Instructions for Use

KBRP, Version 3.01
English

IMPORTANT
Read carefully before use!

Communication Protocol
Kathrein RFID UHF Readers

Quality made in Germany

KATHREIN

Contents

1 About This Guide

These instructions for use describe the communication protocols, commands and configuration parameters required for the software development of the reader control.

This document is valid for all Kathrein RFID readers.

Although the reference in these instructions is made to the RRU4 reader, the same commands are used to operate any other reader.

► For more information, visit our website www.kathrein-solutions.com.
  ☒ The manuals are available for download at the internet product page.

2 Explanation of Symbols and Signal Words

2.1 Signal Words

Tip	This signal word indicates useful tips and recommendations.

2.2 Other Symbols

Symbol	Meaning
Program	frames, block structures, data blocks, result flags etc.

3 Intended Use

The Kathrein-Burgstädt Reader Protocol — subsequently referred to as KBRP — is used to interface the Kathrein UHF RFID reader and its successors to a higher level. It is possible to transfer system commands, data commands and diagnostics commands as well as their responses.

Tip This document applies to all Kathrein RFID readers. Even if it's referred to in the text as RRU4, it is possible to control all other readers using the same commands.

4 Transmission Methods

4.1 UART transmission (RS232, RS422, RS485 or similar)

4.1.1 Physical Layer

A full or half-duplex connection such as RS232, RS422 or RS485 is used for the physical layer.

4.1.2 Data Link Layer

Transmission takes place in frames and blocks. A block comprises a maximum of 256 frames. A frame comprises a maximum of 256 bytes, of which a maximum of 250 bytes can be user data. The result is a maximum block size of 64000 bytes of user data.

The data link layer is used to safeguard the data between the sender and recipient. The sender receives a response from the recipient for each correct frame received. If the sender does not receive a response from the recipient within a time window of 350 milliseconds after sending a frame, the frame sent is repeated until the error counter signals the cancellation of the transmission.

Frame Structure

5A LL SS FF DD ... DD P1 P2

5A	start code for synchronisation
LL	number of bytes in the frame, not including the start code
SS	status byte
FF	frame number
DD	user data
P1	16-bit checksum, low-byte
P2	16-bit checksum, high-byte

Start Code and Synchronisation

The start code is used to synchronise the recipient to the sender. Furthermore, the recipient synchronises to the start of a frame when no data have been received for 100 milliseconds.

Status Byte

The status byte has the following meaning:

50	data packet
A0	response *OK*
LL	response *Memory error* (the recipient was unable to allocate any memory for the received data block)

A response is only 3 bytes long and is not CRC-checked.

OK	*5A 02 A0*
Memory error	*5A 02 A1*

Frame Number

The frame number shows how many more frames belong to this data block. Only the first frame in a data block can be shorter than 256 bytes. Each additional frame must have a length of 256 bytes (length byte LL is FF).

Therefore, it is possible to calculate the block size from the first frame number, see the example below.

A block with 700 bytes of user data is to be transmitted. For this purpose, the block is divided into three frames:

1st frame: *5A CD 50 02* — 200 bytes of user data follow — P1 P2

2nd frame: *5A FF 50 01* — 250 bytes of user data follow — P1 P2

3rd frame: *5A FF 50 00* — 250 bytes of user data follow — P1 P2

The receiver can use the frame number of the first frame (here *02*) and its length byte to calculate the block size (block size = frame number * 250 bytes + length byte -5) (here in the example: 2 * 250 bytes + 205 bytes - 5 bytes = 700 bytes), and reserve an appropriate amount of memory for the data.

User Data

User data are the bytes in a frame that flow into the block transmitted.

Checksum

The checksum is calculated using the polynomial x^16 + x^12 + x^5 + 1 with a pre-initialisation of 0x0000 from the start code to the last user data byte.

Network Layer

As the KBRP is a point-to-point protocol, there is no network layer.

Transport Layer, Session Layer, Presentation Layer,

Do not exist.

Application Layer

The application layer transmits data blocks from 1 to a maximum of 64000 bytes.

4.2 Ethernet Transmission (Generation 2 Reader)

A data transmission layer has been used for communication to our reader over Ethernet, just like in serial communication.

The data transmission layer over Ethernet is a much more simple solution here since the TCP/IP protocol already has a data security layer. As TCP/IP is a stream protocol, only the packet beginning and packet end are necessary.

4.2.1 Frame Set-up

A frame looks as follows:

Start + data block + end

The start is made up of *0xAA 0xBB 0x01 0x01*, whereby the first 1 is the Datetransmit byte and the second 1 is a Stuffbyte. The end is made up of *0xAA 0xCC*. If the byte *0xAA* appears in the KBRP frame, it must be doubled (*0XAA 0xAA 0xAA*).

4.2.2 Port

The TCP communication port is the port 4007.

4.2.3 Example

The frame *ASyncGetEPCs* is shown here as an example. The ID for this command is *0x0111*, which makes the frame look as follows:

0xAA 0xBB 0x01 0x01 0x11 0x01 0xAA 0xCC

4.2.4 Extended Block Structure

If a data block to be transmitted exceeds 16 kB, it is necessary to subdivide into several 16-kB blocks. These blocks receive a block number, and the first block contains the total data length. It is necessary to confirm the reception of each block by means of an answer.

First block:

Start + block number always 0 + 4-byte total data length + data block + end

All other blocks:

Start + block number + data block + end

Answer to confirm the reception:

Start + block number + end

The start consists of *0xAA 0xBB 0x0E 0x01*. The end consists of *0xAA 0xCC*. If the *0xAA* occurs in the KBRP frame, it is necessary to double it: *0xAA → 0xAA 0xAA*.

4.3 Ethernet Transmission (Generation 3 Reader)

There are 3 frame types in the protocol used in the Generation 3 readers.

If no data have been received for 100 milliseconds within a frame, the received part frame is dismissed. The payload (D1 to Dn) is identical to the Generation 2 payload.

4.3.1 Frame Set-up

Data Frame

5A 50 L1 L2 L3 L4 D1 ... Dn A5

5A	start code
50	code for a data frame
L1 – L4	number of data bytes (D1 to Dn) in the data frame (32 bit; LSB first)
D1 – D4	data bytes to be transmitted (payload)
A5	end code

Ping

5A 5F A5

5A	start code
5F	ping code
A5	end code

Response to Ping

5A AF A5

5A	start code
AF	ping response code
A5	end code

4.3.2 Port

The communication takes place via the TCP port 4007 as in case of Generation 2 readers.

4.3.3 Example

The frame *ASyncGetEPCs* is shown here as an example. The ID for this command is *0x0111*, which makes the frame look as follows:

0x5A 0x50 0x02 0x00 0x00 0x00 0x11 0x01 0xA5

4.4 Command and Response Transmission

Commands and their responses are transmitted in individual data blocks. The same applies to asynchronous responses from the reader (responses without an immediately preceding command).

Structure of the data blocks:

I1 I2 DD ... DD

I1	command / response ID low-byte
I2	command / response ID high-byte
DD	command / response data

5 Data Types

5.1 Command/Response IDs

typedef enum

{

// RRU4 commands	
CMD_GetMode = 0x0001,	// 0x0001
CMD_SetMode,	// 0x0002
...	
// RRU4 command responses	
RES_GetMode = 0x8001,	// 0x8001
RES_SetMode,	// 0x8001
...	
// RRU4 asynchronous command responses	
REA_ASyncGetRawEPCs = 0xC110,	// 0xC110
REA_ASyncGetEPCs,	// 0xC111
...	
// RRU4 asynchronous messages	
REA_ReaderMessage = 0xC301	// 0xC301

} tHostCommCommand;

The complete Enum is situated in the *rru4api.h* file in the API files or in the API examples.

5.2 Result Flag

typedef enum
{

RF_NoError = 0,	// 0x00
RF_NoData,	// 0x01
RF_CRCError,	// 0x02
RF_NoLicense,	// 0x03
RF_OutOfRange,	// 0x04
RF_NoStandard,	// 0x05
RF_NoAntenna,	// 0x06
RF_NoFrequency,	// 0x07
RF_NoCarrier,	// 0x08
RF_AntennaError,	// 0x09
RF_NoTag,	// 0x0A
RF_MoreThanOneTagInField,	// 0x0B
RF_WrongLicenseKey,	// 0x0C
RF_FirmwareRejected,	// 0x0D
RF_WrongCommandFunctionMode,	// 0x0E
RF_NoHandle,	// 0x0F
RF_NoProfile,	// 0x10
RF_AccessDenied,	// 0x11
RF_ TransmissionTimeExpired,	// 0x12
RF_DatabaseError,	// 0x13
RF_ParameterError,	// 0x14
RF_SignatureVerificationFailed,	// 0x15
RF_OutOfMemory = 0x7F,	// 0x7F
RF_NonSpecified = 0x80	// 0x80

} tResultFlag;

5.3 Reader Function Mode

typedef enum
{

CFM_Normal = 0,
CFM_Direct,
CFM_Last

} tCommandFunctionMode;

5.4 Communication Standards

typedef enum

{

CS_None = 0,
CS_Autodetect,
CS_ETSI_EN302208,
CS_ETSI_EN302208_LBT,
CS_FCC,
CS_Special,
CS_China,
CS_Thailand,
CS_Brazil,
CS_SouthKorea,
CS_Peru,
CS_Singapore,
CS_Australia,
CS_Japan,
CS_Malaysia,
CS_Indonesia,
CS_Last

} tCommunicationStandard;

5.5 Antenna Modes

typedef enum

{

AM_All = 0,
AM_Antenna1 = 0x10,
AM_Antenna2,
AM_Antenna3,
AM_Antenna4,
AM_Last

} tAntennaMode;

5.6 ETSI Channel Switching Modes

typedef enum

{

EPCSW_Mode0 = 0,
EPCSW_Mode1,
EPCSW_Last

} tETSIPortChannelSwitchingMode;

5.7 Modulation Types

typedef enum
{

MT_DSB = 0,	// double-sideband modulation
MT_PRA,	// PR-ASK modulation
MT_SSB,	// single-sideband modulation (currently not supported)
MT_EXT,	// R1000 external double-sideband modulation (currently not supported)
MT_Last	

} tModulationType;

5.8 On/Off Flag

typedef enum
{

OOF_Off = 0x00
OOF_On = 0xFF

} tOnOffFlag;

5.9 Coming/Going Flag

typedef enum
{

CGF_Going = 0x00,
CGF_DataChanging = 0x80,
CGF_Coming = 0xFF

} tComingGoingFlag;

5.10 Start/Stop/Heartbeat Flag

typedef enum
{

SSHBF_Stop = 0x00,
SSHBF_Heartbeat = 0x80,
SSHBF_Start = 0xFF

} tStartStopHearbeatFlag;

5.11 LED Bar Control Flag

typedef enum

{

RLBCF_Start = 0,
RLBCF_Do,
RLBCF_Stop,
RLBCF_Last

} tRSSILedBarControlFlag;

5.12 Message Numbers

typedef enum

{

EMID_Error_NoFreeChannel = 0,
EMID_Error_Antenna1,
EMID_Error_Antenna2,
EMID_Error_Antenna3,
EMID_Error_Antenna4,
EMID_Error_Antenna5,
EMID_Error_Antenna6,
EMID_Error_Antenna7,
EMID_Error_Antenna8,
EMID_Error_ETSIPowerCheck1,
EMID_Error_ETSIPowerCheck2,
EMID_Error_ETSIPowerCheck3,
EMID_Error_ETSIPowerCheck4,
EMID_Error_ETSIPowerCheck5,
EMID_Error_ETSIPowerCheck6,
EMID_Error_ETSIPowerCheck7
EMID_Error_ETSIPowerCheck8
EMID_Error_KRAIOverCurrent1,
EMID_Error_KRAIOverCurrent
EMID_Error_KRAIOverCurrent3,
EMID_Error_KRAIOverCurrent4,
EMID_Error_KRAIOverCurrent5,
EMID_Error_KRAIOverCurrent6,
EMID_Error_KRAIOverCurrent7,
EMID_Error_KRAIOverCurrent8,
EMID_Error_Last

} tErrorMessageId;

5.13 Extended Result Flag

The Extended Result flag determines which information is transferred within the extended result data. If a bit is set, the corresponding information is transmitted.

```
// Macros for the access to the Extended Result Flag
#define ERF_MASK_A          0x01    // antenna
#define ERF_MASK_R          0x02    // RSSI value
#define ERF_MASK_T          0x04    // timestamp
#define ERF_MASK_PC         0x08    // protocol control word(s)
#define ERF_MASK_FRTP       0x10    // frequency and tag phase
#define ERF_MASK_RDBM       0x20    // RSSI value in dBm
#define ERF_MASK_TMS        0x40    // timestamp in ms
#define ERF_MASK_ALL        0x7F
```

5.14 Tag Error Codes

```
typedef enum
{
    TEC_NoError = 0x00,
    TEC_NoAnswer,
    TEC_AccessDenied,
    TEC_VerifyFailed,
    TEC_Unspecified,
    TEC_Other = 0x80,
    TEC_NotSupported = 0x81,
    TEC_InsufficientPriviliges = 0x82,
    TEC_MemoryLocked = 0x84,
    TEC_CryptoSuiteError = 0x85,
    TEC_CommandNotEncapsulated = 0x86,
    TEC_ResponseBufferOverflow = 0x87,
    TEC_SecurityTimeout = 0x88,
    TEC_InsufficientPower = 0x8B,
    TEC_NonSpecific = 0x8F
} tTagErrorCode;
```

5.15 Tag Memory Bank

typedef enum

{

TMB_RFU = 0,
TMB_EPC,
TMB_TID,
TMB_User

} tTagMemoryBank;

5.16 LED functions with ©KRAI antennas with configurable LEDs

typedef enum

{

KOF_Off = 0,
KOF_On,
KOF_RFOn,
KOF_AntennaError,
KOF_TagFound,
KOF_TagOperationSuccess,
KOF_ProtocolAccess,
KOF_Last

} tKRAIOutputFunction;

5.17 LED Mode

typedef enum

{

LM_Auto = 0,
LM_Off,
LM_Green,
LM_Red,
LM_Orange,
LM_Last

} tLEDMode;

5.18 I/O Card Types

typedef enum
{

IOCT_Autodetect = 0,
IOCT_CardNotUsed,
IOCT_RS232,
IOCT_RS485,
IOCT_Ethernet,
IOCT_CAN,
IOCT_GPIO,
IOCT_GPIORRUMOD,
IOCT_LED,
IOCT_FTDI,

} tIOCardType;

5.19 I/O Communication Protocols

typedef enum
{

IOCS_NotSpecified = 0,	
IOCS_KBP,	// *** out of use ***
IOCS_STDPRTCL,	// *** out of use ***
IOCS_KBRP,	// Kathrein reader protocol (KBRP)
IOCS_KBRPNHS,	// Kathrein reader protocol (KBRP) without handshake
IOCS_KBRPETH,	// Kathrein Reader protocol (KBRP) for Ethernet connections
IOCS_Last	

} tIOCardService;

6 Commands

6.1 System and Configuration Commands

6.1.1 GetMode

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 FM

I1	response ID low-byte
I2	response ID high-byte
FM	function mode; *5.4.3 Reader Function Mode, p. 12* Possible function modes: *CFM_NORMAL* *CFM_DIRECT*

6.1.2 SetMode

Command block structure:
I1 I2 FM

I1	command ID low-byte
I2	command ID high-byte
FM	function mode; *5.4.3 Reader Function Mode, p. 12* Possible function modes: *CFM_NORMAL* *CFM_DIRECT*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*

6.1.3 GetCommunication Standard

Command block structure:
I1 I2 FM

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 CS

I1	response ID low-byte
I2	response ID high-byte
CS	communication standard; see *5.4.4 Communication Standards, p. 13*

6.1.4 SetCommunication Standard

Command block structure:
I1 I2 CS

I1	command ID low-byte
I2	command ID high-byte
CS	communication standard; see *5.4.4 Communication Standards, p. 13*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoLicense* *RF_OutOfRange* *RF_AccessDenied*

6.1.5 GetPortPower

Command block structure:

I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 1 to 4)

Response block structure:

I1 I2 RF PN [PP]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoLicense* *RF_OutOfRange*
PN	antenna port number (in the range from 1 to 4)
PP	port power The power radiated by the antenna is returned if the result flag is *RF_NoError*. The command indicates the power radiation setting for the antenna port in accordance with the cable attenuation and antenna gain in ¼ dBm (erp).

6.1.6 SetPortPower

Command block structure:
I1 I2 PN PP

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 1 to 4)
PP	port power

The transmission power of an antenna port is specified in ¼ dBm (erp) (antenna radiated power).
If the value is zero, the antenna port is cut off. The setting range is between 0x18 (6 dBm (erp)) and 0x88 (34 dBm (erp)).

Response block structure:
I1 I2 RF PN

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoLicense* *RF_OutOfRange* *RF_AccessDenied*
PN	antenna port number (in the range from 1 to 4)

6.1.7 GetCarrierFollowUpTime

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 T1 T2

I1	response ID low-byte
I2	response ID high-byte
T1	carrier follow-up time, low-byte (in seconds)
T2	carrier follow-up time, high-byte (in seconds)

6.1.8 SetCarrierFollowUpTime

Command block structure:
I1 I2 T1 T2

I1	command ID low-byte
I2	command ID high-byte
T1	carrier follow-up time, low-byte (in seconds)
T2	carrier follow-up time, high-byte (in seconds)

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_AccessDenied* *RF_NonSpecified*

6.1.9 GetAntennaMode

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 AS AA

I1	response ID low-byte
I2	response ID high-byte
AS	antenna mode for synchronous commands
AA	antenna mode for asynchronous commands The following antenna modes are possible: *AM_All* *AM_Antenna1* *AM_Antenna2* *AM_Antenna3* *AM_Antenna4*

6.1.10 SetAntennaMode

Command block structure:
I1 I2 AS AA

I1	command ID low-byte
I2	command ID high-byte
AS	antenna mode for synchronous commands
AA	antenna mode for asynchronous commands *AM_All* *AM_Antenna1* *AM_Antenna2* *AM_Antenna3* *AM_Antenna4*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*

6.1.11 GetPortMultiplexSequenceAndExposureTime

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 PN ET1 ET2 [[PN ET1 ET2] ... [PN ET1 ET2]]

I1	response ID low-byte
I2	response ID high-byte
PN	antenna port number (in the range from 1 to 4)
ET1	exposure time low-byte (in milliseconds)
ET2	exposure time high-byte (in milliseconds)

Tip Exposure times are only taken into account with asynchronous reader commands.

6.1.12 SetPortMultiplexSequenceAndExposureTime

Command block structure:
I1 I2 PN ET1 ET2 [[PN ET1 ET2] ... [PN ET1 ET2]]

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 1 to 4)
ET1	exposure time low-byte (in milliseconds)
ET2	exposure time high-byte (in milliseconds)

It is possible to show 1 to 8 ports and their exposure times.

Tip Exposure times are only taken into account with asynchronous reader commands.

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoLicense* *RF_OutOfRange* *RF_AccessDenied*

6.1.13 GetCableLossAndAntennaGain

Command block structure:
I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 1 to 4)

Response block structure:
I1 I2 RF PN [CL AG]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoLicense*
PN	antenna port number (in the range from 1 to 4)
CL	cable loss in ¼ dB
AG	antenna gain in ¼ dBic

6.1.14 SetCableLossAndAntennaGain

Command block structure:
I1 I2 PN CL AG

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 1 to 4)
CL	cable loss in ¼ dB
AG	antenna gain in ¼ dBic

Response block structure:
I1 I2 RF PN

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoLicense* *RF_AccessDenied*
PN	antenna port number (in the range from 1 to 4)

6.1.15 GetETSIPortChannelList

Command block structure:
I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 0 to 4; 0 returns the global channel list – 1 to 4 return the list for the related antenna port)

Response block structure:
I1 I2 RF PN [CH...CH]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoLicense*
PN	antenna port number (in the range from 1 to 4)
CH	ETSI channel in the range from 1 to 15 A channel list is returned only if the result flag is *RF_NoError* and only if the list for the respective port is stored in the reader.

6.1.16 SetETSIPortChannelList

Command block structure:
I1 I2 PN [CH...CH]

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 0 to 4; 0 returns the global channel list – 1 to 4 return the list for the related antenna port)
CH	ETSI channel in the range from 1 to 15 1 to 16 channels can be passed.

Response block structure:
I1 I2 RF PN

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoLicense* *RF_AccessDenied*
PN	antenna port number (in the range from 0 to 4)

6.1.17 GetETSIPortChannelSwitchingMode

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 SM

I1	response ID low-byte
I2	response ID high-byte
SM	switching mode; see *5.4.6 ETSI Channel Switching Modes, p. 13* Possible switching modes: *EPCSW_Mode0* *EPCSW_Mode1*

6.1.18 SetETSIPortChannelSwitchingMode

Command block structure:
I1 I2 SM

I1	command ID low-byte
I2	command ID high-byte
SM	switching mode; see *5.4.6 ETSI Channel Switching Modes, p. 13* Possible switching modes: *EPCSW_Mode0* *EPCSW_Mode1*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*

6.1.19 GetProfileList

Command block structure:

I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:

I1 I2 RF [PL...PL]

I1	response ID low-byte
I2	response ID high-byte
PL	data bytes of the profile list The profile lists comprises 0 to n profile list entries that are sent one after the other. A profile list entry has the following structure: *PN PT...PT 00*

PN	profile number
PT	ASCII-coded profile description text
00	null byte as a terminating character

6.1.20 GetProfile

Command block structure:

I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:

I1 I2 RF PN

I1	response ID low-byte
I2	response ID high-byte
PN	profile number

6.1.21 SetProfile

Command block structure:
I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	profile number

Response block structure:
I1 I2 RF RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*

6.1.22 GetModulationType

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 MT

I1	response ID low-byte
I2	response ID high-byte
MT	modulation type; see *5.4.7 Modulation Types, p. 14* Possible modulation types: *MT_DSB* *MT_PRA*

6.1.23 SetModulationType

Command block structure:
I1 I2 MT

I1	command ID low-byte
I2	command ID high-byte
MT	modulation type; see *5.4.7 Modulation Types, p. 14* Possible modulation types: *MT_DSB* *MT_PRA*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*

6.1.24 GetExtendedResultFlag

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 EF

I1	response ID low-byte
I2	response ID high-byte
EF	extended result flag; see *5.4.13 Extended Result Flag, p. 16*

6.1.25 SetExtendedResultFlag

Command block structure:
I1 I2 EF

I1	command ID low-byte
I2	command ID high-byte
EF	extended result flag; see *5.4.13 Extended Result Flag, p. 16*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*

6.1.26 GetErrorStatus

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 E1 E2 E3 E4

I1	response ID low-byte
I2	response ID high-byte
E1	error flag register LSB
E2	error flag register 2nd byte
E3	error flag register 3rd byte
E4	error flag register MSB

The number of a set bit in the error flag register states the corresponding message number; see *5.4.12 Message Numbers, p. 15*

6.1.27 GetDefaultParameterSet

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 PS

I1	response ID low-byte
I2	response ID high-byte
PS	parameter set number

6.1.28 SetDefaultParameterSet

Command block structure:
I1 I2 PS

I1	command ID low-byte
I2	command ID high-byte
PS	parameter set number (in the range from 0 to 7; > 7 sets the active parameter set as the default value)

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_AccessDenied*

6.1.29 GetActiveParameterSet

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 PS

I1	response ID low-byte
I2	response ID high-byte
PS	parameter set number

6.1.30 SetActiveParameterSet

Command block structure:
I1 I2 PS

I1	command ID low-byte
I2	command ID high-byte
PS	parameter set number (in the range from 0 to 7)

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*

6.1.31 SaveActiveParameterSet

Saves the parameter of the commands *SetCommStandard*, *SetPortPower*, *SetCarrierFollowUpTime*, *SetPortMultiplexSequenceAndExposureTime*, *SetCableLossAndAntennaGain*, *SetETSIPortChannelList*, *SetETSIPortChannelSwitchingMode*, *SetProfile*, *SetModulationType*, *SetExtendedResultFlag*, *SetSelSessionAndTarget*, *SetInitialQValue*, *SetMaxAirCommErrors* and *SetASyncObservedListParameters* in the internal EEPROM. The parameters saved are automatically used again after a restart.

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError*

6.1.32 GetParameterByConfigId

Command block structure:
I1 I2 P1 P2 P3 P4

I1	command ID low-byte
I2	command ID high-byte
P1	config ID LSB
P2	config ID 2nd byte
P3	config ID 3rd byte
P4	config ID MSB

Response block structure:
I1 I2 RF [P1 P2 P3 P4 [V1 [... [V4]]]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
P1	config ID LSB
P2	config ID 2nd byte
P3	config ID 3rd byte
P4	config ID MSB
V1	parameter value LSB
V2	parameter value 2nd byte
V3	parameter value 3rd byte
V4	parameter value MSB

6.1.33 SetParameterByConfigId

Command block structure:
I1 I2 P1 P2 P3 P4 V1 [V2 [V3 [V4]]]

I1	command ID low-byte
I2	command ID high-byte
P1	parameter ID LSB
P2	parameter ID 2nd byte
P3	parameter ID 3rd byte
P4	parameter ID MSB
V1	parameter value LSB
V2	parameter value 2nd byte
V3	parameter value 3rd byte
V4	parameter value MSB

Response block structure:
I1 I2 RF [P1 P2 P3 P4]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*
P1	parameter ID LSB
P2	parameter ID 2nd byte
P3	parameter ID 3rd byte
P4	parameter ID MSB

6.1.34 GetIOCardHwConfig

Command block structure:
I1 I2 CN

I1	command ID low-byte
I2	command ID high-byte
CN	I/O card number (in the range from 0 to 15)

Response block structure:
I1 I2 RF [CN [CT CS CD...CD]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_OutOfRange*
CN	I/O card number
CT	I/O card type
CS	I/O communications protocol
CD	configuration data Configuration data are only returned if the result flag is *RF_NoError*. The explanation of the structure of the configuration data is not part of this document.

6.1.35 SetIOCardHwConfig

Command block structure:
I1 I2 CN CT CS CD ... CD

I1	command ID low-byte
I2	command ID high-byte
CN	I/O card number (in the range from 0 to 15)
CT	I/O card type
CS	I/O communications protocol
CD	configuration data

Response block structure:
I1 I2 RF [CN]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied* *RF_NonSpecified*
CN	I/O card number

6.1.36 GetIOCardProtocolConfig

Command block structure:
I1 I2 CN

I1	command ID low-byte
I2	command ID high-byte
CN	I/O card number (in the range from 0 to 15)

Response block structure:
I1 I2 RF [CN [CD...CD]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
CN	I/O card number
CD	configuration data Configuration data are only returned if the result flag is *RF_NoError*. The explanation of the structure of the configuration data is not part of this document.

6.1.37 SetIOCardProtocolConfig

Command block structure:
I1 I2 CN CD ... CD

I1	command ID low-byte
I2	command ID high-byte
CN	I/O card number (in the range from 0 to 15)
CD	configuration data

Response block structure:
I1 I2 RF [CN]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied* *RF_NonSpecified*
CN	I/O card number

6.1.38 GetIOCardExtendedInfo

Command block structure:
I1 I2 CN

I1	command ID low-byte
I2	command ID high-byte
CN	I/O card number (in the range from 0 to 15)

Response block structure:
I1 I2 RF [CN [EI...EI]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_OutOfRange* *RF_NonSpecified*
CN	I/O card number
EI	extended module information

6.1.39 GetMillisecondsTimestamp

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 T1 T2 T3 T4

I1	response ID low-byte
I2	response ID high-byte
T1	UTC timestamp LSB
T2	UTC timestamp 2nd byte
T3	UTC timestamp 3rd byte
T4	UTC timestamp MSB

The time stamp indicates the milliseconds since the last switch-on of the RFID reader.

6.1.40 GetTime

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 T1 T2 T3 T4

I1	response ID low-byte
I2	response ID high-byte
T1	UTC timestamp LSB
T2	UTC timestamp 2nd byte
T3	UTC timestamp 3rd byte
T4	UTC timestamp MSB

The time stamp gives the UTC time in seconds since 1.1.1970 0:00 hours.

6.1.41 SetTime

Command block structure:
I1 I2 T1 T2 T3 T4

I1	command ID low-byte
I2	command ID high-byte
T1	UTC timestamp LSB
T2	UTC time stamp 2nd byte
T3	UTC timestamp 3rd byte
T4	UTC timestamp MSB

The time stamp gives the UTC time in seconds since 1.1.1970 0:00 hours.

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_AccessDenied*

6.1.42 GetTemperature

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 T1 T2

I1	response ID low-byte
I2	response ID high-byte
T1	reader internal temperature low-byte
T2	reader internal temperature high-byte

The internal temperature of the reader is transmitted as a signed 16-bit integer value with a resolution of 1/100°C.

6.1.43 GetSwVersion

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 H1 H2 N1 N2 U1 U2 B1 B2 B3 B4

I1	response ID low-byte
I2	response ID high-byte
H1	major version number low-byte
H2	major version number high-byte
N1	minor version number low-byte
N2	minor version number high-byte
U1	revision number low-byte
U2	revision number high-byte
B1	build number LSB
B2	build number 2nd byte
B3	build number 3rd byte
B4	build number MSB

Version structure:

Software version: *major version no. point minor version no. point revision no. (***Build**: build number) For example:

1.22.00 (Build: 1914)

The minor version number and the revision number have *two digits* as a minimum.

6.1.44 GetHwVersion

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 CV .. CV 00 PV .. PV 00 [[C0 .. C0 00] .. [CF .. CF 00]]

I1	response ID low-byte
I2	response ID high-byte
CV	identifier text (ASCII-coded) for the CPU module
PV	identifier text (ASCII-coded) for the PA module
00	separating and terminating character
CO to CF	identifier text (ASCII-coded) for the I/O modules 0 to 15 (only as far as available and as far as the modules support this function)

Version structure:

CPU module: *identifier text*
PA module: *identifier text*

For example:
CPU module: *136B211B*
PA module: *136B224B*

6.1.45 GetLicenseKey

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 LK ... LK 00

I1	response ID low-byte
I2	response ID high-byte
LK	license key (ASCII-coded) A license key has the format: *xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx*
PV	identifier text (ASCII-coded) for the PA module
00	terminating character

6.1.46 SetLicenseKey

Command block structure:
I1 I2 LK ... LK 00

I1	command ID low-byte
I2	command ID high-byte
LK	license key (ASCII-coded) A license key has the format: *xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx-xxxxx*
00	terminating character

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_WrongLicenseKey*

6.1.47 GetMaxPowerLimits

Command block structure:
I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range of 1 to 4)

Response block structure:
I1 I2 RF PN [P1 P2 P3]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoLicense* *RF_OutOfRange* *RF_NonSpecified*
PN	antenna port number (in the range of 1 to 4)
P1	maximum radiated power of the antenna (in ¼ dBm (erp))
P2	maximum power at the antenna port (in ¼ dBm)
P3	maximum power at the antenna connector of the reader (in ¼ dBm)

6.1.48 SetConfigPassword

Command block structure:
I1 I2 PO .. PO 00 PN .. PN 00

I1	command ID low-byte
I2	command ID high-byte
PO	old configuration password
00	password terminator
PN	new configuration password
00	password terminator

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied* *RF_NonSpecified*

6.1.49 EnterConfigPassword

Command block structure:
I1 I2 CP .. CP 00

I1	command ID low-byte
I2	command ID high-byte
CP	configuration password
00	password terminator

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied* *RF_NonSpecified*

6.1.50 LoadFactoryDefaults

Command block structure:
I1 I2 [PS]

I1	command ID low-byte
I2	command ID high-byte
PS	parameter set number (for values between 0 and 7: >7 the factory defaults are loaded into the active set of parameters). If no number is given for the set of parameters, all the parameters are set back to factory default.

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_AccessDenied*

Tip If no number is given for the set of parameters, all the parameters are overwritten with the factory defaults.

6.1.51 FirmwareUpdatePrepare

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoLicense* *RF_AccessDenied* *RF_NonSpecified*

6.1.52 FirmwareUpdateTransmitData

Command block structure:
I1 I2 FD [FD ... FD]

I1	command ID low-byte
I2	command ID high-byte
FD	firmware data

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_FirmwareRejected* *RF_NoLicense*

6.1.53 FirmwareUpdateFlash

Command block structure:
I1 I2 [5A A5]

I1	command ID low-byte
I2	command ID high-byte
[5A A5]	identifier to load the default configuration after a reboot **Tip** If the two bytes 0x5A and 0xA5 are attached to the command ID, the reader loads the default configuration after a restart.

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_FirmwareRejected* *RF_NoLicense*

6.1.54 RebootSystem

Command block structure:
I1 I2 5A A5

I1	command ID low-byte
I2	command ID high-byte
5A	constant
A5	constant

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*

6.1.55 GetLEDMode

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF LM

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NonSpecified*
LM	LED mode; see *5.4.17 LED Mode, p. 17*

6.1.56 SetLEDMode

Command block structure:
I1 I2 LM

I1	command ID low-byte
I2	command ID high-byte
LM	LED mode; see *5.4.17 LED Mode, p. 17*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NonSpecified*

6.1.57 ActivateBuzzer

Command block structure:
I1 I2 T1 T2

I1	command ID low-byte
I2	command ID high-byte
T1	buzzer time LSB
T2	buzzer time MSB

The buzzer time sets the length of time in milliseconds for which the buzzer is activated. A value of *0* switches the buzzer off, a value of *0xFFFF* switches the buzzer on permanently.

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*

6.1.58 GetDeviceSerialNumber

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF [[ST ... ST] 00 [HT ... HT] 00]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NonSpecified*
ST	text serial number
HT	text hardware version

6.1.59 GetDeviceTypeNumber

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF [[DT ... DT] 00]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NonSpecified*
DT	text device type number

6.2 Commands in Normal Mode and Direct Mode

6.2.1 GetSelSessionAndTarget

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 SL SN TA

I1	response ID low-byte
I2	response ID high-byte
SL	query Sel (0 to 3; see *EPCglobal Specification V 2.0.0 p. 76*)
SN	query Session (0 to 3; *EPCglobal Specification V 2.0.0 p. 76*)
TA	query Target (0: A; 1: B; see *EPCglobal Specification V 2.0.0 p. 76*)

6.2.2 SetSelSessionAndTarget

Command block structure:
I1 I2 SL SN TA

I1	command ID low-byte
I2	command ID high-byte
SL	query Sel (0 to 3; see *EPCglobal Specification V 2.0.0 p. 76*)
SN	query Session (0 to 3; see *EPCglobal Specification V 2.0.0 p. 76*)
TA	query Target (0: A; 1: B; see *EPCglobal Specification V 2.0.0 p. 76*)

Tip In normal mode, the parameter Q*uery Sel* affects only the commands *...GetEPCs*, *...Any* and *SyncWriteEPC-ToSingleTag,* and only if at least one select filter is active. The parameter *Query Target* does not affect the *...Specific* commands.

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*

6.2.3 GetInitialQ

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 QV

I1	response ID low-byte
I2	response ID high-byte
QV	Q value

6.2.4 SetInitialQ

Command block structure:
I1 I2 QV

I1	command ID low-byte
I2	command ID high-byte
QV	Q value

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_AccessDenied*

6.2.5 GetMaxAirCommErrors

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 AE

I1	response ID low-byte
I2	response ID high-byte
AE	MaxAirCommErrors value

6.2.6 SetMaxAirCommErrors

Command block structure:
I1 I2 AE

I1	command ID low-byte
I2	command ID high-byte
AE	MaxAirCommErrors value

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_AccessDenied*

6.2.7 GetASyncObservedListParameters

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 P1 P2 P3

I1	response ID low-byte
I2	response ID high-byte
P1	glimpsed timeout count
P2	observed threshold count
P3	observed timeout count

6.2.8 SetASyncObservedListParameters

Command block structure:
I1 I2 P1 P2 P3

I1	command ID low-byte
I2	command ID high-byte
P1	glimpsed timeout count (in the range from 0 to 255)
P2	observed threshold count (in the range from 0 to 255)
P3	observed timeout count (in the range from 0 to 255)

Response block structure:
I1 I2 P1 P2 P3

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_AccessDenied*

6.2.9 GetSelectFilterOnOff

Command block structure:
I1 I2 FN

I1	command ID low-byte
I2	command ID high-byte
FN	filter number (in the range from 0 to 31)

Response block structure:
I1 I2 RF FN [OO]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
FN	filter number
OO	on/off flag

6.2.10 SetSelectFilterOnOff

Command block structure:
I1 I2 FN OO

I1	command ID low-byte
I2	command ID high-byte
FN	filter number (in the range from 0 to 31)
OO	on/off flag

Response block structure:
I1 I2 RF FN

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
FN	filter number

6.2.11 GetSelectFilterData

Command block structure:
I1 I2 FN

I1	command ID low-byte
I2	command ID high-byte
FN	filter number (in the range from 0 to 31)

Response block structure:
I1 I2 RF FN [OO TA AC MB P1 P2 P3 P4 ML [MD ... MD]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
FN	filter number
OO	on/off flag
TA	Target (see *EPCglobal Specification V 2.0.0 p. 73*)
AC	action (see *EPCglobal Specification V 2.0.0 p. 73*)
MB	memory bank (see *EPCglobal Specification V 2.0.0 p. 44*)
P1	bit pointer LSB
P2	bit pointer 2nd byte
P3	bit pointer 3rd byte
P4	bit pointer MSB
ML	masked length in bits
MD	masked data

6.2.12 SetSelectFilterData

Command block structure:
I1 I2 FN OO TA AC MB P1 P2 P3 P4 ML [MD ... MD]

I1	command ID low-byte
I2	command ID high-byte
FN	filter number (in the range from 0 to 31)
OO	on/off flag
TA	Target (see *EPCglobal Specification V 2.0.0 p. 73*)
AC	action (see *EPCglobal Specification V 2.0.0 p. 73*)
MB	memory bank (see *EPCglobal Specification V 2.0.0 p. 44*)
P1	bit pointer LSB
P2	bit pointer 2nd byte
P3	bit pointer 3rd byte
P4	bit pointer MSB
ML	masked length in bits
MD	masked data

Tip The bytes of the masked data are transmitted over the air interface in the sequence of their issue from the MSB to the LSB. If the masked data end within a byte with a masked length that is not a multiple of eight, the last bits of the lower range of the last masked data byte are used. Thus, if for instance five bits remain to be transmitted, bits B4 to B0 of the last masked byte are transmitted over the air interface in this sequence.

Response block structure:
I1 I2 RF FN

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
FN	filter number

6.2.13 GPIOGetIOData

Command block structure:
I1 I2 CN

I1	command ID low-byte
I2	command ID high-byte
CN	GPIO card number (in the range from 0 to 15)

Response block structure:
I1 I2 RF [CN [IP1 IP2 IL1 IL2 ID1 ID2 OL1 OL2 OP1 OP2]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
CN	GPIO card number
IP1	LSB physical inputs
IP2	MSB physical inputs
IL1	LSB logical inputs
IL2	MSB logical inputs
ID1	debounced LSB logical inputs
ID2	debounced MSB logical inputs
OL1	LSB logical outputs
OL2	MSB logical outputs
OP1	LSB physical outputs
OP2	MSB physical outputs

Tip Since setting the GPIO outputs and reading the GPIO inputs have a higher priority than executing this command, inconsistencies may arise between the physical and logical inputs and outputs during the course of transmission.

6.2.14 GPIOSetOutput

Command block structure:
I1 I2 CN ON T1 T2

I1	command ID low-byte
I2	command ID high-byte
CN	GPIO card number (in the range from 0 to 15)
ON	output number (starting at 0)
T1	LSB activation time, in milliseconds
T2	MSB activation time, in milliseconds

Tip If the activation time is set to (0xFFFF) milliseconds, the output is permanently activated. To switch an output off before expiry of the activation time, transmit a command with the activation time *0*.

Response block structure:
I1 I2 RF [CN ON]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
CN	GPIO card number
ON	output number

6.2.15 GPIOAddActionToActionlist

Command block structure:
I1 I2 AL AD ... AD

I1	command ID low-byte
I2	command ID high-byte
AL	action list (in the range from 0 to 127)
AD	action data

Tip The *Action data* range consists of a command ID and its associated parameters. At least 2 bytes of *Action data* must be transmitted.

Response block structure:
I1 I2 RF [AL]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
AL	action list

6.2.16 GPIOClearActionlist

Command block structure:
I1 I2 AL

I1	command ID low-byte
I2	command ID high-byte
AL	action list (in the range from 0 to 127)

Response block structure:
I1 I2 RF [AL]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
AL	action list

6.2.17 GPIOAssignInputToActionlist

Command block structure:
I1 I2 CN IN EG AL

I1	command ID low-byte
I2	command ID high-byte
CN	GPIO card number (in the range from 0 to 15)
IN	input number (starting at 0)
EG	edge (0: negative edge; <>0: positive edge)
AL	action list (in the range 0 to 127; from a value of 128 the edge is not assigned to an action list)

Response block structure:
I1 I2 RF [CN IN EG]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
CN	GPIO card number
IN	input number
EG	edge

6.2.18 SetLED

Command block structure:
I1 I2 00 LN T1 T2

I1	command ID low-byte
I2	command ID high-byte
00	reserved (must be 0)
LN	LED number (from 1 to 12)
T1	LSB activation time, in milliseconds
T2	MSB activation time, in milliseconds

Tip If the activation time is set to (0xFFFF) milliseconds, the LED is permanently activated. To switch an LED off before expiry of the activation time, transmit a command with the activation time *0*.

Response block structure:
I1 I2 RF 00 LN

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange*
00	reserved
LN	LED number

6.2.19 KRAIInitializePort

Command block structure:
I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range of 0 to 4) An antenna port number *0* performs a KRAI initialisation on all antenna ports.

Response block structure:
I1 I2 RF [PN]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoAntenna* *RF_NonSpecified*
PN	antenna port number

6.2.20 KRAIGetPortInfo

Command block structure:
I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range of 1 to 4)

Response block structure:
I1 I2 RF [PN [IL IH SC]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoAntenna* *RF_NonSpecified*
PN	antenna port number (in the range of 1 to 4)
IL	antenna ID low-byte
IH	antenna ID high-byte
SC	number of sub-antennas

6.2.21 KRAISetLED

Command block structure:
I1 I2 PN LN TL TH

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range of 1 to 4)
LN	number of the LED (in the range of 0 to 7)
TL	switch-on time low-byte
TH	switch-on time high-byte

Tip A switch-on time of (0xFFFF) milliseconds leads to a permanent *ON* for the specified LED.

Response block structure:
I1 I2 RF [PN LN]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NonSpecified*
PN	antenna port number
LN	number of the LED

6.3 Commands in Normal Mode

6.3.1 SyncGetEPCs

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En]]
...
[[ERD] EW [E1 ... En]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag; see *5.4.13 Extended Result Flag, p. 16*
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB

6.3.2 SyncBulkGetEPCs

Command block structure:
I1 I2 T1 T2

I1	command ID low-byte
I2	command ID high-byte
T1	bulk read time low-byte (in milliseconds)
T2	bulk read time high-byte (in milliseconds)

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En]]
...
[[ERD] EW [E1 ... En]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB

6.3.3 SyncReadDataAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44)*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to read

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]
...
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code; see *5.4.14 Tag Error Codes, p. 16*
WC	number of words to read
D1	response data LSB
...	
Dx	response data MSB

6.3.4 SyncReadDataSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to read

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]
...
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
WC	number of EPC words
D1	response data LSB
...	
Dx	response data MSB

6.3.5 SyncReadDataUntilEndOfBankAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; (see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to read

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]
...
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
WC	number of words to read
D1	response data LSB
...	
Dx	response data MSB

6.3.6 SyncReadDataUntilEndOfBankSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to read

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]
...
[[ERD] EW [E1 ... En] EC [WC D1 ... Dx]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
WC	number of words to read
D1	response data LSB
...	
Dx	response data MSB

6.3.7 SyncWriteDataAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to write
DL	word to write low-byte
DH	word to write high-byte

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.8 SyncWriteDataSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH]

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to write
DL	word to write low-byte
DH	word to write high-byte

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.9 SyncWriteMaskedDataAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH] ML MH [ML MH ... ML MH]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to write
DL	word to write low-byte
DH	word to write high-byte
ML	write mask low-byte
MH	write mask high-byte

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.10 SyncWriteMaskedDataSpecific

Command block structure:

I1 I2 EW [E1 ... En] W1 W2 W3 W4 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH] ML MH [ML MH ... ML MH]

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to write
DL	word to write low-byte
DH	word to write high-byte
ML	write mask low-byte
MH	write mask high-byte

Response block structure:

I1 I2 RF [EF]

[[ERD] EW [E1 ... En] EC]

...

[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.11 SyncBlockEraseAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to erase

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.12 SyncBlockEraseSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to erase

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
EC	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.13 SyncLockAny

Command block structure:
I1 I2 W1 W2 W3 W4 L1 L2 L3

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
L1	payload LSB
L2	payload 2nd byte
L3	payload MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.14 SyncLockSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 L1 L2 L3

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
L1	payload LSB
L2	payload 2nd byte
L3	payload MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.15 SyncKillAny

Command block structure:
I1 I2 W1 W2 W3 W4 RC

I1	command ID low-byte
I2	command ID high-byte
W1	kill password LSB
W2	kill password 2nd byte
W3	kill password 3rd byte
W4	kill password MSB
RC	recommissioning bits; see *EPCglobal Specification V 1.2.0 p. 68*

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.16 SyncKillSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 RC

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
RC	recommissioning bits; see *EPCglobal Specification V 1.2.0 p. 68*

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
EC	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.17 SyncBlockPermalockAny

Command block structure:
I1 I2 W1 W2 W3 W4 00 RL MB P1 P2 P3 P4 BR [LL LH ... [LL LH]]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
00	reserved always 0x00
RL	read/lock; see *EPCglobal Specification V 2.0.0 p. 94*
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
BR	block range; see *EPCglobal Specification V 2.0.0 p. 95*
LL	lock data LSB
LH	lock data MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [BR L1 ... Lx]]
...
[[ERD] EW [E1 ... En] EC [BR L1 ... Lx]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
BR	block range
L1	lock data LSB
...	
Lx	lock data MSB

6.3.18 SyncBlockPermalockSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 00 RL MB P1 P2 P3 P4 BR [LLLH ... [LL LH]]

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
00	reserved always 0x00
RL	read/lock; see *EPCglobal Specification V 2.0.0 p. 94*
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
BR	block range; see *EPCglobal Specification V 2.0.0 p. 95*
LL	lock data LSB
LH	lock data MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [BR L1 ... Lx]]
...
[[ERD] EW [E1 ... En] EC [BR L1 ... Lx]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words

EC	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
BR	block range
L1	lock data LSB
...	
Lx	lock data MSB

6.3.19 SyncAuthenticateAny

Command block structure:
I1 I2 W1 W2 W3 W4 CS L1 L2 [D1 ... Dn]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
CS	crypto suite
L1	number of bits in the message (LSB)
L2	number of bits in the message (MSB)
D1	message first byte
...	
Dn	message last byte

Order of sending: The first bit to be transmitted is located in the MSB of the first byte (D1) of the message data.

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [L1 L2 [D1 ... Dn]]]
...
[[ERD] EW [E1 ... En] EC [L1 L2 [D1 ... Dn]]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode* *RF_NonSpecified*

EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
L1	number of bits in the message (LSB)
L2	number of bits in the message (MSB)
D1	response message first byte
...	
Dn	response message last byte

Order of receiving: The first bit to be received is located in the MSB of the first byte (D1) of the message data.

6.3.20 SyncAuthenticateSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 CS L1 L2 [D1 ... Dn]

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
CS	crypto suite
L1	number of bits in the message (LSB)
L2	number of bits in the message (MSB)
D1	message first byte
...	
Dn	message last byte

Order of sending: The first bit to be transmitted is located in the MSB of the first byte (D1) of the message data.

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC [L1 L2 [D1 ... Dn]]]
...
[[ERD] EW [E1 ... En] EC [L1 L2 [D1 ... Dn]]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode* *RF_NonSpecified*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
L1	number of bits in the message (LSB)
L2	number of bits in the message (MSB)
D1	message first byte
...	
Dn	message last byte

Order of receiving: The first bit to be received is located in the MSB of the first byte (D1) of the message data.

6.3.21 SyncUntraceableAny

Command block structure:
I1 I2 W1 W2 W3 W4 UB EM TM UM RG

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
UB	U-Bit in XPC_W; see *EPCglobal Specification V 2.0.0 p. 107*
EM	hide EPC; see *EPCglobal Specification V 2.0.0 p. 107*
TM	hide TID; see *EPCglobal Specification V 2.0.0 p. 107*
UM	hide user memory; see *EPCglobal Specification V 2.0.0 p. 107*
RG	operating range; see *EPCglobal Specification V 2.0.0 p. 107*

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.22 SyncUntraceableSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 UB EM TM UM RG

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
UB	U-Bit in XPC_W1; see *EPCglobal Specification V 2.0.0 p. 107*
EM	hide EPC; see *EPCglobal Specification V 2.0.0 p. 107*
TM	hide TID; see *EPCglobal Specification V 2.0.0 p. 107*
UM	hide user memory; see *EPCglobal Specification V 2.0.0 p. 107*
RG	operating range; see *EPCglobal Specification V 2.0.0 p. 107*

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.23 SyncWriteEPCSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 NW [N1 ... Nn]

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
NW	number of EPC words of the new EPC (in the range from 0 to 31)
N1	new EPC LSB
...	
Nn	new EPC MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.24 SyncWriteEPCToSingleTag

Command block structure:
I1 I2 W1 W2 W3 W4 EW E1 ... En

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
NW	number of EPC words of the new EPC (in the range from 0 to 31)
N1	new EPC LSB
...	
Nn	new EPC MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW E1 ... En EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoStandard* *RF_MoreThanOneTagInField* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.25 SyncNXPG2XReadProtectAny

Command block structure:
I1 I2 W1 W2 W3 W4

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.26 SyncNXPG2XReadProtectSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.27 SyncNXPG2XResetReadProtectAny

Command block structure:
I1 I2 W1 W2 W3 W4

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.28 SyncNXPG2XResetReadProtectSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.29 SyncNXPG2XChangeEASAny

Command block structure:
I1 I2 W1 W2 W3 W4 AB

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
AB	EAS alarm bit

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.30 SyncNXPG2XChangeEASSpecific

Command block structure:
I1 I2 EW [E1 ... En] W1 W2 W3 W4 AB

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (in the range from 0 to 31)
E1	EPC LSB
...	
En	EPC MSB
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
AB	EAS alarm bit

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En] EC]
...
[[ERD] EW [E1 ... En] EC]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.31 SyncNXPG2XEASAlarm

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF [EF]
[[AP] [RS] [T1 T2 T3 T4] 00]
...
[[AP] [RS] [T1 T2 T3 T4] 00]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError (receive alarm code)* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
AP	antenna port
EW	number of EPC words
RS	RSSI value
T1	timestamp LSB
T2	timestamp 2nd byte
T3	timestamp 3rd byte
T4	timestamp MSB
00	0x00 place holder

6.3.32 SyncRSSILedBarSpecific

Command block structure:
I1 I2 EW [E1 ... En] CF [PN]

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
CF	control flag (0: start; 1: do; 2: stop)
PN	antenna port number

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En]]
...
[[ERD] EW [E1 ... En]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB

6.3.33 ASyncGetRawEPCs

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag; see *5.4.10 Start/Stop/Heartbeat Flag, p. 14*

Asynchronous response block structure:
I1 I2 EF [ERD] EW [E1 ... En]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB

6.3.34 ASyncGetEPCs

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En]

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF); see *5.4.9 Coming/Going Flag, p. 14*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB

6.3.35 ASyncReadDataAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to read

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_OutOfRange* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC [WC D1 ... Dx]

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
WC	number of words to read
D1	response data LSB
...	
Dx	response data MSB

6.3.36 ASyncReadDataUntilEndOfBankAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to read

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_OutOfRange* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC [WC D1 ... Dx]

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF); see *5.4.9 Coming/Going Flag, p. 14*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
WC	number of words to read
D1	response data LSB
...	
Dx	response data MSB

6.3.37 ASyncWriteDataAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to write
DL	word to write low-byte
DH	word to write high-byte

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_OutOfRange* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.38 ASyncWriteMaskedDataAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH] ML MH [ML MH ... ML MH]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to write
DL	word to write low-byte
DH	word to write high-byte
ML	write mask low-byte
MH	write mask high-byte

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_OutOfRange* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.39 ASyncBlockEraseAny

Command block structure:
I1 I2 W1 W2 W3 W4 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
W1	kill password LSB
W2	kill password 2nd byte
W3	kill password 3rd byte
W4	kill password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to erase

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.40 ASyncLockAny

Command block structure:
I1 I2 W1 W2 W3 W4 L1 L2 L3

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
L1	payload LSB
L2	payload 2nd byte
L3	payload MSB

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.41 ASyncKillAny

Command block structure:
I1 I2 W1 W2 W3 W4 RC

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
RC	recomissioning bits; see *EPCglobal Specification V 1.2.0 p. 68*

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.42 ASyncBlockPermalockAny

Command block structure:
I1 I2 W1 W2 W3 W4 00 RL MB P1 P2 P3 P4 BR [LL LH ... [LL LH]]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
00	reserved → always 0x00
RL	read/lock; see *EPCglobal Specification V 2.0.0 p. 94*
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	block word pointer LSB
P2	block word pointer 2nd byte
P3	block word pointer 3rd byte
P4	block word pointer MSB
BR	block range; see *EPCglobal Specification V 2.0.0 p. 95*
LL	lock data LSB
LH	lock data MSB

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_OutOfRange* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC [BR L1 ... Lx]

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB

...	
En	EPC MSB
EC	tag error code
BR	block range
L1	lock data LSB
...	
Lx	lock data MSB

6.3.43 ASyncAuthenticateAny

Command block structure:
I1 I2 W1 W2 W3 W4 CS L1 L2 [D1 ... Dn]

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
CS	crypto suite
L1	number of bits in the message (LSB)
L2	number of bits in the message (MSB)
D1	message of the first byte
...	
Dn	message of the last byte

Order of sending: The first bit to be transmitted is located in the MSB of the first byte (D1) of the message data.
Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_OutOfRange* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC [L1 L2 [D1 ... Dn]]

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code
L1	number of bits in the response message (LSB)
L2	number of bits in the response message (MSB)
D1	response message of the first byte
...	
Dn	response message of the last byte

6.3.44 ASyncUntraceableAny

Command block structure:
I1 I2 W1 W2 W3 W4 UB EM TM UM RG

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
UB	U-bit in XPC_W1; see *EPCglobal Specification V 2.0.0 p. 107*
EM	hide EPC; see *EPCglobal Specification V 2.0.0 p. 107*
TM	hide TID; see *EPCglobal Specification V 2.0.0 p. 107*
UM	hide user memory; see *EPCglobal Specification V 2.0.0 p. 107*
RG	operating range; see *EPCglobal Specification V 2.0.0 p. 107*

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte

RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_OutOfRange* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.45 ASyncNXPG2XReadProtectAny

Command block structure:
I1 I2 W1 W2 W3 W4

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.46 ASyncNXPG2XResetReadProtectAny

Command block structure:
I1 I2 W1 W2 W3 W4

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.47 ASyncNXPG2XChangeEASAny

Command block structure:
I1 I2 W1 W2 W3 W4 AB

I1	command ID low-byte
I2	command ID high-byte
W1	access password LSB
W2	access password 2nd byte
W3	access password 3rd byte
W4	access password MSB
AB	EAS alarm bit

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [ERD] EW [E1 ... En] EC

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
EC	tag error code

6.3.48 ASyncNXPG2XEASAlarm

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF FL

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoStandard* *RF_WrongCommandFunctionMode*
FL	start/stop/heartbeat flag

Asynchronous response block structure (coming message):
I1 I2 CG EF [AP] [RS] [T1 T2 T3 T4] 00

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (coming message: 0xFF)
EF	extended result flag
AP	antenna port
RS	RSSI value
T1	time stamp LSB
T2	time stamp 2nd byte
T3	time stamp 3rd byte
T4	time stamp MSB
00	(0x00: wildcard)

6.3.49 ASyncStopCommand

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_WrongCommandFunctionMode*

6.3.50 ASyncGetTagRate

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF [R1 R2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_WrongCommandFunctionMode*
R1	tag rate low-byte
R2	tag rate high-byte

6.4 Commands in Direct Mode

6.4.1 GetAntennaList

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 PM

I1	response ID low-byte
I2	response ID high-byte
PM	antenna port mask (each 1 bit stands for a configured antenna)

6.4.2 SetAntenna

Command block structure:
I1 I2 PN

I1	command ID low-byte
I2	command ID high-byte
PN	antenna port number (in the range from 1 to 4)

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_WrongCommandFunctionMode*

6.4.3 SetFrequency

Command block structure:
I1 I2 F1 F2 F3

I1	command ID low-byte
I2	command ID high-byte
F1	frequency LSB
F2	frequency 2nd byte
F3	frequency MSB

The frequency is given as a multiple of 1 kHz. Example for 865.7 MHz = 865700 kHz = 0x0D35A4 kHz: *I1 I2 A4 35 0D*

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_WrongCommandFunctionMode*

6.4.4 GetNoiseValue

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF [NV]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoAntenna* *RF_NoFrequency* *RF_WrongCommandFunctionMode*
NV	noise value
N1	noise value in 1/100 dBm LSB (only Generation 3 readers)
N2	noise value in 1/100 MSB (only Generation 3 readers)

6.4.5 GetETSILBTReferenceNoiseValue

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RV

I1	response ID low-byte
I2	response ID high-byte
RV	LBT reference noise value

6.4.6 CarrierOnOff

Command block structure:
I1 I2 OO

I1	command ID low-byte
I2	command ID high-byte
OO	on/off flag

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoAntenna* *RF_NoFrequency* *RF-NoProfile* *RF_AntennaError* *RF_WrongCommandFunctionMode*

6.4.7 RandomModulationOnOff

Command block structure:
I1 I2 OO

I1	command ID low-byte
I2	command ID high-byte
OO	on/off flag

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*

6.4.8 EN302208TestmodulationOnOFF

This command generates a test signal according to Chapter 6.1.1 and Chapter 8.4.2 of the EN 302208-1 V 1.3.1 from Dezember 2009 for the measurement of the reader sending spectrum which conforms to the standards.

Command block structure:
I1 I2 OO

I1	command ID low-byte
I2	command ID high-byte
OO	on/off flag

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*

6.4.9 MeasureTxPwr

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 V1 V2 V3 V4

I1	response ID low-byte
I2	response ID high-byte
V1	measured value LSB
V2	measured value 2nd byte
V3	measured value 3rd byte
V4	measured value MSB

The measured value is a 32-bit floating point number measured in dBm.

6.4.10 MeasureRxPower

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 V1 V2 V3 V4

I1	response ID low-byte
I2	response ID high-byte
V1	measured value LSB
V2	measured value 2nd byte
V3	measured value 3rd byte
V4	measured value MSB

The measured value is a 32-bit floating point number measured in dBm.

6.4.11 MeasureVSWR

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 V1 V2 V3 V4

I1	response ID low-byte
I2	response ID high-byte
V1	measured value LSB
V2	measured value 2nd byte
V3	measured value 3rd byte
V4	measured value MSB

The measured value is a 32-bit floating point number.

6.4.12 StartInventory

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*

6.4.13 OpenNoninventoriedTag

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF [EF [ERD] EW [E1 ... En] [H1 H2]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoCarrier* *RF_NoTag* *RF_WrongCommandFunctionMode* *RF_NoHandle*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
H1	handle low-byte
H2	handle high-byte

6.4.14 OpenSpecificTag

Command block structure:
I1 I2 EW [E1 ... En]

I1	command ID low-byte
I2	command ID high-byte
EW	number of EPC words (between 0 and 31)
E1	EPC LSB
...	
En	EPC MSB

Response block structure:
I1 I2 RF [EF [ERD] EW [E1 ... En] [H1 H2]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoCarrier* *RF_NoTag* *RF_WrongCommandFunctionMode* *RF_NoHandle*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
H1	handle low-byte
H2	handle high-byte

6.4.15 DirectBulkGetEPCs

Command block structure:
I1 I2 T1 T2

I1	command ID low-byte
I2	command ID high-byte
T1	bulk reading time low-byte
T2	bulk reading time high-byte

The bulk reading time is given in milliseconds.

Response block structure:
I1 I2 RF [EF]
[[ERD] EW [E1 ... En]]
...
[[ERD] EW [E1 ... En]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoCarrier* *RF_NoTag* *RF_WrongCommandFunctionMode*
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB

6.4.16 TxGen2CommandSelect

Command block structure:
I1 I2 TA AC MB P1 P2 P3 P4 ML [MD ... MD]

I1	command ID low-byte
I2	command ID high-byte
TA	Target; see *EPCglobal Specification V 2.0.0 p. 73*
AC	action; see *EPCglobal Specification V 2.0.0 p. 73*
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 73*
P1	bit pointer LSB
P2	bit pointer 2nd byte
P3	bit pointer 3rd byte
P4	bit pointer MSB
ML	masked length in bits
MD	masked data

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*

6.4.17 TxGen2CommandACK

Command block structure:
I1 I2 H1 H2

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte

Response block structure:
I1 I2 RF [P1 P2 [X1L X1H [X2L X2H]] [E1 ... En]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
P1	tag PC low-byte
P2	tag PC high-byte
X1L	tag-XPC_W1 low-byte
X1H	tag-XPC_W1 high-byte
X2L	tag-XPC_W2 low-byte
X2H	tag-XPC_W2 high-byte
E1	EPC LSB
...	
En	EPC MSB

6.4.18 TxGen2CommandNAK

Command block structure:
I1 I2

I1	command ID low-byte
I2	command ID high-byte

Response block structure:
I1 I2 RF

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*

6.4.19 TxGen2CommandReqRN

Command block structure:
I1 I2 H1 H2

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte

Response block structure:
I1 I2 RF [R1 R2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
R1	tag random number low-byte
R2	tag random number high-byte

6.4.20 TxGen2CommandAccess

Command block structure:
I1 I2 H1 H2 W1 W2

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte
W1	half-encrypted access password low-byte
W2	half-encrypted access password high-byte

Response block structure:
I1 I2 RF [H1 H2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
H1	handle low-byte
H2	handle high-byte

6.4.21 TxGen2CommandRead

Command block structure:
I1 I2 H1 H2 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to read

Response block structure:
I1 I2 RF [EC [D1 ... Dx] H1 H2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
EC	tag error code
D1	response data LSB
...	
Dx	response data MSB
H1	handle low-byte
H2	handle high-byte

6.4.22 TxGen2CommandWrite

Command block structure:
I1 I2 H1 H2 MB P1 P2 P3 P4 DL DH

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
DL	encrypted word to write low-byte
DH	encrypted word to write high-byte

Response block structure:
I1 I2 RF [EC H1 H2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError RF_NoData* *RF_CRCError* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
EC	tag error code
H1	handle low-byte
H2	handle high-byte

6.4.23 TxGen2CommandKill

Command block structure:
I1 I2 H1 H2 W1 W2 RC SC

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte
W1	half-encrypted kill password low-byte
W2	half-encrypted kill password high-byte
RC	recommissioning bits; see *EPCglobal Specification V 1.2.0 p. 68*
SC	*second kill* flag (the response from the tag varies between the first and second kill)

Response block structure:
I1 I2 RF [EC H1 H2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
EC	tag error code
H1	handle low-byte
H2	handle high-byte

6.4.24 TxGen2CommandLock

Command block structure:
I1 I2 H1 H2 L1 L2 L3

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte
L1	payload LSB
L2	payload 2nd byte
L3	payload MSB

Response block structure:
I1 I2 RF [EC H1 H2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
EC	tag error code
H1	handle low-byte
H2	handle high-byte

6.4.25 TxGen2CommandBlockWrite

Command block structure:
I1 I2 H1 H2 MB P1 P2 P3 P4 WC DL DH [DL DH ... DL DH]

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to write
DL	encrypted word to write low-byte
DH	encrypted word to write high-byte

Response block structure:
I1 I2 RF [EC H1 H2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
EC	tag error code
H1	handle low-byte
H2	handle high-byte

6.4.26 TxGen2CommandBlockErase

Command block structure:
I1 I2 H1 H2 MB P1 P2 P3 P4 WC

I1	command ID low-byte
I2	command ID high-byte
H1	handle low-byte
H2	handle high-byte
MB	memory bank; see *EPCglobal Specification V 2.0.0 p. 44*
P1	word pointer LSB
P2	word pointer 2nd byte
P3	word pointer 3rd byte
P4	word pointer MSB
WC	number of words to erase

Response block structure:
I1 I2 RF [EC H1 H2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_CRCError* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
EC	tag error code
H1	handle low-byte
H2	handle high-byte

6.4.27 TxGen2CommandQuery

Command block structure:
I1 I2 DR MC TE SL SE TA QV

I1	command ID low-byte
I2	command ID high-byte
DR	divide ratio (0: DR = 8; otherwise DR = 64/3)
MC	handle high-byte
MB	Miller coefficient (in the range from 0 to 3; see *EPCglobal Specification V 2.0.0 p. 76*)
TE	TRext (0: no audio tone; see *EPCglobal Specification V 2.0.0 p. 76*)
SL	Sel (in the range from 0 to 3; see *EPCglobal Specification V 2.0.0 p. 76*)
SE	Session (in the range from 0 to 3; see *EPCglobal Specification V 2.0.0 p. 76*)
TA	Target (0: A; otherwise B; see *EPCglobal Specification V 2.0.0 p. 76*)
QV	Q value; see *EPCglobal specification V 2.0.0 p. 76*)

Response block structure:
I1 I2 RF [R1 R2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
R1	tag random number low-byte
R2	tag random number high-byte

6.4.28 TxGen2CommandQueryAdjust

Command block structure:
I1 I2 SE AV

I1	command ID low-byte
I2	command ID high-byte
SE	Session (in the range from 0 to 3; see *EPCglobal specification V 2.0.0 p. 77*)
AV	adjust value (0, 3 or 6; see *EPCglobal specification V 2.0.0 p. 77*)

Response block structure:
I1 I2 RF [R1 R2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
R1	tag random number low-byte
R2	tag random number high-byte

6.4.29 TxGen2CommandQueryRep

Command block structure:
I1 I2 SE

I1	command ID low-byte
I2	command ID high-byte
SE	Session (in the range from 0 to 3; see *EPCglobal specification V 2.0.0 p. 78*)

Response block structure:
I1 I2 RF [R1 R2]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_OutOfRange* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
R1	tag random number low-byte
R2	tag random number high-byte

6.4.30 TxGen2CommandCustomCommand

Command block structure:
I1 I2 BC CD1 [... CDn] RC FR

I1	command ID low-byte
I2	command ID high-byte
BC	number of command bits
CD	command data
RC	number of response bits expected
FR	flag register

Transmit sequence: The first bit to be transmitted is situated in the MSB of the first byte (CD1) of the command data.

Flag register structure:

FR D0	Tx preamble flag (0: FrameSync is sent; 1: preamble is sent)
FR D1	Rx timeout flag (0: normal timeout; 1: e xtended timeout [for write commands])
FR D2 — D7	not used

Response block structure:
I1 I2 RF [RD1 [... RDn]]

I1	response ID low-byte
I2	response ID high-byte
RF	result flag Possible result flags: *RF_NoError* *RF_NoData* *RF_NoCarrier* *RF_WrongCommandFunctionMode*
RD	response data

Receive sequence: The first bit received is in the MSB for the first byte (RD1) of the response data.

6.5 Structure of the Extended Result Data

In case of the tag operations, the extended result data include the antenna port, the RSSI value, the timestamp of the tag access, the Protocol Control Word (s) of the tags, the frequency and the phase angle φ of the tag response. Which information is transmitted, depends on the Extended Result flags (EF); see *5.13 Extended Result Flag, p. 16*.

Structure:

[[AP] [RS] [T1 T2 T3 T4] [P1 P2 [X1L X1H [X2L X2H]]] [F1 F2 F3 PL PH] [R1 R2]] [TMS1 TMS2 TMS3 ... TMS8]]

AP	antenna port
RS	RSSI value
T1	timestamp LSB
T2	timestamp 2nd byte
T3	timestamp 3rd byte
T4	timestamp MSB
P1	tag PC low-byte
P2	tag PC high-byte
X1L	tag-XPC_W1 low-byte
X1H	tag-XPC_W1 high-byte
X2L	tag-XPC_W2 low-byte
X2H	tag-XPC_W2 high-byte
F1	frequency LSB
F2	frequency 2nd byte
F3	frequency MSB
PL	tag phase LSB
PH	tag phase MSB
R1	RSSI value in dBM low-byte
R2	RSSI value in dBM high-byte
TMS1	UTC-Time in milliseconds LSB
TMS2	UTC-Time in milliseconds 2nd byte
TMS3	UTC-Time in milliseconds 3rd byte
TMS4	UTC-Time in milliseconds 4th byte
TMS5	UTC-Time in milliseconds 5th byte
TMS6	UTC-Time in milliseconds 6th byte
TMS7	UTC-Time in milliseconds 7th byte
TMS8	UTC-Time in milliseconds MSB

6.6 Asynchronous Responses

6.6.1 Message Transmission

Response block structure:
I1 I2 T1 T2 T3 T4 MT MN CG

I1	response ID low-byte
I2	response ID high-byte
T1	timestamp LSB
T2	timestamp 2nd byte
T3	timestamp 3rd byte
T4	timestamp MSB
MT	message type (reserved: always 0)
MN	message number
CG	coming/going flag

6.6.2 Tag Data Transmission

The transmission of the tag data (coming message) as a consequence of an asynchronous command is described in the individual asynchronous commands. In the case of going messages, in addition to the extended result flag, only the EPC and a tag counter value are transmitted. The tag counter value shows how often a tag has been detected since its appearance in the field up until the going message has been registered. The tag counter value cannot exceed the value 65535.

Block structure of the asynchronous response for a tag going message:
I1 I2 CG EF [ERD] EW [E1 ... En] C1 C2

I1	response ID low-byte
I2	response ID high-byte
CG	coming/going flag (going message: 0x00)
EF	extended result flag
ERD	extended result data; see *6.4 Structure of the Extended Result Data, p. 140*
EW	number of EPC words
E1	EPC LSB
...	
En	EPC MSB
C1	number of times the tag has been read low-byte
C2	number of times the tag has been read high-byte

6.7 Tips for ©KRAI Antennas

If a reader is connected to antennas with KRAI interface (Kathrein RFID Antenna Interface), and an antenna property, e.g. horizontal, has been configured for the respective antenna, then the antenna property with which the tag has been detected is registered in the response data to tag commands in the upper 4 bits of the antenna information (AP — antenna port).

7 Contact Information

Kathrein Solutions GmbH
Kronstaudener Weg 1
83071 Stephanskirchen
Phone +49 (0) 8036 / 90 831 20
Fax +49 (0) 8036 / 90 831 69
Email: info@kathrein-solutions.com

Kathrein Solutions GmbH, Kronstaudener Weg 1, 83071 Stephanskirchen, Germany, Phone +49 8036 90831 20, Fax +49 8036 90831 69