~ 9)(ex. 2)
Total Pkts	1	Decimal. Total packets (1~9)(ex. 8).
<SUB>	1	<1A, Optional, only if [DO] is existed
DO	Var.	Data Object, format as below
<ETX>	1	<03>
[LRC]	1	Checksum

Data Object:

Each <DO> shall include three data field: [Tag#] || [Format] || [Value], and each field shall delimit with a <FS>. Each data object is delimited by a <SUB> to construct multiple <DO>. The [tag#] defined in EMV 4.1 Book3 Annex A and specific [tag#] defined at Appendix D of this manual have the pre-defined data format and length range, those [tag#] must follow up the rule, otherwise the PIN Pad will reject this data setup.

Data Format: (Please also refer to EMV 4.1 BOOK3, section 4.3)

Format	Description
1	a - Alphabetic data (a ~z, A~Z)
2	b - unsigned binary numbers or bit combinations
3	an - Alphanumeric data (a ~z, A~Z, 0~9)
4	ans - Alphanumeric Special data (Characters defined in ISO8859)
5	cn - Compressed numeric data (0~9, left justify, pad hexadecimal 'F's. Ex. 12 34 56 12 3F FF)
6	n - Numeric data (0~9, right justify, pad leading hexadecimal zeroes. Ex. 00 00 00 01 23 45)
7	var - Variable data (Any bit combination)

Note: Please be careful that only when all data objects send to PIN Pad correctly with sequence packet number within total packets number, these data will be saved to PIN Pad. Any update shall include whole items of <DO>, because previous setup <DO> will be lost when update!

Note. If the data format is '2' (binary), '5'(compressed numeric), '6'(numeric) or '7'(variable data), the [value] correspond to these format can not be sent with these format directly in message T51. It shall be converted to hexadecimal string and with pad char 'F' in the last one for format 'cn' if this tag# has odd chars or with pad char '0' in the first one for format 'b' or 'n' if this tag# has odd chars.

Example: (Clear the terminal configuration data and then setup new data.)
Merchant Category Code: 0000 (Numerical)
Terminal ID: SmartPOS (Ascii)
UI Capability: 0x01 (binary)

<STX>T5111<SUB>9f15<FS>6<FS>0000<SUB>9f1c<FS>3<FS>SmartPOS<SUB>
50000002<FS>2<FS>01<ETX>[LRC]

PIN pad will check if terminal downloads minimum set of terminal-related information into PIN pad. The download process will be failed if there is not enough data in this message. Please refer to appendix E for minimum set of terminal-related data

Message flow:

HOST	Direction	PIN Pad
1st Message T51	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	1st Message T52
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
2nd Message T51	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	2nd Message T52
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
…	…	…
Last one Message T51	→	

	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Last one Message T52
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message T52 Terminal Configuration Setup Response

Format: **<STX>T52[Res][Reason][Err Msg]<ETX>[LRC]**

Message length: Variable.

Usage: The response message of command T51.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T52	3	Message ID
[Res]	1	'0': Ok, '1': Fail
[Reason]	1	<Optional, if Res = '1'> '1': Fatal Error '2': Format Error '3': Invalid Data Object format. '4': Invalid Tag value
[Err Message]	8	Optional, if Reason = '1', Hex decimal string
[Err Tag Number]	Var.	Optional, if Reason = '3' or '4', Hex decimal string
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: Please refer to message T51.

➢ Message T53 Certificate Authority Public Key Setup

Format: **<STX>T53[Pkt No.][Total Pkts][RID][PKI][Key Size][Hash Alg][PK Alg][Exponent Len][Exponent][Hash]<ETX>[LRC]**

<STX> T53[Pkt No.][Total Pkts][Modulus Len][Modulus]<ETX>[LRC]

Message length: Variable.

Usage: Host can use this command to send the **Certificate Authority Public key data** to PIN pad, each command can only setup one key but this command can be sent many times. PIN pad will save those key data inside and use those data when do the transaction. PIN pad will send the message T54 (Certificate Authority Public Key Setup Response) to host. The data installed into PIN pad via this message, PIN pad will save it in internal storage structure with a name same as concatenation of value in [RID] and [PKI] fields. Ex. value in [RID] field is "A000000003", value in [PKI] filed is "90", PIN pad will save these data and give an ID as "A00000000390".

Message element:

1st Packet (Load RSA public key):

Field	Length	Value and description
<STX>	1	<02>
T53	3	Message ID
Pkt No.	1	Decimal. Packet sequence number (1 ~ 9)
Total Pkts	1	Decimal. Total packets (1~9)(ex. 8).
RID	10	Hexadecimal string, the left 5 bytes of EMV Application ID.
PKI	2	Public Key Index, hexadecimal string. (Refer to EMV 4.1, tag '9f22')
Key Size	4	Public Key size, hexadecimal string. Key: [Hash Alg] || [PK Alg] || [Exponent Len] || [Exponent] || [Hash] || [Modulus Len] || [Modulus] The value is displayed as big endian and is half of the key hex string. For example: '00A9'=169 bytes. And key string will be 338 bytes.
Hash Algorithm	2	Hash Algorithm Index, hexadecimal string '01': SHA-1. Now, PIN pad accepts only '01'.
PK Algorithm	2	Public Key Algorithm, hexadecimal string '01': RSA digital signature. Now, PIN pad accepts only '01'.
Exponent Len	2	Public Key Exponent size, hexadecimal string. For example:

		'03' = 3 bytes
Exponent	Var.	Public Key Exponent, hexadecimal '03': 3 '010001': 2^{16}+1
Hash	40	Hash checksum, hexadecimal
<ETX>	1	<03>
[LRC]	1	Checksum

2nd Packet (Load RSA public key):

Field	Length	Value and description
<STX>	1	<02>
T53	3	Message ID
Pkt No.	1	Decimal. Packet sequence number (1 ~ 9)
Total Pkts	1	Decimal. Total packets (1~9)(ex. 8).
Modulus Len	2	Public Key Exponent size, hexadecimal string. For example: '80' = 128 bytes = 1024 bits
Modulus	Var. (Max. 256-byte per transmit)	Public Key Modulus, presented in hexadecimal, data length = 2*[Modulus Len]
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
1st Message T53	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	1st CA Public Key Setup Response Message T54
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
2nd Message T53	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)

	←	2nd CA Public Key Setup Response Message T54
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

Command Example:

1. Visa CA 51, RID: A000000003
   <02>T5313A0000000035100A901010103B9D248075A3F23B522FE45573E04374DC4995D71<03><LRC>
   <02>T532390DB5FA29D1FDA8C1634B04DCCFF148ABEE63C772035C79851D3512107586E02A917F7C7E885E7C4A7D529710A145334CE67DC412CB1597B77AA2543B98D19CF2CB80C522BDBEA0F1B113FA2C86216C8C610A2D58F29CF3355CEB1BD3EF410D1EDD1F7AE0F16897979DE28C6EF293E0A19282BD1D793F1331523FC71A228800468<03><LRC>
   <02>T5333C01A3653D14C6B4851A5C029478E757F<03><LRC>
2. Paypass CA EF, RID: A000000004
   <02>T5313A000000004EF01110101010321766EBB0EE122AFB65D7845B73DB46BAB65427A<03><LRC>
   <02>T5323F8A191CB87473F29349B5D60A88B3EAEE0973AA6F1A082F358D849FDDFF9C091F899EDA9792CAF09EF28F5D22404B88A2293EEBBC1949C43BEA4D60CFD879A1539544E09E0F09F60F065B2BF2A13ECC705F3D468B9D33AE77AD9D3F19CA40F23DCF5EB7C04DC8F69EBA565B1EBCB4686CD274785530FF6F6E9EE43AA43FDB02CE0<03><LRC>
   <02>T53330DAEC15C7B8FD6A9B394BABA419D3F6DC85E16569BE8E76989688EFEA2DF22FF7D35C043338DEAA982A02B866DE5328519EBBCD6F03CDD686673847F84DB651AB86C28CF1462562C577B853564A290C8556D818531268D25CC98A4CC6A0BDFFFDA2DCCA3A94C998559E307FDDF915006D9A987B07DDAEB3B<03><LRC>

➢ Message T54 Certificate Authority Public Key Setup Response

Format: **<STX>T54[Resp][Reason][Err Msg]<ETX>[LRC]**

Message length: Variable.

Usage: The response message of command T53.

Message element:

1st, 2nd Packet:

Field	Length	Value and description
<STX>	1	<02>
T54	3	Message ID
Sequence	1	1 / 2 (first/second part of RSA public key)
[Resp]	1	'0': Ok, '1': Fail
[Reason]	1	Option if [Resp] is '1', '1': Fatal Error '2': Format Error '3': Authentication Fail
[Err Message]	8	Optional, if Reason = '1', Hex String
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: Please refer to message T53.

➢ Message T55 EMV Application Configuration Setup

Format: **<STX>T55[PktNo.][Total Pkts]<SUB>[TranType]<SUB>[KID]<SUB>[AID]<SUB>[DO]<ETX>[LRC]**

<STX>T55[Pkt No.][Total Pkts]<SUB>[DO]<ETX>[LRC]

Message length: Variable.

Usage: Host can use this command to send the **EMV application configuration data** to PIN pad, this command can be sent many times but each command is only for one application. PIN pad will save those data inside and use those data when do the transaction. PIN pad will response the message T56 (EMV Application Configuration Setup Response) to host. The data installed into PIN pad via this message, PIN pad will save it in internal storage structure with a name same as in [AID] field.

Message element:

1st Message:

Field	Length	Value and description
<STX>	1	<02>
T55	3	Message ID
Pkt No.	1	Decimal. Packet sequence number (1 ~ 9)
Total Pkts	1	Decimal. Total packets (1~9)(ex. 8).
<SUB>	1	Optional, if Pkt No is 1 <1A>
[TranType]	2	Optional, if Pkt No is 1. Hexadecimal, Transaction Type.
<SUB>	1	Optional, if Pkt No is 1 <1A>
[KID]	2	Optional, if Pkt No is 1. Hexadecimal, kernel ID.
<SUB>	1	Optional, if Pkt No is 1 <1A>
AID	10~32	Optional, if Pkt No is 1. EMV Application ID, refer to EMV 4.1
<SUB>	1	Optional, only if [DO] is existed
DO	Var. (Max. 220-byte per transmit)	Data Object, format as below
<ETX>	1	<03>
[LRC]	1	Checksum

Rest of Message (If there are 2 more messages):

Field	Length	Value and description
<STX>	1	<02>
T55	3	Message ID

<SUB>	1	Optional, only if [DO] is existed
DO	Var. (Max. 256-byte per transmit)	Data Object, format as below
<ETX>	1	<03>
[LRC]	1	Checksum

Data Format: (Please also refer to EMV 4.1 BOOK3, section 4.3)

Format	Description
1	a - Alphabetic data (a ~z, A~Z)
2	b - unsigned binary numbers or bit combinations
3	an - Alphanumeric data (a ~z, A~Z, 0~9)
4	ans - Alphanumeric Special data (Characters defined in ISO8859)
5	cn - Compressed numeric data (0~9, left justify, pad hexadecimal 'F's. Ex. 12 34 56 12 3F FF)
6	n - Numeric data (0~9, right justify, pad leading hexadecimal zeroes. Ex. 00 00 00 01 23 45)
7	var - Variable data (Any bit combination)

Data Object:

Each <DO> shall include three data field: [Tag#] || [Format] || [Value], and each field shall delimit with a <FS>. Each data object is delimited by a <SUB> to construct multiple <DO>. The [tag#] defined in EMV 4.1 Book3 Annex A and specific [tag#] defined at Appendix D of this manual have the pre-defined data format and length range, those [tag#] must follow up the rule, otherwise the PIN Pad will reject this data setup.

Note. If the data format is '2' (binary), '5'(compressed numeric), '6'(numeric) or '7'(variable data), the [value] correspond to these format can not be sent with these format directly in message T55. It shall be converted to hexadecimal string and with pad char 'F' in the last one for format 'cn' if this tag# has odd chars or with pad char '0' in the first one for format 'b' or 'n' if this tag# has odd chars.

Example:

Default TDOL: 97 07 9f 02 06 95 05 9b 02 (binary)

Threshold Value for Biased Random Selection: 00 00 00 00 40 00(numerical)

Max. Target percentage to be used for Biased Random selection: 100 (decimal) / 0x46 (binary)

<STX>T5511<SUB>00<SUB>03<SUB>A00000031010<SUB>97<FS>2<FS>97079f020695059b02<SUB>40000004<FS>6<FS>000000004000<SUB>40000006<FS>2<FS>46<ETX>[LRC]

PIN pad saves these data and give an ID as "A00000031010" to this group of data.

Special Tag# defined by PIN pad: (Not EMV defined)

Name	Description	Format	Tag	Length
Application Selection Indicator	See below	n	40000001	1
Threshold Value for Biased Random Selection	See below	n	40000004	6
Target Percentage to be used	See below	b	40000005	1

for Biased Random Selection				
Maximum Target Percentage to be used for Biased Random Selection	See below	b	40000006	1
Terminal Action Code - Default	See below	b	40000007	5
Terminal Action Code - Denial	See below	b	40000008	5
Terminal Action Code - Online	See below	b	40000009	5
Data Tags required in Online message (ARQC)	See below	b	4000000A	var.
Data tags required in reversal message	See below	b	4000000D	var.
Data tags for batch data capture	See below	b	40000010	var.
ARC Approve	See below	b	4000001A	var.
ARC Decline	See below	b	4000001B	var.
ARC Referral	See below	b	4000001C	var.

Message flow:

HOST	Direction	PIN Pad
1st Message T55	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	1st Message T56
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
2nd Message T55	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Application Select Response. 2nd Message T56
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	
…	…	…
Last one Message T55	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Application Select Response. Last one Message T56
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

PIN pad will check if terminal downloads minimum set of EMV Application-related information into PIN pad. The download process will be failed if there is not enough data in this message. Please refer to appendix E for minimum set of EMV Application -related data

➢ Message T56 EMV Application Configuration Setup Response

Format: **<STX>T56[Resp][Reason][Err Msg]<ETX>[LRC]**

Message length: Variable.

Usage: The response message of command T55.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T56	3	Message ID
[Resp]	1	'0': Ok, '1': Fail
[Reason]	1	Option if [Resp] is '1', '1': Fatal Error '2': Format Error '3': Invalid Data Object format. '4': Invalid Tag value
[Err Message]	8	Optional, if Reason = '1'Hex String
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: Please refer to message T55.

➢ Message T61 Start Transaction

Format: **<STX>T61<SUB>[AmtAuth]<SUB>[AmtOther]<SUB>[CurExponent][CurCode]<SUB>[TranType]<SUB>[TranInfo]<SUB>[Account Type]<SUB>[Force Online]<SUB>** [**Encrypted Session key]<ETX>[LRC]**

Message length: Variable.

Usage: After receive this message command T61, PIN Pad will perform an completed EMV transaction flow (the flow will cover 'Initiate Application' through 'Completion', see EMV 4.1, book 3, chap 8.2, figure 6 - transaction flow example) based on the selected EMV application that has corrected application name on the T12 (Application Select response), PIN pad will also prompt user to do the appropriated entry when presented, like confirm or enter PIN code. PIN pad will send the message T62 (Start Transaction Response) to host.

If the IC card can't be read (which has known from T12 response code with "T1214" when in message T11 processing and terminal changed to do magnetic stripe card processing; please refer the Notes in message T12), then after receive this message T61, PIN pad will not process the EMV transaction flow but in place of just storing the information provided from this message and return the message T62 with result code "A1" to ask terminal to go line to get the authorization then send back a message T71 to finish this transaction. Terminal could have extra operation on magnetic stripe card transaction (like issue message Q1 to read track data and issue message 70 for PIN entry, and so on.) before issue this message T61.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T61	3	Message ID
<SUB>	1	<1A>
[**AmtAuth]**	12	Hexadecimal, Amount Authrozied, will be stored at tag '0x9f02'
<SUB>	1	<1A>
[**AmtOther]**	12	Hexadecimal, Amount Other, will be stored at tag '0x9f03'
<SUB>	1	<1A>
[**CurExponent**]	1	Hexadecimal, stored at tag '0x5f36'
[CurCode]	3	Hexadecimal, stored at tag '0x5f2a'. For example, USD$ = 0x840
<SUB>	1	<1A>

[**TranType]**	2	Hexadecimal, Transaction Type, will be stored at tag '0x9c'
<SUB>	1	<1A>
[**TranInfo]**	2	Transaction Info, will be stored at tag '0x60000001'
<SUB>	1	<1A>
[**Account Type]**	2	Account Type, stored at tag '0x5f57'
<SUB>	1	<1A>
[**Force Online]**	1	1: Force Online, only valid if this terminal has the capability of support online authorization.
<SUB>	1	<1A>, optional
[**Encrypted Session key]**	16 or 32	Optional, DES or TDES session key that used to encrypt PIN entry when CVM ask online PIN verify. If the CVM ask online PIN verify but this session key does not input, PIN pad will ignore the PIN entry request and indicate that no PIN is entered in the TVR register.
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message T61	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Start Transaction Response. Message T62
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

Note: If [Encrypted Session key] is input, the tag# (for example, 0xDF02) to store this encrypted session key shall be defined at tag#50000005, please refer T01 command's notes how to setup this tag#DF02.

➢ Message T62 Start Transaction Response

Format: **<STX>T62[Status][Reason][Err Message][Result]<SUB>**
<Advice Need> <Reversal Need> <Financial Need><ETX>[LRC]

Message length: Variable.

Usage: The message contains the transaction result on the smart card to be sent to terminal.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T62	3	Message ID
Status	1	0:OK; 1:Fail
Reason	1	Optional. (If Status = 1) 1:Fatal Error 2:Command Format Error
[Err Message]	8	Optional, if Reason = '1'Hex String
[Result]	2	Optional. (If Status = 0) 'Y1': Offline Approved, 'Z1': Offline Declined 'Y3': Unable to go online, Offline Approved 'Z3': Unable to go online, Offline Decline. 'Y4': Online Approved 'Z4': Online Decline 'A1': Online Authorize Request, 'A4': Application reselection.
<Advice Need>	1	Optional, only valid if [Result] is "A1". RFU.
<Reversal Need>	1	0: Terminal does not need to send a reversal to host for this transaction 1: Terminal should send a reversal to host for this transaction.
<Financial Need>	1	Optional, only valid if [Result] is "A1". RFU
<ETX>	1	<03>
[LRC]	1	Checksum

Note: If the [Result] is 'A1', then terminal should send online authorization request to issuer host; and after done, terminal shall send message T71 to PIN pad to continue transaction. (See EMV 4.1, book 3, chap 9, figure 7, 8)

Note: If the transaction is switched to MSR processing, T62 will always return "A1" in [Result] field.

Note: if the previous selected application on IC card can't do the transaction (for example, this application has blocked) but has another application ID within this IC card, PIN pad will response 'A4' to let terminal know and terminal can issue message T13 to select another application ID and issue message T61 to re-start the transaction. Please refer the paragraph of "Ref. 5 Packet command flow for first EMV application is blocked" in the section of "Overall EMV Level 2 transaction flow reference".

Message flow: Please refer to message T61.

➢ Message T63 Get Transaction Result's Data

Format: **<STX>T63[DOL]<ETX>[LRC]**

Message length: Var.

Usage: PIN Pad will retrieve the data that list on the DOL after EMV transaction done. PIN Pad will send the message T64 (Get Transaction Result's Data Response) to host.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T63	3	Message ID
DOL	Var	Data Object List, each object is expressed by tag number, and <SUB> is used to delimit each object. For example, 9F12<SUB>9A<SUB>9F02<SUB>….
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message T63	→	
		<ACK> (Good LRC) <NAK> (Bad LRC) <EOT> after 3 NAKs
	←	Get Transaction Result's Data Response. Message T64
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message T64 Get Transaction Result's Data Response

Format: **<STX>T64[DO]<ETX>[LRC]**

Message length: Var.

Usage: The message contains the transaction result's data to be sent to host.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T64	3	Message ID
[DO]	Var.	Data Object, each data object is expressed by TLV format with an <FS> delimit in each field, and <SUB> is used to delimit each object. For example: 9F12<FS>0F<FS>CREDITO DE VISA<SUB>9A<FS>06<FS>050806<SUB>…….
<ETX>	1	<03>
[LRC]	1	Checksum

Data Object:

PIN pad will return series of data object that list on the [DOL] field in the message T64 with TLV (tag || length|| value) format as below:

[EMV Tag Number (2 ~ 8 byte)] <FS> [Length (2byte)] <FS> [Value] <SUB>
[EMV Tag Number (2 ~ 8 byte)] <FS> [Length (2byte)] <FS> [Value] <SUB>
. . . . . .
[EMV Tag Number (2 ~ 8 byte)] <FS> [Length (2byte)] <FS> [Value].

Note: When PIN Pad response these data object, it will convert these value from binary value to hex decimal string if the data object format is "b" or "cn" or "n".

Message flow: Please refer to message T63.

➢ Message T65 Get Online authorization Data

Format: **<STX>T65<ETX>[LRC]**

Message length: 6.

Usage: Issue this message to get data (EMV) for online authorization.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T65	3	Message ID
<ETX>	1	<03>
[LRC]	1	Checksum

While the EMV transaction must be authorized online, terminal shall issue message T65 to get the necessary data for online authorization from PIN pad. And PIN pad will wait terminal to send one message T71 to tell the PIN Pad the go online authorization result

User shall load a data object's tag list in tag# 4000000A for online authorization data into PIN Pad when do the application configuration setup (message T55), PIN pad will search the corresponding values according to this tag list and return it at message T66 after receive message T65 from terminal. Please refer to Appendix D for more details.

Message flow:

HOST	Direction	PIN Pad
Message T65	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message T66
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message T66 Response of Get Online authorization Data message

Format: **<STX>T66 [online authorization data]<ETX>[LRC]**

Message length: var.

Usage: Return online authorization data to terminal.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T28	3	Message ID
[Online authorization data]	Var.	Hex string. Optional, if this transaction needs to be authorized online.
<ETX>	1	<03>
[LRC]	1	Checksum

Online authorization data:

According to the tag list of online authorization data (refer to appendix D) which defined at tag# 4000000A, PIN pad will search the corresponding values according to this tag list and return series of data object with TLV (tag || length|| value) format as below:

[EMV Tag Number (2 ~ 8 byte)] || [Length (2byte)] || [Value] <SUB>
[EMV Tag Number (2 ~ 8 byte)] || [Length (2byte)] || [Value] <SUB>
. . . . . .
[EMV Tag Number (2 ~ 8 byte)] || [Length (2byte)] || [Value].

PIN pad can return at maximum 256 bytes of data as one record, the right part of data will be ignored if the length record is greater than 256 byte.

When PIN Pad response these data object, it will convert these value from binary value to hex decimal string if the data object format is "b" or "cn" or "n". For example, the data of one record is 0x9F02 || 06 || 000000001100 || 5A || 10 || 11223344556677889900AABBCCDDEEFF, terminal will see "9F0206000000001100<SUB>5A1011223344556677889900AABBCCDDEEFF".

Message flow: Please refer to message T65

➢ Message T71 Send Online Authorized Code

Format: **<STX>T71[Online Res]<SUB>[ARC][IAD]<ETX>[LRC]**

Message length: Fixed 7 or 16 bytes.

Usage: After receive this message T71, PIN Pad will continue to perform the EMV transaction flow if the previous T62 response result is 'A1' (online authorized request, see EMV 4.1, book 3, chap 8.2, figure 6 – transaction flow example). PIN pad will response the message T62 (Start Transaction Response) to this T71 to host.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T71	3	Message ID
Online Res	1	'0': Unable to go online
<ETX>	1	<03>
[LRC]	1	Checksum

OR

Field	Length	Value and description
<STX>	1	<02>
T71	3	Message ID
Online Res	1	'1': Get Online Authorize Response,
<SUB>	1	<1A>
ARC	2	Authorisation Response code, ASCII (0~9, A~Z). Please see **Note 1**.
<SUB>	1	<1A>
[**IAD]**	16~32	Optional, Issuer Authentication Data, Hex string. if there is IAD response from remote host , terminal shall send this to PIN pad.
<ETX>	1	<03>
[LRC]	1	Checksum

Note 1:

These acceptable ARC code shall be matched with the pre-defined code at tag# 4000001A (ARC approval) and tag#4000001B (ARC decline) and tag# 4000001C (ARC referral) (please refer Appendix D) where these tag value can be setup by message T05. Please note the data format in tag# 4000001A to tag# 4000001C are binary but here [ARC] format is ASCII. So there

shall have a convert. For example, if the ARC approval code "Y0", then the binary value in tag# 4000001A shall be 0x5930.

OR (If the transaction is changed to Magnetic stripe card processing)

Field	Length	Value and description
<STX>	1	<02>
T71	3	Message ID
Online Res	1	'3': MSR Online Approve '4': MSR Online Decline,
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message T71	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Start Transaction Response. Message T62
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message T73 Send Issuer Script Command

Format: **<STX>T73[IS]<ETX>[LRC]**

Message length: Var.

Usage: PIN Pad performs the Issuer script processing as in EMV transaction flow after received this command from the host those are the response message when doing online authorization. This command can be send many times if too many script commands need to be processed, but the last one should be send before T71 command.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T73	3	Message ID
IS	Var.	Issuer Script, format as follow. Hex string.
<ETX>	1	<03>
[LRC]	1	Checksum

Issuer Script Format: (see EMV 4.1, book 3, chap 10.10, Figure 10)

T	L	T	L	Script ID	Commands
'71' or '72'	Including Script ID, tags, lengths	'9F18'	'04'	Identifier (4 bytes)	Issuer Script Command Format (see below)

Issuer Script Command Format: (see EMV 4.1, book 3, chap 10.10, Figure 11)

T1	L1	V1	T2	L2	V2	T3	L3	V3	Tx	Lx	Vx
'86'	L(V1)	Cmd	'86'	L(V2)	Cmd	'86'	L(V3)	Cmd	'86'	L(Vx)	cmd

➢ Message T74 Send Issuer Script Command Response

Format: **<STX>T74[Status][Reason][Err Message]<ETX>[LRC]**

Message length: Variable.

Usage: The message response the command T73.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T74	3	Message ID
Status	1	0:OK ; 1:Fail
Reason	1	Optional. (if Status = 1) 1:Fatal Error 2:Command Format Error
[Err Message]	8	Optional, if Reason = '1', Hex String
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: Please refer to message T73.

➢ Message T75 Revocation List Setup

Format: **<STX>T75<SUB>[RID][SN][PKI]<ETX>[LRC]**

Message length: Fixed 25 bytes

Usage: Host can use this command to send **revocation key information** to PIN pad, this command can be sent many times. PIN pad will save those information inside and check those information when do the transaction. PIN pad will send the message T76 (Revocation List Setup Response) to host.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T75	3	Message ID
<SUB>	1	Separator
RID	10	The RID for revocated public key. Present in hexstring.
SN	6	The serial number of the revocated public key. Present in hexstring.
PKI	2	The public key index of the revocated key. Present in hexstring.
<ETX>	1	<03>
[LRC]	1	Checksum

Example:

Revocation list information

RID: A0 00 00 00 03

SN: 00 00 01

PKI: 51

<STX>T75<SUB> A000000003000001 51<ETX>[LRC]

Message flow:

HOST	Direction	PIN Pad
Message T75	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Message T76
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

➢ Message T76 Revocation List Setup Response

Format: **<STX>T76[Res][Reason][Err Msg]<ETX>[LRC]**

Message length: Variable.

Usage: The response message of command T75.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T76	3	Message ID
[Res]	1	'0': Ok, '1': Fail
[Reason]	1	<Optional, if Res = '1'> '1': Fatal error '2': Invalid Data format '3': Revocation list is full '4': The added info exists
[Err Message]	8	Optional, if Reason = '1', Hex decimal string
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: Please refer to message T75.

➢ Message T77 Exception List Setup

Format: **<STX>T77<SUB>[PAN Len][PAN]<ETX>[LRC]**

Message length: Variable.

Usage: Host can use this command to send the **exception pan** to PIN pad. PIN pad will save the information inside and check them when do the transaction. Once the transaction pan is on the exception list, the transaction will be terminated. PIN pad will send the message T78 (Exception List Setup Response) to host.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T77	3	Message ID
<SUB>	1	Separator
PAN Len	2	The length of PAN digits. Present in hexstring.
PAN	var	Numeric string
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow:

HOST	Direction	PIN Pad
Message T77	→	
	←	<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)
	←	Exception List Setup Response Message T78
<ACK> (Good LRC) <NAK> (Bad LRC) (<EOT> after 3 NAKs)	→	

Command Example:

PAN: 47 61 73 90 01 01 00 10

<STX>T77<SUB>1047617390010100 10<ETX><LRC>

➢ Message T78 Exception List Setup Response

Format: **<STX>T78[Resp][Reason][Err Msg]<ETX>[LRC]**

Message length: Variable.

Usage: The response message of command T77.

Message element:

Field	Length	Value and description
<STX>	1	<02>
T78	3	Message ID
[Resp]	1	'0': Ok, '1': Fail
[Reason]	1	Option if [Resp] is '1', '1': Fatal Error '2': Format Error '3': Exception List is full '4': PAN exists
[Err Message]	8	Optional, if Reason = '1', Hex String
<ETX>	1	<03>
[LRC]	1	Checksum

Message flow: Please refer to message T77.

Appendix A Key management

This PIN pad is designed to encrypt Personal Identification Numbers (PIN) as they are entered from the keypad, store the encrypted data in its memory and then transmit it to the HOST as requested.

Because the data-encryption standard (DES) and RSA algorithm are in the public domain, the security of the functions of the PP190 depend on the security of the key that is used in processing the algorithm. Therefore, after you load cryptographic keys into the PP190, the keys cannot be read. They are placed AES encrypted by a randomly generated AES key, resident in a battery-powered register. Once security is breached, the AES key will be erased, and all encrypted DES master keys become unusable.

You can design a secure method for handling your keys when you are isolated from the PIN keypad, using the provisions for loading the keys. Randomly generate your keys, and store and distribute your keys in a secure, controlled manner that you can audit.

An independent Tamper Resistant Security Module (TRSM) is required for secure key injection process. UIC provides a software key injection utility (UICKIT for Windows) as demo for safely and manageable key injection procedure. Please refer to UICKIT programming manual for detail.

PP190 supports following management schemes:

1. **Master/session key (MK/SK):**

   PP190 can store 16 master keys, key ID 0 to 9 are for MK/SK PIN entry (They can be PIN master key or PIN key), key ID B to E is for generate or verify MAC, depend on its usage and mode settings. (They can be MAC master key or MAC key), F for master key transportation (It can be only key loading key). These master keys cannot be used in other ways (e.g. designer cannot use PIN entry keys for MAC generation.) Session keys (working keys) are transmitted from the HOST, encrypted by the master key for every transaction. Customer's PIN is encrypted using the decrypted working key or by selected master key (If the selected one is with key usage "P0"). Thus the master keys must exist before any transaction can take place. PP190 can use 8 bytes DES key format or 16 / 24 bytes Triple DES key format, the working key can also be 16 bytes TDES key. When doing transactions using MK/SK scheme, firmware of PP190 applies a DES calculation count limiter (***only 10 transactions are allowed in 5 minutes period***.) to comply with PCI PED security requirement (average one transaction per 30 seconds.) This constraint is set to deter attacker using huge saturation DES transaction to detect master key in PP190.

2. **ANSI TR31 Specified Key Bundle**

   Key Attribute:

   When loading master keys into PP190 in encrypted format, the key data is wrapped by a key bundle specified in ANSI TR-31 2010 specification.

   1. Key usage: indicate what usage of a key.
      "K0", indicates that this key is used for key transportation;
      "P0", indicates that this key is used for PIN entry directly;

"D0", indicates that this key is used for data transportation;

"M1" indicates that this key is used for MAC calculation directly by ISO 9797-1 method 1.

"M3" indicates that this key is used for MAC calculation directly by ISO 9797-1 method 3.

"B1", indicates that this key is used for DUKPT initial key (ANSI TR-31 2010).

2. Algorithm: indicate what algorithm will be used with the key.

   "D" : DES algorithm

   "T": TDES algorithm (double or triple length key)

   "A": AES algorithm (RFU)

3. Mode: indicate what cryptograph operation will be applied with the key

   "D": Decryption

   "E": Encryption

   "G": MAC generation

   "V": MAC verification

   "X": Key derivation (DUKPT)

4. Version (RFU): It should be 00.
5. Export (RFU): It should be "N".

If the key usage is "K0", the length of key must be 16 bytes or 24 bytes (algorithm must be "T").

Key Architecture and limitation

Group	Key ID	Length	Usage	Algorithm	Mode	Encrypt under
PIN	0~9	8~24	P0	D or T	E	KLK
		16~24	K0	T	D	KLK
Data	A	RFU	RFU	RFU	RFU	RFU
MAC	B~E	8	M1	D	G	KLK
		16	M3	T	G / V	KLK
		16~24	K0	T	D	KLK
KLK	F	16~24	K0	T	D	KLK

Key attribute and limitation for IPEK

IPEK	Length	Usage	Algorithm	Mode	Encrypt under
IPEK0 or IPEK1	8 or 16	B1	D or T	X	KLK

1. All the keys injected in cipher-text must be encrypted by key derived from KLK and calculate a MAC value by key derived from KLK
2. For key with usage "K0", the length must be 16 bytes or 24 bytes.
3. For MAC key with usage "M1", the length of key must be 8 bytes (DES-MAC).
4. For MAC key with usage "M3", the length of key must be 16 bytes (TDES-MAC).

5. Duplicate key injection is not allowed. (except IPEK0 and IPEK1)
6. The length of injected key in cipher-text should be equal to or less then the length of KLK.

Key Injection

To inject clear-text key (Key ID: 0~9, A~F) into PP190, the default attributes will be as following, Key usage = "K0", Algorithm = "T", Mode = "D", Version = "00" and export = "N"

To inject cipher-text key into PP190, user has to assign these attributes.
For Key 0 ~ 9,
Key usage should be "K0" or "P0", algorithm should be "T" or "D", mode should be "D" (If for "K0" usage) or "E" (If for "D0" usage).
For Key B ~ E,
Key usage should be "K0", "M1" or "M3", algorithm should be "T" (If for "K0" or "M3" usage) or "D" (If for "M1" usage), mode should be "D" (If for "K0" usage) or "G" (If for "M1" or "M3" usage).
For Key F,
Key usage should be "K0", algorithm should be "T".
For IPEK 0~1
Key usage could be any 2 bytes data, algorithm should be "D" or "T", mode should be "E".

Inject key in cipher-text (TR31 format)

For Key 0~9, A~F
<SI>02[Key ID][KBH][Encrypted KEY][MAC]<SO>, where [KBH] + [Encrypted KEY] + [MAC] is TR31 block.
For IPEK0
<STX>90[KBH][Optional KBH][DUKPT0][MAC]<ETX>, where [KBH and Optional KBH] + [IPEK0] + [MAC] is TR31 block.
For IPEK1
<STX>94[KBH][Optional KBH][DUKPT1][MAC]<ETX>, where [KBH and Optional KBH] + [IPEK1] + [MAC] is TR31 block.

KBH (Key Block Header – ASCII format):

A[4byte – length of TR31 block][2byte - Usage][1byte - Algorithm] [1byte - Mode][2byte - Version][1byte - Export][2byte - option][2byte - rfu]

Optional KBH (For DUKPT use only):

[2byte: Optional Block ID, fixed as "KS"][2byte: Optional Block Length, fixed as "18"][20byte: Optional Block Data, put key serial number (refer to ANSI X9.24 SMID) in this field]

Encrypted KEY Block:

1. Derive Key1 by XOR KLK with 0x45

2. Generate new key block, [2byte number indicate the key in bits][key][random padding]
3. Encrypt the new key block by Key1 with first 8byte of KBH as IV in CBC mode and get encrypted key block.

MAC:

1. Derive Key2 by XOR KLK with 0x4D
2. Concatenate KBH with Optional KBH (if any) and encrypted key block and get new key block 2.
3. Encrypt the new key block 2 by Key2 without IV in CBC mode and get the last 8 bytes.
4. Get the first 4 bytes of result as MAC value

Example 1:

KLK: 0123456789ABCDEFFEDCBA9876543210

New MK (Key ID = 1, key usage = "K0"): 89E88CF7931444F334BD7547FC3F380C

Generate KBH:

KBH = A | 0072 | K0 | T | D | 00 | N | 0000

Generate Encrypted KEY Block:

1. Derive K1 for encryption: 44660022CCEE88AABB99FFDD33117755
   K2 for MAC value: 4C6E082AC4E680A2B391F7D53B197F5D
2. Key length = 16 bytes (128 bits = 0x80), 6 byte random padding = 720DF563BB07,
   New key block = 008089E88CF7931444F334BD7547FC3F380C720DF563BB07.
3. IV = first 8 byte of KBH ("A0072K0T") = 41303037324B3054, apply TDES-CBC on new key block by K1 with IV and get encrypted key block = D078A2657E5B57972CD3 D308E05E1FE519B316309AA6354A

MAC:

1. Concatenate KBH and encrypted key block = 41303037324B30544430304E303 03030D078A2657E5B57972CD3 D308E05E1FE519B316309AA6354A
2. Apply TDES-CBC on new key block 2 by K2 without IV and get last 8 byte result = 668071B5B73CC024
3. MAC value = 668071B5
4. The final TR31 block = A0072K0TD00N0000 - D078A2657E5B57972CD3D308E05E 1FE519B316309AA6354A - 668071B5

Send message 02 to load this new key in cipher-text:

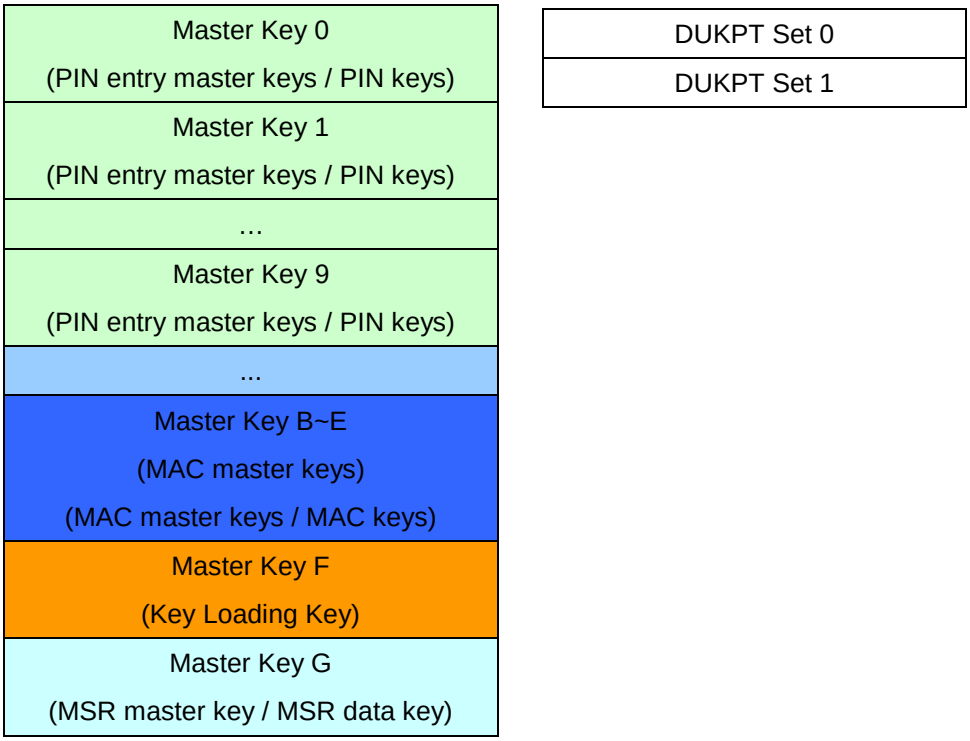
<SI>021A0072K0TD00N0000D078A2657E5B57972CD3D308E05E1FE519B316309AA6354A66 8071B5<SO>[LRC]

3. **Derived Unique Key Per Transaction (DUKPT):**

PP190 Implements ANSI X9.24-2002 and ANSI TR31 key management scheme for DUKPT.

Authorized personnel can load 8bytes/16bytes Initial keys (also known as IPEK) and Key serial number (also known as 'Security Management Information Data-SMID' in ANSI X9.24). Every time when PP190 finished a PIN entry transaction, a new key will be calculated. Every single transaction will use different key in order to prevent attacker to detect specific keys in any transactions.

The symmetric keys (MKSK/DUKPT) structure is shown as following:



4. **RSA public key:**

   PP190 supports RSA encryption when processing EMV level 2 offline transactions with smart cards.

5. **Second DUKPT Key Set of PP190:**

   PP190 provides 2nd key set of DUKPT operation for customer's scalability. Message 90 is used to initialize first key set, with message 94 to initialize second key set. User must issue message 96 to select preferred key set before doing DUKPT transactions. These two key set are independent with each other, and both accepts double length key for TDES capability. Ether key set reaches 1million transaction limit will lock down PP190.

   In real operation, authorized user can load a 8byte DES initial key to key set 1 and a 16byte TDES initial key to key set 2 before PIN pad is deployed. At first use can transact with key set 1. When backbone system ready, user can use message 96 to select key set 2 to switch to TDES transaction immediately.

6. **Triple DES (TDES) capability:**

TDES means that DES algorithm is applied three times on the data to be encrypted before it is sent over the line. PP190 can detect key length when loading keys (message 02 for Master/Session key and message 90/94 for DUKPT) and doing transactions (Master/Session key message 70, Z60, Z62). If a 32 or 48 characters (16 or 24 byte) key is used, PP190 will treat all transactions using this key as TDES enabled, else PP190 use DES operation.

TDES algorithm needs a 16-byte key, which separated as L-key (leftmost 8 bytes) and R-key (rightmost 8 bytes). PP190 defaults EDE order for TDES encrypting operation as follows:

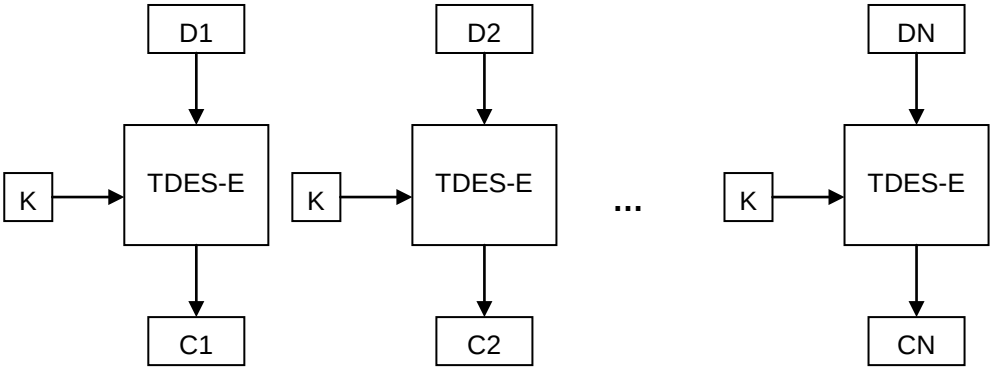
Clear Text	→	DES encryption via L-key	→	DES decryption via R-key	→	DES encryption via L-key	→	Ciphered Text

EDE order of TDES operation – 16 byte key. (Data decrypting process is the reverse of encrypting process.)

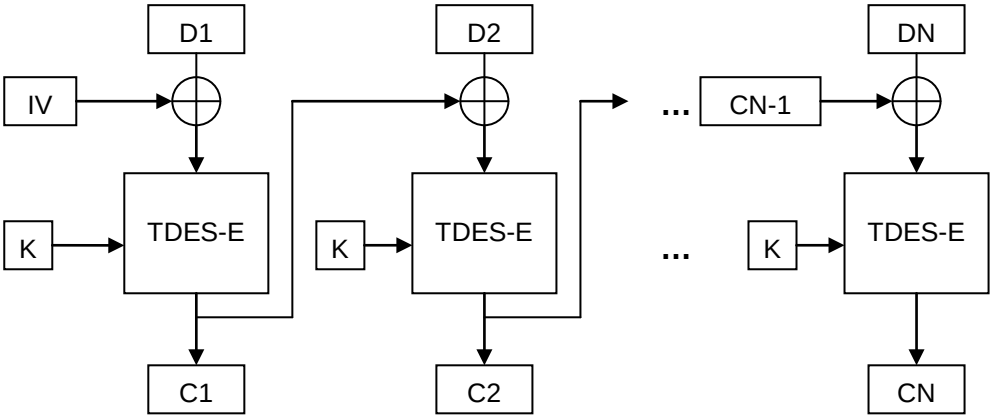
Clear Text	→	DES encryption via L-key	→	DES decryption via Middle-key	→	DES encryption via R-key	→	Ciphered Text

EDE order of TDES operation – 24 byte key. (Data decrypting process is the reverse of encrypting process.)

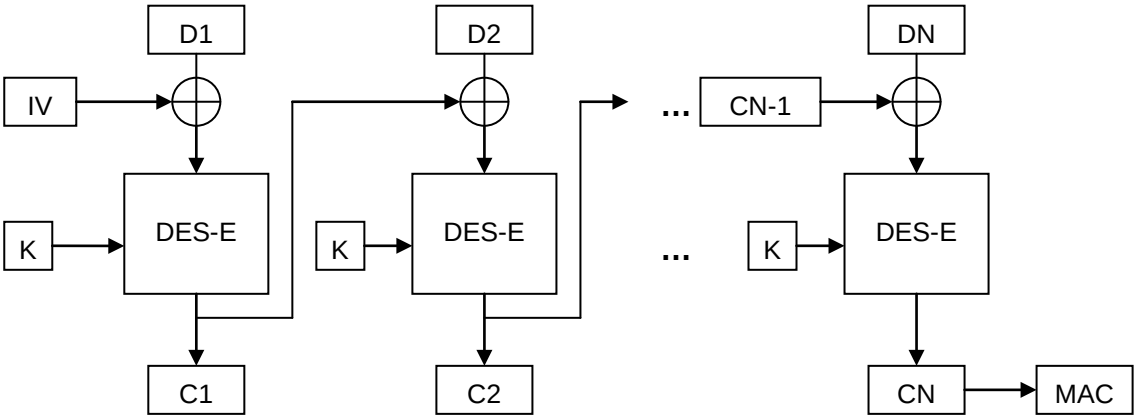
7. TDES – ECB Encryption:



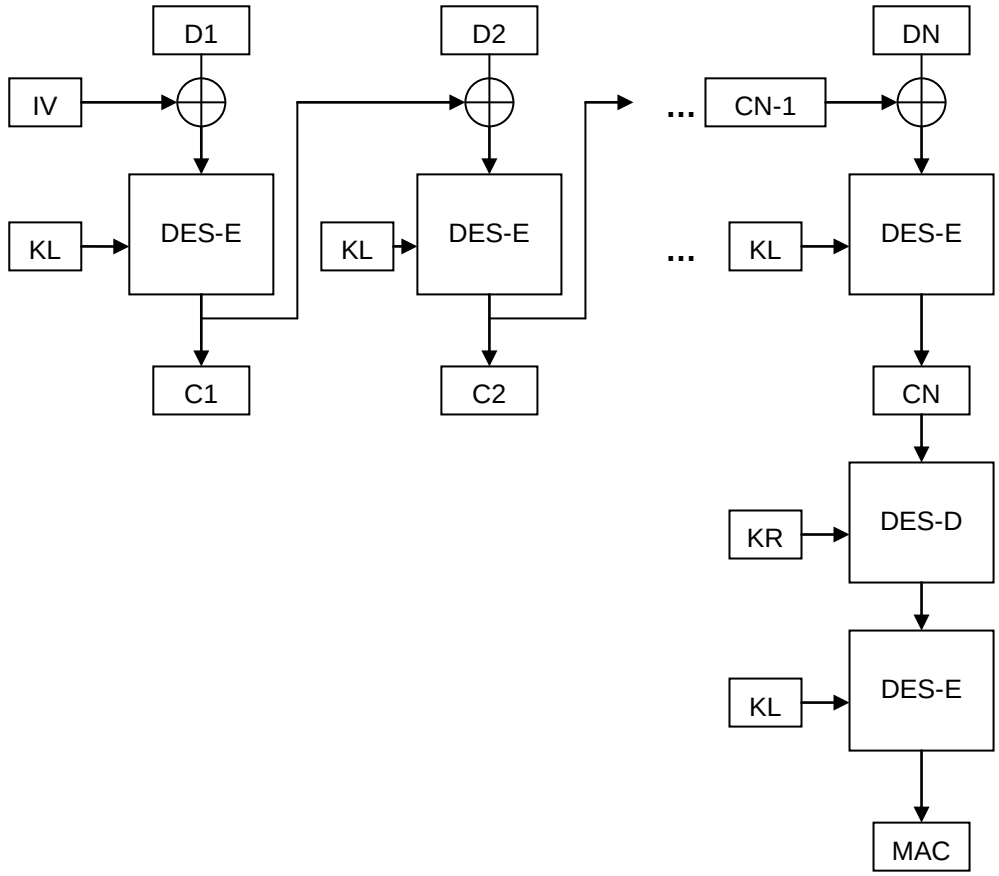
8. **TDES – CBC Encryption:**



9. **DES-MAC (ISO 9797-1 method 1)**



10. **TDES – MAC (ISO 9797-1 method 3)**



Appendix B PIN Block Format

➢ ANSI x9.8 format (MK/SK, DUKPT, and Offline clear text PIN entry)

PP190 outputs ANSI X9.8 PIN blocks. Its format as follows:

PIN Block Format

Bit	0-3	4-7	8-11	12-15	16-19	20-23	24-27	28-31	32-35	36-39	40-43	44-47	48-51	52-55	56-59	60-63
Value	C	N	P	P	P	P	P/F	P/F	P/F	P/F	P/F	P/F	P/F	P/F	F	F

Bit field explanation:

C - Control field (Format number). Value = 0000 (Does not support Format 1 or Format 3)

N - PIN length entered field. Value = 0100 to 1100 (4-12) (0x4 – 0xC)

P - PIN digit. Value = 0000 to 1001 (0-9)

F - Fill digit. Value = 1111 (F)

P/F - Pin digit or fill digit, as determined by PIN Length N. PIN Length is 4 to 12

Primary Account Number Block (PANB) Format

Bit	0-3	4-7	8-11	12-15	16-19	20-23	24-27	28-31	32-35	36-39	40-43	44-47	48-51	52-55	56-59	60-63
Value	0	0	0	0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12

Bit field explanation:

A - The twelve rightmost digits of the primary account number (PAN), excluding the check digit. A1 is the most significant digit and A12 is the digit immediately preceding the PAN's check digit. If the primary account number excluding the check digit is less than twelve digits, the digits are right justified and padded on the left with zeroes. Permissible values are 0000 to 1001.

0 - Pad digit = 0000. The first four digits of the account number block are always padded with this value.

Formatted Clear-Text PIN Block

The PIN and account number blocks are Exclusive ORed before being assembled in the DES (Data Encryption Standard) input register. When the account number is not available, only the PIN block is assembled in the DES input register. PP190 will output DES/TDES encrypted PIN block with message 71 and delete clear-text PIN block immediately after transaction completed.

Example:

Account Number: 1234567890-6 (6 is check number and will be ignored)

PIN: 8780

The PIN block = `04 87 80 FF FF FF FF FF`

The PANB = `00 00 12 34 56 78 90 00`

Formatted PIN block = `04 87 92 CB A9 87 6F FF (Data to be encrypted)`

Appendix C Fixed Prompts for Z2/Z3 authenticated mode

Prompt ID	Display
001	ACCOUNT NUMBER
002	AIRCRAFT TAIL NO
003	BADGE NUMBER
004	CARD NUMBER
005	CARD SEC CODE
006	CASH BACK AMOUNT
007	CID CODE
008	CVC CODE
009	CVN CODE
010	CVV CODE
011	CUSTOMER CODE
012	CUSTOMER DATA
013	CUSTOMER ID
014	CUSTOMER NUMBER
015	CUSTOMER REF
016	CUSTOMER REF NO.
017	DATE OF BIRTH
018	DEPARTMENT NO.
019	DRIVER ID
020	DRIVER LICENSE
021	DRIVER NUMBER
022	EMPLOYEE ID
023	EMPLOYEE NUMBER
024	ENTER
025	ENTER ACCOUNT #
026	ENTER AIR TAIL #
027	ENTER BADGE #
028	ENTER BIRTH DATE
029	ENTER CARD #
030	ENTER CASH BACK
031	ENTER CUST #
032	ENTER CUST CODE
033	ENTER CUST DATA
034	ENTER CUST ID

Prompt ID	Display
035	ENTER CUST REF
036	ENTER CUST REF #
037	ENTER CID CODE
038	ENTER CVC CODE
039	ENTER CVN CODE
040	ENTER CVV CODE
041	ENTER DEPT #
042	ENTER DOB
043	ENTER DRIVER #
044	ENTER DRIVER ID
045	ENTER DRIVER LIC
046	ENTER EMP ID
047	ENTER EMPLOYEE #
048	ENTER EXP DATE
049	ENTER FLEET #
050	ENTER FLEET DATA
051	ENTER HOME PHONE
052	ENTER ID #
053	ENTER JOB #
054	ENTER ODOMETER
055	ENTER PHONE #
056	ENTER PO #
057	ENTER REF #
058	ENTER ROUTE #
059	ENTER SEC CODE
060	ENTER SERIAL #
061	ENTER SOC SEC #
062	ENTER SSN
063	ENTER STREET #
064	ENTER TRAILER #
065	ENTER USER ID
066	ENTER V-CODE
067	ENTER VEH CARD #
068	ENTER VEHICLE #

Prompt ID	Display
069	ENTER VEHICLE ID
070	ENTER WORK PHONE
071	ENTER ZIP CODE
072	EXPIRATION DATE
073	FLEET DATA
074	FLEET NUMBER
075	HOME PHONE NO.
076	ID NUMBER
077	JOB NUMBER
078	MMDDYY
079	MMDDYYYY
080	MMYY
081	ODOMETER READING
082	OR PHONE #
083	OR PHONE NUMBER
084	PHONE NUMBER
085	PLEASE
086	PLEASE ENTER
087	PLEASE RE-ENTER
088	PO NUMBER
089	RE-ENTER
090	REFERENCE NUMBER
091	RESTRICTION CODE
092	ROUTE NUMBER
093	SECURITY CODE
094	SERIAL NUMBER
095	SOCIAL SEC NO.
096	STREET NUMBER
097	SWIPE CARD
098	SWIPE CARD OR
099	TRAILER NUMBER
100	USER ID
101	V-CODE
102	VEHICLE CARD NO.
103	VEHICLE ID
104	VEHICLE NUMBER
105	WORK PHONE NO.
106	ZIP CODE
107	ENTER CARD
108	ENTER ST ADDRESS
109	STREET ADDRESS
110	SWIPE/TAP CARD
111	SWIPE/INSERT CRD
112	TAP/INSERT CARD
113	OR ENTER ACCT #
114	OR ENTER
115	TAP CARD
116	TAP CARD OR
117	INSERT CARD
118	INSERT CARD OR
119	SELECT CARD TYPE
120	CREDIT
121	DEBIT
122	EBT
123	GIFT
124	LOYALTY
125	GIFT/LOYALTY
126	HGM
127	STORED VALUE
128	GSB
129	ONECARD
130	GSB/ONECARD

Appendix D Fixed Prompts for Z2/Z3 PIN entry mode

Prompt ID	Display
001	ENTER PIN
002	ENTER YOUR PIN
003	PLEASE ENTER PIN
004	THEN PRESS ENTER
005	THANK YOU