

5E9020.29-FCC

Technical documentation

Version: **1.00 (September 2014)**
Model no.: **5E9020.29-FCC**

All information contained in this manual is current as of its creation/publication. B&R reserves the right to change the contents of this manual without notice. The information contained herein is believed to be accurate as of the date of publication; however, Bernecker + Rainer Industrie-Elektronik Ges.m.b.H. makes no warranty, expressed or implied, with regard to the products or documentation contained within this manual. In addition, Bernecker + Rainer Industrie-Elektronik Ges.m.b.H. shall not be liable for any incidental or consequential damages in connection with or arising from the furnishing, performance or use of the product(s) in this documentation. Software names, hardware names and trademarks are registered by their respective companies.

1 Views.....	3
2 General information.....	4
2.1 User information.....	4
2.1.1 Description.....	4
2.1.2 Version information.....	4
2.2 Organization of safety notices.....	4
2.3 Guidelines.....	4
3 Technical data.....	6
4 Command set.....	7
4.1 General information (command set).....	7
4.2 Getting started.....	8
4.3 Commands.....	9
4.3.1 ISO 15693 memory organization.....	10
4.4 Access rights.....	11
4.5 MIFARE command examples.....	12
4.6 Rights and memory organization.....	14
4.7 Value block.....	14
4.8 Updating firmware.....	14
4.9 Error messages and numbers.....	16

1 Views

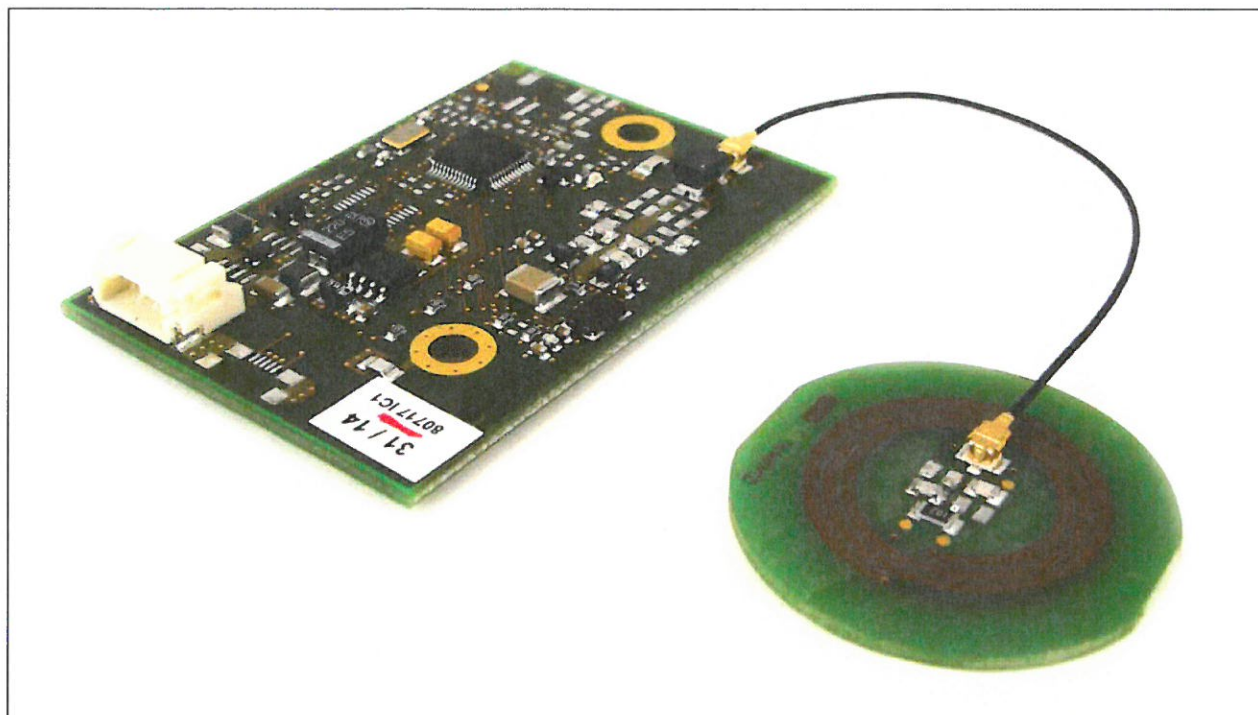


Figure 1: 5E9020.29-FCC - Oblique view

2 General information

Information:

B&R keeps the printed version of technical descriptions as current as possible. The latest version of this technical description can be downloaded in PDF format from the B&R website at www.br-automation.com.

2.1 User information

User Information in accordance with FCC 15.21

- Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

Part 15 Statement in accordance with FCC 15.19/RSS Gen Issue 3 Sect. 7.1.3

- This device complies with Part 15 of the FCC Rules and with Industry Canada license-exempt RSS standard(s). 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.

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes : (1) l'appareil ne doit pas produire de brouillage, et (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

2.1.1 Description

The 5E9020.29-FCC transponder is a read/write unit with the following specifications:

- 13.56 MHz read/write transponder unit
- ISO 15693 and MIFARE compatible
- USB interface

2.1.2 Version information

Version	Date	Comment	Responsible
1.00 (starting with Rev. A0)	25.09.2014	First edition	Anna Sigl

Table 1: Version information

2.2 Organization of safety notices

Safety notices in this manual are organized as follows:

Safety notice	Description
Danger!	Disregarding these safety guidelines and notices can be life-threatening.
Caution!	Disregarding these safety guidelines and notices can result in severe injury or substantial damage to equipment.
Warning!	Disregarding these safety guidelines and notices can result in injury or damage to equipment.
Information:	This information is important for preventing errors.

Table 2: Organization of safety notices

2.3 Guidelines



European dimension standards apply to all dimension diagrams in this document.

All dimensions are specified in mm.

Range of nominal sizes	General tolerance according to DIN ISO 2768 (medium)
Up to 6 mm	± 0.1 mm
For 6 to 30 mm	± 0.2 mm
For 30 to 120 mm	± 0.3 mm
For 120 to 400 mm	± 0.5 mm
For 400 to 1000 mm	± 0.8 mm

Table 3: Range of nominal sizes

3 Technical data

Product ID	5E9020.29-FCC
General information	
Certification CE	Yes
Interfaces	
USB	
Type	USB 2.0
Design	Type A
Transfer rate	Low speed (1.5 Mbit/s), full speed (12 Mbit/s)
RFID transponder read/write unit	
Type	For I-Code SLI transponder with amplitude modulation and MIFARE classic, carrier frequency 13.56 MHz
Read/Write range in air	Approx. 1 to 3 cm
Electrical characteristics	
Power consumption	Max. 0.5 W
Supply voltage	5 VDC $\pm 10\%$ (via USB)
Environmental conditions	
Temperature	
Operation	0 to 50°C
Storage	-20 to 60°C
Transport	-20 to 60°C
Relative humidity	
Operation	5 to 90%, non-condensing
Storage	5 to 90%, non-condensing
Transport	5 to 90%, non-condensing
Mechanical characteristics	
Dimensions	
Width	625 mm
Height	427 mm
Diameter	35mm (antenna)
Weight	TBD

Table 4: 5E9020.29-FCC - Technical data

4 Command set

4.1 General information (command set)

The transponder reader reads and writes MIFARE and ISO 15693 tags. The corresponding commands and parameters are valid depending on the tag being used. When a tag comes within range of the antenna, the PiccSelect message and serial number of the tag are output. When the tag is removed, the PiccRemove message and serial number of the tag are output.

***** SW_ResetSTART *****

** Mifare **

** B&R Automation **

** S/W Rev. 0.7 **

ENTER 'helpme' for information

PiccSelect: 3B7CF4E0

PiccRemove: 3B7CF4E0

An executed command appears as follows:

1. "Command syntax" command ->

2nd The executed command in plain text

3. Response

read,a,ff ff ff ff ff,5

"read" command ->

Data in block 5 in HEX=00000000000000000000000000000000

Error messages are constructed as follows:

"Error: <ErrorNumber> (error syntax)"

Operating errors and RFID stack errors are differentiated. See the tables "Operating errors" and "RFID stack LOW byte error messages".

4.2 Getting started

Before using the reader, the driver for the USB communication device class (CDC) must be installed.

Note:

This driver can be downloaded from the B&R website [B&R Homepage \(http://www.br-automation.com\)](http://www.br-automation.com).

Port settings:

Bits per second: 115200

Data bits: 8

Parity: None

Stop bits: 1

Flow control: None

The keys needed for authentication can be stored either in the reader's EEPROM or temporary memory. Alternatively, the authentication keys can be sent with the command.

The storage location and whether or not the key is sent with the command are defined using the commands Key_EEPROM, Key_TEMP and Key_CMD.

Note:

Authentication is only necessary for MIFARE.

The following commands are used to store the authentication key on the reader.

MIFARE commands - Authentication key storage					
Command	Para1	Para2	Para3	Para4	Para5
Store_key_eeprom	"A" or "B"	Sector for key in EEPROM reader	6-byte key	--	--
Store_key_temp	6-byte key	--	--	--	--

Table 5: MIFARE commands - Authentication key storage

Example:

```
store_key_eeprom,a,0,ff ff ff ff ff
```

"store_key_eeprom" command ->

Key in Reader-EEProm sector 000 written

Default key (when tags delivered): 0xFF FF FF FF FF FF

Up to 16 keys (0-15) can be stored in the EEPROM for each key (A and B).

4.3 Commands

General commands

Command	Description	Para1	Para2	Para3	Para4	Para5
Info_On	Outputs command confirmation	--	--	--	--	--
Info_Off	Command confirmation not output (DEFAULT)	--	--	--	--	--
Show_Config	Displays current settings	--	--	--	--	--
Show_Revision	Displays software and hardware revision information	--	--	--	--	--
Show_Status	Displays the RFID stack error	--	--	--	--	--
Show_SN	Displays the active tag number	--	--	--	--	--
Show_Key	Displays the active key	--	--	--	--	--
Echo_On	Displays an input echo for the output	--	--	--	--	--
Echo_Off	Suppresses the input echo for the output (DEFAULT)	--	--	--	--	--
Startup	Displays the startup message	--	--	--	--	--
Life	Life command --> Returns OK	--	--	--	--	--
Helpme	Displays a list of all commands	--	--	--	--	--

Table 6: General commands

Update commands

Command	Description	Para1	Para2	Para3	Para4	Para5
Restarting	Restarts the reader	--	--	--	--	--
Firmware_Update	Switches the µC to X-modem update mode	--	--	--	--	--
Firmware_Upgrade	Switches the µC to USB mass memory update mode	--	--	--	--	--
Firmware_Info	Displays information about the current firmware	--	--	--	--	--

Table 7: Update commands

MIFARE commands

Command	Description	Para1	Para2	Para3	Para4	Para5
Read	Reads a 16-byte block from PICC	"A" or "B"	See table "MIFARE parameters 2".	Source block	--	--
Read_Blocks	Reads the specified range of blocks from PICC	"A" or "B"	See table "MIFARE parameters 2".	Start block	End block	--
Write	Writes a 16-byte block to PICC	"A" or "B"	See table "MIFARE parameters 2".	Target block	16-byte data in hex 00h	--
Decr	Takes the value of the source VALUE block, subtracts the specified value and writes the result to the destination block	"A" or "B"	See table "MIFARE parameters 2".	Source block	Target block	Value
Incr	Takes the value of the source value block, adds the specified value and writes the result to the destination block	"A" or "B"	See table "MIFARE parameters 2".	Source block	Target block	Value
Restore	Copies a VALUE block	"A" or "B"	See table "MIFARE parameters 2".	Source block	Target block	--
InitZero	Initializes a VALUE block with the value 0	"A" or "B"	See table "MIFARE parameters 2".	Target block	--	--
Store_Key_EEPROM	Stores a 6-byte key in EEPROM memory	"A" or "B"	Sector for key in EEPROM reader	6-byte key		
Store_Key_Temp	Stores a 6-byte key in the µC temporarily until the reader is turned off	6-byte key	--	--	--	--
Connect	Connects manually to a specific tag	4-byte SNr	--	--	--	--
Disconnect	Automatically restores a connection to the best tag	--	--	--	--	--
Key_CMD	Sends a key directly with each command (DEFAULT)	--	--	--	--	--
Key_EEPROM	Uses the key stored in EEPROM memory	--	--	--	--	--
Key_TEMP	Uses the temporary key	--	--	--	--	--

Table 8: MIFARE commands

ISO15693 commands

Command	Description	Parameter1	Parameter2
Read	Reads a 4-byte block from PICC	Source block	--
Read_Blocks	Reads the specified range of blocks from PICC	Start block	End block
Write	Writes a 4-byte block to PICC	Target block	4-byte data in hex 00h
Sys_Info	Reads the AFI, DSFID, number of bytes and bytes/block	--	--
Security	Displays the write protection status of individual blocks	Start block	End block
Set_AFI	Writes the AFI value	1-byte value	--
Set_DSFID	Writes the DSFID value	1-byte value	--
Lock_Block	Enables write protection for a block	Target block	--
Lock_AFI	Sets AFI write protection	--	--
Lock_DSFID	Sets DSFID write protection	--	--
Connect	Connects manually to a specific tag	8-byte SNr	--
Disconnect	Automatically restores a connection to the best tag	--	--

Table 9: ISO15693 commands

Commands have the following structure

[Command] , [Parameter1] , ... [ParameterN]

Parameters are separated by commas; the command ends with "\r".

Case sensitivity is ignored within commands.

Note:

When a setting is changed, all settings are output.

4.3.1 ISO 15693 memory organization

Source and target block = 0–254 (depending on the tag)

Data and key = 00–FF (no leading "0x", bytes can optionally be separated by a space)

Value = 1 byte including sign bit

All commands and parameters are in ASCII. Data is in hex 00h format.

4.4 Access rights

There are multiple options for Parameter 2 depending on the setting.

MIFARE access rights (Parameter2)			
Config	Key_CMD	Key_EEPROM	Key_TEMP
Para2	6-byte key	Sector for key in EEPROM reader	Uses the temporarily stored key – dummy value (0-63)

Table 10: MIFARE access rights (Parameter2)

"A" = authentication with key A, "B" = authentication with key B

Sector = 0–63

Source and target block = 0-254 (depending on type: MIFARE 1K 0-63, MIFARE 4K 0-254;

for "Decrement", "Increment" and "Restore", the source block must be of type "Value block")

Data and key = 00–FF (no leading "0x", bytes can optionally be separated by a space)

Value = 4 bytes including sign bit

All commands and parameters are in ASCII. Data is in hex 00h format.

MIFARE access rights (5-byte parameter)

Para1	Para2	Para3	Para4	Para5
Block (typically 3 – sector trailer)	Access rights for block 0 (data block)	Access rights for block 1 (data block)	Access rights for block 2 (data block)	Access rights for block 3 (sector trailer)

Table 11: MIFARE access rights (5-byte parameter)

MIFARE access rights - Parameters 2, 3 and 4 (data blocks)

Value	RD	WR	INCR	DECR / TRANSFER / RESTORE	Comment
0x00	A / B	A / B	A / B	A / B	Full access with every key (default factory setting)
0x01	A / B	B	-	-	
0x02	A / B	-	-	-	
0x03	A / B	B	B	A / B	Value block
0x04	A / B	-	-	A / B	Value block
0x05	B	-	-	-	
0x06	B	B	-	-	
0x07	-	-	-	-	No access

Table 12: MIFARE access rights - Parameters 2, 3 and 4 (data blocks)

If key B is read in a particular sector trailer, then it cannot be used for authentication (all of the yellow cells in the table above).

Consequences: If the reader attempts to authenticate a block in a sector with Key B (access conditions marked in gray), then the card will refuse all subsequent memory accesses following authentication.

MIFARE access rights - Parameter 5 (sector trailer)

Value	Key A		Access bits		Key B		Comment
	RD	WR	RD	WR	RD	WR	
0x00	-	A	A	-	A	A	Key B can be read
0x01	-	B	A / B	-	-	B	
0x02	-	-	A	-	A	-	Key B can be read
0x03	-	-	A / B	-	-	-	No access
0x04	-	A	A	A	A	A	Key B can be read (factory default)
0x05	-	-	A / B	B	-	-	
0x06	-	B	A / B	B	-	B	
0x07	-	-	A / B	-	-	-	No access

Table 13: MIFARE access rights - Parameter 5 (sector trailer)

The cells marked in yellow are access conditions where key B can be read and used for data.

4.5 MIFARE command examples

Writing data to block 5 with key A from the reader's EEPROM sector 0: Write,A,0,5,01 23 45 67 89 AB CD EF 01 23 45 67 89 AB CD EF

```
write,a,0,5,0123456789ABCDEF0123456789ABCDEF
```

"write" command ->

Data in sector 1 block 5 written=0123456789ABCDEF0123456789ABCDEF

Reading data from block 5 with key A from the reader's EEPROM sector 0: Read,A,0,5

```
read,a,0,5
```

"read" command ->

Data in block 5 in HEX=0123456789ABCDEF0123456789ABCDEF

Initializing block 8 with key A from EEPROM sector 0 as a VALUE block

```
initzero,a,0,8
```

"initzero" command ->

Data in sector 2 block 8 written=00000000FFFFFFFF0000000000FF00FF

Incrementing block 8 by 2 with key A from sector 0 of the reader's EEPROM:

```
Incr,a,0,8,8,2
```

"incr" command ->

Block 008 successfully implemented by 0000000002 and written to block 008

Querying from block 5 with a key in the command

```
read,a,ff ff ff ff ff ff,5
```

"read" command ->

Data in block 5 in HEX=FFFF0000000000000000000000000000

Connecting manually to a specific tag

```
connect,0b a1 78 98
```

"connect" command ->

```
*****INFO*****
```

```
Key_mode = Key_eeprom
```

```
Info_text = Info_on
```

```
Tag_connected = Manual_mode
```

```
Tag_sn = 0BA17898
```

```
*****END_OF_INFO*****
```

Reading blocks 0 to 3

```
read_blocks,a,0,0,3
```

"read_blocks" commands ->

Data in block 0 in HEX=3B7CF4E05388040046B9949745302809

Data in block 1 in HEX=00000000000000000000000000000000

Data in block 2 in HEX=00000000000000000000000000000000

Data in block 3 in HEX=000000000000FF078069FFFFFFFFFFFF

4.6 Rights and memory organization

The tag is divided into 16 sectors of 4 blocks each. Each block has 16 bytes. The 4th block in each sector contains the keys and the rights for the respective sector (i.e. the "sector trailer"). Rights can be assigned individually to each block. Each sector can be assigned one of two different keys (A or B).

These keys and rights must be stored in a defined format (according to the MIFARE® specification). To change these rights, use the "write" command to write to the respective sector trailer using the correct format.

This document will provide no further details regarding the organization of memory and TAG rights. For more information please see the data sheet "mifare Standard Card IC MF1 IC S50 Functional Specification".

4.7 Value block

A block can be used as a value block. A value block is 4 bytes, including the sign bit. When a command (read, increment, decrement, restore) is used, the tag automatically increments/decrements the value without requiring the value to be read. In order to use a block as a value block, it must adhere to a defined format.

This document does not go into any more details regarding TAG value blocks. For more information please see the PDF "mifare Standard Card IC MF1 IC S50 Functional Specification".

4.8 Updating firmware

Firmware can be updated using HyperTerminal or a similar program (RealTerm, Tera Term, etc.). HyperTerminal is no longer supported in Windows 7; in this case, we recommend Tera Term.

Complete the following steps to update the firmware:

- Enter the command "firmware_update".
- Upload the firmware (e.g. in HyperTerminal -> Transfer -> Send file -> Select file ->).
- Transfer the file.
- The update is carried out.
- The reader restarts.
- Check the information.
- The update is complete.

The terminal looks like this:



Figure 2: Terminal view 1

```

**          B&R Automation          **
          S/W Rev. 65535.65535
**ENTER 'Helpme' for information**
*****
!Error:-1 (PiccRequest)
firmware_update
Kommando FIRMWARE_UPDATE ->
FIRMWARE_UPDATE
CCCCCCCCFirmware-Update complete with 27648 Bytes
***
***** Bootloader *****
*****
New Firmware found: S/W Rev. 1.3
Start Firmware copy
Firmware copy finished
Firmware found-> Version 1.3
Info:09.06.2011 12:50:53
**          Mifare          **
**          B&R Automation          **
          S/W Rev. 1.3
**ENTER 'Helpme' for information**
*****
!Error:-1 (PiccRequest)
-

```

Figure 3: Terminal view 2

4.9 Error messages and numbers

Operating errors

Operating errors		
Value	Name	Description
0	ERR_NONE	No errors (OK)
1	ERR_OTHER	Other operating error, unrecognized command
2	ERR_PARAMETER	Incorrect number of parameters
3	ERR_AUTH	Authentication error (A or B)
4	ERR_SECTOR	Invalid range for sector (0-63)
5	ERR_BLOCK_SRC	Invalid range for source block (0-255)
6	ERR_WRITE	No data found to write
7	ERR_KEY_EE	No key found for writing
8	ERR_KEY_STORE	Key could not be saved
9	ERR_BLOCK_DST	Invalid range for target block (0-255)
10	ERR_NO_TAG	No tag in range
11	ERR_BLOCK_NUM	Incorrect values for source/target blocks
12	ERR_TAG_NUM	Invalid tag number
13	ERR_KEY	Invalid key
14	ERR_REMOTE	X-modem firmware update canceled (not used)
15	ERR_SYNC	X-modem firmware update packet start not found (not used)
16	ERR_RETRY	X-modem firmware update number of retry attempts (not used)
17	ERR_READ	Data could not be read
18	ERR_INCDECREAS	Error accessing the VALUE block
19	ERR_LOCK	Could not lock block
20	ERR_AFI	Error writing the AFI value
21	ERR_DSFDID	Error writing the DSFDID value
22	ERR_SYSINFO	Could not read system information

Table 14: Operating errors

RFID stack error messages

RFID stack errors can be queried using the "show_status" command. Each error number consists of a HIGH byte (stack components) and a LOW byte (errors).

RFID stack error messages		
Value	Name	Description
0xXX00	PH_ERR_SUCCESS	Returned if no error
0xXX71	PH_ERR_SUCCESS_CHAINING	Rx chaining not complete, further action needed
0xXX72	PH_ERR_SUCCESS_INCOMPLETE_BYTE	Incomplete byte received
0xXX01	PH_ERR_IO_TIMEOUT	No reply received, e.g. PICC removal
0xXX02	PH_ERR_INTEGRITY_ERROR	Wrong CRC or parity detected
0xXX03	PH_ERR_COLLISION_ERROR	Collision occurred
0xXX04	PH_ERR_BUFFER_OVERFLOW	Attempt to write beyond buffer size
0xXX05	PH_ERR_FRAMING_ERROR	Invalid frame format
0xXX06	PH_ERR_PROTOCOL_ERROR	Received response violates protocol
0xXX07	PH_ERR_AUTH_ERROR	Authentication error
0xXX08	PH_ERR_READ_WRITE_ERROR	Read or write error in RAM/ROM or flash memory
0xXX09	PH_ERR_TEMPERATURE_ERROR	RC sensors signaling overheating
0xXX0A	PH_ERR_RF_ERROR	Error on RF interface
0xXX0B	PH_ERR_INTERFACE_ERROR	RC communication error
0xXX0C	PH_ERR_LENGTH_ERROR	Length error
0xXX7F	PH_ERR_INTERNAL_ERROR	Internal error
0xXX20	PH_ERR_INVALID_DATA_PARAMS	Invalid data parameters supplied (layer ID check failed)
0xXX21	PH_ERR_INVALID_PARAMETER	Invalid parameter supplied
0xXX22	PH_ERR_PARAMETER_OVERFLOW	Reading/Writing a parameter would produce an overflow
0xXX23	PH_ERR_UNSUPPORTED_PARAMETER	Parameter not supported
0xXX24	PH_ERR_UNSUPPORTED_COMMAND	Command not supported
0xXX25	PH_ERR_USE_CONDITION	Condition of use not satisfied
0xXX26	PH_ERR_KEY	Key error

Table 15: RFID stack error messages

Error messages - RFID stack HIGH byte		
Value	Name	Description
0x00XX	PH_COMP_GENERIC	Generic component code
0x01XX	PH_COMP_BAL	BAL component code
0x02XX	PH_COMP_HAL	HAL component code
0x03XX	PH_COMP_PAL_ISO14443P3A	ISO14443-3A PAL component code
0x04XX	PH_COMP_PAL_ISO14443P3B	ISO14443-3A PAL component code
0x05XX	PH_COMP_PAL_ISO14443P4A	ISO14443-4A PAL component code
0x06XX	PH_COMP_PAL_ISO14443P4	ISO14443-4 PAL component code
0x07XX	PH_COMP_PAL_MIFARE	MIFARE(R) PAL component code
0x08XX	PH_COMP_PAL_FELICA	Open FeliCa PAL component code
0x09XX	PH_COMP_PAL_EPCUID	ICode EPC/UID PAL component code
0x0AXX	PH_COMP_PAL_SLI15693	ICode SLI/ISO15693 PAL component code
0x0BXX	PH_COMP_PAL_I18000P3M3	ISO18000-3 Mode3 PAL component code
0x0CXX	PH_COMP_PAL_I18092MPI	ISO18092 passive initiator mode PAL component code
0x10XX	PH_COMP_AL_MFC	MIFARE(R) Classic AL component code
0x11XX	PH_COMP_AL_MFUL	MIFARE(R) Ultralight AL component code
0x12XX	PH_COMP_AL_MFP	MIFARE(R) Plus AL component code
0x13XX	PH_COMP_AL_VCA	Virtual Card Architecture AL component code
0x14XX	PH_COMP_AL_FELICA	Open FeliCa AL component code
0x15XX	PH_COMP_AL_I15693	ISO15693 AL component code
0x16XX	PH_COMP_AL_SLI	ICode SLI AL component code
0x18XX	PH_COMP_AL_I18000P3M3	ISO18000-3 Mode3 AL component code
0x19XX	PH_COMP_AL_MFDF	MIFARE DESFIRE EV1 AL component code
0x1AXX	PH_COMP_AL_P40CMDPRIV	P40 command libraryAL component code
0x1BXX	PH_COMP_AL_P40CMDPUB	P40 command libraryAL component code
0x30XX	PH_COMP_DL_AMP	Amplifier DL component code
0x31XX	PH_COMP_DL_THSTRM	Thermostream DL component code
0x32XX	PH_COMP_DL_OSCI	Oscilloscope DL component code
0x33XX	PH_COMP_DL_RDFPGA	Reader FPGA Box DL component code
0x34XX	PH_COMP_DL_MSTAMPOSC	Master Amplifier Oscilloscope DL component code
0x35XX	PH_COMP_DL_STEPPER	Stepper DL component code
0xE0XX	PH_COMP_CIDMANAGER	Cid Manager component code
0xE1XX	PH_COMP_CRYPTOSYM	CryptoSym component code
0xE2XX	PH_COMP_KEYSTORE	KeyStore component code
0xE3XX	PH_COMP_TOOLS	Tools component code
0xE4XX	PH_COMP_CRYPTORNG	CryptoRng component code
0xEFXX	PH_COMP_LOG	Log component code

Table 16: Error messages - RFID stack HIGH byte

Figure index

Figure 1:	5E9020.29-FCC - Oblique view.....	3
Figure 2:	Terminal view 1.....	14
Figure 3:	Terminal view 2.....	15

Table 1:	Version information.....	4
Table 2:	Organization of safety notices.....	4
Table 3:	Range of nominal sizes.....	5
Table 4:	5E9020.29-FCC - Technical data.....	6
Table 5:	MIFARE commands - Authentication key storage.....	8
Table 6:	General commands.....	9
Table 7:	Update commands.....	9
Table 8:	MIFARE commands.....	9
Table 9:	ISO15693 commands.....	10
Table 10:	MIFARE access rights (Parameter2).....	11
Table 11:	MIFARE access rights (5-byte parameter).....	11
Table 12:	MIFARE access rights - Parameters 2, 3 and 4 (data blocks).....	11
Table 13:	MIFARE access rights - Parameter 5 (sector trailer).....	11
Table 14:	Operating errors.....	16
Table 15:	RFID stack error messages.....	16
Table 16:	Error messages - RFID stack HIGH byte.....	17