OBID®
OBID classic-pro

MANUAL

ID CPR40.xx - Family

RFID Reader with ISO/IEC14443-A and -B support

Up From Firmware Version 03.04.00

final - public (B)
2013-04-29 - H71100-13e-ID-B.docx

FEIG ELECTRONIC

Note

FEIG ELECTRONIC GmbH
Lange Strasse 4
D-35781 Weilburg (Germany)
Tel.: +49 6471 3109-0
http://www.feig.de

With the edition of this manual, all previous editions become void. Indications made in this manual may be changed without previous notice.

Composition of the information in this manual has been done to the best of our knowledge. FEIG ELECTRONIC GmbH does not guarantee the correctness and completeness of the details given in this manual and may not be held liable for damages ensuing from incorrect or incomplete information. Since, despite all our efforts, errors may not be completely avoided, we are always grateful for your useful tips.

The installation instructions given in this manual are based on advantageous boundary conditions. FEIG ELECTRONIC GmbH does not give any guarantee promise for perfect function in cross environments.

FEIG ELECTRONIC GmbH assumes no responsibility for the use of any information contained in this manual and makes no representation that they free of patent infringement. FEIG ELECTRONIC GmbH does not convey any license under its patent rights nor the rights of others.

OBID® and OBID i-scan® is a registered trademark of FEIG ELECTRONIC GmbH.
All cited brand names, product names, or trademarks belong to their respective holders.

General information's regarding this manual

- If bits within one byte are filled with "-", these bit spaces are reserved for future extensions or for internal testing- and manufacturing-functions. These bit spaces must not be changed, as this may cause faulty operation of the Reader.
- The following figure formats are used:
  0...9: for decimal figures
  0x00...0xFF: for hexadecimal figures,
  b0...1 for binary figures.
- The hexadecimal value in brackets "[ ]" indicates a command.

Content

Revision History of documentation

Revision	Description
0	Described Firmware: 01.00.00 • First edition
1	Described Firmware: 01.00.00 • Command [0x75] Adjust Antenna added • Parameter CFG3.MIN_LVL redefined • Parameter CFG4.MOD-IDX removed
2	Described Firmware: 01.01.00 • CFG3.ISO14443 STUPT uses 5 ms increments
3	Described Firmware: 01.04.00 • Adding of ANNEX F: Supported SAM Baud Rates • SAM command [0xC0][0x01]: Adding of optional parameter EXT_TA1 • New status byte 0x37: Unsupported SAM Baudrate
4	Described Firmware: 01.05.00 • Revised description of chapter 2.4. Data Format and Protocol Frames for bi-directional communication • Changed naming of reader types with USB interface • Command [0xC0][0x01] SAM Activate / Deactivate: New option GetATR • New parameter CFG3.ISO14443_FTUR.UID_ORDER
5	Described Firmware: 01.07.00 • Support of mifare Ultralight C (MF0ICU2): added • Command [0xA3] Write DES/AES Reader Keys: added • Command [0xA2] Write mifare Keys: maximum number of keys reduced to 8 • Command [0xB2] Authent Mifare Ultralight C: added • New parameter CFG6.SCAN-DATA2.INDPD
6	Described Firmware: 01.09.00 • Command [0xB0][0x25] Select, new option DRV_SEL for explicite Transponder driver selection. • Command [0xBD] ISO14443A Transparent Command: new mode • Command [0xBE] ISO14443B Transparent Command: new mode 1
7	Described Firmware: 01.10.00 • Support of my-d move (SLE 66R01P) added • Command [0xB0][0x25] Select, new TR-DRIVER-SELECTION 0x0B for my-d move • Annex F: appendixed
8	Described Firmware: 01.11.00 • Command [0xA0] Reader login added • Configuration Blocks CFG15 (Password) and CFG14 (Customer Parameters) added

Revision	Description
9	Described Firmware: 02.00.00 • CFG1, PSST changed default value (0x00) • CFG1, TR-RESPONSE-TIME changed default value (0x14) • CFG2 changed – new I/O control parameters added • CFG8, and CFG9 added • Description of SoftCrypto functions for mifare DESFire, mifare Ultralight C and mifare PLUS transferred to separate manual H01110-#e-ID-B.doc (Command [0xA3],[0xB2], [0xC1] and [0xC2] are not longer documented in this manual) • Description of SAMCrypto functions for mifare DESFire documented in separate manual H01111-#d-ID-B.doc • Command [0x66] - Mode 0x04 displays support of mifare DESFire and mifare PLUS - Mode 0x11 added to check if a SAM is inserted or removed while operation. (Command [0x65] cannot be used to check if a SAM is inserted or removed while operation). • Command [0x72] SetOutput added • Command [0xC0] [0x22] GetMoreData added • Command [0xC0] [0xBF] ISO7816 APDU added • Chapters 6.4. [0xC1] / [0xC3] ISO Host Commands for mifare DESFire Communication, 6.5. [0xC2] ISO Host Commands for mifare Plus Communication added. • ANNEX D updated • ANNEX F: changed bitrates for SAM communication • ANNEX B: ID CPR52.xx added
10	Described Firmware: 03.00.00 • Chapter 2.2.1. ISO/IEC 14443 Standard Mode added • CFG1: Default value of TR-Response-Time changed (6 Seconds) • CFG3: Default value of 14443 bitrate changed (106 kBit/s) • CFG3: Default value of ISO14443 STUPT changed (0 ms) • CFG8 and CFG9 adjusted • CFG10: Scan Mode - Mifare DESFire Settings added • CFG11, CFG12 and CFG13 added as reserved • Command [0x66] (Get Reader Info): Mode 0x04: Parameter FNC_LST1 added • Command [0x75] Adjust Antenna removed • [0xC0] SAM Commands: Description updated • Command [0xC0] [0x01] Activate/Deactivate: New parameters added • Command [0xC0] [0x23] AckAbortRequest added • ANNEX D updated
11	Described Firmware: 03.02.00 • CFG10: Scan Mode - Mifare DESFire Settings: New Parameter added (support of reading record files)

Revision	Description
12	Described Firmware: 03.03.00 • Documentation of REQUEST-DATA and RESPONSE-DATA changed and aligned for all commands. • Chapter 2.2. ISO Host Commands for Transponder Communication revised editorially • Chapter 2.4. Data Format and Protocol Frames for bi-directional communication revised editorially • CFG3: New parameter TAG-DRV.SEL added • CFG11: New parameters TAG DRIVER x added • Command [0x66] - New Mode 0x08 displays the supported tag drivers • Command [0xB0][0x01] Inventory: Chapter 6.1.1.6. Response-Data – Innovatron [14443-B'] (TR-TYPE = 0x10) added. Chapter 6.1.1.7. Response-Data – CTx (TR-TYPE = 0x11) added. • Command [0xB0][0x23] Read Multiple Blocks: Max number of bytes to read increased to 256. • Command [0xB0][0x24] Write Multiple Blocks: The maximum number of bytes to write increased to the reader´s maximum receive buffer size. • Command [0xB0][0x25] Select: New CARD_INFO of an Innovatron radio protocol (ISO14443B') Transponder added. • Command [0xB2] ISO 14443 Special Commands: Chapter 6.3.3. [0xBB] ISO 14443-B' (Innovatron) Data Exchange added. • Supported ISO Host Commands: Chapters 9.4.3. ASK – CTS256B 9.4.4. ASK – CTx512B 9.5. Innovatron radio protocol (ISO14443-B') compliant Transponder added. • ANNEX A: Codes of Transponder Types modified added.
13	Described Firmware: 03.04.00 • CFG6 Scan-Mode 1: Bit SCAN-DATA1.ByteOrder[Processor Cards] added • CFG6 Scan-Mode 1: Default value of SCAN-DATA1 changed to 0x11 • Chapter 7.1. [0xB9] Jewel Transparent Command added • Chapter 7.2. [0xBD] ISO14443A Transparent Command updated • Chapter 7.3. [0xBE] ISO14443B Transparent Command updated

Abbreviations

ADR	Address
AFI	Application Family Identifier
ASK	Amplitude Shift Keying
CFG	Configuration Parameter Block
CRC	Cyclic Redundancy Check
DB	data block
frq	Frequency
FSK	Frequency Shift Keying
h	Hour
Hz	Hertz
ID	Identification
IN	Input
LEN	Length
LOC	Location
LSB	Least Significant Byte
min	Minutes
ms	Milliseconds
MSB	Most Significant Byte
N	Number
OUT	Output
R/W	Read / Write Access
RD	Read
REL	Relay
RF	Radio Frequency
TR	Transponder
TS	Timeslot
UID	Unique Identifier (read only Serial Number)
WO	Write Only Access
WR	Write

1. Introduction

1.1. About this Manual

This Manual describes the interface commands, functions and parameters, which are supported by the OBID® *classic-pro* ID CPR40.xx RFID reader.

This manual is intended for system integrators, software developers, system designers and all others who are working with the ID CPR40.xx host interface. The structure and elements of this document are represented in a very similar way also in the Windows® software tool "CPRStart" which is available via download from FEIG ELECTRONIC's web site (www.feig.de).

1.2. The OBID® ID CPR40 Family

Readers of the OBID® *classic-pro* ID CPR40.xx reader family are multi-tag readers which mean that they are able to identify transponders of different manufacturers and ISO-standards.

The functionality of the ID CPR40.xx reader family is based on the well known ID CPR-family, like the reader module ID CPR.M02.VP/AB-x and the ID CPR.02.VP/AB-x reader for wall installation and are compatible with them mainly.

The OBID® *classic-pro* ID CPR40.xx reader family is able to process (read and write) Transponders according ISO/IEC 14443 type -A and type -B.

The Readers are supporting also the NXP mifare classic security. Additional some reader models are available with 2 sockets for attachable Security Access Module (SAM), which makes it even suitable for applications such as ticketing, banking, transportation, accounting systems etc.

For data transfer with an host computer the ISO-host mode (polling) or the Scan-mode (unidirectional) are available and can be configured in the reader.

Apart from this, the data-/clock interface of some OBID® *classic-pro* ID CPR40.xx Readers enables them to be used in access control systems.

The use of OBID® ISO-host commands guarantees a easy creation of user software as well as the module's compatibility with OBID *i-scan*® Reader family.

Beside the **CPRStart** software for demonstration and configuration the reader capabilities and the **OBID® Firmware Update Tool** a lot of different **Software Development Kits** (SDK) and drivers are available to support a easy integration into the customers application.

NOTICE:
The described functionality in this document represents a summary of OBID® classic-pro ID CPR40.xx Family. The table in ANNEX D: Compendium of Supported Commands and Functions ***gives differentiated overview capabilities of each reader type and the supported functions. Also the separate firmware history file for each reader type informs you about the supported functionality.***

The following table gives a overview about the hardware similarities and differences within the OBID® ID CPR40.xx-xx reader family.

	ID CPR40.00-CD 3.3	**ID CPR40.01-CD 3.3**	**ID CPR40.00-A**	**ID CPR40.01-A**	**ID CPR40.00-CDUSB**	**ID CPR40.01-CDUSB**	**ID CPR40.30-USB**	**ID CPR40.30-SUSB**	**ID CPR40.30-A**
Housing	-						Plastic		
Dimensions (L x D x H)	50 x 50 x 14 mm						144x84x18 mm		
Protection class	-						IP42		
Power supply	3.3 V / DC		5 V / DC		USB				5 V / DC
Antenna:									
internal	●	-	●	-	●	-	●	●	●
external	-	●	-	●	-	●	-	-	-
SAM Socket	-	-	-	-	-	-	-	2	-
Signaler									
• LED	2	2	2	2	2	2	2	2	2
• Buzzer	-	-	-	-	-	-	-	-	-
Digital outputs	-	-	-	-	-	-	-	-	-
Digital inputs	-	-	-	-	-	-	-	-	-
Interface									
• RS232	–	–	●	●	–	–	–	–	●
• RS232-TTL	–	–	–	–	●	●	–	–	–
• RS232-LVTTL	●	●	–	–	–	–	–	–	–
• Data-/Clock (Mag. Stripe)	●	●	–	–	●	●	–	–	–
• Wiegand	●	●	–	–	●	●	–	–	–
• USB full-speed (12Mbit/s)	–	–	–	–	●	●	●	●	–

● included in standard device
○ optional, according to the model
(○) in development
– not available

1.3. High Level Security Card Interface

The firmware offers a high level interface which simplifies data exchange with encrypted NXP chips like mifare DESFire, mifare PLUS and mifare Ultralight C which offers fast processing and easy implementation.

This high level interface is called SoftCrypto for such OBID® *classic-pro* readers without SAM socked and SAMCrypto for OBID® *classic-pro* readers without SAM socked.

SAMCrypto

The SAMCrypto functions supporting SAM based authentication, ciphering and deciphering and security functions of mifare DESFire. The advantage of SAMCrypto firmware is the firmware internally processing and handling of crypto functions between mifare DESFire and mifare SAM AV1 or AV2.

SoftCrypto

The SoftCrypto functions supporting authentication, ciphering and deciphering and security functions of mifare DESFire, mifare PLUS and mifare Ultralight C like DES, TDES, 3KTDES and AES algorithms for fully enciphered and CMAC protected files as well as read and write operations on plain files. For this purpose the necessary crypto graphical algorithms are implemented in firmware. The keys can be stored in the EEPROM permanently or in the RAM temporary.

2. Data Transmission between OBID® ID CPR-Reader and Host

Four different ways of data transmission between OBID® *classic-pro* Readers and host (terminal, PC) are possible. The ISO Host Commands and the Scan-Mode are used for the data exchange between Transponder and host, whereas the Configuration Commands and the Control Commands are for adapting the Reader parameters to the individual range of applications. The following chart shows which method of data transmission is supported by which interface:

	interface		
	asynchronous (RS232)	**synchronous Data-/Clock**	**USB**
Configuration and control commands	●	-	●
ISO Host Commands	●	-	●
Scan-Mode	●	●	●

2.1. Configuration Commands and Control Commands

This method of data transmission is used for Reader configuration and the diagnosis via the asynchronous interface.

The Reader-configuration parameters will be stored in the Reader memory. To store the current configuration during a power down of the Reader the Reader configuration has to be stored in the EEPROM. After power up the Reader reads the configuration out of the EEPROM.

The Reader control is immediately processed and the answer from the Reader contains status or data information of the control command.

Host (Terminal / PC / ....)		**Reader**	
parameter- / control command	→	parameter received and stored / control command processed	
		yes	no
	←	status / data	error status
	←		

2.2. ISO Host Commands for Transponder Communication

The ISO Host Commands provide the exchange of data between a host and Transponders via the Reader as long as the Transponder remains in the detection range of the Reader.

NOTICE:
During the writing of data on a Transponder, it must be ensured that the Transponder is located within the detection range of the Reader during the entire process. If the Transponder is removed from detection range of the Reader during a writing process, this will cause a loss of data.

Transponder Addressing Modes:

The Reader supports the two different transponder addressing modes "Selected" and "Addressed" which are specified by the relevant technical standards or the transponder chip Manufacturer

Addressed Transponder Communication:

Before reading or writing data in addressed mode, the UID of the Transponder has to be known. This is executed by sending the command “6.1.1. [0x01] Inventory“. If a Transponder is located within the detection range of the Reader at that time, it answers with its UID. For all following read- / write command the Transponder must be addressed with its UID.

The following chart shows the necessary steps for the communication with a Transponder in addressed mode:

<table>
<tr><th>Host (Terminal / PC / ....)</th><th></th><th colspan="2">Reader</th></tr>
<tr><td rowspan="4">Inventory
to get the UID</td><td>→</td><td colspan="2">Transponder in antenna field ?</td></tr>
<tr><td></td><td>Yes</td><td>No</td></tr>
<tr><td>←</td><td>status /
number of Transponders / UID</td><td>status =
no Transponder</td></tr>
<tr><td>←</td><td colspan="2"></td></tr>
<tr><td rowspan="4">Data exchange by addressing the
with its UID</td><td>→</td><td colspan="2">Transponder with
correct UID in antenna field ?</td></tr>
<tr><td></td><td>Yes</td><td>No</td></tr>
<tr><td>←</td><td>status /
Transponder read data</td><td>status =
no Transponder
in Reader field</td></tr>
<tr><td>←</td><td colspan="2"></td></tr>
</table>

Selected Transponder Communication:

In this mode the Reader communicates only with the one, selected Transponder.

Before reading or writing data in selected mode, the UID of the Transponder has to be known. This is executed by sending at first the command “6.1.1. [0x01] Inventory“. In a second step the Transponder must be selected with the select command (see: 6.1.2. [0x25] Select) which must include its UID.

The following chart will show the necessary steps for the communication with a Transponder in selected mode:

<table>
<tr><th>Host (Terminal / PC / ....)</th><th></th><th colspan="2">Reader</th></tr>
<tr><td rowspan="4">Inventory
to get the UID</td><td>→</td><td colspan="2">Transponder in antenna field ?</td></tr>
<tr><td></td><td>Yes</td><td>No</td></tr>
<tr><td>←</td><td>status /
number of Tran-
sponders / UID</td><td rowspan="2">status =
no Transponder</td></tr>
<tr><td>←</td><td></td></tr>
<tr><td rowspan="4">Select Transponder with UID</td><td>→</td><td colspan="2">Transponder with the
correct UID in antenna field ?</td></tr>
<tr><td></td><td>Yes</td><td>No</td></tr>
<tr><td>←</td><td>status /
Transponder read
data</td><td rowspan="2">status =
no Transponder
in Reader field</td></tr>
<tr><td>←</td><td></td></tr>
<tr><td rowspan="4">Data Exchange with the selected
Transponder</td><td>→</td><td colspan="2">selected Transponder
in antenna field ?</td></tr>
<tr><td></td><td>Yes</td><td>No</td></tr>
<tr><td>←</td><td>status /
Transponder read
data</td><td rowspan="2">status =
no Transponder
in Reader field</td></tr>
<tr><td>←</td><td></td></tr>
</table>

2.2.1. ISO/IEC 14443 Standard Mode

In ISO/IEC 14443 Standard Mode the reader handles timeout control and error handling for ISO14443 RFID cards in conformance with ISO/IEC 14443.

In ISO/IEC 14443 standard mode the reader RFID Interface can be adapted by various parameters to the application requirements (see chapters 3.4. CFG3: RF-Interface, 3.5. CFG4: Transponder Parameters and 3.6. CFG5: Anticollision).

The adjacent flowchart shows a typical command flow to process a transponder in ISO/IEC 14443 standard mode.

Inventory Command

The Inventory command has to be used to check if a transponder is in the detection range of the reader. The response includes beneath the UID/PUPI a information about the ISO14443 layer supported by the card. More details are described in chapter 6.1.1. [0x01] Inventory

Select Command (see: 6.1.2. [0x25] Select)

The Select command has to be used for most common ISO/IEC 14443 transponder (see also addressing mode in chapter 2.2. ISO Host Commands).

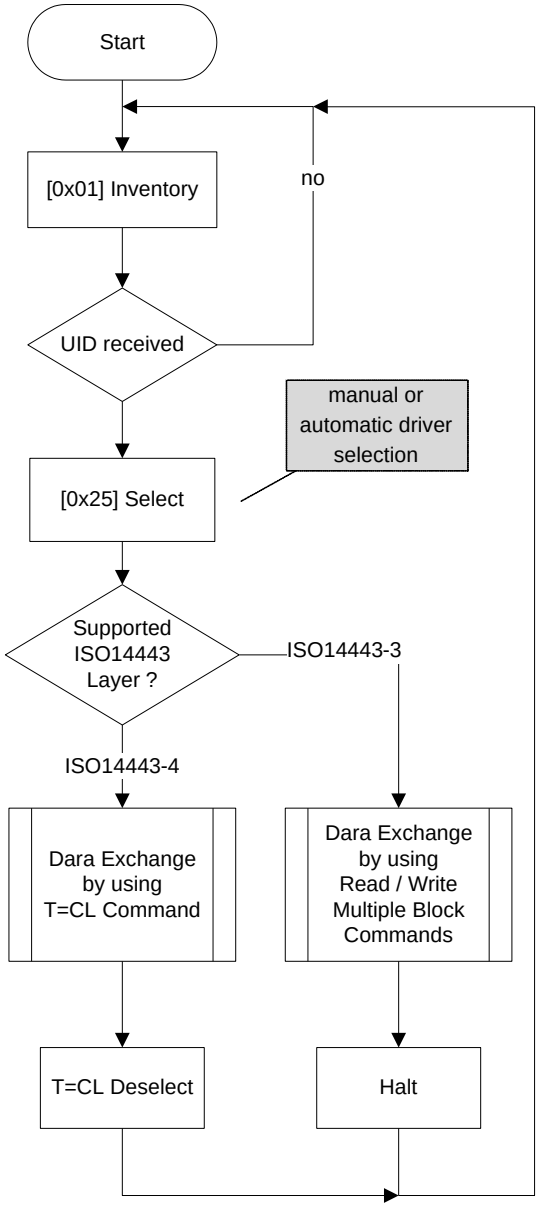
The response includes information's about the selected transponder type.

The select command offers the option of an automatic type identification or to select explicit a Transponder driver.

Data Exchange:

Depending on the ISO14443 Layer supported by the transponder different commands are offered for data exchange.

Details about the supported commands are described in chapter 9. Supported ISO Host commands.



2.3. Scan-Mode for Transponder Communication

In this operation-mode the Reader autonomously sends out data to the host as soon as a Transponder is within the detection range and valid data could be read.

In Scan-Mode the contents of the message block can be adapted to the user-application.

Scan-Mode is available via the asynchronous interface, the Data-/Clock interface or the USB interface depending on the reader hardware. The Scan-Mode interface can be enabled by parameters in CFG1 (see chapter 3.2. CFG1: Interface)

Data-/Clock interface:

Scan-Mode can be configured as Magnetic Stripe Emulation and Wiegand Emulation (see chapter 3.2. CFG1: Interface).

Also the kind of data, the data coding and the data volume can configured in a wide range (see 3.7. CFG6: Scan-Mode1 and 3.8. CFG7: Scan-Mode2)).

Asynchronous interface:

The kind of data, the data coding the data volume can be configured (see 3.7. CFG6: Scan-Mode1 and 3.8. CFG7: Scan-Mode2) if the asynchronous interface is used.

USB interface:

If an USB-Reader is used in scan mode, the reader sends its data automatically in HID interface mode (keyboard emulation) to the host. In this case you cannot catch the data with the FEUSB.DLL or any other libraries.

The configuration of the kind of data and the data volume can be done by settings in 3.7. CFG6: Scan-Mode1 and 3.8. CFG7: Scan-Mode2).

NOTICE:

- ***Scan-Mode is only available on readers without SAM.***
- ***If configuration commands shall be sent to the Reader while the Scan-Mode is active, no Transponder should be within the detection range of the Reader during this time.***
- ***Only read operations are available in Scan-Mode.***

2.4. Data Format and Protocol Frames for bi-directional communication

The communication between Reader and connected host (terminal, PC, etc.) is executed by means of fixed protocols. The used protocol is intended for data bus use and offers an address byte for addressing devices within one data bus.

During data transfer the Reader supplies the required data or a status byte. The response contains the transmitted command byte.

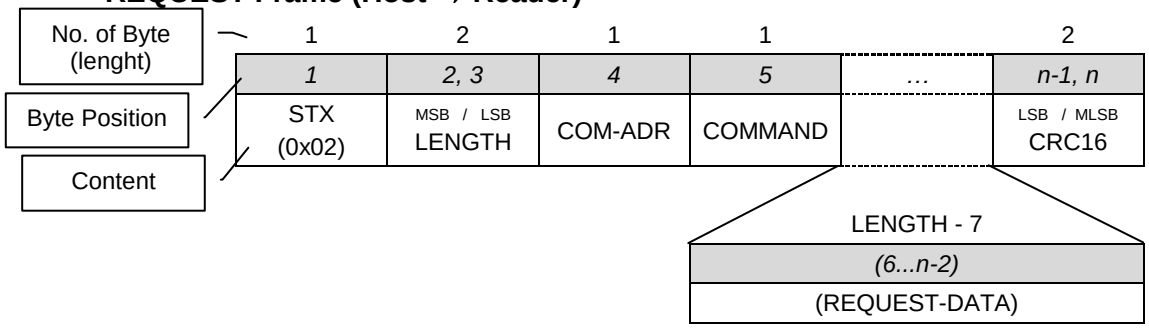
There is no reply from the Reader in case of a protocol frame failure.

The Reader supports two different Protocol frames which are the standard and the advanced protocol frame.

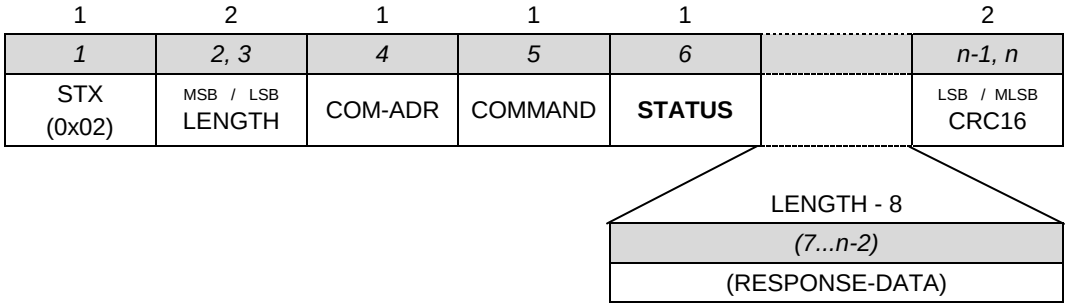
Advanced Protocol Frame

This frame is recommend for all new applications where OBID® RFID readers with Advanced Protocoll Frame support are used. The Advanced Protocol Frame can transfer up to 65535 Byte per frame and uses a clear defined STX character.

REQUEST Frame (Host → Reader)



RESPONSE Frame (Host ← Reader)



NOTICE:

- ***In this document only the REQUEST-DATA and RESPONSE-DATA block is documented for each command without the protocol frame.***
- ***Optional parameters are documented inside of* round brackets "(Optional)"**

Standard Protocol Frame

The reader supports this frame to ensure the backward compatibility to older OBID® RFID reader devices only. The Standard Protocol Frame is limited to a 255 byte protocol length and is not recommended for new applications.

Request: Host → Reader

1	1	1	LENGTH-5	2
1	2	3	4...n-2	n-1, n
LENGTH (n)	COM-ADR	COMMAND	REQUEST-DATA	LSB / MLSB CRC16

Response: Host ← Reader

1	1	1	1	LENGTH-6	2
1	2	3	4	(5...n-2)	n-1, n
LENGTH (n)	COM-ADR	COMMAND	STATUS	(RESPONSE-DATA)	LSB / MLSB CRC16

Frame selection by the reader

If the host application chose advanced protocol frame the Reader will always response with advanced protocol frames.

If the host application chose the standard protocol frame the Reader's response will depend on the length of the response data. If the response data will result a protocol frame with more than 255 Byte the Reader chose the advanced protocol frame otherwise the Reader chose the standard protocol frame for responding.

2.4.1. Protocol Elements

STX:

The STX sign (0x02) at the start of protocol indicates an Advanced Protocol-Frame.

LENGTH:

Standard Protocol Frame (6...255)

Number of protocol bytes including LENGTH and CRC16.

Advanced Protocol Frame (7...65535)

Number of protocol bytes including STX, LENGTH and CRC16

COM-ADR:

0..253 address of device in bus mode

NOTICE:

The Reader can be addressed via COM-ADR 255 at any time!

COMMAND:

Defines the Command which the Reader shall operate.

STATUS [1]:

Includes the status message or protocol data from or to the Reader.

DATA:

Is a optional data field with variable length. The number of DATA byte depends on the command. The data will be sent always as MSB first if the Reader is in the ISO-Host Command Mode.

CRC16:

Cyclic redundancy check of the protocol bytes from 1 to n-2, as specified by CCITT-CRC16

Polynomial $x^{16} + x^{12} + x^5 + 1$

Start Value 0xFFFF

In case of USB communication the CRC16 value is not checked by the reader.

[1] see ANNEX C: Index of Status Bytes

2.4.2. Timing Conditions

Protocol Start Synchronization Time (PSST):

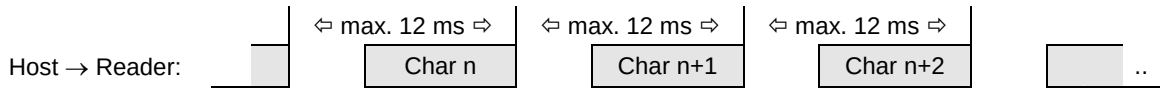
Before starting a new request protocol there must be a gap without any communication of normally 5 ms after the reception of the last byte of the response protocol. The PSST is configurable by the parameter PSST in CFG1.

Block timeout:

Defines the time within the reader response can be expected by the host. The host block timout shall be set to value longer than the time configured in CFG1.TR-RESPONSE-TIME.

Character timeout:

Within one protocol, the characters have to follow each other in intervals of maximum 12 ms.



2.4.3. CRC16 Calculation Algorithm

Polynom: $x^{16} + x^{12} + x^5 + 1 \Rightarrow$ CRC_POLYNOM = 0x8408;

Start Value: 0xFFFF ⇒ CRC_PRESET = 0xFFFF;

C-Example:

```
unsigned int crc = CRC_PRESET;

for (i = 0; i < cnt; i++)  // cnt = number of protocol bytes without CRC
{
  crc ^= DATA[i];
  for (j = 0; j < 8; j++)
  {
    if (crc & 0x0001)
      crc = (crc >> 1) ^ CRC_POLYNOM;
    else
      crc = (crc >> 1);
  }
}
```

3. Configuration Parameters (CFG)

The configuration memory of the Reader is organized in configuration blocks of 16 byte each. These are divided into 14-byte configuration parameters and a 2-byte CRC16 checksum. Each of these configuration blocks takes a number (CFG 0...CFG n).

Structure of a configuration block in Reader configuration memory and Reader EEPROM (CFG):

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Contents	PARAMETER														CRC16	

The parameters are stored in two different configuration memory locations:

- Reader RAM
- Backup EEPROM (used for storing parameter over power down)

Multiple configuration memory locations can be addressed by the value of the parameter CFG-ADR used in chapter 4. Commands for Reader Configuration

CFG-ADR:

CFGn: memory-address of the required configuration block
LOC: specifies the location of the configuration block (RAM / EEPROM)
MODE: specifies one or all configuration blocks

Bit:	7	6	5	4	3	2	1	0
Function	LOC	MODE	CFGn: address of configuration block					

The EEPROM configuration blocks are protected by a 16 bit CRC-checksum. The examination of these checksums is executed after each reset of the Reader. If an faulty checksum is found, the Reader goes into an error status "EE-Init-Mode" and sets the configuration block which is faulty to the default values.

While the EE-Init-Mode is active, the LED blinks alternately red and green and the Reader answers external commands with the status "0x10 EEPROM Failure". The "EE-Init-Mode" can be exited now by a new reset (cold start or 5.3. [0x63] CPU Reset command). If after this the checksums of all data records are correct, the Reader shifts to the configured operation mode.

Notes:

- ***Malfunctions may occur if parameters are configured outside their described range or if unspecified parameters have been changed!***
- ***A firmware update resets the EEPROM to default settings and the Reader goes into the error status "EE-Init-mode".***

Structure of configuration parameter description.

Byte	0	1	2	.....n
contents	RAM-eff.	EEPROM-eff.	00 res	.....

not marked

Changing of this parameter becomes immediately effective after writing / saving this configuration block to RAM

gray marked

Changing of this parameter only becomes effective after writing / saving this configuration block to EEPROM and a Reader reset

marked with “00“

these bits or bytes are reserved for future extensions or for internal testing and manufacturing-functions. These bits or bytes and also any not described bits and bytes **must not be changed**, as this may cause faulty operation of the Reader.

3.1. CFG0: Reserved

The configuration block CFG0 is reserved for future use.

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

3.2. CFG1: Interface

The parameters of the CFG1 configuration block contain the data communication settings.

Byte	0	1	2	3	4	5	6
Contents	COM-ADR	0x00	BAUD	TRANS-FORM	0x00	PSST	TR-RESPONSE-TIME
Default	0x00	0x00	0x08 *38400 Baud*	0x01 *e,8,1*	0x00	0x00 *0 ms*	0x00

Byte	7	8	9	10	11	12	13
Contents	TR-RESPONSE-TIME	0x00	0x00	0x00	0x00	0x00	READER - MODE
Default	0x3C *6 sec.*	0x00	0x00	0x00	0x00	0x00	0x00

COM-ADR:

Bus address of the Reader (0 .. 254) for communication via the asynchronous interface, especially for applications with the RS485 interface.

Notes:

- ***Do not configure address 255!***
- ***Via the COM-ADR 255 in the send protocol, the Reader is able to be addressed at any time. It answers then with the configured address.***

BAUD[1]**:**

By means of this byte the baud rate of the asynchronous interface can be defined.

BAUD	ID CPR40.0x- Ax / Cx	
0x05	4.800	bit/s
0x06	9.600	bit/s
0x07	19.200	bit/s
0x08	38.400	bit/s
0x09	57.600	bit/s
0x0B	115.200	bit/s
0x0D	230.400	bit/s
0x80	-	bit/s
0x81	-	bit/s
0x82	-	bit/s
0x83	-	bit/s

NOTICE:

- ***Make sure that your host system supports the selected baud rate. If not it's impossible to communicate with the reader any longer after the baud rate was changed!***
- ***Changing of BAUD only becomes effective after writing / saving configuration block CFG1 to EEPROM and a reset of the Reader.***
- ***The Reader set the baud rate to 38400 bit/s, if the user set an undefined baud rate.***

[1] A plausibility check is performed by writing this parameter to the Reader. If an error occurs the Reader answers with STATUS = 0x11.

TRANS-FORM[1]**:**

By means of this byte, several parameters for the data transmission format of the asynchronous interface can be defined.

Bit:	7	6	5	4	3	2	1	0
Function:	0	0	0	0	S	D	P	

P: Kind of Parity
- b00: no parity
- b01: even parity
- b10: odd parity
- b11: **- do not use -**

D: Number of data bits
- b0: 8 data bits
- b1: **- do not use -**

S: Number of stop bits
- b0: 1 stop bit
- b1: **- do not use -**

NOTICE:

- ***Changing of TRANS-FORM only becomes effective after writing / saving configuration block CFG1 to EEPROM and reset of the Reader.***
- ***Always 8 Data Bits and 1 Stop Bits should be used***

PSST (Protocol Start Synchronization Time) 0...5 ms

By means of this parameter the duration of the minimum communication gap between the reception of the last byte of the response protocol and the first byte of a new protocol can be defined in 1 ms steps (see also 2.4. Data Format and Protocol Frames)

The parameter could be used to speed up the communication via the asynchronous interface. In case of a RS485 Interface it's not recommended to decrease PSST.

0: The Reader response starts as soon as possible

5: Maximum value for PSST (5 ms)

[1] A plausibility check is performed by writing this parameter to the Reader. If an error occurs the Reader answers with STATUS = 0x11.

TR-RESPONSE-TIME:

By means of this parameter the maximum duration for the Transponder command can be defined.

The TR-RESPONSE-TIME starts after the Reader has received a new command. At the latest after the TR-RESPONSE-TIME elapsed the Reader will be sent an answer protocol. In this case, the current commands between Reader and Transponder are aborted.

	max. response duration
TR-RESPONSE-TIME	0...65535 * 100 ms

NOTICE:

- ***TR-RESPONSE-TIME has no effect with the protocols for Reader Configuration and the protocols for Reader Control.***
- ***The block receive timeout of host computer must set to a value ≥ TR-RESPONSE-TIME.***

READER-MODE:

By means of this byte, the Reader mode can be defined.

Bit:	7	6	5	4	3	2	1	0
Function:				DC-FORMAT			SCAN-IF	SCAN-E

SCAN-E:

By setting of this bit the Scan-Mode can be enabled

b0: **ISO Host Mode** (see chapter 6. ISO Host Commands)
b1: **Scan-Mode** (see chapter 3.7. CFG6: Scan-Mode1)

SCAN-IF:

This bit selects the interface for Scan-Mode

b0: Scan-Mode: via asynchronous or USB interface
b1: Scan-Mode: via data-/clock interface

DC-FORMAT:

By means of this parameter the kind of data transmission via data-/clock interface could by selected:

b000: Wiegand emulation (see 3.2.2. Wiegand Emulation)
data format: binary 1:1, according written to Transponder.

b001: magnetic stripe (see 3.2.1. Magnetic Strip Emulation)
data format: binary 1:1, according written to Transponder.

b010: magnetic stripe (see 3.2.1. Magnetic Strip Emulation)
data format: according ISO 7811-2, track 2+3 (5 Bit)

b011: magnetic stripe (see 3.2.1. Magnetic Strip Emulation)
data format: according ISO 7811-2, track 1 (7 Bit)

b100: Wiegand emulation (see 3.2.2. Wiegand Emulation)
data format: Wiegand formatted protocol frame with start and stop sign.

b101: magnetic stripe (see 3.2.1. Magnetic Strip Emulation)
data format: according ISO 7811-2, track 2+3 (5 Bit)
additional a prefix of 16 leading zero clocks before the start character and additional a trailer of 16 attached zero clocks following to the LRC character.

b110: Wiegand emulation (see 3.2.2. Wiegand Emulation)
data format: Wiegand formatted protocol frame

3.2.1. Magnetic Strip Emulation

Data Format:

The following table shows data coding depending on DC-FORMAT

For cutting the length of data output the parameters D_LGT and D_START (see chapter: 3.7. CFG6: Scan-Mode1) can be use.

DC-FORMAT raw data	b001 binary 1:1 MSB.....LSB	b010 b101 according ISO 7811-2 (5 bit) P / MSB.....LSB	b011 according ISO 7811-2 (7 bit) P / MSB.....LSB
0x0	b 0 0 0 0	b 1 / 0 0 0 0	b 0 / 0 1 0 0 0 0
0x1	b 0 0 0 1	b 0 / 0 0 0 1	b 1 / 0 1 0 0 0 1
0x2	b 0 0 1 0	b 0 / 0 0 1 0	b 1 / 0 1 0 0 1 0
0x3	b 0 0 1 1	b 1 / 0 0 1 1	b 0 / 0 1 0 0 1 1
0x4	b 0 1 0 0	b 0 / 0 1 0 0	b 1 / 0 1 0 1 0 0
0x5	b 0 1 0 1	b 1 / 0 1 0 1	b 0 / 0 1 0 1 0 1
0x6	b 0 1 1 0	b 1 / 0 1 1 0	b 0 / 0 1 0 1 1 0
0x7	b 0 1 1 1	b 0 / 0 1 1 1	b 1 / 0 1 0 1 1 1
0x8	b 1 0 0 0	b 0 / 1 0 0 0	b1 / 0 1 1 0 0 0
0x9	b 1 0 0 1	b 1 / 1 0 0 1	b 0 / 0 1 1 0 0 1
0xA	b 1 0 1 0	b 1 / 1 0 1 0	b 1 / 1 0 0 0 0 1
0xB	b 1 0 1 1	b 0 / 1 0 1 1	b 1 / 1 0 0 0 1 0
0xC	b 1 1 0 0	b 1 / 1 1 0 0	b 0 / 1 0 0 0 1 1
0xD	b 1 1 0 1	b 0 / 1 1 0 1	b 1 / 1 0 0 1 0 0
0xE	b 1 1 1 0	b 0 / 1 1 1 0	b 0 / 1 0 0 1 0 1
0xF	b 1 1 1 1	b 1 / 1 1 1 1	b 0 / 1 0 0 1 1 0
Start „%“	-	b 0 / 1 0 1 1	b 1 / 0 0 0 1 0 1
Stop „?“	-	b 1 / 1 1 1 1	b 0 / 0 1 1 1 1 1

Example: Output of raw data 0x19BF

Sign / DC-FORMAT	prefix (16*0)	Start %	0x1	0x9	0xB	0xF	Stop ?	LRC	trailer (16*0)
b001	-	-	0001	1001	1011	1111	-	-	-
b010	-	1101/0	1000/0	1001/1	1101/0	1111/1	1111/1	0001/0	-
b101	000...000	1101/0	1000/0	1001/1	1101/0	1111/1	1111/1	0001/0	000...000
b011	-	101000/1	100010/1	100110/0	010001/1	011001/0	111110/0	011010/0	-

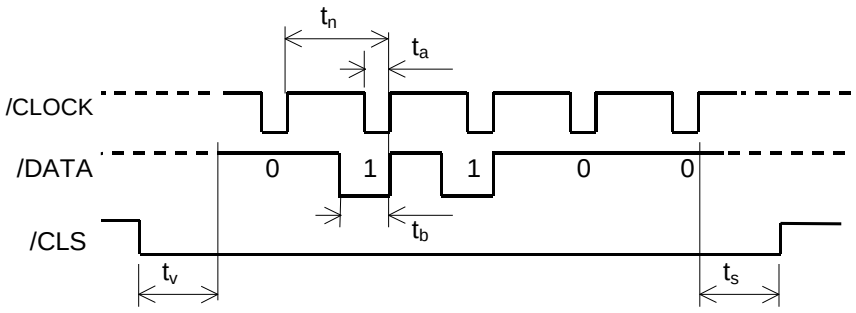
Time →

LRC

XOR operation on Start-, Data and Stop-sign

Timing

The following diagram represents the signal response of the 3 data lines of the data-/clock-interface in magnetic strip emulation.



t_n = 0,5 ms
$t_{a(n)} \approx t_n / 3$
$t_{b(n)} \approx t_n / 2$
$t_v = t_s$ = 10...12 ms

3.2.2. Wiegand Emulation

Data Format:

The following description represents the data coding depending on DC-FORMAT

For cutting the length of data output the parameters D_LGT and D_START (see chapter: 3.7. CFG6: Scan-Mode1) can be use.

DC-FORMAT = b000 ⇒ binary 1:1

In this configuration the output data format is equal to the data coding on the Transponder. The Reader doesn't add a protocol frame e.g. parity Bits or start or stop signs across the data stream.

DC-FORMAT = b100 ⇒ Wiegand formatted protocol frame with start and stop sign

In this configuration the Reader build the protocol frame with one even parity bit at the beginning and one odd parity bit at the end and one start and one stop sign:

4 Bit	1 Bit	n Bit	1 Bit	4 Bit
START	EVEN	DATA	ODD	STOP

START: b1011

EVEN: Even parity bit calculated across the first half DATA bits.

DATA: Data bits as read from the Transponder and defined in scan-mode settings.

ODD: Odd parity bit calculated across the last half DATA bits.

STOP: b1111

DC-FORMAT = b110 ⇒ Wiegand formatted protocol frame

In this configuration the Reader build the protocol frame with one even parity bit at the beginning and one odd parity bit at the end

1 Bit	n Bit	1 Bit
EVEN	DATA	ODD

EVEN: Even parity bit calculated across the first half DATA bits.

DATA: Data bits as read from the Transponder and defined in scan-mode settings.

ODD: Odd parity bit calculated across the last half DATA bits.

Example of parity calculation

Example 1: 18 DATA bit

```
DATA bin     011110110010110101101001

OUTPUT     1 011110110010110101101001 0
           `- Even Parity Bit         `- Odd Parity Bit
```

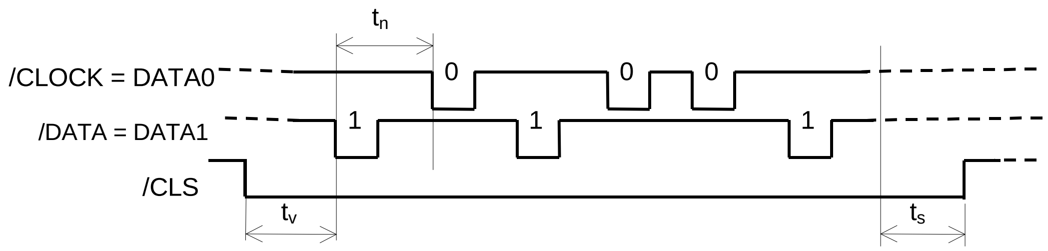
Example 1: 19 DATA bit

```
DATA bin     0111101100101101011010011

OUTPUT     0 0111101100101101011010011 1
           `- Even Parity Bit          `- Odd Parity Bit
```

Timing

The following diagram represents the signal response of the 3 data lines of the data-/clock-interface in Wiegand emulation.



t_n = 0,5 ms
t_v = t_s = 10..12 ms

3.3. CFG2: Inputs / Outputs general

Via the following parameters the operation mode of the LED can be configured individual separate for offline, online and tag-detect conditions.

NOTICE:

The ID CPR40.30 has only the LEDs V1(GRN) and V2(BL)

The ID CPR40.0x has only the LEDs V1(GRN) and V3(RED)

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	ONLINE-STATE		0x00	0x00	0x00
Default	0x00	0x00	0x0001		0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	OFFLINE-STATE		0x00	OFFLINE-DELAY	TAGDETECT-STATE		TAG DETECT ACTIVATION TIME
Default	0x0002		0x00	0x14 *2 sec.*	0x0015		0x04 *400 ms*

ONLINE-STATE:

This Parameter defines the behavior of the signal transmitters if they are not activated by any other event.

Bit:	15	14	13	12	11	10	9	8
Function:	-		-		-		-	

Bit:	7	6	5	4	3	2	1	0
Function:	-		V3(RED)		V2(BL)		V1(GRN)	

V1 / V2 / V3

The bit combination defines the behavior of the signal transmitter

b00: OFF
b01: ON
b10: FLASHING SLOW
b11: FLASHING FAST

OFFLINE-STATE:

This parameter defines the behavior of the signal transmitter, in case of the reader has detected an offline state. The following cases are possible:

Polling-Mode:

In case of polling mode the reader starts to signalize the offline state if it has received no command from the host for more than the time defined by the parameter OFFLINE-TIME.

Bit:	15	14	13	12	11	10	9	8
Function:	-		-		-		-	

Bit:	7	6	5	4	3	2	1	0
Function:	-		V3(RED)		V2(BL)		V1(GRN)	

V1 / V2 / V3

The bit combination defines the behavior of the signal transmitter

b00: OFF
b01: ON
b10: FLASHING SLOW
b11: FLASHING FAST

OFFLINE-DELAY:

This parameter defines the delay in 100 ms increments, after the Reader will signalize the offline state if he had not received a command via his asynchronous interface.

0 ... 255 x 100 ms ⇒ 0 ... 25,5 sec

TAGDETECT-STATE:

This parameter defines the behavior of the signal transmitter if a new transponder was detected by the reader

Bit:	15	14	13	12	11	10	9	8
Function:	-		-		-		-	

Bit:	7	6	5	4	3	2	1	0
Function:	-		V3(RED)		V2(BL)		V1(GRN)	

V1 / V2 / V3

The bit combination defines the behavior of the signal transmitter

b00: OFF
b01: ON
b10: FLASHING SLOW
b11: FLASHING FAST

TAGDETECT-ACTIVATION-TIME:

This parameter defines the duration in 100 ms increments, the Reader signalize a transponder was detected.

0 ... 255 x 100 ms ⇒ 0 ... 25,5 sec

3.4. CFG3: RF-Interface

The parameters of the CFG3 configuration block contain global Transponder driver and Reader settings.

Byte	0	1	2	3	4	5	6
Contents	TAG-DRV		ISO14443-DRV		0x00	0x00	MIN_LVL
Default	0x0D30		0x000F		0x00	0x00	0x05

Byte	7	8	9	10	11	12	13
Contents	ISO14443 BIT RATE	0x00	0x00	0x00	0x00	ISO14443 STUPT	ISO14443 FTUR
Default	0x00	0x00	0x00	0x00	0x00	0x00 0 ms	0x1A

TAG-DRV[1]**:**

Defines the Transponder types that are operated by the Reader.

Byte:	0								1							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Driver	SEL	0	0	0	L	K	0	I	0	G	F	E	0	0	0	0
Default	0	0	0	0	1	1	0	1	0	0	1	1	0	0	0	0

b0: Driver for the Transponder type is disabled

b1: Driver for the Transponder type is activated

.E: Driver for ISO14443A

.F: Driver for ISO14443B

.I Driver for Jewel

.K Driver for SR176

.L Driver for SRIxx

.SEL Selector

This parameter offers the option to enable further drivers which may not part of bit field TAG-DRV and the option to define the call order of the enabled transponder drivers.

b0: Select drivers from CFG3.TAG-DRV with fixed call order.

b1: Select drivers from 3.12. CFG11: Tag Driver Priority List with user defined call order. Drivers from CFG3.TAG-DRV are ignored.

Only those Transponder drivers should be active that are used in the current application. Thus, the reaction time of the Reader for Transponder read- / write-operations is reduced and the danger of a parasitic Transponder access is minimized.

[1] A plausibility check is performed by writing this parameter to the Reader. If an error occurs the Reader answers with STATUS = 0x11.

ISO14443-DRV:

Defines the ISO 14443 Transponder types that are read/write operated by the Reader. Reading of the UID is also possible if the driver is inactive, because of the standardized ISO14443 access conditions.

If more than one Transponder driver is activated the Reader attempted by means of some indications to decide about the Transponder type.

To guarantee that the Reader only processes the correct Transponder type the not required drivers should be disabled.

Byte:	2								3							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Driver	0	0	0	0	0	0	0	0	0	0	0	0	L4	C	B	A

b0: Driver for the Transponder type is disabled

b1: Driver for the Transponder type is activated

- A: Driver for mifare classic
- B: Driver for my-d proximity SLE55Rxx
- C: Driver for mifare Ultralight / NFC Card Type 2
- L4 Driver for ISO14443A, Part 4 compatible Transponders

MIN_LVL:

This parameter defines the sensitivity of the RFID receiver. The value depends on the electromagnetic environment noise.

Bit:	7	6	5	4	3	2	1	0
Function	MAN				LVL			

LVL (0x0...0x0F):

Level which could be set in automatic or manual mode. A low level could increase the reading distance but also the probability of interruptions because of noise.

- 0x0: highest sensitivity (mostly impracticable)
- 0xF: lowest sensitivity

MAN:

b0: default
If this setting is used the reader runs with his default sensitivity.

b1: manual mode
This setting is necessary if the reader shall use the setting of LVL.

ISO14443 BIT RATE:

This parameter defines the highest Bit-Rate which shall used by the Reader. The actual used Bit-Rate depends on the capabilities of the present Transponder. If the adjusted Bit-Rate is not support by the Transponder the Reader select the highest supported Bit-Rate of the Transponder.

Bit:	7	6	5	4	3	2	1	0
Function	Tx BIT RATE		Rx BIT RATE		-	-	-	-

TX BIT RATE

Used for bit rate selection from Reader to Transponder

b00: 106 kbit / s
b01: 212 kbit / s
b10: 424 kbit / s
b11: 848 kbit / s

RX BIT RATE

Used for bit rate selection from Transponder to Reader

b00: 106 kbit / s
b01: 212 kbit / s
b10: 424 kbit / s
b11: 848 kbit / s

NOTICE:

- ***A high Bit-Rate could effect a reduction of the reading distance.***
- ***It is recommended to use identical Bit-Rates for RX and TX.***

ISO14443 STUPT (1 ... 255 * 5 ms = 5 ms ... 1,275 sec)**:**

The Startup Time defines a delay-time which is required by a ISO14443 Transponder for startup after the RF-Field was switched on (e. g. after a command [0x69] RF Reset).

NOTICE:
The value of ISO14443 STUPT must be considered for calculating the TR-RESPONSE-TIME (see CFG1)

ISO14443 FTUR:

In this parameter byte are some special features combined.

Bit:	7	6	5	4	3	2	1	0
Function	UID-ORDER			OPTI	ERROR_RETRY		PLIC	BSLCT

BSLCT (only ISO 14443B Transponder)

This bit selects the response behavior for ISO 14443B Transponder with Bit-Rates above 106 kBit / s.

The Reader principally use 106 kBit / for the first communication cycle. If the Transponder supports a higher Bit-Rate and this is configured by the parameter ISO14443 BIT RATE the Reader selects the highest possible Bit-Rate.

Unfortunately the reception from the Transponder could be on 106 kBit / s ore on the new higher Bit-Rate.

b0: The first reception after a Bit-Rate change is expected with 106 kBit / s.

b1: The first reception after a Bit-Rate change is expected with the selected higher Bit-Rate.

PLIC (only ISO 14443-4 Transponder)

This bit enables the **power level indicator check** function of the Reader.

b0: Power level check is disabled.

b1: Power level check is enabled.
The power level indicator of ISO 14443-4 Transponders will be interpreted by the Reader if it is supported by the Transponder.

If a Transponder response indicates insufficient power the reader breaks the present command and send an error status.

ERROR_RETRY (only ISO 14443-4 Transponder)

This parameter defines the maximum number of automatic retry loops in case of transmission or protocol errors as described in ISO 14443-4.

b00: disables retry loop

b01: 1 retry loop

b10 2 retry loops

b11: 3 retry loops

OPTI (only ISO14443A Transponder)

By means of this bit some optional information's could be displayed for ISO14443A in the [0x01] inventory response byte OPT_INFO (see also 6.1.1. [0x01] Inventory)

b0: The OPT_INFO byte in [0x01] inventory response is always set to 0.

b1: The OPT_INFO byte in [0x01] inventory response includes further Information's.

UID_ORDER (only ISO14443A Transponder)

By means of this bit the byte order of the UID of ISO14443A Transponder can be swapped.

b0: The UID will be transferred as described in 6.1.1.1. Response-Data - ISO 14443A (TR-TYPE = 0x04).

b1: The byte order of the transferred UID will be swapped (UID transfer will be carried out like described in ISO14443).

3.5. CFG4: Transponder Parameters

The parameters of the CFG4 configuration block contain general Transponder settings.

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	0x00	ISO14443B-AFI	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

ISO14443B-AFI: (only ISO14443B Transponders)

Application Family Identifier for ISO14443 type B Transponder. For more information's refer to ISO14443-3.

3.6. CFG5: Anticollision

The parameters of the CFG5 configuration block contain anticollision settings.

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	0x00	0x00	0x00	0x00	ONT	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x0C	0x00	0x00

ONT:

This parameter configures the reply behavior of the Inventory command [0x01]. It defines which Transponder will reply to the host.

Bit:	7	6	5	4	3	2	1	0
Driver	0	0	0	0	AORB_REQ	ACOLL	0	ONT

ONT:

b0: All Transponder in Reader field
The response of the Inventory command [0x01] includes the UID of all detected Transponders in the detection range of the reader.

The Reader performs a RF-Reset before any command reads a UID.

b1: Only new Transponder in Reader field
The response of the Inventory command [0x01] includes only the UID of new detected Transponders.

If the Reader has detected a new Transponder, the Transponder will be automatically set to into the halt state (ISO14443, but not Jewel) by the Reader. In this state the Transponder does not send back a response for the next Inventory command.

The UID of a Transponder will replied only after the Transponder reenters into the detection range of the reader. Otherwise the Reader replies the Status "No Transponder" (0x01).

ACOLL:

This bit activates Anticollision Mode. In Anticollision Mode the Reader automatically sets Transponder-specific communication parameters.

b0: disabled
In this case the Reader doesn't processes any anticollision procedure with the Transponders inside the antenna field.

If anticollision is disabled, the Reader automatically selects the Transponder. The Select command [0x25] is not necessary for further communication with the Transponder.

If more than one Transponder of the same type is in the detection range the Reader replies an error status.

b1: enabled (default)
In this case the Reader processes the anticollision procedure with the Transponders inside of the antenna field and replies the UID of all detected Transponder's.

AORB_REQ:

This parameter defines the abort conditions of the Inventory command [0x01] for ISO14443 Transponder if the ISO14443A and ISO14443B Transponder drivers are activated.

b0: disabled (default)
The Inventory command runs while not all UIDs of ISO14443A and ISO14443B Transponders in the detection range are read.

b1: The Inventory command stops if the UID of all ISO14443A or of all ISO14443B Transponders in the detection range are read. So the Inventory command returns either the presence of ISO14443A or ISO14443B Transponders.

3.7. CFG6: Scan-Mode1

The parameters of the CFG6 configuration block contains Scan-Mode settings. To enable Scan-Mode the SCAN-MODE bit in the configuration block CFG1 (3.2. CFG1: Interface) has to be set.

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	0x00	SCAN-DATA1	SCAN-DATA2	0x00	SCAN-LOCK-TIME
Default	0x02	0x00	0x00	0x11	0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	SCAN-LOCK-TIME	MAD_AID		SCAN-KEY_ADR	DB_ADR	D_LGT	D_START
Default	0x0A	0x00	0x00	0x00	0x05	0x04	0x00

SCAN-DATA1

selects the data types to be sent in the Scan Mode.

Bit:	7	6	5	4	3	2	1	0
Function	Byte Order	COM-Prefix	MAD	Byte Order [ProcessorCards]	0	BCD_ UID	DB	UID

NOTICE:

- ***If Scan-Mode via data-/clock interface is selected the reader can only transmit the UID or a Data Block. If both options are activated the reader only transmits the UID.***
- ***If Scan-Mode via asynchronous interface is selected reading of UID and Data-Block can be configured at the same time.***

UID = Serial No.

Setting of this bit activates the output of the UID

b0 Output of the UID inactive (inactivates the scan-mode)
b1 Output of the UID active

DB = Data Block

Setting of this bit activates the output of a specified data block.

b0 Output of a data block inactive
b1 Output of a data block active

In case of memory cards like mifare classic the reader internaly uses commands equvalent to the ISO-Host command Read-Multible Block [0x23] whereby the parameter DB_ADR, D_LGT and D-START becomes to relevant to define the data which shall be read.

In case of mifare classic data block shall be read see also the parameter SCAN_KEY_ADR, MAD_ID and MAD Bit

In case of mifare DESFire the data file to be read is defined by parameters in CFG10 (see 3.11. CFG10: Scan Mode - Mifare DESFire Settings)

BCD_UID = Serial No. in BCD format

Setting of this bit activates the output of the UID in BCD format if the UID Bit set. In this case the least significant 4 hexadecimal Bytes of the UID are transformed into their 10 digit decimal equivalent value.

b0 Output of the BCD_UID inactive (inactivates the scan-mode)
b1 Output of the BCD_UID active

Depending on the selected scan mode interface (see CFG1) and data format (see CFG7, DB-FORMAT) the output of the BCD transformed UID could be configured in different ways.

Example:
The hexadecimal UID is 0x38 F3 7B 29
The decimal value is: 0955480873

Scan-Mode: via asynchronous interface

unformatted hex data
output 0x09 55 48 08 73

ASCII formatted hex data
output: 0x30 0x39 0x35 0x35 0x34 0x38 0x30 0x38 0x37 0x33

Scan-Mode: via data-/clock interface
If Data-/Clock Interface is used the output data format depends on the setting of the DC-FORMAT parameter.

Byte Order [Processor Cards]

Defines the byte Order within frame

b0 MSB first
b1 LSB first

MAD: (Mifare Application Identifier)

Setting of this bit activates the MAD function for reading data blocks of mifare classic Transponders. It becomes only effect if the DB bit is set to 1.

b0 MAD function is inactive

b1 MAD function is active
In this operation mode the parameter MAD_ID becomes effect and some other scan-mode becomes a different function.

NOTICE
Further details about the MAD functionality are described in Mifare Application Directory documentation issued by NXP Semiconductors.

COM Prefix

When this option is on, the Reader will transmit the COM-ADR before each data set.

b0 COM-ADR of the Reader will not transmit
b1 COM-ADR of the Reader will transmit

Byte Order

Defines the byte Order within frame

b0 MSB first
b1 LSB first

SCAN-DATA2

selects the data types to be sent in the Scan Mode.

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	INDPD

INDPD

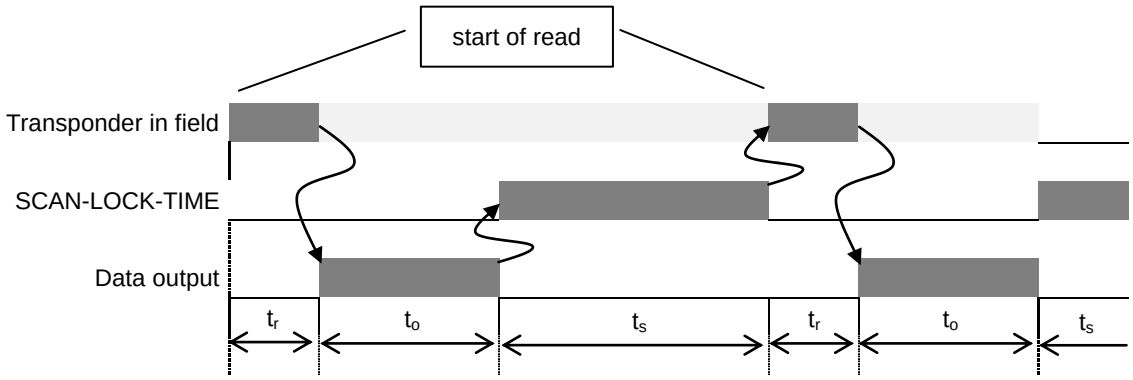
Via this bit an independent transmission of the UID can be configured if the reading of UID and DB is activated.

b0: In this case the reader starts the output of the UID and DB data block as soon as all required data (UID and DB) have been read from the Transponder successful. If the reader can not read the DB data block no UID will be transmitted.

b1: In this case the Reader transmits the UID only independent form a successful reading of the DB data block.

SCAN-LOCK-TIME: (1 ... 65535 * 100 ms = 100 ms ... 6553,5 sec)

The SCAN-LOCK-TIME defines the period in which the Reader does not transmit the Transponder data a second time, after the Reader had transmitted it the first time (regardless whether the Transponder is in the detection range of the reader during SCAN-LOCK-TIME or not). The SCAN-LOCK-TIME starts after the data transmission from the Transponder to the Reader.



t_r: Time to read the Transponder data

t_o: Data Transmission from the Reader to the host

t_s: SCAN-LOCK-TIME
As long as the SCAN-LOCK-TIME is active, the Transponder can be in the detection range of the reader or outside of it.

MAD_AID

Parameter to configure the 2 byte AID (Application Identifier) of the MAD function.

SCAN-KEY_ADR

Defines the mifare key address to be used for authentication at the mifare block which shall be read in scan-mode.

This parameter is designed to be used if the mifare block is directly addressed via DB_ADR or indirect addressed via MAD function.

The command 4.5. [0xA2] Write Mifare Reader Keys describes how to store a key in the reader.

Bit:	7	6	5	4	3	2	1	0
Function	KEY-TYPE	0	0	0	KEY-ADR			

KEY-TYPE:

Defines how the key will be used in authentication process.

0 KEY-A
1 KEY-B

KEY-ADR: (0x0 .... 0xF)

Address of the Key which will be used for authentication.

DB_ADR (0x00 .... 0xFF):

depending on the setting of the MAD bit in parameter SCAN-DATA this parameter can have two functions

case MAD = 0

DB_ADR defines the absolute Transponder address of the first data block which will be transferred in Scan-Mode. The maximum address depends on the memory size and organization of the respective Transponder (see 9. Supported ISO Host commands

case MAD = 1

DB_ADR defines the relative data block address within one mifare sector which will be transferred in Scan-Mode in MAD function, if a mifare classic Transponder is currently detected by the reader. The maximum address range depends on the memory size and organization of the respective mifare Transponder as displayed in the following table.

	Sector 0...15	Sector 16...31	Sector 32...39
mifare 1k	0...2(3)	-	
mifare 4k	0...2(3)		0...14(15)

Values in brackets () includes the mifare sector trailer block.

D_LGT:

D_LGT defines the length of raw data which are transmitted in the Scan-Mode. Depending on the selected READER-MODE (see: 3.2. CFG1: Interface) D_LGT will be interpreted in different ways. The Parameter D_LGT has only effect to the transmission of a data blocks.

NOTICE:
In case of a mifare classic Transponder the maximum range of D_LGT and D_START are limited by the end of the mifare sector.

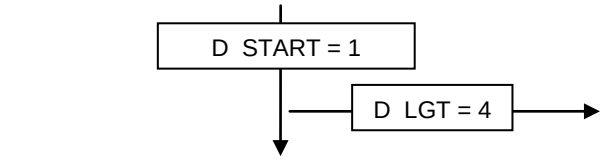
Case Scan-Mode via asynchronous interface:

D_LGT = Number of **data bytes** to be transferred, started with the D_START.

NOTICE:
D_LGT must be less than 128 byte. Otherwise the reader truncates the supernumerary bytes.

Example: Data Block

Byte	0	1	2	3	4	5	6	7
Data	0x01	0x23	0x45	0x67	0x89	0xAB	0xCD	0xEF



Output Data	0x23	0x45	0x67	0x89

Case Scan-Mode via data-/clock interface:

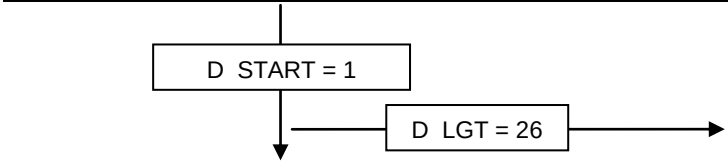
D_LGT = Number of **data bits** to be transferred, started with the D_START.

0: D_LGT = 256 bit.
1...255: D_LGT = Parameter value

In case if DB-FORMAT = ASCII format, the number of D_LGT data bits must be multiplied with 2 to get the whole data block

Example: data block

Byte	0	1	2	3	4	5	6	7
Data	0x01	0x23	0x45	0x67	0x89	0xAB	0xCD	0xEF



Output Data	b0010 0011	b0100 0101	b0110 0111	b10

D_START:

This parameter defines the starting byte in the raw data on which D_LGT starts. The Parameter D_START has only effect to the transmission of a Data Block, defined by DB_ADR.

3.8. CFG7: Scan-Mode2

The CFG7 configuration block contains additional parameters to format the data output in Scan Mode.

NOTICE:
If an USB-Reader is used:

- ***The data are transferred in HID interface mode (keyboard emulation) to the host like an additional keyboard.***
- ***In some cases it might be necessary to switch in the NUM-LOCK function on the host PC to receive the scan-mode data.***
- ***All data's will be transformed into UNICODE format and then transferred with USB-Keycode.***
- ***Independent of any setting the hex raw data of the UID will separated into their nibbles and then transformed into ASCII signs according Table 1: Hex data to ASCII conversion table.***
- ***Special characters like separators or end characters are not influenced by any format setting.***

Byte	0	1	2	3	4	5	6
Contents	DB-USE	SEP-CHAR	SEP-USR	END-CHAR	END-USR	0x00	0x00
Default	0x02	0x20	0x2C	0x01	0x0D	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

DB-USE:

Defines the output data format for scan mode data (Data Block and UID)

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	DB-FORMAT			

DB-FORMAT

depending on the SCAN-MODE interface of the kind of data interpretation and the data format can be configured.

USB interface:

b0000 unformatted hex data
In this case the raw data of a data block first are separated into their nibbles and then transformed into ASCII char according Table 1: Hex data to ASCII conversion table and are transmitted in UNICODE format.

This setting is recommended for numeric information's

b0010 ASCII interpreted hex data
In this case each raw data byte is interpreted as an ASCII char and is transformed UNICODE format.

This setting is recommended for alpha-numeric information's

Asynchronous or data-/clock interface

b0000 unformatted hex data
In this case the data are transferred as they were read from the Transponder.

b0010 ASCII formatted hex data
In this case the raw data bytes from the Transponder first are separated into their nibbles and then changed into ASCII chars according Table 1: Hex data to ASCII conversion table.

raw data (hex / binary)		ASCII data (ASCII / hex)	
0x0	b0000	'0'	0x30
0x1	b0001	'1'	0x31
0x2	b0010	'2'	0x32
0x3	b0011	'3'	0x33
0x4	b0100	'4'	0x34
0x5	b0101	'5'	0x35
0x6	b0110	'6'	0x36
0x7	b0111	'7'	0x37
0x8	b1000	'8'	0x38
0x9	b1001	'9'	0x39
0xA	b1010	'A'	0x41
0xB	b1011	'B'	0x42
0xC	b1100	'C'	0x43
0xD	b1101	'D'	0x44
0xE	b1110	'E'	0x45
0xF	b1111	'F'	0x46

Table 1: Hex data to ASCII conversion table

SEP-CHAR:

Selects the separation character between two data types for the send data.

Bit:	7	6	5	4	3	2	1	0
Function	USER	‘ ‘	‘,‘	‘;’	TAB	CR	LF	CR+LF

SEP-CHAR	ASCII	Hex
b0000 0000	none	none
b0000 0001	CR+LF	0x0D and 0x0A
b0000 0010	LF	0x0A
b0000 0100	CR	0x0D
b0000 1000	TAB	0x07
b0001 0000	‘;’	0x3B
b0010 0000	‘,’	0x2C
b0100 0000	‘ ‘	0x20
b1000 0000	USER	user defined in SEP-USR

Note:

Only one option can be selected.

SEP-USR:

User defined separation character.

END-CHAR:

selects the end character between two data types for the send data.

Bit:	7	6	5	4	3	2	1	0
Function	USER	‘ ‘	‘,‘	‘;’	TAB	CR	LF	CR+LF

END-CHAR	ASCII	Hex
b0000 0000	none	none
b0000 0001	CR+LF	0x0D and 0x0A
b0000 0010	LF	0x0A
b0000 0100	CR	0x0D
b0000 1000	TAB	0x07
b0001 0000	‘;’	0x3B
b0010 0000	‘,’	0x2C
b0100 0000	‘ ‘	0x20
b1000 0000	USER	user defined in END-USR

NOTICE:

Only one option could be selected.

END-USR:

User defined end character.

3.9. CFG8: ISO14443-A Transponder Settings

The CFG8 configuration block contains additional parameters to adjust the ISO 14443-A settings.

Notice:

This configuration block contains hardware relevant parameters which should be modified only from experts. A separate application note (N01110-1e-ID-C) with description of these parameters is available on request.

Byte	0	1	2	3	4	5	6
Contents	GsNReg	CWGsPReg	RFCfgReg	0x00	ModWidth 106 kBaud	ModWidth 212 kBaud	ModWidth 424 kBaud
Default	0xF4	0x3F	0x50	0x00	0x26	0x11	0x09

Byte	7	8	9	10	11	12	13
Contents	ModWidth 848 kBaud	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x04	0x00	0x00	0x00	0x00	0x00	0x00

3.10. CFG9: ISO14443-B Transponder Settings

The CFG9 configuration block contains additional parameters to adjust the ISO 14443-B settings.

Notice:

This configuration block contains hardware relevant parameters which should be modified only from experts. A separate application note (N01110-1e-ID-C) with description of these parameters is available on request.

Byte	0	1	2	3	4	5	6
Contents	GsNReg	CWGsPReg	ModIndex	RFCfgReg	TypeBReg [C1]	TypeBReg [C2]	0x00
Default	0xFF	0x3F	0x0E	0x40	0x80	0xC0	0x00

Byte	7	8	9	10	11	12	13
Contents	0x00	0x00	0x00	0x00	0x00	MOD-IDX LowerLimit	MOD-IDX UpperLimit
Default	0x00	0x00	0x00	0x00	0x00	0x08	0x0E

3.11. CFG10: Scan Mode - Mifare DESFire Settings

The configuration block CFG10 contains setting which has to be defined, if a data or a record file from a mifare DESFire shall be read in Scan-Mode.

To read a mifare DESFire file in scan-mode the scan-mode has to enabled in CFG1 (see 3.2. CFG1: Interface) and the reading of Data-Blocks has to be enabled (SCANTDATA1, bit DB = b1, 3.7. CFG6: Scan-Mode1). Please consider that the Driver for ISO14443-A (see 3.4. CFG3: RF-Interface, Parameter TAG-DRV) and also the ISO14443-DRV, L4 Bit has to enabled.

The reader internally uses the [0xC1] SoftCrypto commands (see 6.4. [0xC1] / [0xC3] ISO Host Commands for mifare DESFire Communication) to read and decipher the DESFire data.

Byte	0	1	2	3	4	5	6
Contents	DF_AID			DF_KEY_NO	READER_ KEY_IDX	DF_FILE_ID	DF_FILE _SETTINGS
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00
	LSB		*MSB*				

Byte	7	8	9	10	11	12	13
Contents	DF_FILE_OFFSET			DF_FILE_ LENGTH	RFU	RFU	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00
	LSB		*MSB*				

DF_AID:

DESFire Application ID in which the requested file (see DF_FILE) is stored.

DF_KEY_NO (0x0...0xE)

Defines the number of the key to which the authentication shall be processed on the DESFire to get access to the data. The KEY-No on DESFire is specified in the access rights for each file.

0x0....0xD:

DESFire Key no for enciphered access
If a data exchange with a enciphered or MACed file shall be processed the respective KEY-NO has to used here

0xE:

DESFire Key No for free access.

This setting is possible for "read", "write" and "read&write" access rights for each file on the card. If this setting is used for "read&write" access rights a different key setting for "read" access rights or "write" access rights becomes obsolete.

NOTICE:
Depending on the DESFire KEY-NO and FILE-COMM-SETTINGS different security conditions can be configured. The following table shows the possible combinations and how the security functions are influenced by this settings.

DESFire KEY-NO	FILE-COMM-SETTINGS	file access conditions	
		valid authentication required	data exchange
0x0...0xD	0x00	yes	plain
	0x01		plain + MAC
	0x03		enciphered
0xE	0x00 0x01 0x03	no	plain

READER-KEY-IDX (0...3)

Index of the key which is stored in the reader and which shall be used for authentication for the current command.

Notice
The key which is addressed with READER-KEY-IDX defines the authentication command and method. The addressed application has to configured in the same way.

DF_FILE

This parameter defines which file from a DESFire shall be read.

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	**FILE-NO**				

FILE-NO 0x0...0x1F:

No of the file which shall be read.

DF_FILE_SETTINGS

This parameter defines how a file from a DESFire shall be read.

Bit:	7	6	5	4	3	2	1	0
Function	**FILE-TYPE**		-	-	-	-	**FILE-COMM-SETTINGS**	

FILE-TYPE

coding of the file type of the dedicated file.

b00: Standard Data or Backup Data File

b01: Cyclic or Linear Record File

FILE-COMM-SETTINGS

coding of the communication mode of the dedicated file.

b00: Plain communication

b01: Plain communication secured by MACing

b11: Fully enciphered communication

DF_FILE_OFFSET:

Defines the position within a DESFire file where the read process shall start.

FILE-TYPE	FILE-	DF_FILE_OFFSET
b00	Standard Data File	Defines the byte position within a DESFire Data file where the read process shall start.
	or	
	Backup Data File	
b01	Cyclic Record File	Defines the record no. which shall be read out. 0x00 00 00 defines the latest written record 0x01 00 00 defines next the latest written record
	or	
	Linear Record File	

DF_FILE_LENGTH (1 ...128 Byte):

FILE-TYPE	FILE-	DF_FILE_LENGTH
b00	Standard Data File	Defines the number of bytes which shall be read.
	or	
	Backup Data File	
b01	Cyclic Record File	Defines the size of the record (in bytes) which shall be read. Data are only transmitted if parameter DF_FILE_LENGHT is equal to the DESFire record length
	or	
	Linear Record File	

Notice:

If parameter SCAN-MODE1, D_LGT < DF_FILE_LNGTH

superfluous data are truncated

If parameter SCAN-MODE1, D_LGT > DF_FILE_LNGTH

remaining data are filled with 0

3.12. CFG11: Tag Driver Priority List

The configuration block CFG11 contains a user definable call order of the tag drivers during the command [0xB0][0x01] Inventory.

The settings of CFG11 are only active, if bit CFG3.TAG-DRV.SEL is set.

Byte	0	1	2	3	4	5	6
Contents	TAG DRIVER 1	TAG DRIVER 2	TAG DRIVER 3	TAG DRIVER 4	TAG DRIVER 5	TAG DRIVER 6	TAG DRIVER 7
Default	0x04	0x05	0xFF	0xFF	0xFF	0xFF	0xFF

Byte	7	8	9	10	11	12	13
Contents	TAG DRIVER 8	TAG DRIVER 9	TAG DRIVER 10	TAG DRIVER 11	TAG DRIVER 12	TAG DRIVER 13	TAG DRIVER 14
Default	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

A list of supported tag drivers can be found in ANNEX A: Codes of Transponder Types.

A plausibility check is performed by writing these parameters to the Reader. If an error occurs the Reader answers with STATUS = 0x11.

3.13. CFG12: Reserved

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

3.14. CFG13: Reserved

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00	0x00	0x00

3.15. CFG14: Customer Parameters

CFG14 configuration block can be used to store any kind of customer identifiers.

Byte	0	1	2	3	4	5	6
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default							

Byte	7	8	9	10	11	12	13
Contents	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Default							

3.16. CFG15: Passwords

The parameters of the CFG15 configuration block contain the access conditions for the Reader configuration.

The Reader configuration can be protected by a 32-bit password, the "READER-ID". This means that only after a "Login" with a valid READER-ID with the command 5.9. [0xA0] Reader-Login configuration parameters in the EEPROM of the Reader may be read or written.

Byte	0	1	2	3	4	5	6
Contents	READER-ID				CFG_ACCESS_RD_WR		
Default	0x00000000				0x00	0x00	0x00

Byte	7	8	9	10	11	12	13
Contents	CFG_ACCESS_RD_WR					0x00	0x00
Default	0x00	0x00	0x00	0x00	0x00		

READER-ID:

Defines the password with which the host logs into the Reader for a read / write access to the configuration parameter blocks.

The READER-ID can be changed only immediately after a valid Login with the command 5.9. [0xA0] Reader-Login.

Notes:

- ***A READER-ID = 0x00000000 disables the password function.***
- ***A changed password becomes valid after a Reader reset.***

CFG_ACCESS_RD_WR:

Defines the Configuration blocks which are read/write accessible only after a successful login to the Reader.

Byte:	4								5							
Bit:	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
CFG No.	3		2		1		0		7		6		5		4	

Byte:	6								7							
Bit:	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
CFG_NO.	11		10		9		8		15		14		13		12	

Byte:	8								9							
Bit:	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
CFG_NO.	RFU		RFU		RFU		RFU		RFU		RFU		RFU		RFU	

Byte:	10								11							
Bit:	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
CFG_NO.	RFU		RFU		RFU		RFU		RFU		RFU		RFU		RFU	

CFG_NO

The Bit in CFG_NO defines if the access to the configuration block is free or protected by reader login.

b00 ⇒ Access is free

b01 ⇒ Read Access login protected

b10 ⇒ Write Access login protected

b11 ⇒ Read and Write Access login protected

Notes:

The command 4.4. [0x83] Set Default Configuration ***don't change the CFG15 register if all configuration blocks are used.***

4. Commands for Reader Configuration

Via the command protocols for the Reader configuration, the Reader may be adapted to individual conditions of application within wide limits.

4.1. [0x80] Read Configuration

By using the Read Configuration command the actual configuration of the Reader can be read out. The configuration is organized in blocks of 14 bytes each and addressed by CFGn in the byte CFG-ADR.

REQUEST-DATA

1
CFG-ADR

RESPONSE-DATA

14
CFG-REC

CFG-ADR[1]:

Bit:	7	6	5	4	3	2	1	0
Function	LOC	0	CFGn: Address of Configuration Block					

CFGn:

Memory-address of the required configuration block.

LOC:

Specifies the location of the configuration block.

b0: RAM

b1: EEPROM

CFG-REC:

14-byte configuration block read from address CFGn in CFG-ADR.

NOTICE:

A read configuration from EEPROM with reserved configuration blocks will cause error code 0x15.

[1] see Chapter 3. Configuration Parameters (CFG)

4.2. [0x81] Write Configuration

Via the command Write Configuration the configuration of the Reader can be changed. The configuration memory organized in 14 bytes long blocks and addressed by CFGn in the byte CFG-ADR. The description of parameters can be taken from chapter 3. Configuration Parameters (CFG)

REQUEST-DATA

1	14
CFG-ADR	CFG-REC

RESPONSE-DATA

0
-

CFG-ADR[1]**:**

Bit:	7	6	5	4	3	2	1	0
Function	LOC	0	CFGn: Address of Configuration Block					

CFGn: Memory-address of the required configuration block.

LOC: Specifies the location of the configuration block.
b0 RAM
b1 EEPROM and RAM

CFG-REC:

14-byte configuration block stored in the configuration memory of the Reader at address CFGn.

NOTICE:
A write configuration to EEPROM with reserved configuration blocks will cause error code 0x16.

[1] see chapter 3. Configuration Parameters (CFG)

4.3. [0x82] Save Configuration

By the command Save Configuration each configuration block of the RAM can be stored in EEPROM.

REQUEST-DATA

1
CFG-ADR

RESPONSE-DATA

0
-

CFG-ADR[1]**:**

Bit:	7	6	5	4	3	2	1	0
Function	0	MODE	CFGn					

CFGn: Memory-address of the required configuration block.

MODE: Specifies one or all configuration blocks.
- b0: configuration block specified by CFGn
- b1: all configuration blocks

NOTICE:
A write configuration to EEPROM with reserved configuration blocks will cause error code 0x16.

[1] see chapter 3. Configuration Parameters (CFG)

4.4. [0x83] Set Default Configuration

Using the command Set Default Configuration each configuration block can be reset to the manufacturer's setting.

REQUEST-DATA

1
CFG-ADR

RESPONSE-DATA

0
-

CFG-ADR:

Bit:	7	6	5	4	3	2	1	0
Function	LOC	MODE	CFGn					

CFGn: Memory-address of the required configuration block.

MODE: Specifies one or all configuration blocks.
- b0: configuration block specified by CFGn
- b1: all configuration blocks

LOC: Specifies the location of the configuration block.
- b0: RAM
- b1: EEPROM

Notes:

- ***To store RAM configuration over power down see chapter*** 4.3. [0x82] Save Configuration
- ***A set default configuration with reserved configuration blocks will cause an error code.***

4.5. [0xA2] Write Mifare Reader Keys

The keys which are required by the Reader in order to authentify itself to a Mifare classic Transponder (mifare classic mini, 1k, 4k) will be stored by this command. Only if the keys of the Reader and of the Transponder correspond, the data exchange between Reader and Transponder can be effected.

REQUEST-DATA

1	1	6
KEY-TYPE	KEY-ADR	KEY

RESPONSE-DATA

0
-

KEY-TYPE:

Defines the key for the authentication.

0x00 KEY-A
0x01 KEY-B

KEY-ADR: (0x00 .... 0x07)

Address where the key is stored in the reader. The address can be any value between 0 and 7.

KEY:

Mifare: 6 byte Key

Notes:

- ***It is not possible to read back the keys off the Reader. After having changed the keys these should be stored at a secured place.***
- ***The factory adjustment of the keys on KEY-ADR 0x00 is:***
  KEY-A: 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF
  KEY-B: 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

5. Command for Reader Control

5.1. [0x52] Baud Rate Detection

This command serves to determine the actual baud rate of the Reader's asynchronous interface.

REQUEST-DATA

1
0x00

RESPONSE-DATA

0
-

NOTICE:

- ***The return protocol will only be sent if the inquiry is executed with the baud rate and actual parity of the Reader.***

5.3. [0x63] CPU Reset

This command allows you to reset the CPU on the Reader.

REQUEST-DATA

0
-

RESPONSE-DATA

0
-

NOTICE:

The RF-field will be switched off while a CPU Reset.

5.4. [0x65] Get Software Version

This command allows you to determine the software version of the Reader, its type and the types of the Transponders which are supported by the software.

REQUEST-DATA

1
MODE

RESPONSE-DATA

2	1	1	1	2
SW-REV	D-REV	HW-TYPE	SW-TYPE	TR-TYPE

SW-REV:

Version of the firmware.

D-REV:

Revision status of the firmware.

HW-TYPE:

Displays options which are supported by the Reader hardware

Bit:	7	6	5	4	3	2	1	0
Function:	ANT	MODEL	-	SD2	HWE		SD1	SE

SE[1]:

b0: The Reader is not equipped with any SAM socket.
b1: The Reader is equipped with one or more SAM sockets.

SD1[2]:

This bit indicates if a SAM is inserted into SAM socket 1.
b0: No SAM inserted.
b1: SAM inserted

SD2[3]:

This bit indicates if a SAM is inserted into SAM socket 2.
b0: No SAM inserted.
b1: SAM inserted

HWE:

RF-Decoder type of the Reader.

[1] Status after power up sequence.
[2] Status after power up sequence.
[3] Status after power up sequence.

MODEL:

	b0	b1
ID CPR40.xx	-	-Ax / -Cx / -Ux

ANT:

This bit indicates if the reader is equipped with an internal antenna or it is a model for an external antenna.

b0: external antenna (depending on reader type 1 or more external antennas)

b1: internal antenna

SW-TYPE:

Displays the type / model of the Reader
(see: ANNEX B: Codes of Reader Types)

TR-TYPE:

Displays the Transponders supported by the Reader.

Bit:	15	14	13	12	11	10	9	8
Function:	-	-	-	-	SRIx	SR176	-	Jewel

Bit:	7	6	5	4	3	2	1	0
Function:	-	-	ISO 14443B	ISO 14443A	-	-	-	-

5.5. [0x66] Get Reader Info

This command allows you to determine, a lot of firmware and hardware options and version from the reader. Most information's are only required for service and support questions.

REQUEST-DATA

1	1
[0x66]	MODE

RESPONSE-DATA

Depending on the MODE Parameter the reader response has different structures with several information's which are described in the next chapters for each MODE separate.

MODE:

Via the Parameter MODE different information could requested from the Reader.

0x00: General hard- and firmware information's

0x03: RF-decoder information's for factory diagnostic.

0x04: Additional firmware functionality.

0x05: Bootloader version information.

0x05: Bootloader version information.

0x08: Supported Tag Drivers information.

0x11: SAM Information

0x80: Device-ID: Necessary Information's for firmware updates and firmware upgrades.

0xFF: Read all info modes

5.5.1. Mode = 0x00 (RF Controller Firmware)

RESPONSE-DATA

2	1	1	1	2	2	2
SW-REV	D-REV	HW-TYPE	SW-TYPE	TR-TYPE	RX-BUF	TX-BUF

SW-REV / D-REV / HW-TYPE / SW-TYPE / TR-TYPE:

see: This command allows you to reset the CPU on the Reader.

REQUEST-DATA

0
-

RESPONSE-DATA

0
-

NOTICE:

The RF-field will be switched off while a CPU Reset.

5.4. [0x65] Get Software Version

RX-BUF:

RX-BUF is the maximum receive buffer size of the Reader. If a protocol from the host exceed the RX-BUF size the Reader response with error code 0x81 PROTOCOL LENGTH ERROR.

TX-BUF:

TX-BUF is the maximum transmit buffer size of the Reader. The host has to take in to account that a response protocol of the Reader can have this length.

5.5.2. Mode = 0x03 (RF-decoder information's)

RESPONSE-DATA

2	1	5
DEC_TYPE	SELF_TST	<RFU>

DEC_TYPE

Information's about the functionality and revision of the RF-decoder for service and support.

SELF_TST

This byte gives information's about the self test result, which is performed automatically by the reader after a power on reset.

0x00: Self test not OK
The reader has detected an internal failure.

0x01: Self test OK.

5.5.3. Mode = 0x04 (Additional firmware functionality)

RESPONSE-DATA

1	1	1	6	2
TEMPLATE	FNC_LST0	FNC_LST1	<RFU>	FW_IDENT

TEMPLATE:

Indicates how to interpret the following content depending on the reader type

0x01: ID CPR-Family

FNC_LST0:

Each bit represents a firmware functionality.

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	MFP	MFD	MAD	TCL

TCL:

Indicates the support of the T=CL Function (Command [0xB2][0xBE])

b0: T=CL function is not supported

b1: T=CL function is supported

MAD:

Indicates the support of MAD (Mifare Application Directory) in Scan-Mode

b0: MAD is not supported

b1: MAD is supported

MFD (Mifare DESFire high level crypto function):

Indicates the support of crypto functions with DESFire Commands [0xC1][0x##] and [0xC3][0x##].

b0: Mifare DESFire crypto functions are not supported

b1: Mifare DESFire crypto functions are supported

MFP (Mifare PLUS high level crypto function):

Indicates the support of crypto functions with Mifare Plus Commands [0xC2][0x##].

b0: Mifare Plus crypto functions are not supported

b1: Mifare Plus crypto functions are supported

FNC_LST1:

Each bit represents a firmware functionality.

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0			EMV Slot2	EMV Slot1	EMVco

EMVco:

Indicates EMVco support (contactless) [Level 1]

b0: EMVco is not supported

b1: EMVco is supported

EMV Slot 1:

Indicates EMV support (contact) of contact Slot 1 [Level 1]

b0: EMV is not supported

b1: EMV is supported

EMV Slot 2:

Indicates EMV support (contact) of contact Slot 2 [Level 1]

b0: EMV is not supported

b1: EMV is supported

5.5.4. Mode = 0x05 (Bootloader version information)

RESPONSE-DATA

1	1	9
BL_VERSION	BL_REF	<RFU>

BL_VERSION:

Bootloader Version

BL_REV:

Revision of Bootloader Version

5.5.5. Mode = 0x08 (Supported Tag Drivers)

RESPONSE-DATA

30
TAG DRIVERS

TAG_DRIVERS:

List of supported Tag Drivers by the reader [starting with lowest number]
A list of supported tag drivers can be found in ANNEX A: Codes of Transponder Types.

5.5.6. Mode = 0x11 (SAM Information)

This mode can be used to check how many SAM slots are available and if a SAM is inserted into the SAM slot.

RESPONSE-DATA

1	NO_OF_ SLOTS

NO_OF_ SLOTS	SLOT_INFO

NO_OF_SLOTS:

No of available SAM slots.

SLOT_INFO:

Bit:	7	6	5	4	3	2	1	0
CFG_NO	Slot-Type				-	-	-	inserted

Slot-Type:

0000: Universal SAM socket for ID000 SAM

inserted:

b0 No SAM inserted

b1 SAM is inserted

5.5.9. Mode = 0x80 (Device_ID)

RESPONSE-DATA

4	12
DEV_ID	<RFU>

DEV_ID:

Individual device identifier of the Reader.

5.5.10. Mode = 0xFF (Read all info modes)

MODE 0xFF returns all Info modes within one response.

RESPONSE-DATA

1	1	30
DATASETS	MODE	DATA

Repeated DATASETS times

DATASETS:

Number of transferred datasets (each with a length of 31 byte)

MODE:

Mode byte

DATA:

Data record according to the definition in previous sections.
The data record is always 30 byte long.
Unused bytes are filled with 0x00.

5.6. [0x69] RF Reset

The RF-field of the Reader antenna can be switched off for approx **6 ms** by the command RF Reset. Thus, all Transponders which are within the antenna field of the Reader will be reset to their base setting.

REQUEST-DATA

0
-

RESPONSE-DATA

0
-

NOTES:

- ***After the RF Reset command the Reader is not able to receive any new Transponder before expiration of t_{rf} .***
- ***After a RF Reset a Transponder which is located within the field has to be re-selected.***

5.7. [0x6A] RF Output ON/OFF

The command RF ON/OFF switches the RF field of the Reader antenna ON or OFF.

REQUEST-DATA

1
RF_OUTPUT

RESPONSE-DATA

0
-

RF_OUTPUT:

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	ANT_OUT	

ANT_OUT:

This parameter could be used to select one antenna.

b00: switches off RF power at all antennas.

b01: switches on the RF power at antenna 1.
This setting has to be used for readers with only one internal or one external antenna.

5.8. [0x72] Set Output

The command Set Output serves temporary limited or unlimited activation of the digital reader outputs.

Each output takes on the state defined by the byte OUTx-mode for the period of time (OUT-TIME) defined in this command. If the Reader receives a Set Output command, all times that have been active until then are being overwritten by the new times if they are > 0.

REQUEST-DATA

1	1	1	1	1	2
[0x72]	Mode	OUT-N	OUT-NR	OUT-S	OUT-TIME
			Repeated OUT-N times		

RESPONSE-DATA

0
-

Mode: 0x00

OUT-N:

Defines the number of output records.

OUT-NR:

Defines the type and the number of the output

Bit:	7	6	5	4	3	2	1	0
Function:	OUT-Typ			0	OUT-Number			

OUT-Typ:

b001 LED

OUT-Number:

b0001 LED V1 (green)
b0010 LED V2 (blue)[1]
b0011 LED V3 (red)[2]

[1] Only supported by ID CPR40.3x
[2] Only supported by ID CPR40.0x

OUT-S:

OUT-S (Output State) defines the status of the output during the time defined in OUT-TIME and provides the possibility to allocate its own flashing-frequency to each output.

Bit:	7	6	5	4	3	2	1	0
Function:	0	0	0	0	0	0	OUTx-mode	

OUTx-mode:

b00: OFF
b01: ON
b10: FLASHING SLOW
b11: FLASHING FAST

OUT-TIME:

By the values defined by “OUT-TIME” the outputs can be activated temporary limited or unlimited.
Accepted are the time values 0 and 65535 (0xFFFF) (see following table).

0x0001	1 x 100ms	-> 100ms
...	...	
0xFFFE	65534 x 100ms	-> 1:49:13 h
0xFFFF	continuously active	

Notes:

- ***In order to reset a continuously active time, OUT-TIME = 1 has to be sent to the Reader, which effects a change to the idle status after 100 ms.***
- ***The continuous activation is being set back after a reset or a power failure.***
- ***The command is not influenced by the settings of*** 3.3. CFG2: Inputs / Outputs general

5.9. [0xA0] Reader-Login

The Reader-Login must be executed after every power up or *5.3.* [0x63] CPU Reset command, if an access to the configuration parameters is desired.

REQUEST-DATA

4
READER-ID

RESPONSE-DATA

0
-

READER-ID:

The READER-ID is a password which protects the configuration parameters from any read and write access.
The READER-ID can be changed in the configuration block 3.16. CFG15: Passwords.

NOTE:

- ***A Reader-Login with wrong READER-ID cause a "Logout".***
- ***A "Logout" can be effected by the command*** *5.3.* [0x63] CPU Reset.

6. ISO Host Commands for Transponder Communication

In the following chapters the Host commands for communication with a Transponder according are described. Notice that not all commands are available for each Transponder type. Detailed information about the supported ISO Host commands are described in chapter 9. Supported ISO Host commands. for each Transponder type separate.

6.1. [0xB0] ISO Standard Host Commands

This command sends standard RF commands to the Transponder.

REQUEST-DATA

1	(X)
SUB-COMMAND	PARAMETER

RESPONSE-DATA

(X)
RESPONSE-DATA

SUB-COMMAND, PARAMETER:

Command specific request with variable length. The content is described in the next chapters.

RESPONSE-DATA:

Command specific response with variable length. The content is described in the next chapters.

Notes:

* ***Data are only transferred if STATUS = 0x00, 0x83, 0x94, 0x95.***
* ***These commands are not available if Scan-Mode is active.***

6.1.1. [0x01] Inventory

This command reads the UID of Transponders which are located in the detection range of the reader. The reply behavior of this command depends on several settings.

REQUEST-DATA

1	1	(1)
[0x01]	MODE	NTFC_TIME

NOTES:

- ***The operating behavior of the Inventory command depends on some settings in CFG5, parameter ONT and on settings in MODE byte***
  - ***If the CFG5.ONT Bit ONT = b1 only the response of those Transponders are read which came into the antenna field since the last Inventory command.***

    In this case The Reader response includes an UID only if:
    - ***the Transponder has left the antenna field and reentered the antenna field or***
    - ***the command MODE 0xFF returns all Info*** modes within one response.

RESPONSE-DATA

1	1	30
DATASETS	MODE	DATA

Repeated DATASETS times

DATASETS:

Number of transferred datasets (each with a length of 31 byte)

MODE:

Mode byte

DATA:

Data record according to the definition in previous sections.
The data record is always 30 byte long.
Unused bytes are filled with 0x00.

- 5.6. [0x69] RF Reset ***was send to the Reader meanwhile or***
- ***the Transponder in the antenna field is a Jewel***

- ***If the CFG5.ONT Bit ONT = b0 a RF-Reset is performed to read the UID of all Transponders inside the antenna field.***
- ***If CFG5.ONT Bit ACOLL = b0 (anticollision function is disabled) the Reader selects the Transponder itself.***
- ***If MODE bit PRESC = b1 the response includes the Transponder UIDs without performing a RF-Reset.***

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	MORE	NTFC	PRESC	-	-	-	-	-

PRESC:

Settling of this bit activates the presence check mode of the Inventory command. This setting is suitable to perform a presence check of all Transponder in detection range of the reader without influencing the UID of Transponder with a random UID

b0: Presence check is deactivated

b1: The response of the Inventory command [0x01] includes the UID of all detected Transponders in the reader detection range.

NOTICE:
The* PRESC = *b1 can only used if CFG5.ONT, ONT bit and ACOLL bit is set to b1 (see CFG5.ONT)

NTFC:

Settling of this bit activates the notification mode of the Inventory command.

b0: Standard Inventory command

b1: Inventory with notification:
In this case the optional parameter NTFC_TIME must send to the reader.

In notification mode the Inventory command runs internally while one ore more Transponders are detected by the reader or while the time defined by NTFC_TIME elapsed.

MORE:

this bit can be used, to read out the whole UIDs, after the Reader had signalized more data sets with status 0x94 (see: ANNEX C: Index of Status Bytes)**.**

b0: new Inventory requested
The reader carries out a new inquiry, which Transponder are in his detection range.

b1: more data requested
The reader response contain the UIDs which are not transferred with the last response because of the status 0x94.

NOTICE:
The MORE and NTFC function can be used only exclusive.

NTFC_TIME:

This optional parameter defines the maximum duration of the Inventory command in notification mode (see NTFC bit in MODE Byte).

	max. response duration
NTFC_TIME	0...255 * 100 ms

NOTICE:

- ***The NTFC_TIME overwrites the TR-RESPONSE-TIME which is defined in CFG1. The receive block timeout of the host computer must set to a value ≥ NTFC_TIME.***
- ***A running Inventory command with NTFC option couldn't be interrupted by any other command while NTFC_TIME.***

DATA-SETS:

Number of Transponder data sets to be transferred in this reader response.

TR-TYPE:

Bit:	7	6	5	4	3	2	1	0
Function	RF_TEC		-	-	TYPE_NO			

RF_TEC:

Indicates the RFID - Technology of the present Transponder:
b00: 13,56 MHz Transponder
b10: UHF Transponder

TYPE_NO

Displays the Transponder type of the present Transponder
(see: ANNEX A: Codes of Transponder Types).

RESPONSE-DATA:

Depending on the Transponder type the response data of the Reader are different as described in the following chapters.

6.1.1.1. Response-Data - ISO 14443A (TR-TYPE = 0x04)

Response data of ISO 14443 Type A compliant Transponder:

RESPONSE-DATA - Case CFG3. ISO14443 FTUR.OPTI = b0 ⇒ OPT_INFO is disabled

1	1	1	1	8 (10)
DATA-SETS	TR-TYPE	TR_INFO	0	UID
	Repeated DATA-SETS times			

RESPONSE-DATA - Case CFG3. ISO14443 FTUR.OPTI = b1 ⇒ OPT_INFO is enabled

1	1	1	1	8 (10)
DATA-SETS	TR-TYPE	TR_INFO	OPT_INFO	UID
	Repeated DATA-SETS times			

TR_INFO (only ISO 14443A Transponder)**:**

This byte represent some information's from the SAK byte as described in ISO14443-3 ([1].

Bit:	7	6	5	4	3	2	1	0
Function	-	-	L4	-	-	CL3	-	-

CL3: Displays the UID length of the present Transponder.

b0 The UID is transmitted as a 7 byte field
(Transponder with Cascade Level 1 or Level 2)

b1 The UID is transmitted as a 10 byte field
(Transponder with Cascade Level 3)

L4: Displays the compliance of the Transponder with ISO 14443-4 according ISO 14443-3, SAK, b6

b0 Not compliant with ISO 14443-4

b1 Compliant with ISO 14443-4

[1] In case of NXP mifare chips this byte also indicates the chip type. Further information's are given in the NXP Application Note "mifare Interface Platform, Type Identification Procedure" M018412.

OPT_INFO (only ISO 14443A Transponder)**:**

Depending on the setting of CFG3.ISO14443 FTUR.OPTI this byte could optional display further information's about the present Transponder.

It's recommend to use this information if ISO14443-4 Transponder or Transponder with more the 4 byte UID length shall be handled by the reader.

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	L4_SLCT	C_LEVEL	

C_LEVEL:

This 2 bits displays the Cascade Level of the Transponder UID

b00: Cascade Level 1 (4 byte UID)

b01: Cascade Level 2 (7 byte UID)

b10: Cascade Level 3 (10 byte UID)

L4_SLCT:

This bit displays the select status of the present Transponder.

b0: The Transponder is not selected in ISO14443-4 level.

b1: The Transponder is selected on ISO14443-4 level by the reader now. A further select command is not necessary for data exchange with this Transponder.

UID:

ISO 14443A UID could have different lengths. This depends on the Cascade Level of the Transponder (see also TR_INFO byte). It is transmitted by the reader with a length of 7 or 10 byte.

The following table shows the structure of the UID in relation to ISO14443-3

transmitted byte	9	10	11	12	13	14	15	16	17	18
Cascade-Level 1	0	0	0	UID3	UID2	UID1	UID0	-	-	-
Cascade-Level 2	UID6	UID5	UID4	UID3	UID2	UID1	UID0(*	-	-	-
Cascade-Level 3	UID9	UID8	UID7	UID6	UID5	UID4	UID3	UID2	UID1	UID0(*

* UID0: Manufacturer ID according ISO/IEC7816-6/AM1

In case of a shorter UID the redundant bytes are filled with 0 at the most significant digits.

Notice:

The UID byte order can be swapped by using the parameter CFG3.ISO14443 FTUR.UID_ORDER

6.1.1.2. Response-Data - ISO 14443B (TR-TYPE = 0x05)

Response data of ISO 14443 Type B compliant Transponder:

RESPONSE-DATA

1	1	1	4	4
DATA-SETS	TR-TYPE	PROTO INFO	APP DATA	PUPI LSB / MSB
	Repeated DATA-SETS times			

PROTO_INFO

This parameter is extracted from the protocol Info bytes as described in ISO14443-3.

Bit:	7	6	5	4	3	2	1	0
Function	Max_Frame_Size				Protocol_Type			

Max_Frame_Size (according ISO14443-3:2001)

Value	0	1	2	3	4	5	6	7	8	9-F
Frame Size (Byte)	16	24	32	40	48	64	96	128	256	RFU > 256

Protocol_Type (according ISO14443-3:2001)

7	6	5	4	Meaning
0	0	0	1	PICC compliant with ISO/IEC 14443-4
0	0	0	0	PICC not compliant with ISO/IEC 14443-4

APP_DATA

4 byte Application Data (according ISO 14443-3:2001).

1	2	1
Number of Applications	CRC_B (AID) LSB / MSB	AFI

PUPI

4 byte Pseudo-Unique PICC Identifier (according ISO 14443-3:2001).

This information is required to select the Transponder.

6.1.1.3. Response-Data - Jewel (TR-TYPE = 0x08)

Response data of Jewel Transponder:

RESPONSE-DATA

1	1	1	1	6					
DATA-SETS	TR-TYPE	0	0	UID					
				HR0	HR1	UID0	UID1	UID2	UID3
	Repeated DATA-SETS times								

UID: Read-only UID of the Transponder.

In case of a shorter UID the redundant bytes are filled with 0 at the most significant digits.

HR0, HR1:

metal-mask data selected. 0x01 0x3C for IRT5001W and IRT5001E.

6.1.1.4. Response-Data - SR176 (TR-TYPE = 0x0A)

Response data of STM SR176 Transponder

RESPONSE-DATA

1	1	1	8							
DATA-SETS	TR-TYPE	CHIP_ID	UID							
			UID0	UID1	UID2	UID3	UID4	UID5	UID6	UID7
	Repeated DATA-SETS times									

CHIP_ID:

Optional fixed Chip_ID from block address 15.

UID:

64Bit UID of the Transponder.

6.1.1.5. Response-Data - SRIx (TR-TYPE = 0x0B)

Response data of STM SRI512, SRI4K SRIX4k Transponder:

RESPONSE-DATA

1	1	1	8							
DATA-SETS	TR-TYPE	CHIP_ID	UID							
			UID0	UID1	UID2	UID3	UID4	UID5	UID6	UID7
	Repeated DATA-SETS times									

CHIP_ID:

Optional fixed Chip_ID from block address 255.

UID:

64Bit UID of the Transponder.

6.1.1.6. Response-Data – Innovatron [14443-B'] (TR-TYPE = 0x10)

Response data of a transponder, that supports the Innovatron radio protocol:

RESPONSE-DATA

1	1	8	1	1	1	ATR-LEN
DATA-SETS	TR-TYPE	UID	VERLOG	CONFIG	ATR-LEN	ATR
	Repeated DATA-SETS times					

UID:

4 LSBytes of card´s serial number (filled with leading zeros).

VERLOG:

Software version and type of REPGEN.

Bit:	7	6	5	4	3	2	1	0
Function	Type	1	1	Software version				1

Type:

b0 "Short REPGEN" (No additional bytes following)
b1 "Long REPGEN" (Additional bytes following)

Software version:

Software version of the tag (0…14)

CONFIG:

Configuration of REPGEN

Bit:	7	6	5	4	3	2	1	0
Function	WE	TAB	RFU = 0					

WE (Wait enble):

b0 tag doesn´t accept WAIT command frames
(useless in batteryless mode)
b1 tag accepts WAIT command frames
(useless in batteryless mode)

TAB:

b0 no tag ATR bytes are present
b1 tag ATR bytes are present

ATR LENGTH:

Length of the following ATR

ATR:

ATR (according to ISO7816-3) of the transponder

6.1.1.7. Response-Data – CTx (TR-TYPE = 0x11)

Response data of CTx transponder.

RESPONSE-DATA

1	1	1	1	1	1	8
DATA-SETS	TR-TYPE	PRODUCT CODE	FAB CODE	APP CODE	EMBEDDER CODE	UID
	Repeated DATA-SETS times					

PRODUCT CODE:

Produce code of ASK CTx transponder.

FAB CODE:

Fabrication code of ASK CTx transponder.

APP CODE:

Application code of ASK CTx transponder

EMBEDDER CODE:

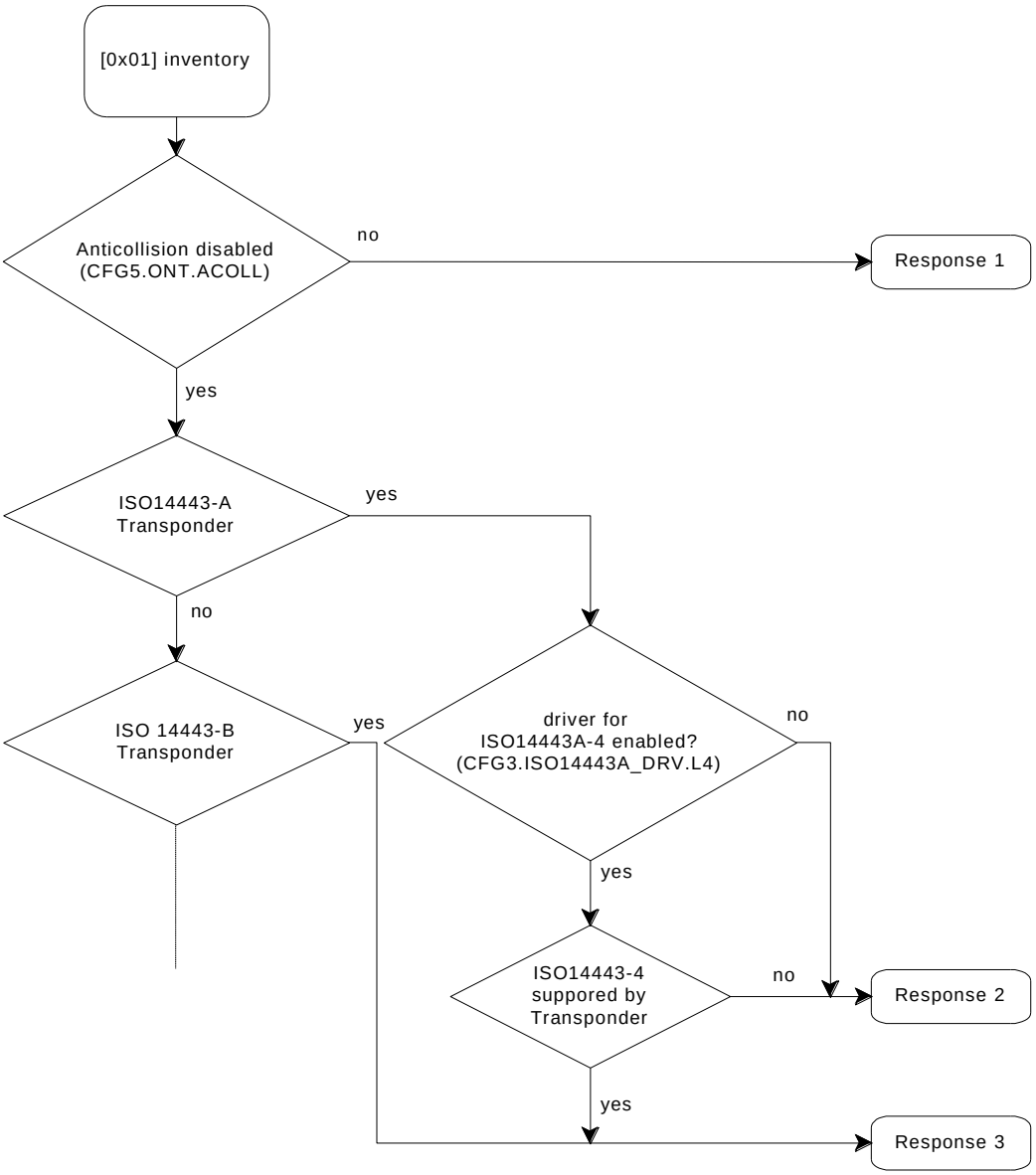
Embedder code of ASK CTx transponder

UID:

32 Bit card´s serial number. (LSB First)

6.1.1.8. Sequences of Inventory Command and ISO14443 Transponder

The following chart displays the sequences and effects after a inventory command depending on the Transponder type and the Reader configurations.



	Transponder(s) are selected	**No of announced Transponder**	**next possible commands**
Response 1	no	> 1 (possibly)	6.1.2. [0x25] Select
Response 2	yes one Transponder on ISO14443-3 level	1	Proprietary or standard commands for ISO14443 Transponders (see 6.3. [0xB2] ISO14443 Special Host Commands)
Response 3	yes one Transponder on ISO14443-4 level	1	ISO14443-4 Commands (see 6.3. [0xB2] ISO14443 Special Host Commands

6.1.2. [0x25] Select

This command sets one Transponder to the Select State. Only one ISO Transponder can be selected at once.

The supported ISO Host commands depends on the Transponder types, they are described in chapter 9. Supported ISO Host commands.

REQUEST-DATA

1	1	(1)	(1)	(8)/(UID_LEN)
[0x25]	MODE	TR_DRV	UID_LEN	UID

RESPONSE-DATA if STATUS = 0x95

1
ISO-ERROR

RESPONSE-DATA if STATUS = 0x00 and REQEST-DATA bit MODE.CINF = 1.

(1)	(X)
FORMAT	CARD_INFO

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	0	DRV_SEL	CINF	UID_LF	0	ADR		

ADR:

b001 addressed

UID_LF:

If this bit is set the parameter UID_LEN must inserted into the command.

b0: The request data doesn't include the UID_LEN byte and the UID field has a fix length of 8 byte.

b1: The request data includes the parameter UID_LEN. The UID has a variable length as defined in UID_LEN.

CINF:

b0: response data doesn’t include optional CARD_INFO

b1: response data includes optional CARD_INFO

DRV_SEL:

b0: No extra byte TR-DRV included in request data

b1: Extra byte TR-DRV included in request data

TR_DRV:

This optional extra byte offers the possibility to select explicit a Transponder driver for the current select command.

This option may be helpfull in such cases where the reader can not detect the correct type of the transponder e.g. if a processor card emulates any kind of memory card or if NFC devices are presented.

NOTICE
Make shure that the relevant TAG-DRV and ISO14443-DRV is enabled in CFG3.

Bit:	7	6	5	4	3	2	1	0
Function	TR-DRIVER-SELECTION							

TR-DRIVER-SELECTION

By means of this parameter the host application explicit selects an internal driver to handle the data exchange with the addressed transponder.

TR-DRIVER-SELECTION	Selected Transponder driver
0x01	ISO 14443-4
0x02	mifare classic
0x03	NFC Card Type 2 (e.g. mifare Ultralight, my-d move, etc.)
0x04	mifare plus SL1
0x05	mifare plus SL1 - ISO14443-4
0x06	mifare plus SL2
0x07	mifare plus SL2 - ISO14443-4
0x08	mifare plus SL3
0x09	mifare DESFire
0x0A	my-d proximity
0x0B	my-d move (use this if proritary my-d move command shall be used)

UID_LEN:

Is an optional parameter and depends on the setting of UID_LF (see MODE). UID_LEN defines the length of the following UID field.

NOTICE:
The maximum UID_LEN is limited depending on the reader type. If UID_LEN exceeds the internal buffer size the reader responses a error massage.

UID:

UID, Serial-Number or pseudo unique identifier of the Transponder.

Depending on the UID_LF and UID_LEN the UID field could have a fixed ore a variable length and a variable position in the protocol.

case UID_LF = 0:

If UID_LEN is not used, the following definitions are mandatory depending on the Transponder type.

ISO 14443A

1	1	1	7
[0x25]	b000 0001	0x00	UID

ISO 14443B

1	1	4	4
[0x25]	b000 0001	0x00	PUPI

case : UID_LF = 1

If UID_LEN is activated the specific UID length of the Transponder should be used in the protocol.

1	1	1	UID_LEN
[0x25]	MODE	UID_LEN	UID

ISO-ERROR:

Additional error code if STATUS = 0x95.

FORMAT:

Indicates the format of the CARD_INFO field:

0x00: No further CARD_INFO field available.

0x01: CARD_INFO of an ISO14443-4 Type-A Transponder.

0x02: CARD_INFO of an ISO14443-4 Type-B Transponder.

0x03: CARD_INFO of an ISO14443-3 Type-A Transponder.

0x04: CARD_INFO of an Innovatron radio protocol (ISO14443B') Transponder.

CARD_INFO:

Depending on the FORMAT parameter this data field contains different data's

case FORMAT = 0x01

CARD_INFO contains the ATQA and SAK and parts of the Answer to select (ATS) of the ISO14443 Type-A Transponder as defined in ISO14443-4. The length of CARD_INFO depends on the TL parameter. The response length depends on the TL parameter of the Transponder ATS.

1				1
ATQA				
RFU	Proprietary coding	UID size	RFU	Bit frame anticollision

1	1	TL
SAK	TL	T0, TA(1), TB(1), TC(1) T1, Tk

case FORMAT = 0x02

CARD_INFO contains parts of the answer ATQB response ATTRIB response of the ISO14443 Type-B Transponder as defined in ISO14443-3.

1	1	1			
ATQB Response Protocol Info					
Bit_Rate_capability	Max_Frame_ Size	Protocol_Type	FWI	ADC	F0

1	
1 th Byte of Answer to ATTRIB	
MBLI	CID

case FORMAT = 0x03

CARD_INFO contains the ATQA and SAK parameter after the anticollision loop has finished of ISO14443 Type-A Transponder as defined in ISO14443-3.

1				1
ATQA				
RFU	Proprietary coding	UID size	RFU	Bit frame anticollision

1
SAK

case FORMAT = 0x04

CARD_INFO contains the transponder address and the validator address of Innovatron radio protocol Transponder.

1	
TAG ADDRESS	VALIDATOR ADDRESS

6.1.3. [0x23] Read Multiple Blocks

This command reads one or more data blocks from a memory card.

The supported ISO Host commands depends on the different Transponder types, they are described in chapter 9. Supported ISO Host commands.

REQUEST-DATA

1	1	(8)	1	1
[0x23]	MODE	(UID)	DB_ADR	DB-N

RESPONSE-DATA if STATUS = 0x00

1	1	1	DB-SIZE
DB-N	DB-SIZE	SEC-STATUS	DB
		Repeated DB-N times	

RESPONSE-DATA if STATUS = 0x95

1
ISO-ERROR

RESPONSE-DATA in case of STATUS = any other

0
-

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	SEC	ADR		

ADR:

b000 non-addressed
b001 addressed
b010 selected

SEC:

Requests optional the security status of the followed data block
b0 security status not requested (SEC-STATUS always = 0x00)
b1 security status is requested

UID:

Read-only UID of the Transponder. The UID is required only in the addressed mode.

DB_ADR:

First block number to be read. First block can be any value between 0 and 255.

DB-N:

Number of data blocks to be read from the Transponder, starting at DB_ADR. The maximum number of DB-N, depends on DB-Size. The maximum number of bytes is 256 byte.

DB-Size	Max. DB-N
1	256
4	64
8	32
x	= 256 / x

ISO-ERROR:

Additional error code if STATUS = 0x95.

DB-SIZE:

Number of bytes of one data block. This value depends on the specification of the Transponder manufacturer, see chapter 9. Supported ISO Host commands.

SEC-STATUS:

Block security status of followed data block.
If SEC-STATUS is not requested or not supported, this value will return 0x00.

DB:

Requested data block. The block size is defined by DB-SIZE.

Notes:

- ***If a Transponder does not support Read Multiple Blocks commands several Read Single Block commands are used for this Transponder.***
- ***Jewel Transponders are only supported in Addressed Mode***
- ***A read of 1 byte from a JEWEL Transponder uses the JEWEL READ Instruction***
  A read of more than 1 byte from a JEWEL Transponder uses the JEWEL READ-ALL instruction

6.1.4. [0x24] Write Multiple Blocks

This command writes one or more data blocks.

The supported ISO Host commands depends on the different Transponder types, which are described in chapter 9. Supported ISO Host commands.

REQUEST-DATA

1	1	(8)	1	1	1	DB-SIZE
[0x24]	MODE	(UID)	DB_ADR	DB-N	DB-SIZE	DB
						Repeated DB-N times

RESPONSE-DATA if STATUS = 0x00

0
-

RESPONSE-DATA if STATUS = 0x03

1
DB_ADR

RESPONSE-DATA if STATUS = 0x95

1	1
ISO-ERROR	DB_ADR

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	WR-NE	ADR		

ADR:

b000 non-addressed
b001 addressed
b010 selected

WR-NE (Only JEWEL and ASK CTx):

b0 Write-Erase
b1 Write-No-Erase
This settling is necessary for write operations on OTP Bytes.

NOTICE:
To perform write operation on JEWEL Transponder it is recommended to set MIN_LVL (see 3.4. CFG3: RF-Interface) manual to a value > 8.

UID:

Read-only UID of the Transponder. The UID is required only in the addressed mode.

DB_ADR:

Address of the first data block to be written to the Transponder. First block can be any value between 0 and 255.

DB-N:

Number of data blocks to be written to the Transponder, starting at DB_ADR. The maximum number of DB-N depends on DB-Size. The maximum number of bytes is limited through the reader´s maximum receive buffer size.

DB-SIZE:

Number of bytes of one data block. This value depends on the specification of the Transponder manufacturer, see chapter 9. Supported ISO Host commands.

DB:

Data of the data block to be written to the Transponder. The required block size is defined by DB-SIZE. The number of the expected bytes are DB-N * DB-SIZE.

ISO-ERROR:

Additional error code if STATUS = 0x95.

DB_ADR-E:

Block number where the error occurred.

Notes:

- ***If a Transponder does not support Write Multiple Blocks commands several Write Single Block commands are used for this Transponder.***
- ***If an error occurred during a write command, the number of the block were the error occurred will be send to host***

6.2. [0xB0] ISO 14443 Standard Host Commands

6.2.1. [0xC0] Halt - ISO 14443-3 Transponder

This command sets one ISO14443-3 Transponder into Halt State.

The supported ISO Host commands depends on the different Transponder types, they are described in chapter 9. Supported ISO Host commands.

REQUEST-DATA

0	0
[0xC0]	MODE

RESPONSE-DATA

0
-

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	ADR		

ADR:

b010 selected

NOTICE:

- ***The reader response is always "OK" independent if the transponder is further one in the detection field of the reader or not.***

6.3. [0xB2] ISO14443 Special Host Commands

The [0xB2] commands are supposed to send special ISO14443 defined commands and proprietary ISO14443 RF commands to the Transponder.

REQUEST-DATA

1	(X)
SUB-COMMAND	PARAMETER

RESPONSE-DATA

(X)
RESPONSE-DATA

SUB-COMMAND, PARAMETER:

Command specific request with variable length. More details are described in the next chapters.

RESPONSE-DATA:

Command specific response with variable length. More details are described in the next chapters.

Notes:

- ***This command available only if the scan mode is switched off.***

6.3.1. [0x30] Mifare Value Commands

This command provides the Mifare value functions INCREMENT, DECREMENT, TRANSFER and RESTORE of an value formatted Mifare sector block. The command returns an error if the block is not in value block format (details about the Mifare value block format are described in Mifare standard data sheet provided by NXP). The command loads the value from a source address (DB_ADR), operates the value function and stores the result at the destination address (DEST_ADR).

NOTICE:

- ***A previous authentication (see:*** 6.3.2. [0xB0] Authent Mifare***) is needed to process the command.***
- ***The Mifare value block format could be written with the reader command*** 6.1.4. [0x24] Write Multiple Blocks

REQUEST-DATA

1	1	1	1	4	1
[0x30]	MODE	MF_CMD	DB_ADR	OP_VALUE *MSB … LSB*	DEST_ADR

RESPONSE-DATA

0
-

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	ADR		

ADR:

b010 selected

MF_CMD

This parameter defines the value operation which shall be operated on the Mifare Transponder.

0x01 INCREMENT
Adds the value OP_VALUE to the value specified by address DB_ADR.

0x02 DECREMENT
Subtracts the value OP_VALUE from the value specified by address DB_ADR.

0x03 COPY
Transfers the value structure from address DB_ADR to address DESTIN_ADR without changing the value.

DB_ADR:

Source Mifare block address of the value formatted data. A formula to calculate DB_ADR could be found in Chapter 9.2.3. NXP - Mifare classic: mini, 1k, 4k / mifare plus (Level 1)

NOTICE:
The specified Mifare block must have been formatted as a Mifare value block.

OP_VALUE:

This parameter contains the 32 Bit value which shall be calculated with the value at DB_ADR.

NOTICE:
In case of the COPY function the content of OP_VALUE has no effect.

DEST_ADR:

Destination address where the result of the value operation shall be stored.

NOTICE:
DEST_ADR and DB_ADR must be in the same Mifare sector.

Example:

- *Formatting of Mifare Sector 2, Block 1 in Mifare value block format with Value = 2 and Adr = 5 by using the command [0x24] Write Multiple Blocks.*

1	*1*	*1*	*16*													
		mifare Byte:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DB_ADR	*DB-N*	*DB-SIZE*	*DB*															
0x09	*0x01*	*0x10*	*0xFA Adr.*	*0x05 Adr.*	*0xFA Adr.*	*0x05 Adr*	*0x00000002 Value*				*0xFFFFFFFD Value*				*0x00000002 Value*			

NOTICE:
make sure that the access conditions in the Mifare Sector Trailer for this block are also configured as value block.

- *Formatting of Mifare Sector Trailer by using the command [0x24] Write Multiple Blocks*

1	*1*	*1*	*16*															
		mifare Byte:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DB_ADR	*DB-N*	*DB-SIZE*	*Key A*						*Access Bits*				*Key B*					
0x0B	*0x01*	*0x10*	*0xFFFFFFFFFFFF*						*0x69 8F 77 08*				*0xFFFFFFFFFFFF*					

- *Increment Value at Mifare Sector 2, Block 1 with OP_VALUE = 3*

1	*1*	*4*	*1*
MF_CMD	*DB_ADR*	*OP_VALUE*	*DEST_ADR*
0x01	*0x09*	*0x00000003*	*0x05*

6.3.2. [0xB0] Authent Mifare

Before access is given to the data stored in the memory of a mifare classic Transponder, the user have to prove his permission for the requested operation. Depending on the MODE.KL bit this command offers to possibilities for key handling. It is possible to use a key which is stored into the readers EERPOM (see: 4.5. [0xA2] Write Mifare Reader Keys) or a temporary key can transferred within the request data.

REQUEST-DATA

case MODE = bxxxx 0010

1	1	1	1	1
[0xB0]	MODE	DB_ADR	KEY-TYPE	KEY-ADR

case MODE = bxxxx 1010

1	1	1	1	6
[0xB0]	MODE	DB_ADR	KEY-TYPE	KEY

RESPONSE-DATA

0
-

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	KL	ADR		

ADR:

b010 selected

KL:

This bit indicates the key location

b0: EEPROM Key, defined by KEY-TYPE and KEY-ADR is used for authentication process .

b1: KEY-TYPE and temporary KEY which are transferred within the request data are used for authentication process.

DB_ADR:

Address of the first data block on which an access is requested .

NOTICE:
The Reader uses a linear addressing mode. For calculating the block address (DB_ADR) the expected mifare Sector and the mifare Block in this sector must be known. A formula to calculate DB_ADR could be found in Chapter 9.2.3. NXP - Mifare classic: mini, 1k, 4k / mifare plus (Level 1)

An authentication to one mifare Block inside a sector has effect to the whole sector.

KEY-TYPE:

Defines the key for the authentication.

0x00: KEY A

0x01 KEY B

KEY-ADR:

EEPROM Address (0x00 ... 0x0F) where the key is stored in the Reader (see: 4.5. [0xA2] Write Mifare Reader Keys).

KEY:

6 byte Mifare Key which shall used for the current authentication process.

6.3.3. [0xBB] ISO 14443-B' (Innovatron) Data Exchange

This command provides the data exchange between a host and the Transponder on ISO 14443-B' (Innovatron) layer. It is special designed for easy APDU data exchange.

NOTICE:

The maximum buffer size of the Reader for data exchange has to be considered and can be determined with command [0x66], Mode = 0x00.

REQUEST-DATA

1	1	x
[0xBB]	MODE	(DATA)

RESPONSE-DATA

4	x
STATUS	(DATA)

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	-	-	DISC

DISC:

b0 "DISCONNECT"
Sends the command "DISCONNECT" to the present Transponder.

b1 "APDU"
Instructs the Reader to send the APDU which is included in the DATA Block to the Transponder.

DATA:

ISO7816 format for commands: CLA/INS/P1/P2/P3/DataIn

STATUS

0x00 OK
This status shows that APDU exchange is completed.
DATA contains further information's

Other

DATA are not present.
This response is given by the Reader if the present command could not be finished, because of transmission errors.
see **ANNEX C: Index of Status Bytes**

6.3.4. [0xBE] ISO 14443-4 T=CL

This command provides the data exchange between a host and the Transponder on ISO 14443-4 layer. It is special designed for easy APDU data exchange.

NOTICE:

The maximum buffer size of the Reader for data exchange has to be considered and can be determined with command [0x66], Mode = 0x00.

REQUEST-DATA

1	1	(X)
[0xBE]	MODE	(PARAMETER)

RESPONSE-DATA

(1)	(2)	(X)
(PSTAT)	(BLK_CNT)	> depends on PSTAT <

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	FIRST	MORE	-	-	PING	NAD_E	CID_E	INF

MODE bit setting rules

MODE	DATA			
	(1)	(1)	(X)	
b1000 0001	(INF)			APDU without CID or NAD (single block)
b1100 0001	(INF)			APDU without CID or NAD (first chained block)
b0100 0001	(INF)			APDU without CID or NAD (further chained block)
b0000 0001	(INF)			APDU without CID or NAD (last chained block)
b1000 0011	CID	(INF)		APDU with CID (single block)
b1100 0011	CID	(INF)		APDU with CID (first chained block)
b1000 0101	NAD	(INF)		APDU with NAD (single block)
b1100 0101	NAD	(INF)		APDU with NAD (first chained block)
b1000 0111	CID	NAD	(INF)	APDU with CID and NAD (single block)
b1100 0111	CID	NAD	(INF)	APDU with CID and NAD (first chained block)
b1000 0000	-			DESLECT without CID or NAD
b1000 0010	CID	-		DESLECT with CID
b1000 100x	-			PING without CID or NAD
b1000 101x	CID	-		PING with CID

INF:

b0 "DESELECT"
Sends the S-block command "DESELECT" to the present Transponder.

b1 "APDU"
Instructs the Reader to send the INF Block (APDU) which is included in the DATA Block to the Transponder.

CID_E:

b0 The DATA Block includes no CID

b1: The DATA Block includes an optional 1 byte CID Parameter
The CID has to be placed in DATA directly behind the MODE Parameter

NOTICE:
In case of command chaining (see Bit "MORE") only the CID in the first command block is accepted by the Reader.

NAD_E:

b0 The DATA Block includes no NAD

b1: The DATA Block includes an optional 1 byte NAD Parameter
The NAD parameter is only supported in conjunction with INF = b1

NOTICE:
In case of command chaining (see Bit "MORE") only the NAD in the first command block is accepted by the Reader.

PING:

By means of this bit a presence check to the current Transponder can be operated by the host. The response includes only a status message.

b0: PING will not be operated

b1: PING will be operated by the Reader.

NOTICE:
PING is an exclusive function and can not combined with an APDU command. It can used with or without CID.

FIRST:

This bit indicates the first protocol of a new command. It is necessary for single commands and chained commands.

b0: The present protocol block is the second or further part of a chained command.

b1: The present protocol block is a single command or the first part of a chained command.

MORE:

By means of this bit a data chaining from the host to the Reader could realized if the number of data bytes which shall send beyond the receive buffer size of the Reader.

b0 **No downlink chaining (Host ⇒ Reader)**
The present protocol block includes the complete command.

b1 **downlink chaining (Host ⇒ Reader)**
The present protocol block includes not the complete command.
After the reader has acknowledged the protocol block the host can send further parts of the command.

NOTICE:

- ***If an error status is responded by the Reader the downlink chaining shall stopped by the host.***
- ***If a MORE status (0x94) is responded by the Reader the host have to handle this message.***

Protocol examples for Error-free operation with 3 blocks and 1 MORE response

	DATA	
MODE: b11xx 0xx1	(CID), (NAD), INF	Host ⇒ Reader (1. protocol block)
STATUS: 0x94 (MORE)		Host ⇐ Reader
STATUS: 0x00 (OK)		Host ⇐ Reader
b01xx 0001	INF	Host ⇒ Reader (2. protocol block)
STATUS: 0x00 (OK)		Host ⇐ Reader
b00xx 0001	INF	Host ⇒ Reader (last protocol block)
STATUS: 0x00 (OK)		Host ⇐ Reader

DATA:

The DATA Field could be used to transfer the optional CID, NAD and INF Field of the ISO14443-4 communication protocol.

In most cases the INF Field carries an APDU to the Transponder.

STATUS

0x00 OK

This status shows that APDU exchange is completed.
PSTAT and BLK_CNT has to be evaluated

0x94 MORE

This status shows that further exchange cycles has to be executed.
PSTAT and BLK_CNT has to be evaluated

NOTICE:
In case of STATUS = 0x94 the host has to continue its receiving procedure without sending any further request, because the reader transmits further response data later.

0x96 ISO14443-ERROR

This status shows that an additional ISO14443-ERROR has to be evaluated

RESPONSE-DATA

1
ISO14443-ERROR

ISO14443-ERROR

Additional error code if STATUS = 0x96 (see ANNEX C2: ISO14443-Error, Error-Codes

Other

PSTAT and BLK_CNT should not be evaluated.
This response indicates that the present command could not be finished, because of transmission errors.
see ANNEX C: Index of Status Bytes

PSTAT

This parameter represents the processing status of the present command. PSTAT must be evaluated in conjunction with the STATUS byte of the Reader response.

Depending on PSTAT and STATUS the response data of the Reader are different.

0x01 WTXM

This response is given by the Reader if the Transponder needs more time than defined in parameter TR-RESPONSE-TIME (see CFG1) to proceed the present command.

After receiving this response the host shall align his receive timeout to a value grater than indicated by WTXM.

RESPONSE-DATA

1	2	1	1
PSTAT 0x01	BLK_CNT	WTXM	FWI

WTXM and FWI:

refer to ISO 14443-4

The minimum receive timeout could calculated by the following formula:

TIMEOUT = 302µsec * 2^{FWI} * WTXM

WTXM: 1...59
FWI: 0...14

0x02 INF

This response is given by the Reader if the protocol includes data's from the Transponder.

RESPONSE-DATA

1	2	X
PSTAT 0x02	BLK_CNT	APDU-RESPONSE

APDU-RESPONSE:

Response to the APDU from the Transponder (if any).

0xFF BUSY

This response is given by the Reader to re-trigger the receive timeout of the host. This response could occur if an error in data exchange between Transponder and Reader had happened and the Reader retries the process by it self.

RESPONSE-DATA

1	2
PSTAT 0xFF	BLK_CNT

BLK_CNT

The BLK_CNT is a block counter which indexes each transmission from the Reader to the Host. On basis of the BLK_CNT the host could proof and sort the received protocols.

6.3.5. [0xBF] ISO 14443-4 Container Command

This command encapsulates and transports the ISO 14443-4 commands to the Transponder. The Command enables the transparent data exchange between host and Transponder as described in ISO 14443-4.

REQUEST-DATA

1	1	x
RSP	TIMEOUT (FWI)	REQUEST-BLOCK

RESPONSE-DATA

x
RESPONSE-BLOCK

NOTICE:
The maximum buffer for the RESPOSE-DATA is 256 byte (FSDI = 8).

RSP:

0	The Reader will send the command to the Transponder but do not wait for any response from the Transponder. This option should only used if the command doesn't have any response.
> 0	The Reader will send the command and is waiting for a response from the Transponder while the time period defined in TIMEOUT is running or the Transponder had send a response.

TIMEOUT (FWI):

With this parameter the Frame waiting time (FWT) according ISO14443-4 could be select by the user

TIMEOUT (FWI)	approx. Frame waiting time (FWT)
0	1 ms
1	1 ms
2	2 ms
3	3 ms
4	5 ms
5	10 ms
6	20 ms
7	39 ms
8	78 ms
9	155 ms
10	310 ms
11	619 ms
12	1237 ms
13	2474 ms
14	4948 ms
15..254	- not allowed -
255	automatically

NOTICE:

- ***If TIMEOUT = 255 is chosen the Reader used the FWI as transmitted from the Transponder.***
- ***The value of TIMEOUT must be considered for calculating the TR-RESPONSE-TIME (see CFG1)***

REQUEST-BLOCK

This Parameter with variable length is provided for the transparent data transfer to the Transponder. Refer to ISO 14443-4, chapter "Block Format"

Prologue filed			Information field	Epilogue field
PCB	[CID]	[NAD]	[INF]	EDC
1 Byte	1 Byte	1 Byte	x Byte	2 Byte

NOTICE:
PCB is mandatory for each command whereas the parameters in [ ] brackets are optional. For further Information please see ISO/IEC ISO14443-4

***The max. size of a* REQUEST-BLOCK is 256 byte.**

6.3.6. [0x2B] ISO14443-4 Transponder-Info

This command could be helpful to get further information's about the capabilities of the present ISO14443-4 Transponder. The included information are transferred from the Transponder. (For further Information please see ISO/IEC ISO14443-4)

NOTICE:

- ***This command could be used only after the Transponder was selected (see*** 6.1.2. [0x25] Select***).***

REQUEST-DATA

1
[0x2B]

RESPONSE-DATA

1	1	1	1	1	1
FSCI	FWI	DSI	DRI	NAD	CID

FSCI:

Transponder Frame-Size

FSCI	0	1	2	3	4	5	6	7	8	9..255
Bytes	16	24	32	40	48	64	96	128	256	RFU

FWI:

Frame Waiting Time Integer of the Transponder.

Frame Waiting Time (FWT) = 302µsec * 2^{FWI} (FWI_{max} = 14 ⇒ 4949 ms)

DSI (Divisor send Integer)**:**

Displays the present supported data transfer rate from Reader to Transponder.

DSI	**b00**	**b01**	**b10**	**b11**
kBit / s	106	212	424	847

DRI (Divisor receive Integer)**:**

Displays the present supported data transfer rate from Transponder to Reader.

DRI	**b00**	**b01**	**b10**	**b11**
kBit / s	106	212	424	847

NAD:

b1: NAD (Node Address) supported, if bit is set to 1.

CID:

b1: CID (Card Identifier) supported, if bit is set to 1.

6.4. [0xC1] / [0xC3] ISO Host Commands for mifare DESFire Communication

Mifare DESFire can be operated on APDU command level or with high level commands, implemented in firmware.

Chapter 9.1.1. mifare DESFire gives an overview about the available commands for mifare DESFire.

High level commands for mifare DESFire communication are described in separate manuals.

6.5. [0xC2] ISO Host Commands for mifare Plus Communication

Security Level 0, 2 and 3

Mifare Plus in Security Level 0 and 3 can be operated on APDU command level or with high level commands, implemented in firmware.

Chapter 9.1.2. NXP - mifare Plus (Security Level 0, 2, 3) gives an overview about the available commands for this security levels.

Security Level 1

Mifare Plus in Security Level 1 can be operated with commands for mifare classic.

Chapter 9.2.3. NXP - Mifare classic: mini, 1k, 4k / mifare plus (Level 1) gives an overview about the available commands for mifare Plus Security Level 1

7. Special Commands for Transponder Communication

7.1. [0xB9] Jewel Transparent Command

This command sends user transparent commands to Jewel transponder.

REQUEST-DATA

1	2	1	1	1	1
MODE	RSP-LENGHT	TIMEOUT	CRC-INFO	TX-SETTINGS	RX-SETTINGS

2	X
REQ-BITS	TR-REQUEST

RESPONSE-DATA

2	X
RSP-BITS	TR-RESPONSE

MODE:

Options for frame format request.

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	0

RSP-LENGTH

If RSP-LENGTH is set to “0” the Reader will send the command but not wait for any response.

If RSP-LENGTH is not equal to “0”, it must be set to the number of expected bits **including parity bits** ! The Reader will send the command and return the response data of the transponder without SOF and EOF.

TIMEOUT:

The TIMEOUT value defines the time for receiving the whole Transponder response. If the TIMEOUT it exceeded the command will be abort and the Status “NO TRANSPONDER” is returned.

Bit:	7	6	5	4	3	2	1	0
Function	Timeout-Value							

The Timeout-Value can be adjusted in 1ms steps. (0ms ...255ms)

NOTICE:

- ***The value of TIMEOUT must be considered for calculating the HOST-TIMEOUT.***
- ***The value of TIMEOUT must be considered for calculating the TR-RESPONSE-TIME (see CFG1).***

CRC-INFO:

Selects kind and mode of checking the data integrity of the RF-channel.

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	RxCRC En	TXCRC En	-	-

TxCRCEn

b0: No CRC is inserted/transmitted

b1: A CRC is calculated over the transmitted data and the CRC byte(s) are appended to the data stream

RxCRCEn

b0: No CRC is checked

b1: The last byte(s) of a received frame is/are interpreted as CRC byte/s

TX-SETTINGS:

Selects kind and mode of transmission settings.

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	0

RX-SETTINGS:

Selects kind and mode of reception settings.

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	0

REQ-BITS:

Number of valid Bits in TR-REQUEST

TR-REQUEST:

Complete transponder request without SOF and EOF. If "**TxCRCEn"** is "1" the reader appended a calculated CRC to the date stream. If "**TxCRCEn"** is "0" the application should send the CRC within the **TR-REQUEST**, if the CRC is needed.

RSP-BITS:

Number of valid Bits in TR-RESPONSE

TR-RESPONSE:

Complete transponder response without SOF and EOF. A CRC check is performed inside the reader if "**RxCRCEn"** is "1". However if "**RxCRCEn"** is "0"the transponder CRC is transferred with the response data.

7.2. [0xBD] ISO14443A Transparent Command

This command sends user transparent commands to ISO14443A transponder.

REQUEST-DATA

Mode0, Mode 1

1	2	1	1	x
MODE	RSP-LENGHT	TIMEOUT	CRC-INFO	TR-REQUEST

Mode 2

1	2	1	1	1	x
MODE	RSP-LENGHT	TIMEOUT	CRC-INFO	REQ-BITS	TR-REQUEST

Mode 3

1	2	1	1	1	1
MODE	RSP-LENGHT	TIMEOUT	CRC-INFO	TX-SETTINGS	RX-SETTINGS

2	X
REQ-BITS	TR-REQUEST

RESPONSE-DATA

Mode 0, Mode 1

X
TR-RESPONSE

Mode 2

1	X
RSP-BITS	TR-RESPONSE

Mode 3

2	X
RSP-BITS	TR-RESPONSE

MODE:

Options for frame format request.

The following frame types are defined:
- short frames for commands like REQA, WUPA, ...
- standard frames for regular commands;
- bit oriented anticollision frame for anticollision command

0 = short frame

A short frame is used to initiate communication and consists of, in the following order:

- start of communication;
- 7 data bits transmitted LSB first
- end of communication.
- No parity bit is added.

1 = standard frame

Standard frames are used for data exchange and consist of:

- start of communication;
  - n * (8 data bits + odd parity bit), with n _ 1. The LSB of each byte is transmitted first. Each byte is followed by an odd parity bit. The parity bit P is set such that the number of 1s is odd in (b1 to b8, P);
  - end of communication.

2 = bit oriented frame

Bit oriented Frames are used for anticollision.

RSP-LENGTH

If RSP-LENGTH is set to “0” the Reader will send the command but not wait for any response. If RSP-LENGTH is not equal to “0” the Reader will send the command and return the response data of the Transponder without SOF and EOF.

TIMEOUT:

The TIMEOUT value defines the time for receiving the whole Transponder response. If the TIMEOUT it exceeded the command will be abort and the Status “NO TRANSPONDER” is returned.

Bit:	7	6	5	4	3	2	1	0
Function	FWI-VALUE	Timeout-Value						

FWI-VALUE: b0: The Timeout-Value can be adjusted in 1ms steps. (0ms ...127ms)

b1: The Timeout Value is equivalent to the FWI value according to ISO14443-4. (0...14)

With this parameter the Frame waiting time (FWT) according ISO14443-4 could be select by the user

TIMEOUT (FWI)	**approx. Frame waiting time (FWT)**
0	1 ms
1	1 ms
2	2 ms
3	3 ms
4	5 ms
5	10 ms
6	20 ms
7	39 ms
8	78 ms
9	155 ms
10	310 ms
11	619 ms
12	1237 ms
13	2474 ms
14	4948 ms
15..254	- not allowed -

NOTICE:

- ***The value of TIMEOUT must be considered for calculating the HOST-TIMEOUT.***
- ***The value of TIMEOUT must be considered for calculating the TR-RESPONSE-TIME (see CFG1).***

CRC-INFO:

Selects kind and mode of checking the data integrity of the RF-channel.

Bit:	7	6	5	4	3	2	1	0
Function	-	CRC MSB First	-	-	RxCRC En	TXCRC En	-	Parity En

ParityEn

b0: No parity bit is inserted or expected

b1: An odd parity bit is inserted in the transmitted data stream after each byte and expected in the received data stream after each byte (standard ISO14443A)

TxCRCEn

b0: No CRC is inserted/transmitted

b1: A CRC is calculated over the transmitted data and the CRC byte(s) are appended to the data stream

RxCRCEn

b0: No CRC is checked

b1: The last byte(s) of a received frame is/are interpreted as CRC byte/s

CRCMSBFirst

b0: CRC-calculation starts with the LSB bit (standard ISO14443A)

b1: CRC-calculation starts with the MSB bit

TX-SETTINGS:

Selects kind and mode of transmission settings.

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	-	BAUDRATE	

BAUDRATE

b00: 106 kBaud

b01: 212 kBaud

b10: 424 kBaud

b11: 848 kBaud

RX-SETTINGS:

Selects kind and mode of reception settings.

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	-	BAUDRATE	

BAUDRATE

b00: 106 kBaud

b01: 212 kBaud

b10: 424 kBaud

b11: 848 kBaud

REQ-BITS:

Number of valid bits in REQUEST-DATA.

REQUEST-DATA:

Complete transponder request without SOF and EOF. If "**TxCRCEn"** is "1" the reader appended a calculated CRC to the date stream. If "**TxCRCEn"** is "0" the application should send the CRC within the **Request-Data**, if the CRC is needed.

NOTICE:

The max. size of REQUEST-DATA is 256 Byte

TR-RESPONSE:

Complete transponder response without SOF and EOF. A CRC check is performed inside the reader if "**RxCRCEn"** is "1". However if "**RxCRCEn"** is "0"the transponder CRC is transferred with the response data.

RSP-BITS:

Number of valid Bits in TR-RESPONSE

NOTICE:

- ***Data is only transferred if STATUS = 0x00, 0x02, 0x83, 0x84, 0x94.***
- ***The response data ever contains the in RSP-LENGTH defined number of data bytes.***

7.3. [0xBE] ISO14443B Transparent Command

This command sends user transparent commands to ISO14443B transponder.

REQUEST-DATA

Mode0

1	2	1	1	1	x
MODE	RSP-LENGHT	TIMEOUT	FRAME	CRC-INFO	TR-REQUEST

Mode1

1	2	1	1	1	1	1
MODE	RSP-LENGHT	TIMEOUT	FRAME	CRC-INFO	TX-SETTINGS	RX-SETTINGS

2	X
REQ-BITS	TR-REQUEST

RESPONSE-DATA

x
TR-RESPONSE

RSP-LENGTH:

Length of the transponder response in bit without SOF, CRC and EOF.

TIMEOUT:

The TIMEOUT value defines the time for receiving the whole Transponder response. If the TIMEOUT it exceeded the command will be abort and the Status “NO TRANSPONDER” is returned.

Bit:	7	6	5	4	3	2	1	0
Function	FWI-VALUE	Timeout-Value						

FWI-VALUE: b0: The Timeout-Value can be adjusted in 1ms steps. (0ms ...127ms)

b1: The Timeout Value is equivalent to the FWI value according to ISO14443-4. (0...14)

With this parameter the Frame waiting time (FWT) according ISO14443-4 could be select by the user

TIMEOUT (FWI)	**approx. Frame waiting time (FWT)**
0	1 ms
1	1 ms
2	2 ms
3	3 ms
4	5 ms
5	10 ms
6	20 ms
7	39 ms
8	78 ms
9	155 ms
10	310 ms
11	619 ms
12	1237 ms
13	2474 ms
14	4948 ms
15..254	- not allowed -

NOTICE:

- ***The value of TIMEOUT must be considered for calculating the HOST-TIMEOUT.***
- ***The value of TIMEOUT must be considered for calculating the TR-RESPONSE-TIME (see CFG1).***

FRAME:

Defines the framing for ISO 14443B transponders.

Bit:	7	6	5	4	3	2	1	0
Function	RxSOF Req	RxEOF Req	-	EOFSOF Width	No TxSOF	No TxEOF	TxEGT	

TxEGT:

These bits define the lenght of the EGT

b00: 0 Bit
b01: 1 Bit
b10: 2 Bit
b11: 3 Bit

NoTxEOF

b0: The frame includes EOF
b1: TxCoder suppresses the EOF

NoTxSOF

b0: The frame includes SOF
b1: TxCoder suppresses the SOF

EOFSOFWidth

b0: Set the SOF to a length of 10 ETU Low and 2 ETU High
Set the EOF to a length of 10 ETU

b1: Set the SOF to a length of 11 ETU Low and 3 ETU High
Set the EOF to a length of 11 ETU

RxEOF Req:

b0: A data stream with and without EOF is accepted
b1: A EOF is required in data stream

RxSOF Req:

b0: A data stream with and without SOF is accepted
b1: A SOF is required in data stream

CRC-INFO:

Selects kind and mode of checking the data integrity of the RF-channel.

Bit:	7	6	5	4	3	2	1	0
Function	-	CRC MSB First	-	-	RxCRC En	TXCRC En	-	Parity En

ParityEn

b0: No parity bit is inserted or expected (standard ISO14443B)

b1: A parity bit is inserted in the transmitted data stream after each byte and expected in the received data stream after each byte

TxCRCEn

b0: No CRC is inserted

b1: A CRC is calculated over the transmitted data and the CRC byte(s) are appended to the data stream

RxCRCEn

b0: No CRC is checked

b1: The last byte(s) of a received frame is/are interpreted as CRC byte/s

CRCMSBFirst

b0: CRC-calculation starts with the LSB bit (standard ISO14443B)

b1: CRC-calculation starts with the MSB bit

TX-SETTINGS:

Selects kind and mode of transmission settings.

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	-	BAUDRATE	

BAUDRATE

b00: 106 kBaud

b01: 212 kBaud

b10: 424 kBaud

b11: 848 kBaud

RX-SETTINGS:

Selects kind and mode of reception settings.

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	-	BAUDRATE	

BAUDRATE

b00: 106 kBaud

b01: 212 kBaud

b10: 424 kBaud

b11: 848 kBaud

REQ-BITS:

Number of valid Bits in TR-REQUEST.

TR-REQUEST:

Complete transponder request without SOF and EOF. If "**TxCRCEn"** is "1" the reader appended a calculated CRC to the date stream. If "**TxCRCEn"** is "0" the application should send the CRC within the **Request-Data**, if the CRC is needed.

NOTICE:

The max. size of REQUEST-DATA is 256 Byte

TR-RESPONSE:

Complete transponder response without SOF and EOF. A CRC check is performed inside the reader if "**RxCRCEn"** is "1". However if "**RxCRCEn"** is "0" the transponder CRC is transferred with the response data.

NOTICE:

- ***Data is only transferred if STATUS = 0x00, 0x02, 0x83, 0x84, 0x94.***
- ***The response data ever contains the in RSP-LENGTH defined number of data bytes.***

7.4. [0xBC] Command Queue

This command can be used for sending multiple commands within one protocol frame to the reader to speed up the total processing time. It is like a container for a queue of different commands which shall be processed by the reader sequentially.

REQUEST-DATA

1	2	X
MODE	CMD_NO	CMD_QUEUE

RESPONSE-DATA

1	1	X
0x00	CMD_CNT	QUEUE_RESPONSE

NOTICE:
This command can be used only with commands for Transponder communication as described in chapters 6.1. [0xB0] ISO Standard Host Commands, 6.2. [0xB0] ISO 14443 Standard Host Commands and 6.3. [0xB2] ISO14443 Special Host Commands.

MODE:

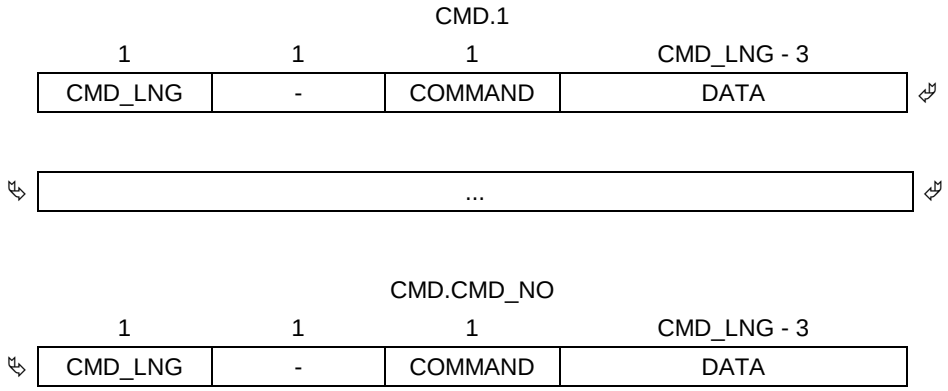
Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	0

CMD_NO:

Specifies the number of commands which are included in the queue.

CMD_QUEUE:

This field contains the command queue which shall be processed by the reader. The structure of a command in queue is identical with the structure of the not queued command, as described in this manual, but without the both CRC16 characters.



CMD_LNG:

Number of command bytes including CMD_LNG.

COMMAND:

Defines the command which shall operated by the reader. .

DATA:

Optional data field with variable length. The number of DATA byte depends on the command.

CMD_CNT:

Indicates the processing step where the reader had stopped the queue processing.

QUEUE_RESPONSE:

The QUEUE_RESPONSE field includes the response of the at last operated command of the command queue. This means that the QUEUE_RESPONSE includes the status and/or data of that command which could be operated at last. If an error occurs while operation of any queued command the queue proceeding will be interrupted and the error status of this last command is send back in the QUEUE_RESPONSE field.

The structure of the QUEUE_RESPONSE is identical with the structure of the not queued QUEUE_RESPONSE as documented in this manual, but without the both CRC16 characters.

1	1	1	1	RSP_LNG - 4
RSP_LNG	COM_ADR	COMMAND	Status[1]	DATA

RSP_LNG:

Number of response bytes including RSP_LNG.

COMMAND:

Command which was operated by the reader at least.

DATA:

Optional data field with variable length. The number of DATA byte depends on the command.

[1] see: ANNEX C: Index of Status Bytes

EXAMPLE:

The commands
1. [0xB0][0x25] select,
2. [0xB2] [0xB0] authent mifare and
3. [0xB0][0x23] read multiple blocks
shall be operated within one queue.

Host → Reader

1	2	1	3	1	5
STX	LENGTH	COM-ADR	COMMAND	MODE	CMD_NO
0x02	0x002B	0x00	[0xBC]	0x00	0x03

SELECT

1	1	1	10
CMD_LNG	-	COMMAND	DATA
0x0D	0x00	[0xB0]	[0x25] 0x01 0x00 0x00 0x00 0x00 0x6C 0x29 0xA7 0x62

Authent Mifare, DB_ADR: 4, KEY_TYP: A, KEY_ADR: 0

1	1	1	5
CMD_LNG	-	COMMAND	DATA
0x08	0x00	[0xB2]	[0xB0] 0x02 0x00 0x00 0x00

Read Multiple Blocks, DB_ADR: 4, DB_N: 1

1	1	1	4	2
CMD_LNG	-	COMMAND	DATA	CRC16
0x07	0x00	[0xB0]	[0x23] 0x02 0x04 0x01	

Host ← Reader

1	2	1	1	1	1
STX	LENGHT	COM_ADR	COMMAND	STATUS	CMD_CNT
0x02	0x0020	0x00	[0xBC]	0x00	0x03

23	2
RESPONSE-DATA	CRC16
0x17 0x00 0x[B0] 0x00 0x01 0x10 0x00 0x74 0x73 0x65 0x54 0x20 0x6E 0x69 0x65 0x20 0x74 0x73 0x69 0x20 0x73 0x61 0x44	

8. [0xC0] SAM Commands

The [0xC0] commands are supposed for communication with SAMs (security application modules) which could be installed inside the reader, if the reader type is equipped with a SAM socket. CPU-based SAM with T0= and/or T=1 protocol are supported as defined in ISO7816-4.

REQUEST-DATA

1	1	1	(X)
SLOT	SC_TIMEOUT	SUB-COMMAND	PARAMETER

RESPONSE-DATA

(X)
SAM-RESPONSE

SLOT:

Defines the physical address of the requested smartcard slot.
1: addresses the smartcard in Slot SD1
2: addresses the smartcard in Slot SD2

SC_TIMEOUT:

Defines the individual timeout for the current command. If the reader can not finish the current command within the defined SC_TIMEOUT it will respond an timeout error.

0: - do not use -
1..255: timeout in 100 ms increments.

The host application has to consider the SC_TIMOUT for setting the timeout on host side.

SUB-COMMAND, PARAMETER:

Command specific request with variable length

SAM-RESPONSE:

Command specific response with variable length.

8.1. [0x01] SAM Activate / Deactivate

With This command a smartcard can be activated and deactivated and also the protocol selection can be performed.

The activation and protocol selection is the first smartcard command which has to be proceeded in a communication cycle with a smartcard.

REQUEST-DATA

1	1	1	1	1	1	1
SLOT	SC_TIMEOUT	[0x01]	MODE	EXT_TA1	MODE2	EXT_CLK

1	1	1
EXT_TC1	EXT_TC2	EXT_TBx

RESPONSE-DATA

(X)
ATR

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	SET_ MODE2	0	CLASS		SET_TA1	ACTIVATE		

ACTIVATE

With this parameter the smartcard activation or deactivation and the protocol selection can be done.

In case of an activation command the reader response includes the ATR (Answer to Reset) of the card and/or an error status. The supported baudrates are shown in ANNEX F: Supported SAM Baud Rates.

b000 Deactivation
This setting powers off the addressed smartcard.

b001 T=0 protocol activation [Cold Reset]
This setting powers up the smartcard and performs a smartcard reset and attempts to activate the T=0 protocol of the card, if this protocol is supported by the card.

b011 T=1 protocol activation [Cold Reset]
This setting powers up the smartcard and performs a smartcard reset and attempts to activate the T=1 protocol of the card, if this protocol is supported by the card.

b010 GetATR
This setting activates the addressed smartcard temporary and can be used to evaluate ATR string of the inserted smartcard.

Notice:
This mode deactivates an activated smartcard.

b100 Activate first offered transmission protocol [Cold Reset]
This setting powers up the smartcard and performs a smartcard reset and attempts to activate the first offered transmission protocol of the card.
(T=0 or T=1 support)

b101 T=0 protocol activation [Warm Reset]
This setting performs a smartcard warm reset and attempts to activate the T=0 protocol of the card, if this protocol is supported by the card.

b110 Activate first offered transmission protocol [Warm Reset]
This setting performs a smartcard reset and attempts to activate the first offered transmission protocol of the card.
(T=0 or T=1 support)

b111 T=1 protocol activation [Warm Reset]
This setting performs a smartcard warm reset and attempts to activate the T=1 protocol of the card, if this protocol is supported by the card.

CLASS

With this parameter the smartcard activation of different voltage classes can be done.

b00 AutoMode

This setting tries to activate the smartcard using all voltage classes. The order of activation attempts is Class C, Class B and Class A.

b01 Class A (5V)

This setting tries to activate the smartcard using voltage Class A.

b10 Class B (3V)

This setting tries to activate the smartcard using voltage Class B.

b11 Class C (1.8V)

This setting tries to activate the smartcard using voltage Class C.

SET_TA1:

If this bit is set the optional parameter EXT_TA1 must inserted into the command

SET_MODE2:

If this bit is set the additional byte MODE2 must inserted into the command.

EXT_TA1:

By using this optional parameter it is possible for the application to select an explicit SAM Baud Rate. The structure of EXT_TA1 is equal to the TA(1) byte of ISO 7816-3.

The supported baudrates are shown in ANNEX F: Supported SAM Baud Rates.

Bit:	7	6	5	4	3	2	1	0
Function	Fi				Di			

Fi:

Indicator value of the clock rate conversion factor according ISO 7816-3.

Di:

Indicator value of the baud rate adjustment factor according ISO 7816-3.

MODE2:

By Using this optional parameter additional settings can be enabled.

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	SET_TBx	SET_TC2	SET_TC1	SET_CLK

SET_CLK:

If this bit is set, the optional parameter EXT_CLK must be inserted into the command.

SET_TC1: Global Interface Character TC1: ExtraGuardTime

If this bit is set, the optional parameter EXT_TC1 must be inserted into the command.

SET_TC2: Specific Interface Character [for T=0] TC2: WI (Work Waiting Time Index)

If this bit is set, the optional parameter EXT_TC2 must be inserted into the command.

SET_TBx: Specific Interface Character [for T=1] TBx: BWI|CWI

(Block Waiting Time Index)| (Character Waiting Time Index)

If this bit is set, the optional parameter EXT_TC1 must be inserted into the command.

EXT_CLK:

By using this optional parameter it will be possible for the application to select an explicit SAM card clock. Please refer to the following table.

EXT_CLK	Card Clock
	SAM1, SAM2
0…1	12,0 MHz
2	6,0 MHz
3…4	3,0 MHz
5…48	1,5 MHz

EXT_TC1:

By using this optional parameter it will be possible for the application to select an explicit extra guard time. The structure of EXT_TC1 is equal to the TC(1) byte of ISO 7816-3.

Bit:	7	6	5	4	3	2	1	0
Function	N							

N:

Indicator value of the extra guard time factor according ISO 7816-3.

EXT_TC2:

By using this optional parameter it will be possible for the application to select an explicit work waiting time for T=0 protocol. The structure of EXT_TC2 is equal to the TC(2) byte of ISO 7816-3.

Bit:	7	6	5	4	3	2	1	0
Function	WI							

WI:

Indicator value of the work waiting time factor according ISO 7816-3.

EXT_TBx:

By using this optional parameter it will be possible for the application to select an explicit block waiting time and character waiting time for T=1 protocol. The structure of EXT_TBx is equal to the TB(x) (for x>2) byte of ISO 7816-3.

Bit:	7	6	5	4	3	2	1	0
Function	BWI				CWI			

BWI:

Indicator value of the block waiting time factor according ISO 7816-3.

CWI:

Indicator value of the character waiting time factor according ISO 7816-3.

8.2. [0x22] GetMoreData

This command has to be executed, if status = 0x94 and PSTAT = 0x01, 0x02, 0x04 or 0xFF is responded by the following commands:

- 8.6. [0xBF] ISO7816 APDU Exchange
- 8.2. [0x22] GetMoreData
- 8.3. [0x23] AckAbortRequest

REQUEST-DATA

1	1	1	1
SLOT	SC_TIMEOUT	[0x22]	MODE

RESPONSE-DATA

1	2	X
PSTAT	BLK_CNT	> depends on PSTAT <

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	-	-	-

STATUS

0x00 OK
This status shows that APDU exchange is completed.
PSTAT and BLK_CNT have to evaluated.

0x94 MORE
This status shows that further exchange cycles has to be executed.
PSTAT and BLK_CNT have to evaluated.

Other
PSTAT and BLK_CNT should not be evaluated.
This response indicates that the present command cannot be finished, because of transmission errors. see ANNEX C: Index of Status Bytes

PSTAT

This parameter represents the processing status of the present command. PSTAT must be evaluated in conjunction with the STATUS byte of the Reader response.

Depending on PSTAT and STATUS the response data of the Reader are different.

0x01 WTXM

This response is given by the Reader if the smartcard needs more time than defined in parameter TIMEOUT (see 8. [0xC0] SAM Commands) to proceed the present command.

After receiving this response the host shall align his host timeout to a value greater than indicated by WTXM.

RESPONSE-DATA

1	2	1
PSTAT 0x01	BLK_CNT	WTXM

Calculating the host timeout:

refer to ISO 7816-3:2006 (3rd edition)

The minimum host timeout could be calculated by the following formula:

TIMEOUT = BWT * WTXM

WTXM: 1...255
BWI: 0...9

BWT is calculated by the formula:

$BWT = 11\ etu + 2^{BWI} * 960 * Fd / f$

$= 11 + 2^{BWI} * 960 * 372 / (Fi/Di)\ etu$

$BWT = (2^{BWI} * 960 * 372 / f_{CardClock} + 11 * (1 / Di * Fi / f_{CardClock}))$

Example: BWI = 4; $f_{CardClock}$ = 3,5712 MHz; Di = 1; Fi = 372

$BWT = 1,6 + 0,001146$

$BWT = 1,601146\ s = 1601,146\ ms$

PSTAT

This parameter represents the processing status of the present command. PSTAT must be evaluated in conjunction with the STATUS byte of the Reader response.

Depending on PSTAT and STATUS the response data of the Reader are different.

0x01 WTXM

This response is given by the Reader if the smartcard needs more time than defined in parameter TIMEOUT (see 8. [0xC0] SAM Commands) to proceed the present command.

After receiving this response the host shall align his host timeout to a value greater than indicated by WTXM.

RESPONSE-DATA

1	2	1
PSTAT 0x01	BLK_CNT	WTXM

Calculating the host timeout:

refer to ISO 7816-3:2006 (3rd edition)

The minimum host timeout could be calculated by the following formula:

TIMEOUT = BWT * WTXM

WTXM: 1...255
BWI: 0...9

BWT is calculated by the formula:

$BWT = 11\ etu + 2^{BWI} * 960 * Fd / f$

$= 11 + 2^{BWI} * 960 * 372 / (Fi/Di)\ etu$

$BWT = (2^{BWI} * 960 * 372 / f_{CardClock} + 11 * (1 / Di * Fi / f_{CardClock}))$

Example: BWI = 4; $f_{CardClock}$ = 3,5712 MHz; Di = 1; Fi = 372

$BWT = 1,6 + 0,001146$

$BWT = 1,601146\ s = 1601,146\ ms$

0x02 INF

This response is given by the Reader if the protocol includes data's from the smartcard.

RESPONSE-DATA

1	2	X
PSTAT 0x02	BLK_CNT	APDU-RESPONSE

APDU-RESPONSE:

Response to the ISO7816-4 APDU from the card (if any).

Activated Protocol	Maximum APDU-RESPONSE size	Description
T=0	258 bytes	256 bytes (Data) + 2 Byte (SW1, SW2)
T=1	254 bytes	maximum INF Block Size

0x03 ABORT REQUEST

The response is given by the reader if the smartcard aborts a chaining process. To acknowledge the request the command 8.3. [0x23] AckAbortRequest should be sent.

RESPONSE-DATA

1	2
PSTAT 0x03	BLK_CNT

0x04 IFSC CHANGE REQUEST

This response is given by the reader if the smartcard changes its maximum blocksize (IFSC). The user must adjust the size of the following request APDUs. To acknowledge the request the command 8.2. [0x22] GetMoreData should be sent.

RESPONSE-DATA

1	2	1
PSTAT 0x04	BLK_CNT	IFSC

0xFF BUSY

This response is given by the reader to re-trigger the host timeout of the host. This response could occur if an error in data exchange between smartcard and Reader had happened and the Reader retries the process by itself.

RESPONSE-DATA

1	2
PSTAT 0xFF	BLK_CNT

BLK_CNT

The BLK_CNT is a block counter which indexes each transmission from the Reader to the Host. On basis of the BLK_CNT the host could proof and sort the received protocols.

8.3. [0x23] AckAbortRequest

This command has to be executed, if status = 0x94 and PSTAT = 0x03 is responded by the following commands:

- command 8.6. [0xBF] ISO7816 APDU Exchange
- command 8.2. [0x22] GetMoreData

This command is only needed if chaining is done.

Chaining process from device to smartcard:

After the response of this command, the host should restart the chaining using the first block by using the command 8.6. [0xBF] ISO7816 APDU Exchange.

Chaining process smartcard to device:

The host should discard all previously received data and analyze the response of this command.

REQUEST-DATA

1	1	1	1
SLOT	SC_TIMEOUT	[0x23]	MODE

RESPONSE-DATA

1	2	X
PSTAT	BLK_CNT	> depends on PSTAT <

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	-	-	-	-	-	-	-	-

RESPONSE-DATA

Please refer to 8.2. [0x22] GetMoreData

8.4. [0xBD] T=0 Data Exchange

This command is to exchange APDU (Application Protocol Data Unit) command and response pairs with the smartcard by using the T=0 protocol.

Before performing this command the T=0 protocol has to be selected and the smartcard has to activated by using the SAM Activate / Deactivate command (see **SLOT:**

Defines the physical address of the requested smartcard slot.

1: addresses the smartcard in Slot SD1

2: addresses the smartcard in Slot SD2

SC_TIMEOUT:

Defines the individual timeout for the current command. If the reader can not finish the current command within the defined SC_TIMEOUT it will respond an timeout error.

0: - do not use -

1..255: timeout in 100 ms increments.

The host application has to consider the SC_TIMOUT for setting the timeout on host side.

SUB-COMMAND, PARAMETER:

Command specific request with variable length

SAM-RESPONSE:

Command specific response with variable length.

8.1. [0x01] SAM Activate / Deactivate).

REQUEST-DATA

1	1	1	1	1	1	1
SLOT	SC_TIMEOUT	[0xBD]	MODE	CLA	INS	P1

1	1	P3
P2	P3	DATA

RESPONSE-DATA

P3	1	1
DATA	SW1	SW2

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	TR_REC

TR_REC:

This bit specifies the data transfer direction of the current command.

b0: Transmit data to SAM

b1: Receive data from SAM

CLA:

APDU instruction class byte.

INS:

APDU instruction

P1:

APDU parameter byte 1

P2:

APDU parameter byte 2

P3:

Definition according ISO7816 part 3

>0 Defines the number of bytes to be transferred during the command.

=0 depending on TR_REC:

TR_REC: = b0 (transmit data)
Introduces no data transfer.

TR_REC: = b1 (receive data)
Introduces a 256 byte data transfer from the smartcard.

DATA:

APDU command or response data.

SW1, SW2:

Status Code returned by the card.

8.5. [0xBE] T=1 Data Exchange

This command is to exchange APDU (Application Protocol Data Unit) command and response pairs with the smartcard by using the T=1 protocol.

Before performing this command the T=1 protocol has to be selected and the smartcard has to activated by using the SAM Activate / Deactivate command (see **SLOT:**

Defines the physical address of the requested smartcard slot.

1: addresses the smartcard in Slot SD1

2: addresses the smartcard in Slot SD2

SC_TIMEOUT:

Defines the individual timeout for the current command. If the reader can not finish the current command within the defined SC_TIMEOUT it will respond an timeout error.

0: - do not use -

1..255: timeout in 100 ms increments.

The host application has to consider the SC_TIMOUT for setting the timeout on host side.

SUB-COMMAND, PARAMETER:

Command specific request with variable length

SAM-RESPONSE:

Command specific response with variable length.

8.1. [0x01] SAM Activate / Deactivate).

The SAM interface has implemented only the common functionality of a smartcard reader. Smartcard functions as WTX handling, chaining and some error recovering proceedings are not implemented into the reader firmware and have do done by the host computer or device driver on host side.

REQUEST-DATA

1	1	1	1	X
SLOT	SC_TIMEOUT	[0xBE]	MODE	(REQ_DATA)

RESPONSE-DATA

(X)
(RSP_DATA)

MODE:

Bit:	7	6	5	4	3	2	1	0

Function	0	0	0	0	0	0	0	0

REQ_DATA:

The REQ_DATA Field could be used to transfer T=1 APDU and has to build by the optional NAD, PCB, LEN, INF Field of the communication protocol.

In most cases the INF Field carries an APDU to the smartcard.

The EDC field is build internally by the reader

RSP_DATA:

Response to the T1 block from the card (if any).

The EDC field is not included in RSP_DATA.

8.6. [0xBF] ISO7816 APDU Exchange

This command is to exchange APDU (Application Protocol Data Unit) command and response pairs with the smartcard.

Before performing this command the T=0 or the T=1 protocol has to be selected and the smartcard has to activated by using the SAM Activate / Deactivate command (see 8.1. [0x01] SAM Activate / Deactivate).

REQUEST-DATA

1	(X)
MODE	APDU-DATA

RESPONSE-DATA

Please refer to 8.2. [0x22] GetMoreData

MODE:

Bit:	7	6	5	4	3	2	1	0
Function	FOLLOWING	MORE (Downlink) (I-Block Chaining) (only T=1)	-	-	-	NAD_E (only T=1)	-	-

MODE bit setting rules

MODE	REQ-DATA		
	6	7...n-2	
b0000 0000	(INF)		APDU without NAD (single block)
b0100 0000	(INF)		APDU without NAD (first chained block)
b1100 0000	(INF)		APDU without NAD (further chained block)
b1000 0000	(INF)		APDU without NAD (last chained block)
b0000 0100	NAD	(INF)	APDU with NAD (single block)
b0100 0100	NAD	(INF)	APDU with NAD (first chained block)

NAD_E (only T=1):

b0 The REQ-DATA Block includes no NAD

b1: The REQ-DATA Block includes an optional 1 byte NAD Parameter

FOLLOWING:

This bit indicates the second or further protocol of a command. It is necessary for chained commands.

b0: The present protocol block is a single command or the first part of a chained command.

b1: The present protocol block is the second or further part of a chained command.

MORE (only T=1):

By means of this bit a I-Block data chaining from the host to the Reader could be realized..

b0 **No downlink chaining (Host ⇒ Reader)**
The present protocol block includes the complete command.

b1 **downlink chaining (Host ⇒ Reader)**
The present protocol block includes not the complete command.
After the reader has acknowledged the protocol block the host can send further I-Blocks of the command.

NOTICE:

- ***If an error status is responded by the Reader the downlink chaining should stopped by the host.***
- ***If a MORE status (0x94) is responded by the Reader the host have to handle this message.***

Protocol examples for Error-free operation with 3 blocks and 1 MORE response

	DATA	
MODE: b0100 0x00	(NAD), INF	Host ⇒ Reader (1. protocol block)
STATUS: 0x94 (MORE)		Host ⇐ Reader
[0xC0] [0x22]		Host ⇒ Reader (Get MORE Data)
STATUS: 0x00 (OK)		Host ⇐ Reader
B1100 0000	INF	Host ⇒ Reader (2. protocol block)
STATUS: 0x00 (OK)		Host ⇐ Reader
B1000 0000	INF	Host ⇒ Reader (last protocol block)
STATUS: 0x00 (OK)		Host ⇐ Reader

APDU-DATA:

The APDU-DATA Field could be used to transfer ISO7816-4 APDUs

Activated Protocol	Maximum APDU-DATA size	Description
T=0	260 bytes	CLA, INS, P1, P2, P3, 255 bytes (Data)
T=1	254 bytes / 255[1] bytes	maximum INF Block Size

In T=0 protocol only short APDUs could be sent to the smartcard.

Please refer to ISO7816-3:2006 3rd edition, how to map extended APDUs to short APDUs.

Another possibility is to use the fescr function library (which would handle extended APDUs).

If the T=1 protocol is activated, the APDU-DATA Field could be used to transfer the optional NAD and INF Field of the ISO7816-4 communication protocol.

In most cases the INF Field carries an APDU to the SAM.

Field	Description	Number of bytes
Command header	Class byte denoted CLA	1
	Instruction byte denoted INS	1
	Parameter denoted P1-P2	2
L_c field	Absent for encoding $N_c = 0$, present for encoding $N_c > 0$	0, 1
Command data field	Absent if $N_c = 0$, present as string of N_c bytes if $N_c > 0$	N_c
L_e field	Absent for encoding $N_e = 0$, present for encoding $N_e > 0$	0, 1

RESPONSE-DATA

Please refer to 8.2. [0x22] GetMoreData

[1] 255 bytes, if NAD is used

9. Supported ISO Host commands

The command codes listed in the following chapters gives an overview of the various Transponder commands and operations that are available for each Transponder type.

NOTICE:

Detailed data sheets and information's about the functions and capabilities of each Transponder type are not supplied by FEIG ELECTRONIC. For detailed information's we refer to the original data sheets of the chip manufacturer.

9.1. ISO14443A & B Part 4 compliant Transponder

Memory organization:
Depends on the type and implementation of the used Transponder.

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x25]	Select	-	√	-	
[0xB2] [0xBE]	ISO14443-4 T=CL	-	-	√	
[0xB2] [0xBF]	ISO14443-4 Container	-	-	√	
[0xB2] [0x2B]	ISO14443-4 Transponder-Info	-	-	√	

9.1.1. mifare DESFire

mifare DESFire (MF3 IC D40)

mifare DESFire EV1 (2k: MF3 IC D21, 4k: MF3 IC D41, 8k: MF3 IC D81)

Standard Commands

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x25]	Select	-	√	-	
[0xB2] [0xBE]	ISO14443-4 T=CL	-	-	√	
[0xB2] [0xBF]	ISO14443-4 Container	-	-	√	
[0xB2] [0x2B]	ISO14443-4 Transponder-Info	-	-	√	

[0xC1] High Level Commands

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xC1] [0xFA]	DESFire Authent	-	-	√	①
[0xC1] [0xBD]	DESFire Read Standard Data	-	-	√	①
[0xC1] [0x3B]	DESFire Write Standard Data	-	-	√	①
[0xC1] [0x6C]	DESFire Get Value	-	-	√	①
[0xC1] [0x0C]	DESFire Credit	-	-	√	①
[0xC1] [0xDC]	DESFire Debit	-	-	√	①
[0xC1] [0x1C]	DESFire Limited Credit	-	-	√	①
[0xC1] [0x3B]	DESFire Write Records	-	-	√	①
[0xC1] [0xBB]	DESFire Read Records	-	-	√	①
[0xC1] [0xEB]	DESFire Clear Record File	-	-	√	①
[0xC1] [0x5F]	DESFire Change File Settings	-	-	√	①
[0xC1] [0x54]	DESFire Change Key Settings	-	-	√	①
[0xC1] [0xC4]	DESFire Change Key	-	-	√	①

[0xC3] High Level Commands

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xC3] [0xFA]	DESFire Authent	-	-	√	① / ②
[0xC3] [0x54]	DESFire Change Key Settings	-	-	√	① / ②

Command Code	Function	Mode			Comment
[0xC3] [0x45]	DESFire Get Key Settings	-	-	√	① / ②
[0xC3] [0xC4]	DESFire Change Key	-	-	√	① / ②
[0xC3] [0x54]	DESFire Get Key Version	-	-	√	① / ②
[0xC3] [0xCA]	DESFire Create Application	-	-	√	① / ②
[0xC3] [0xDA]	DESFire Delete Application	-	-	√	① / ②
[0xC3] [0x6A]	DESFire Get Application IDs	-	-	√	① / ②
[0xC3] [0x6D]	DESFire Get DF Names	-	-	√	① / ②
[0xC3] [0x5A]	DESFire Select Application	-	-	√	① / ②
[0xC3] [0xFC]	DESFire Format PICC	-	-	√	① / ②
[0xC3] [0x60]	DESFire Get Version	-	-	√	① / ②
[0xC3] [0x6E]	DESFire Free Mem	-	-	√	① / ②
[0xC3] [0x5C]	DESFire Set Configuration	-	-	√	① / ②
[0xC3] [0x51]	DESFire Get Card UID	-	-	√	① / ②
[0xC3] [0x6F]	DESFire Get File IDs	-	-	√	① / ②
[0xC3] [0x61]	DESFire Get ISO File IDs	-	-	√	① / ②
[0xC3] [0xF5]	DESFire Get File Settings	-	-	√	① / ②
[0xC3] [0x5F]	DESFire Change File Settings	-	-	√	① / ②
[0xC3] [0xCD]	DESFire Create Standard Data File	-	-	√	① / ②
[0xC3] [0xCB]	DESFire Backup Data File	-	-	√	① / ②
[0xC3] [0xCC]	DESFire Create Value File	-	-	√	① / ②
[0xC3] [0xC1]	DESFire Create Linear Record File	-	-	√	① / ②
[0xC3] [0xC0]	DESFire Create Cyclic Record File	-	-	√	① / ②
[0xC3] [0xDF]	DESFire Delete File	-	-	√	① / ②
[0xC3] [0xBD]	DESFire Read Standard Data	-	-	√	① / ②
[0xC3] [0x3B]	DESFire Write Standard Data	-	-	√	① / ②
[0xC3] [0x6C]	DESFire Get Value	-	-	√	① / ②
[0xC3] [0x0C]	DESFire Credit	-	-	√	① / ②
[0xC3] [0xDC]	DESFire Debit	-	-	√	① / ②
[0xC3] [0x1C]	DESFire Limited Credit	-	-	√	① / ②
[0xC3] [0x3B]	DESFire Write Records	-	-	√	① / ②
[0xC3] [0xBB]	DESFire Read Records	-	-	√	① / ②
[0xC3] [0xEB]	DESFire Clear Record File	-	-	√	① / ②
[0xC3] [0xC7]	DESFire Commit Transaction	-	-	√	① / ②
[0xC3] [0xA7]	DESFire Abort Transaction	-	-	√	① / ②

① See manual H01110-0e-ID-B.doc SoftCrypto functions for reader types without SAM

② See manual H01111-0e-ID-B.doc SAMCrypto functions for reader types with SAM

9.1.2. NXP - mifare Plus (Security Level 0, 2, 3)

For mifare PLUS security Level 1 see: 9.2.3. NXP - Mifare classic: mini, 1k, 4k / mifare plus (Level 1)

Memory organization:

mifare Plus 2k (MF1SPLUS60, MF1PLUS60)

Number of blocks	64	user area: 47
Block size	16 byte	

mifare Plus 4k (MF1SPLUS80, MF1PLUS80)

Number of blocks	256	user area: 215
Block size	16 byte	

mifare Plus (MF1SPLUS60, MF1SPLUS80)

Command Code	Function	ISO14443 Level	Mode		Comment
			addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x25]	Select	-	√	-	
[0xB2] [0xBE]	ISO14443-4 T=CL	-4	-	√	
[0xB2] [0xBF]	ISO14443-4 Container	-4	-	√	
[0xB2] [0x2B]	ISO14443-4 Transponder-Info	-4	-	√	
[0xC2] [0x77]	MFP_SL1_Authent	-3	-	√	①
[0xC2] [0x70]	MFP_FirstAuthentication	-4	-	√	①
[0xC2] [0x76]	MFP_FollowingAuthentication	-4	-	√	①
[0xC2] [0x78]	MFP_SL3_ResetAuthnetication	-4	-	√	①
[0xC2] [0x33]	MFP_SL3_ReadPlainMaced	-4	-	√	①
[0xC2] [0xA1]	MFP_SL3_WriteEncryptedMaced	-4	-	√	① AES Keys only
[0xC2] [0xA3]	MFP_SL3_WritePlainMaced	-4	-	√	①

mifare Plus (MF1PLUS60, MF1PLUS80)

Command Code	Function	ISO14443 Level	Mode		Comment
			addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x25]	Select	-	√	-	
[0xB2] [0xBE]	ISO14443-4 T=CL	-4	-	√	
[0xB2] [0xBF]	ISO14443-4 Container	-4	-	√	
[0xB2] [0x2B]	ISO14443-4 Transponder-Info	-4	-	√	
[0xC2] [0x77]	MFP_SL1_Authent	-3	-	√	①
[0xC2] [0x70]	MFP_FirstAuthentication	-4	-	√	①

Command	Function	ISO14443	Mode		Comment
[0xC2] [0x76]	MFP_FollowingAuthentication	-4	-	√	①
[0xC2] [0x75]	MFP_SL2_AESandCRYPTO1Authent	-3	-	√	①
[0xC2] [0x38]	MFP_SL2_MultiBlockRead	-3	-	√	①
[0xC2] [0xA8]	MFP_SL2_MultiBlockWrite	-3	-	√	①
[0xC2] [0x78]	MFP_SL3_ResetAuthnetication	-4	-	√	①
[0xC2] [0x30]	MFP_SL3_ReadEncrypted	-4	-	√	①
[0xC2] [0x31]	MFP_SL3_ReadEncryptedMaced	-4	-	√	①
[0xC2] [0x32]	MFP_SL3_ReadPlain	-4	-	√	①
[0xC2] [0x33]	MFP_SL3_ReadPlainMaced	-4	-	√	①
[0xC2] [0x34]	MFP_SL3_ReadEncryptedUnmaced	-4	-	√	①
[0xC2] [0x35]	MFP_SL3_ReadEncryptedUnmacedRespMaced	-4	-	√	①
[0xC2] [0x36]	MFP_SL3_ReadPlainUnmaced	-4	-	√	①
[0xC2] [0x37]	MFP_SL3_ReadPlainUnmacedRespMaced	-4	-	√	①
[0xC2] [0xA0]	MFP_SL3_WriteEncrypted	-4	-	√	①
[0xC2] [0xA1]	MFP_SL3_WriteEncryptedMaced	-4	-	√	①
[0xC2] [0xA2]	MFP_SL3_WritePlain	-4	-	√	①
[0xC2] [0xA3]	MFP_SL3_WritePlainMaced	-4	-	√	①
[0xC2] [0xB0]	MFP_SL3_IncrementEnrypted	-4	-	√	①
[0xC2] [0xB1]	MFP_SL3_IncrementEnryptedMaced	-4	-	√	①
[0xC2] [0xB2]	MFP_SL3_DecrementEnrypted	-4	-	√	①
[0xC2] [0xB3]	MFP_SL3_DecrementEnryptedMaced	-4	-	√	①
[0xC2] [0xB4]	MFP_SL3_Transfer	-4	-	√	①
[0xC2] [0xB5]	MFP_SL3_TransferMaced	-4	-	√	①
[0xC2] [0xB6]	MFP_SL3_IncrementTransferEncrypted	-4	-	√	①
[0xC2] [0xB7]	MFP_SL3_IncrementTransferEncryptedMaced	-4	-	√	①
[0xC2] [0xB8]	MFP_SL3_DecrementTransferEncrypted	-4	-	√	①
[0xC2] [0xB9]	MFP_SL3_DecrementTransferEncryptedMaced	-4	-	√	①
[0xC2] [0xC1]	MFP_SL3_Restore	-4	-	√	①
[0xC2] [0xC2]	MFP_SL3_RestoreMaced	-4	-	√	①

① See manual H01110-0e-ID-B.doc SoftCrypto functions

To find the AES Key to the chosen Data-Block you have to use the following calculation:

AES Sector Keys for sector 0 to 39 (40 00h to 40 4Fh)

Key A = sector number multiplied by 2

Key B = sector number multiplied by 2 +1

E.g. Key A for sector 2 has number: 40 04

9.2. ISO14443A Part 3 compliant Transponder

9.2.1. Infineon - my-d move SLE66R01P

Memory organization (SLE66R01P): 38 x 4 byte = 152 byte

Number of blocks	38	user area: 12 ... 32
Block size	4 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	Security Status is always 0x00
[0xB0] [0x24]	Write Multiple Blocks	-	-	√	
[0xB0] [0x25]	Select	-	√	-	
[0xB0] [0xC0]	Halt	-	-	√	

Commands for password protected memory and multiple block read/write commands can be carried out by using [0xBD] ISO14443A Transparent Command or by with using the OBID® C++ Class Library ID FEDM up from version 3.01.00 (see document H10202-##-ID-B) of OBID® software development kit.

9.2.2. Infineon - my-d proximity SLE55Rxx

Memory organization:

SLE55R04: 616 bytes

Number of blocks	82	max. user area: 5...81 min. user area: 32...81
Block size	8 / (10) byte	

SLE55R08: 1024 bytes

Number of blocks	133	max. user area: 5...132 min. user area: 32...132
Block size	8 / (10) byte	

SLE55R16: 2048 bytes

Number of blocks	261	max. user area: 5...260 min. user area: 32...260
Block size	8 / (10) byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	DB-Size = 8
[0xB0] [0x24]	Write Multiple Blocks	-	-	√	DB-Size = 8
[0xB0] [0x25]	Select	-	√	-	
[0xB0] [0xC0]	Halt	-	-	√	

9.2.3. NXP - Mifare classic: mini, 1k, 4k / mifare plus (Level 1)

Memory organization:

mifare classic mini (MF1 S20)

Number of blocks	20	user area: 14
Block size	16 byte	

mifare classic 1k (MF1 IC S50)

Number of blocks	64	user area: 47
Block size	16 byte	

mifare classic 4k (MF1 IC S70)

Number of blocks	256	user area: 215
Block size	16 byte	

mifare plus 2k (MF1PLUS60)

Number of blocks	128	user area: 95
Block size	16 byte	

mifare plus 4k (MF1PLUS80)

Number of blocks	256	user area: 215
Block size	16 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks*	-	-	√	Security Status is always 0x00
[0xB0] [0x24]	Write Multiple Blocks*	-	-	√	
[0xB0] [0x25]	Select	-	√	-	
[0xB0] [0xC0]	Halt	-	-	√	
[0xB2] [0x30]	Mifare value Commands*			√	
[0xB2] [0xB0]	Authent Mifare*	-	-	√	

**** The Reader uses a linear addressing mode. To calculate the Data-Block-Address (DB_ADR) the expected mifare Sector and the mifare Block in the sector must be known.***

DB_ADR calulation methode:

MF1 IC S20	Sector 0 ... 4: DB_ADR = MIFARE_SECTOR * 4 + MIFARE_BLOCK
MF1 IC S50	Sector 0 ... 15: DB_ADR = MIFARE_SECTOR * 4 + MIFARE_BLOCK
MF1 IC S70 **MF1 PLUS 80**	Sector 0 ... 31: DB_ADR = MIFARE_SECTOR * 4 + MIFARE_BLOCK Sector 32 ... 39: DB_ADR = (MIFARE_SECTOR - 32) * 16 + MIFARE_BLOCK + 128
MF1 PLUS 60	Sector 0 ... 31: DB_ADR = MIFARE_SECTOR * 4 + MIFARE_BLOCK

9.2.4. NXP - Mifare Ultralight

Memory organization (MF0U10 / MF0U11): 16 x 4 byte = 64 byte

Number of blocks	16	user area: 12
Block size	4 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	Security Status is always 0x00
[0xB0] [0x24]	Write Multiple Blocks	-	-	√	
[0xB0] [0x25]	Select	-	√	-	
[0xB0] [0xC0]	Halt	-	-	√	

9.2.5. NXP (Mifare ultralight C)

Memory organization (MF0ICU2): 48 x 4 byte = 192 byte

Number of blocks	48	user area: 4...39, (2...3, 40...47)
Block size	4 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	Security Status is always 0x00
[0xB0] [0x24]	Write Multiple Blocks	-	-	√	
[0xB0] [0x25]	Select	-	√	-	
[0xB0] [0xC0]	Halt	-	-	√	
[0xB2][0xB2]	Authent Mifare Ultralight C	-	-	√	①

① See manual H01110-0e-ID-B.doc SoftCrypto functions

9.3. ISO14443A Part 2 compliant Transponder

9.3.1. Innovision - Jewel

Memory organization:

IRT5001W / IRT5001E

Number of blocks	120	user area: 8...104; (113...120)
Block size	1 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	√	-	
[0xB0] [0x24]	Write Multiple Blocks	-	√	-	WRITE-ERASE and WRITE-NO-ERASE

NOTICE:
In case of write operations closely to the reader antenna it could be helpful to increase the MIN_LVL Parameter (see: 3.4. CFG3: RF-Interface).

9.4. ISO14443-B Part 2/ -3 compliant Transponder

9.4.1. STMicroelectronics - SR176

Memory organization:

Number of blocks	16	user area: 4...14; (15)
Block size	2 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	
[0xB0] [0x24]	Write Multiple Blocks*	-	-	√	
[0xB0] [0x25]	Select	-	√	-	

* The reader internally performs a read after write before acknowledging the command.

9.4.2. STMicroelectronics - SRIxx

Memory organization:

SRI512 / SRIX512:

Number of blocks	17	user area: 0...15; (255)
Block size	4 byte	

SRI4K / SRIX4K:

Number of blocks	129	user area: 0...127; (255)
Block size	4 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	
[0xB0] [0x24]	Write Multiple Blocks*	-	-	√	
[0xB0] [0x25]	Select	-	√	-	

* The reader internally performs a read after write before acknowledging the command.

9.4.3. ASK – CTS256B

Memory organization:

Number of blocks	16	user area: 5...15
Block size	2 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	
[0xB0] [0x24]	Write Multiple Blocks*	-	-	√	
[0xB0] [0x25]	Select	-	√	-	
[0xB0] [0xC0]	Halt	-	-	√	

* The reader internally performs a read after write before acknowledging the command.

9.4.4. ASK – CTx512B

Memory organization:

Number of blocks	32	user area: 5...31
Block size	2 byte	

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x23]	Read Multiple Blocks	-	-	√	
[0xB0] [0x24]	Write Multiple Blocks	-	-	√	
[0xB0] [0x25]	Select	-	√	-	
[0xB0] [0xC0]	Halt	-	-	√	

9.5. Innovatron radio protocol (ISO14443-B') compliant Transponder

Memory organization:
Depends on the type and implementation of the used Transponder.

Command Code	Function	Mode			Comment
		non-addressed	addressed	selected	
[0xB0] [0x01]	Inventory	-	-	-	
[0xB0] [0x25]	Select	-	√	-	
[0xB2] [0xBB]	ISO14443B' (Innovatron) Data Exchange	-	-	√	

ANNEX

ANNEX A: Codes of Transponder Types

TYPE_NO	Transponder Type	Supported by ID CPR.40.xx
0x00	NXP I-Code1	-
0x03	Transponder according ISO15693	-
0x04	Transponder according ISO14443A	●
0x05	Transponder according ISO14443B	●
0x06	NXP I-Code EPC	-
0x08	Jewel	●
0x0A	STMicroelectronics SR176	●
0x0B	STMicroelectronics SRIxx (SRI512, SRIX512, SRI4K, SRIX4K)	●
0x10	Innovatron (14443-B')	●
0x11	ASK CTx	●
0xFF	"free"	●

ANNEX B: Codes of Reader Types

No.	Reader Type
80	ID CPR.M02
81	ID CPR.02
82*	ID CPR40.xx-U with USB interface
83*	ID CPR40.xx- with asynchronous (RS232) interface
84	ID CPR.50.xx
85	ID CPR44.xx
86	ID CPR30.xx
87	ID CPR52.xx
88	ID CPR.04-USB (USB-Version; 596/#)

* if a reader is equipped with both interfaces the reader type is switched dynamical depending on the currently used interface.

ANNEX C: Index of Status Bytes

Hex-value	General
0x00	**OK:** • Data / parameters have been read or stored without error • Control command has been executed

Hex-value	Transponder Status
0x01	**No Transponder:** • No Transponder is located within the detection range of the Reader. • The Transponder in the detection range has been switched to mute. • The communication between Reader and Transponder has been interfered and the Reader is not able to read the Transponder anymore.
0x02	**Data False:** • CRC, parity or framing error at received data.
0x03	**Write-Error:** Negative plausibility check of the written data: • Attempt to write on a read-only storing-area. • Too much distance between Transponder and Reader antenna. • Attempt to write in a noise area.
0x04	**Address-Error:** The required data are outside of the logical or physical Transponder-address area: * The address is beyond the max. address space of the Transponder. * The address is beyond the configured address space of the Transponder.
0x05	**Wrong Transponder-Type:** This command is not applicable at the Transponder: * Attempt to write on or read from a Transponder. * A special command is not applicable to the Transponder.
0x08	**Authent-Error** The reader could not identify itself to the transponder as authorized: * reader- and transponder Keys do not correspond
0x0E	**General-Error** • The Transponder answered with an undefined or general error code
0x83	**RF Communication Error:** • Anticollision could not be finished by the reader. • Corrupted or faulty data exchange between reader and Transponder
0x93	**Data Buffer Overflow:** • There are more Transponders in reader field than could be handled by the reader (refer ANNEX D: Compendium of Supported Commands and Functions).
0x94	**More Data:** • There are more Transponder data sets requested than the response protocol can transfer at once.
0x95	**ISO15693-Error:** • An additional error code for ISO15693 Transponders is sent with response data.
0x96	**ISO14443-Error:** • An additional error code for ISO14443 Transponders is sent with response data. (see: ANNEX C2: ISO14443-Error, Error-Codes)
0x97	**Crypto Processing Error** • An additional code for source and reason of the error is sent with response data (See: ANNEX C1: Crypto Processing Error - ERROR-CODE

Hex-value	Parameter Status
0x10	**EEPROM-failure:** • The EEPROM of the Reader is not able to be written on. • Before writing onto the EEPROM a faulty checksum of parameters has been detected.
0x11	**Parameter-Range-Error:** • The value range of the parameters was exceeded.

Hex-value	Interface Status
0x80	**Unknown Command:** • The Reader does not support the selected function.
0x81	**Length-Error:** • The received protocol contains not the expected content.
0x82	**Command (currently) not available:** • The reader is configured in scan-mode and had received an ISO Host-mode command.

Hex-value	Reader Status
0xF1	**Hardware Warning:** • The Firmware is incompatible with the hardware

Hex-value	SAM Status
0x31	**No SAM detected** • The reader get no response from the Smart Card
0x32	**Requested SAM is not activated** • The requested SAM is not activated by the SAM Activate command
0x33	**Requested SAM is already activated**
0x34	**Requested protocol is not supported by the SAM** • Check if T=0 or T=1 protocol is supported by the SAM
0x35	**SAM communication error** • A data transmission error occurred while communication with the SAM
0x36	**Timeout** • The Reader got no response from SAM within the defined timout
0x37	**Unsupported SAM Baudrate** • The used parameter of Fi and/or Di are not supported by the reader

ANNEX C1: Crypto Processing Error - ERROR-CODE

ERROR-SOURCE = 1:

ERROR-CODE	Status
0x96xx	• ISO14443-Error: An additional error code for ISO14443 Transponders is sent with response data. (see: H71100-#e-ID-B, ANNEX C2: ISO14433-Error, Error-Codes)
0x6581	• Buffer Overflow, because the received data volume exceeds the reader internal buffer size
0x00##	• DESFire Error The ERROR-CODE was received form the DESFire Card (see NXP mifare DESFire functional specification)

ERROR-SOURCE = 3:

ERROR-CODE	Status
0x901E	• an error occurs while authentication, MAC calculation or CRC calculation. The reason can be a not satisfied security status or any kind of transmission errors.

ANNEX C2: ISO14443-Error, Error-Codes

Hex-value	Response error code definition
0x01	Lowlevel Error: CRC, Framing or EGT error
0x02	Timeout
0x03	Protocol error
0x04	block-no error (Chaining)
0x05	**Insufficient power: The present Transponder indicates** insufficient power * Maybe is distance between reader antenna and Transponder is high. * To many Transponders in the detection range of the Reader. • The power consumption of the Transponder exceed the antenna power of the Reader.

ANNEX D: Compendium of Supported Commands and Functions

This annex is a short compendium of the supported commands and functions of each type of reader depending on reader hardware and firmware version

Description	ID CPR40.00-CD	ID CPR40.01-CD	ID CPR40.00-A	ID CPR40.01-A	ID CPR40.00-UCD	ID CPR40.01-CDUSB	ID CPR40.30-USB	ID CPR40.30-A	ID CPR40.30-SUSB
5.1. [0x52] Baud Rate Detection	●	●	●	●	●	●	●	●	●
5.2. [0x55] Start Flash Loader	●	●	●	●	●	●	●	●	●
5.3. [0x63] CPU Reset	●	●	●	●	●	●	●	●	●
5.4. [0x65] Get Software Version	●	●	●	●	●	●	●	●	●
5.5. [0x66] Get Reader Info	●	●	●	●	●	●	●	●	●
5.6. [0x69] RF Reset	●	●	●	●	●	●	●	●	●
5.7. [0x6A] RF Output ON/OFF	●	●	●	●	●	●	●	●	●
5.8. [0x72] Set Output	●	●	●	●	●	●	●	●	●
5.9. [0xA0] Reader-Login	●	●	●	●	●	●	●	●	●
4.1. [0x80] Read Configuration	●	●	●	●	●	●	●	●	●
4.2. [0x81] Write Configuration	●	●	●	●	●	●	●	●	●
4.3. [0x82] Save Configuration	●	●	●	●	●	●	●	●	●
4.4. [0x83] Set Default Configuration	●	●	●	●	●	●	●	●	●
4.5. [0xA2] Write Mifare Reader Keys	●	●	●	●	●	●	●	●	●
*) [0xA3] Write DES/AES Reader Keys	●	●	●	●	●	●	●	●	-
ISO Host-Mode (see 3.2. CFG1: Interface)	●	●	●	●	●	●	●	●	●
Scan-Mode (see 3.2. CFG1: Interface)	●	●	●	●	●	●	●	●	-
6.1. [0xB0] ISO Standard Host Commands									
6.1.1. [0x01] Inventory	●	●	●	●	●	●	●	●	●
6.1.2. [0x25] Select	●	●	●	●	●	●	●	●	●
6.1.3. [0x23] Read Multiple Blocks	●	●	●	●	●	●	●	●	●
6.1.4. [0x24] Write Multiple Blocks	●	●	●	●	●	●	●	●	●
6.2. [0xB0] ISO 14443 Standard Host Commands									
6.2.1. [0xC0] Halt	●	●	●	●	●	●	●	●	●
6.3. [0xB2] ISO14443 Special Host Commands									
6.3.1. [0x30] Mifare Value Commands	●	●	●	●	●	●	●	●	●
6.3.2. [0xB0] Authent Mifare	●	●	●	●	●	●	●	●	●
*) [0xB2] Authent Mifare Ultralight C	●	●	●	●	●	●	●	●	-
6.3.3. [0xBB] ISO 14443-B' (Innovatron) Data Exchange	●	●	●	●	●	●	●	●	●
6.3.4. [0xBE] ISO 14443-4 T=CL	●	●	●	●	●	●	●	●	●
6.3.5. [0xBF] ISO 14443-4 Container Command	●	●	●	●	●	●	●	●	●
6.3.6. [0x2B] ISO14443-4 Transponder-Info	●	●	●	●	●	●	●	●	●
7.2. [0xBD] ISO14443A Transparent Command	●	●	●	●	●	●	●	●	●
7.3. [0xBE] ISO14443B Transparent Command	●	●	●	●	●	●	●	●	●

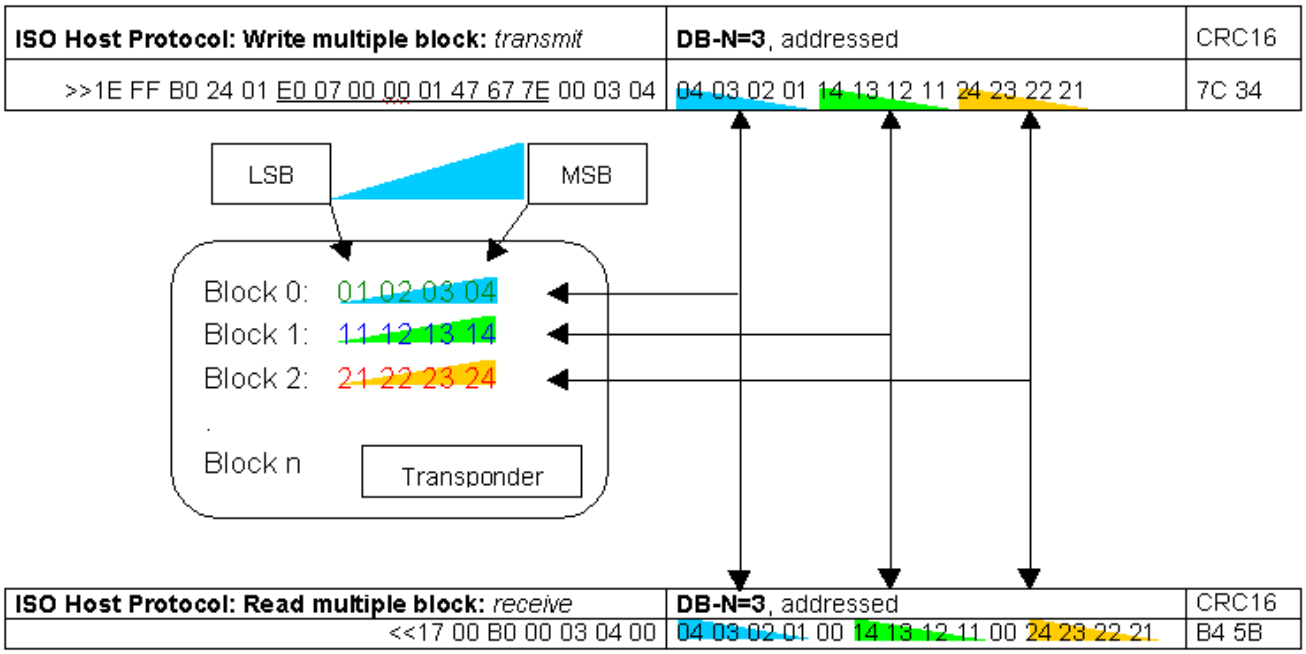
Description	ID CPR40.00-CD	ID CPR40.01-CD	ID CPR40.00-A	ID CPR40.01-A	ID CPR40.00-UCD	ID CPR40.01-CDUSB	ID CPR40.30-USB	ID CPR40.30-A	ID CPR40.30-SUSB
7.4. [0xBC] Command Queue	●	●	●	●	●	●	●	●	●
8. [0xC0] SAM Commands									
8.1. [0x01] SAM Activate / Deactivate	-	-	-	-	-	-	-	-	●
8.2. [0x22] GetMoreData	-	-	-	-	-	-	-	-	●
8.3. [0x23] AckAbortRequest	-	-	-	-	-	-	-	-	●
8.4. [0xBD] T=0 Data Exchange	-	-	-	-	-	-	-	-	●
8.5. [0xBE] T=1 Data Exchange	-	-	-	-	-	-	-	-	●
8.6. [0xBF] ISO7816 APDU Exchange	-	-	-	-	-	-	-	-	●
[0xC1] ISO Host Commands for DESFire Communication	①	①	①	①	①	①	①	①	-
[0xC3] ISO Host Commands for DESFire Communication	①	①	①	①	①	①	①	①	②
[0xC2] ISO Host Commands for mifare Plus Communication	①	①	①	①	①	①	①	①	-

● supported
– not available
① Supported with SoftCrypto functions (see manual H01110-0e-ID-B.doc SoftCrypto functions)
② Supported with SAMCrypto functions (see manual H01111-0e-ID-B.doc SAMCrypto functions)

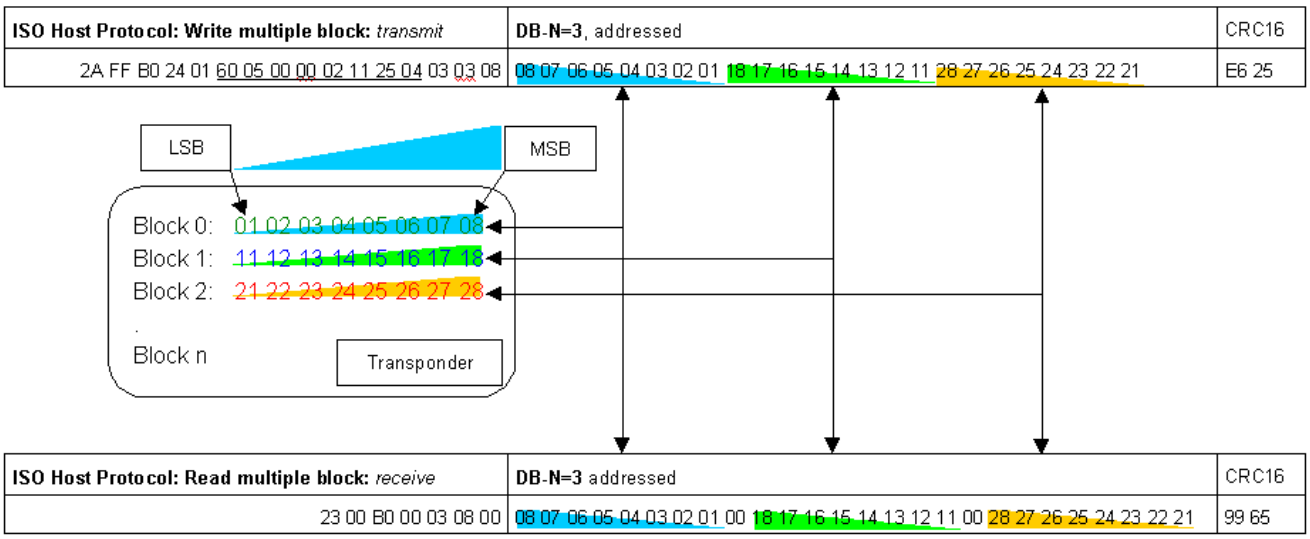
ANNEX E: Examples for Read Data

The setting "**LSB first**" and "**MSB first**" gives the direction of the received data bytes

ISO-Host Command (DB-Size of the Transponder = 4Byte)



ISO-Host Command (DB-Size of the Transponder = 8Byte)



ANNEX F: Supported SAM Baud Rates

The following baud rates are supported by the reader depending on transmission factors FI and DI of TA(1) parameter according ISO7816-3.

DI	FI											
	b0000 0x0	**b0001 0x1**	**b0010 0x2**	**b0011 0x3**	**b0100 0x4**	**b0101 0x5**	**b0110 0x6**	**b1001 0x9**	**b1010 0xA**	**b1011 0xB**	**b1100 0xC**	**b1101 0xD**
b0001 0x1	10753	10753	10753	8065	5376	4032	3226	7813	7813	5859	3906	2930
b0010 0x2	21505	21505	21505	16129	10753	8065	6452	15625	15625	11719	7813	5859
b0011 0x3	43011	43011	43011	32258	21505	16129	12903	31250	31250	23438	15625	11719
b0100 0x4	86022	86022	-	64516	43011	32258	25806	62500	62500	46875	31250	23438
b0101 0x5	-	-	-	129032	-	64516	-	125000	125000	93750	62500	46875
b0110 0x6	-	-	-	-	-	129032	-	250000	-	187500	125000	93750
b0111 0x7	-	-	-	-	-	-	-	*)	*)	375000	*)	187500
b1000 0x8	129032	129032	129032	96774	64516	48387	38710	-	93750	-	46875	-
b1001 0x9	-	-	-	-	-	-	64516	-	-	-	-	-
Card Clock	4 MHz	4 MHz	6 MHz	6 MHz	6 MHz	6 MHz	6 MHz	4 MHz	6 MHz	6 MHz	6 MHz	6 MHz

*) Are supported with baud rate 9600 (7813), if the card works in negotiated mode.

RFU parameters according ISO7816-3 are not shown in the table