I. Introduction

Id&DC's OEM Mifare reader terminals (Rdr11M, Rdr23M, Usb11M) are an ideal choice for your system applications with Mifare Classic and Ultra-light Cards, because of their affordable price, small size, convenient binary communication protocol and rich command set. Some major application areas are listed below:

- Access control systems
- Time and attendance systems
- Keyless door control
- Data collection/Storage/Processing Systems
- Cashless payment systems
- Smart ticketing for public transportation systems
- Mixed applications

II. Readers

1. Block diagrams

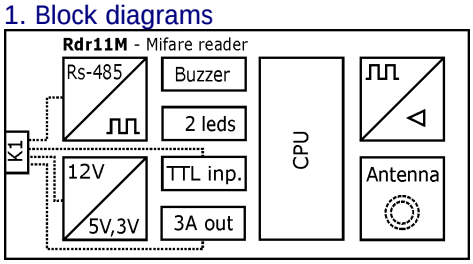

Fig.01: Rdr11M block diagram

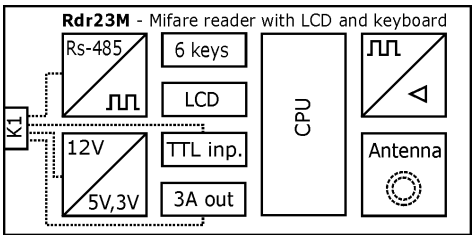

Fig.02: Rdr23M block diagram

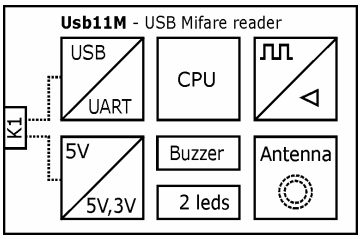

Fig.03: Usb11M block diagram

2. Features

Rdr11M	Terminal with Mifare reader
Communication	Rs-422 or Rs-485, 19200 bps, max. 300m cable length
Communication Protocol	CRC checked binary frames. 32 bits encryption. (On request)
Reader	Internal Mifare 13.56Mhz (ISO14443A) reader
Reading Distance	Up to 5 cm
Voltage & Consumption	DC 12÷20 V, 2 W (15V / 150ma)
Digital outputs	1 MOSFET over-current protected output, max. 3A
Digital inputs	1 over-voltage protected TTL input, function assignment possible
Indications	Piezoelectric buzzer, green and red LEDs
Dimensions & Weight	88 x 48 x 20 mm, 0.2 kg
Software	Access control or factory automation. On-line updates thru own communication port possible.

Table 01: Rdr11M - features

Rdr23M	Terminal with 2 lines x 16 characters LCD, keyboard and Mifare reader
Communication	Rs-422 or Rs-485, 19200 bps, max. 300m cable length
Communication Protocol	CRC checked binary frames. 32 bits encryption. (On request)
Reader	Internal Mifare 13.56Mhz (ISO14443A) reader
Reading Distance	Up to 7 cm
Voltage & Consumption	DC 12÷20 V, 2 W (15V / 150ma)
Keyboard	6 keys (<-, ->, +, -, Clr, OK)
Screen	2 lines x 16 characters (totally 32) LED illumination LCD screen
Digital outputs	1 MOSFET over-current protected output, max. 3A
Digital inputs	1 over-voltage protected TTL input, function assignment possible
Software	Access control or factory automation. On-line updates thru own communication port possible.
Indications	Piezoelectric buzzer
Dimensions & Weight	104 x 72 x 30 mm, 0.3 kg

Table 02: Rdr23M - features

Usb10M	Mifare reader with Usb interface (on-line)
Communication	USB 1.1 (virtual serial port driver available - free)
Communication Protocol	CRC checked binary frames. 32 bits encryption. (On request)
Reader	Internal Mifare 13.56Mhz (ISO14443A) reader
Reading Distance	Up to 5 cm
Voltage & Consumption	No need for external DC adaptor (powered from USB port)
Indications	Piezoelectric buzzer, green and red LEDs
Software	On-line updates thru USB possible
Dimensions & Weight	88 x 48 x 20 mm, 0.2 kg

Table 03: Usb11M – features

3. Typical access control system connection diagram

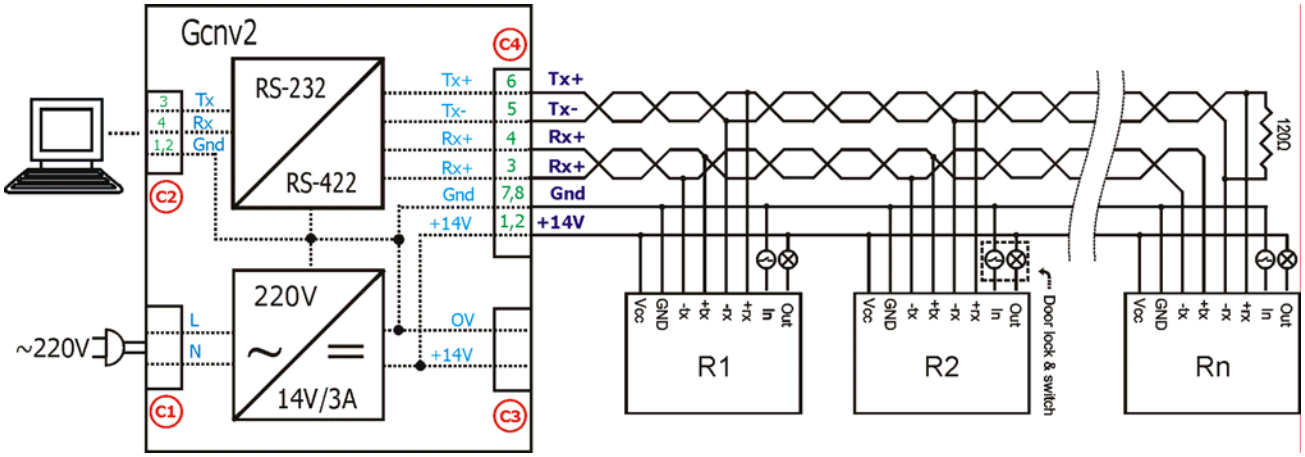

Fig.04: Typical access control system connection diagram (4 wire Rs-485)

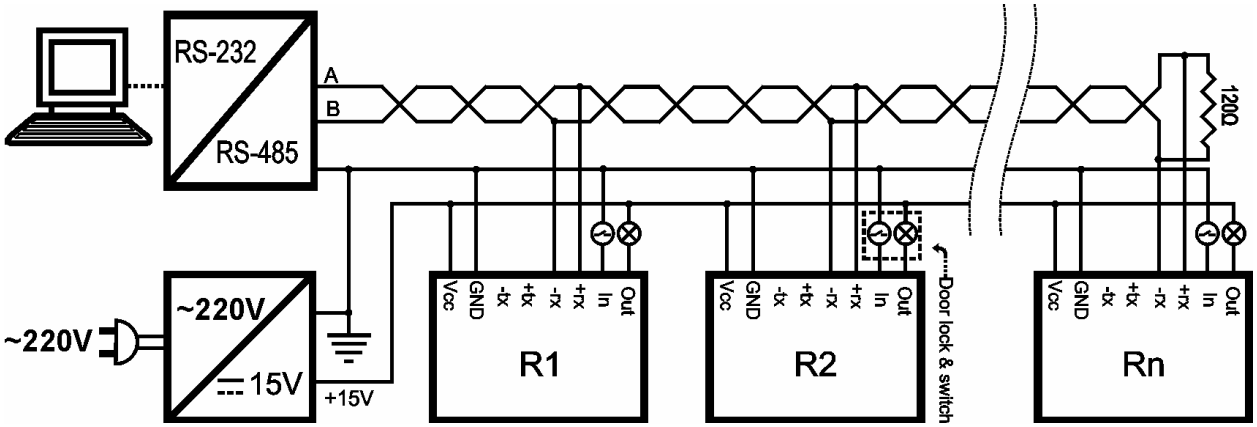

Fig.05: Typical access control system connection diagram (2 wire Rs-485)

III. Mifare cards - structures, principles, configurations

1. Mifare cards: introduction
2. Mifare cards: memory organization
3. Mifare cards: communication principle
4. Mifare cards: recommended configurations

1. Mifare cards: introduction

Mifare cards consist of Philips patented high security smart memory chip and coil. Philips has developed the mifare card systems to operate at 13.56Mhz, according to ISO/IEC 14443 A. The system offers optimal user convenience. These cards, when entered in the effective electro-magnetic field of the Mifare reader, along with absorbing induced voltage on the coil, are able to transmit and receive data with 106kbit/s. Such high communication speed allows high-speed transactions handling. Thus, Mifare card users are not forced to stop at reader and wait the transaction to finish. The card may also remain in wallet, even if there are coins in it.

Special emphasis has been placed on security against fraud. The communication layer complies with parts 2 and 3 of the ISO/IEC 14443A standard. The security layer supports the field-proven CRYPTO1 stream cipher for secure data exchange of the Mifare family. Mutual challenge and response authentication, data ciphering and message authentication checks protect the system from any kind of tampering and thus make it attractive for electronic purse applications. Serial numbers, which cannot be altered, guarantee the uniqueness of each card.

The independent memory sectors and two different keys for each sector allow **multiple hierarchical applications** with the same card.

Some widespread mifare applications are: cashless payment systems, ticketing in public transportation systems, Access Control Systems, Timing and Attendance Systems, data collection, process control systems, ….

2. Mifare cards: memory organization

Mifare card, depending on its type, consist of a number of sectors (Mifare Standard 1kB – 16 sectors, Mifare Classic 4kB – 40 sectors). Each of the first 32 sectors consist of 4 blocks (16 bytes per block), the remaining 8 sectors consist of 16 blocks each. The last block of the sector, named sector trailer, contains two 6 bytes access keys A and B (henceforth they will be referred as **"key A"** and **"key B"**) and 4 bytes **Access data**. Arrangement of sectors and blocks start from 0, namely, a 1kB Mifare Standard card consists of 16 sectors (0...15) and 64 blocks (0...63). Block 0 is a "Read Only" block, initial 4 bytes, being unique in each card, is the serial number of the card. Remaining 12 bytes represents the manufacturer data. As mentioned above, the last block of each sector is the **key A, key B** and **access data** block. Henceforth, it will be referred as **access block**. Therefore, Block 3, 7, 11, 15, ... are the **access blocks** of the related sectors (Access Block No = 3 + Sector No x 4). The 16 bytes memory block apart from the access block, as to access data, can be configured as 16 bytes **data domain** or 4 bytes **value domain**.

Sector No	Block No		Byte No															
			B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	B15
0	0	0	Card Serial No									Manufacturer Data						
	1	1	16 bytes data or 4 bytes value															
	2	2	16 bytes data or 4 bytes value															
	3	3	Key A					Access data							Key B			
1	0	4	16 bytes data or 4 bytes value															
	1	5	16 bytes data or 4 bytes value															
	2	6	16 bytes data or 4 bytes value															
	3	7	Key A					Access data							Key B			
.		.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.		.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.		.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.		.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
15	0	60	16 bytes data or 4 bytes value															
	1	61	16 bytes data or 4 bytes value															
	2	62	16 bytes data or 4 bytes value															
	3	63	Key A					Access data							Key B			

Table 04: Mifare card memory structure

When configured as **data domain**, the 16 bytes of the block can be freely read and altered with **block read** and **block write** commands.
When configured as **value domain**, the relevant value is stored in the whole block in a special manner. This configuration can be used in cashless payment systems. Following operations (commands) are applicable to the **value domain**: **block read, block write, write value** (initialization), **read value**, **increment value**, **decrement value**, **copy value** (copy a value to another block – to prevent loss of data). It is not recommended to use **block write** command due to the possibility of damaging special value format.

3. Mifare cards: communication principle

When a Mifare card enters the influence field of a reader, it starts to store the energy needed. After this process, the card is ready for communication with the reader and starts waiting for commands.

Select card command initiates packet of operations (transaction) with certain card. This command returns card's ID (serial number). After successful **select card** command, card reader must gain access to a certain sector by performing ***s*ector login** command with key A or key B. Only one sector can be accessed at a time. Depending on the access data, some commands to a block of a sector require sector to be logged in only with **key A**, some - only with **key B**, and some - with any key **(A or B)**. After a successful **sector login** with a key, relevant to the blocks of that sector, all the operations with the appropriate key (single operation or sequence of operations), unless a bad acknowledgement, can be processed. When the card enters the influence field of the reader at the first time or when a bad acknowledgement is returned after a command sequence to a block or blocks of a sector, to continue the operations with the blocks of that sector a ***s*elect card** operation should be repeated and if successful, depending on the operation, a **sector login** command with the right key should be send. Working with a block of an other sector, requires a **select card** and afterwards a **login sector** with the right key operations to be repeated.

If belonging sector of a block is not properly logged in, the reader replies with an "**operation failed**" message to data request to this block, or the information do not reflect real content of the block.
To prevent any sniffing attempts over the cable connecting the reader and the main computer, 32 (0...31) keys are stored in the mifare reader IC EEPROM. These keys, named **master key**s, along with the related keys on the cards, should be written (prepared) in secure environments. In possible sniffing cases, when **login sector** needed, **"sector login with master key"** command is sent to the reader. The command, along with the sector number, contains only the **master key** number (0...31), and the type of a key (A or B). The key itself is not included. Perceiving this command, the reader

IC reads the key from its internal memory, then automatically sends the **sector login** command to the card and thus has not been sent over the cable.

Configuring access data for each sector, if **sector login** with the right key is executed, is totally convenient. However, in some cases the related sector may be **partially or totally inaccessible**. It is a special case of the 4 bytes access data, thus, before altering it the user must learn very detailed information about Mifare cards. Such information, being confidential, can be obtained only after signing a privacy mutual understanding with PHILIPS. In this paper, besides factory default keys and Access data, high security level 8 (eight) different **access data** formats are explained, which are also supported and configurable by the explained readers.

4. Mifare cards: recommended configurations

Mifare cards, depending on the manufacturer, contain **factory access data** and **keys** configured at factory stage. Samples:
Philips: **key A** = FFH, FFH, FFH, FFH, FFH, FFH, **B** = FFH, FFH, FFH, FFH, FFH, FFH; Infineon: **key A** = A0H, A1H, A2H, A3H, A4H, A5H, **B** = B0H, B1H, B2H, B3H, B4H, B5H.

All blocks (except for the sector trailers) are arranged as **data blocks**, all blocks (except for the sector trailers), after login operation with proper **key A** or **key B** executed, ***block read and block write*** can be executed.

In sector trailers **key B** and **access data,** can be read and written with **key A** (after **sector login** to the block's sector), **key A** is not accessible in any manner, with **key A** (after **sector login** to the block's sector) write operation can be done. In high security needed applications, changing the **factory access data** is highly recommended!

As stated above, it is recommended that, the sectors supported and configurable with commands sent from readers, depending on the application, be configured with one of the 8 (eight) very secure different **access data**. In the following table, for each sector form the block types, key usage and allowed operations are explained in details.

Form	Block 0								Block 1								Block 2							
	Type	Operation							Type	Operation							Type	Operation						
		r	w	+	-	=	vw	vr		r	w	+	-	=	vw	vr		r	w	+	-	=	vw	vr
0	V	A|B	B	B	A|B	B	B	A|B	v	A|B	B	B	A|B	B	B	A|B	v	A|B	B	B	A|B	B	B	A|B
1	V	A|B	B	B	A|B	B	B	A|B	v	A|B	B	B	A|B	B	B	A|B	d	A|B	B	x	x	x	x	x
2	V	A|B	B	B	A|B	B	B	A|B	d	A|B	B	x	x	x	x	x	v	A|B	B	B	A|B	B	B	A|B
3	V	A|B	B	B	A|B	B	B	A|B	d	A|B	B	x	x	x	x	x	d	A|B	B	x	x	x	x	x
4	D	A|B	B	x	X	x	x	x	v	A|B	B	B	A|B	B	B	A|B	v	A|B	B	B	A|B	B	B	A|B
5	D	A|B	B	x	X	x	x	x	v	A|B	B	B	A|B	B	B	A|B	d	A|B	B	x	x	x	X	x
6	D	A|B	B	x	X	x	x	x	d	A|B	B	x	x	x	x	x	v	A|B	B	B	A|B	B	B	A|B
7	D	A|B	B	x	X	x	x	x	d	A|B	B	x	x	x	x	x	d	A|B	B	x	x	x	X	x
F	V	A	A	A	A	A	A	A	v	A	A	A	A	A	A	A	v	A	A	A	A	A	A	A

Table 05: Recommended sector forms 0...7 and 0F (factory form): types, keys and operations for Block 0, 1, 3.

Form No	Access Data Block: Block 3																				
	Key A: operations							Access Data: operations							Key B: operations						
	r	w	+	-	=	vf	vr	r	w	+	-	=	vf	vr	r	w	+	-	=	vf	vr
0..7	x	B	x	x	x	x	x	A|B	B	x	x	x	x	x	x	B	x	x	x	x	x
0F (factory)	x	A	x	x	x	x	x	A	A	x	x	x	x	x	A	A	x	x	x	x	x

Table 06: Recommended sector forms 0...7 and 0F (factory form): required keys for access block operations.

Abbreviations: **v** - value, **d** – data; **r** – read block, **w** – write block, **+** - increment value, **-** - decrement value, **=** – copy value, **vw** – write(format) value, **vr** – read value; **A** – key A, **B** – key B, **A|B** – key A or key B (either key can be used), **x** – invalid operation.

As seen in the tables, the recommended 8 sector forms (0...7) and all possible block 0, 1 and 2 value and data combinations are covered. For example, if certain application requires **Block 52 to be value domain; Block 53 - data domain; Block 54 - value domain,** then access data of sector 13 must be configured as **Form 2**. In the **access block**, the key cannot be accessed in any manner. The 4 bytes access data can be read with either key, namely, when executed a sector login with any key and block 55 is read, the 6 byte data each for key A and B is read as 0, the 4 byte access data will be read as last configured. Besides, in order to configure this sector **a sector login with key B** should have been executed.

In all **value blocks** **value increment**, **value format**, and **value copy** operations can be executed only with **key B**; **value decrement** and **value read** operations can be executed only with **key A or B**.

For all **data blocks**, **value operations** are **not allowed** (when tried, invalid operation response is retrieved), **block read** operation is executed with key A or key B; **block write** is executed only with **key B**.

IV. Communication Protocol

With some minor exceptions, communication is in query form. The host sends commands or requests data to the reader, using the appropriate address, and in a result, the addressed reader responds back to the computer. Request is done with two purposes:

Purpose 1: Main computer (from now on, will be referred as **host**) sends a **command** to the **reader** and the reader, after performing requested action, responds back the result of the executed command, or

Purpose 2: The host requests information (data, records, ...) from **the reader.** The addressed reader answers by sending the requested data.

The data sequence, sent by the host to the reader (command or data request), from now, will be referred as **RPkt** (request packet), and the acknowledgement, sent back by the reader to the host, will be referred as **APkt** (acknowledge packet). All the readers in the **RS485** network are in client mode and are waiting for **RPkt**'s from the host. Upon receipt, only the addressed one (target reader, the reader, which has physical address equal to the value in the **RPkt**'s address field) takes the **Rpkt** in account, and after performing the appropriate action, responds back with **APkt**. This rule is not valid for:

1. **RPktB -** broadcast request packets (address field = FF) type commands. Because such urgent commands are concerned with all the readers in the network, to prevent data collision, none of the readers respond back.
2. **RPktC** - command-only packets. Addressed reader does not respond to such requests. These commands will be explained later.

The readers, concerned in this document, can be configured to send automatically data packets (without being queried by the host) when suitably formatted card enters the active reader area. Such data packets will be referred as **APktA** – Automatic Answer Packets. These configurations and situations will be described in more details in the related topic.

RPkt, **RPktB**, **RPktC**, **APkt, APktA**, are binary packets and their structures are similar:

Field	STX	Address	Length = N	Data	CSum	ETX
Length (in bytes)	1	1	1	N	1	1

Table 07: Binary packet structure

In the following table the packet structure is explained in more details:

Field	Length	Value	Description
Stx	1	2	**RPkt:** packet start indicator (start of text).
			APkt: packet (start of text).
Address	1	1-254	**RPkt:** target reader address.
			APkt: Always = 0, because slave's responds always targets host
Length	1	1-255	**RPkt:** varies with the command sent.
			APkt: varies with the answer.
Data	N	0-255	**RPkt:** command or data request contents (varies with command).
			APkt: response content (varies with the answer).
Csum	1	0-255	**RPkt, APkt:** data verification sign. The client calculates the Csum of the received packet, and in case the calculated Csum's are different then the received packet is considered as invalid. CSum calculation: **CSum=Address** xor **Length** xor **Data1** xor **...** xor **DataN**
Etx	1	3	**RPkt:** packet end indicator (end of text).
			APkt: packet end indicator (end of text).

Table 08: Binary packet structure in details

All the readers in the network are in client mode and waiting for RPkt's from host. In some accidental situations, to prevent undesired behaviors, readers free their data buffers and turn to the initial mode (Stx wait). These situations are as follows:

1. Faulty Stx retrieval (packet start indicator);
2. The duration between two bytes exceeds 20ms;
3. Retrieving more data than expected data bytes;
4. Mismatch of calculated and retrieved checksums;
5. Retrieval of faulty Etx (packet ending indicator);
6. When faulty or senseless command or data exist in the data domain;

V. Reader Commands

1. Select Card (Read Card Serial Number)

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	01h	's' (73H)	...	03H

APkt:

1. Select Card successful: s1,...,s4 – serial number of selected card.

Stx	Address	Length	Data	CSum	Etx
02H	00H	04H	s1, s2, s3, s4	...	03H

2. No card:

Stx	Address	Length	Data	CSum	Etx
02H	00H	01H	'N' (4EH)	4FH	03H

2. Extended Select Card (Card Type + Card Serial Number)

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	02h	's', 'x' (73H,78H)	...	03H

APkt:

1. Select Card successful: t – type of selected card, s1,...,s4 – serial No.

Stx	Address	Length	Data	CSum	Etx
02H	00H	05H	t, s1, s2, s3, s4	...	03H

2. No card:

Stx	Address	Length	Data	CSum	Etx
02H	00H	01H	'N' (4EH)	4FH	03H

3. Sector Login

RPkt:

a. With Philips factory key A

Stx	Address	Length	Data	CSum	Etx
02H	...	04H	'l' (6CH), Sector N, FFH, 0DH	...	03H

b. With Infineon factory key A

Stx	Address	Length	Data	CSum	Etx
02H	...	04H	'l' (6CH), Sector N, AAH, 0DH	...	03H

c. With Infineon factory key B

Stx	Address	Length	Data	CSum	Etx
02H	...	04H	'l' (6CH),Sector N, BBH, 0DH	...	03H

d. With factory key A (manufacturer detected automatically and required key used)

Stx	Address	Length	Data	CSum	Etx
02H	...	03H	'l' (6CH),Sector N, 0DH	...	03H

e. Directly with key A

Stx	Address	Length	Data	CSum	Etx
02H	...	09H	'l', S.No, AAH, a1, a2, a3, a4, a5, a6	...	03H

f. Directly with key B

Stx	Address	Length	Data	CSum	Etx
02H	...	09H	'l', S.No, BBH, a1, a2, a3, a4, a5, a6	...	03H

g. With master key, key type A

Stx	Address	Length	Data	CSum	Etx
02H	...	03H	'l', S.No, 10H+Key No (0..31)	...	03H

h. With master key, key type B

Stx	Address	Length	Data	CSum	Etx
02H	...	03H	'l', S.No, 30H+Key No (0..31)	...	03H

APkt:

Sector Login Successful:

Stx	Address	Length	Data	Csum	Etx
02H	00H	01H	'L'	4DH	03H

Sector Login Fail: No Card

Stx	Address	Length	Data	Csum	Etx
02H	00H	01H	'N'	4FH	03H

Sector Login Fail: Faulty Key

Stx	Address	Length	Data	Csum	Etx
02H	00H	01H	'F'	47H	03H

Sector Login Fail: Faulty command or key type

Stx	Address	Length	Data	Csum	Etx
02H	00H	01H	'E'	44H	03H

Sector Login RPkt	Login type
02, 05, 04, 6C ,0A, FF, 0D, 95, 03	With Philips factory key A
02, 05, 04, 6C, 0A, AA, 0D, C0, 03	With Infineon factory key A
02, 05, 04, 6C, 0A, BB, 0D, D1, 03	With Infineon factory key B
02, 05, 03, 6C, 0A, 0D, 6D, 03	Factory key A
02, 05, 09, 6C, 0A, AA, 53, 62, B2, 4D, 8E, 9C, 1C, 03	Direct key A=53, 62, B2, 4D, 8E, 9C
02, 05, 09, 6C, 0A, BB, 53, 62, B2, 4D, 8E, 9C, 0D, 03	Direct key B=53, 62, B2, 4D, 8E, 9C
02, 05, 03, 6C, 0A, 2E, 4E, 03	Master key No: 1E(30), key A
02, 05, 03, 6C, 0A, 3B, 5B, 03	Master key No: 0B(11), key B

Table 09: Sector Login RPkt Samples: all commands address 5th reader.

4. Configure Sector

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	0FH	'f' (66H), Sector No, Form No, a1, .., a6, b1, .., b6	...	03H

APkt:

Stx	Address	Length	Data		CSum	Etx
02H	00H	01H	'B'	Sector Format Successful	...	03H
			'N'	No Card		
			'F'	Fail ()		
			'E'	Erroneous Command form		

5. Read Sector Form

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	03H	'rf' (72H, 66H), Sector No (00H..0FH)	...	03H

APkt:

Stx	Address	Length	Data		CSum	Etx
02H	00H	01H	00H..0FH	Form No	...	03H
			'U'	Unknown Form No		
			'N'	No Card		
			'F'	Fail ()		
			'E'	Erroneous command form		

6. Read Block

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	02H	'r' (72H), Block No	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	10H or 01H	B1, B2, ...,B16	...	03H

ByteNo	Data	Description
1	B1 or read failure sign	If Length 10H : Block No N Byte 1 If Length 01H : 'F' – read fail 'N' – no card
2	B2	Block No N Byte 2
3	B3	Block No N Byte 3
...	.....	.....
16	B16	Block No N Byte 16

Table 10:

Sample Block Read RPkt

Block Read RPkt	Description
02, 05, 02, 72, 2B, 5E, 03	Read 43(2B) th block of 5th reader command

Sample successful block read APkt data: C0,C1,C2,C3,C4,C5,C6,C7,C8,C9,CA,CB,CC,CD,CE,CF

Block Read Apkt
02, 00, 10, C0, C1, C2, C3, C4, C5, C6, C7, C8, C9, CA, CB, CC, CD, CE, CF, 10, 03

Bad Block Read APkt: 'F' response – read fail, related *"sector login"* is not completed

Block Read Apkt
02, 00, 01, 46, 47, 03

Bad Block Read APkt: 'N' response – read fail (or card not present)

Block Read Apkt
02, 00, 01, 4E, 4F, 03

7. Block Write

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	12H	'w' (77H), Block No, B1, B2, ...,B16	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	10H or 01H	B1, B2, ...,B16	...	03H

ByteNo	Data	Description
1	B1 or write failure sign	If Length 10H: Block No N Byte 1 If Length 01H: 'F' – write failure 'N' – no card present
2	B2	Block No N Byte 2
3	B3	Block No N Byte 3
...	.....	.....
16	B16	Block No N Byte 16

Table 11:

8. Write Value

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	07H	'wv' (77H,76H), Value Block No, B1, B2, B3, B4	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	04H or 01H	B1, B2, B3, B4	...	03H

ByteNo	Data	Description
1	B1 or write failure sign	If Length 04H: Value Block No N - Byte 1 If Length 01H: 'F' – write failure 'N' – no card
2	B2	Value Block No N - Byte 2
3	B3	Value Block No N - Byte 3
4	B4	Value Block No N - Byte 4

Table 12:

9. Read Value

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	03H	'rv' (72H,76H), Value Block No	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	04H or 01H	B1, B2, B3, B4	...	03H

ByteNo	Data	Description
1	B1 or read failure sign	If Length 04H : Value Block No N - Byte 1 If Length 01H : 'F' – read failure 'N' – no card
2	B2	Value Block No N - Byte 2
3	B3	Value Block No N - Byte 3
4	B4	Value Block No N - Byte 4

Table 13:

10. Copy Value – copy source value block to the target value block

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	03H	'=' (3DH), Source Value Block No, Target Value Block No	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	04H or 01H	B1, B2, B3, B4	...	03H

ByteNo	Data	Description
1	B1 or read failure sign	If Length 04H: Target Value Block No N new value- Byte 1 If Length 01H : 'F' – copy failure 'N' – no card
2	B2	Target Value Block No N new value - Byte 2
3	B3	Target Value Block No N new value - Byte 3
4	B4	Target Value Block No N new value - Byte 4

Table 14:

11. Increment Value – Increments the value of the Value Block by a number

RPkt:

Stx	Address	Length	Data	Csum	Etx
02H	...	06H	'+' (2BH), Block No, B1,B2,B3,B4 (B1..B4-inc. value)	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	04H or 01H	B1, B2, B3, B4	...	03H

ByteNo	Data	Description
1	B1 or read failure sign	If Length 04H: Value Block No N new value- Byte 1 If Length 01H: 'F' – increment failure 'N' – no card
2	B2	Value Block No N new value - Byte 2
3	B3	Value Block No N new value - Byte 3
4	B4	Value Block No N new value - Byte 4

Table 15:

12. Decrement Value – Decrements the value of the Value Block by a number

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	06H	'-' (2DH), Block No, B1,B2,B3,B4 (B1..B4-dec. Value)	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	04H or 01H	B1, B2, B3, B4	...	03H

ByteNo	Data	Description
1	B1 or read failure sign	If Length 04H: Value Block No N new value- Byte 1 If Length 01H: 'F' – decrement failure 'N' – no card
2	B2	Value Block No N new value - Byte 2
3	B3	Value Block No N new value - Byte 3
4	B4	Value Block No N new value - Byte 4

Table 16:

13. Write Master Key

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	09H	'wm' (77H, 6DH), Key No, B1, B2, ..., B6	...	03H

APkt:

Stx	Address	Length	Data		CSum	Etx
02H	00H	01H	'B'	Reader key write successful	...	03H
			'F'	Reader key write failure		
			'E'	Erroneous command form		

14. Set Digital IO's On/Off + Read Input/Output + Select Card

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	0Ah	'p'(70H), 'w'(77H), K1, K2, S1, S2, S3, S4, S5, S6	...	03H

ByteNo	Data	Description / Assigned values		
		Bit	Value	Definition
		7,6	00	IO4: Do not change
			01	IO4: On
			10	IO4: Blink (100 ms on, 100 ms off)
			11	IO4: Off
		5,4	00	IO3: Do not change
			01	IO3: On
			10	IO3: Blink (100 ms on, 100 ms off)
3	K1 (Commands)		11	IO3: Off
		3,2	00	IO2: Do not change
			01	IO2: On
			10	IO2: Blink (100 ms on, 100 ms off)
			11	IO2: Off
		1,0	00	IO1: Do not change
			01	IO1: On
			10	IO1: Blink (100 ms on, 100 ms off)
			11	IO1: Off
		Bit	Value	Definition
		7,6	00	Disable Select Card
			01	Enable Select Card
			10	Enable Extended Select Card
			11	Disable Select Card
		5,4		
4	K2 (Commands)			
		3,2	00	IO6: Do not change
			01	IO6: On
			10	IO6: Blink (100 ms on, 100 ms off)
			11	IO6: Off
		1,0	00	IO5: Do not change
			01	IO5: On
			10	IO5: Blink (100 ms on, 100 ms off)
			11	IO5: Off
5	S1	IO1 on (or blink) time x 100ms. =0 means permanently on		
6	S2	IO2 on (or blink) time x 100ms. =0 means permanently on		
7	S3	IO3 on (or blink) time x 100ms. =0 means permanently on		
8	S4	IO4 on (or blink) time x 100ms. =0 means permanently on		
9	S5	IO5 on (or blink) time x 100ms. =0 means permanently on		
A	S6	IO6 on (or blink) time x 100ms. =0 means permanently on		

Table 17:

APkt:

1. **Length=1.**

Stx	Address	Length	Data	CSum	Etx
02H	00H	01H	DIO	...	03H

2. **Length=2: No Card Present**

Stx	Address	Length	Data	CSum	Etx
02H	00H	02H	DIO, 'N' (4EH)	...	03H

3. **Length=5: Select Card Successful: s1,...,s4 – serial No**

Stx	Address	Length	Data	CSum	Etx
02H	00H	05H	DIO, s1, s2, s3, s4	...	03H

4. **Length=6: Select Card Successful: t – selected card type; s1,...,s4 – serial No.**

Stx	Address	Length	Data	CSum	Etx
02H	00H	06H	DIO, t, s1, s2, s3, s4	...	03H

15. Read EEProm

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	03h	're' (72H, 65H), EEP Addr(max.1FH)	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	01H	Value at specified address in RPkt	...	03H

16. Write EEProm

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	04h	'we' (77H, 65H), EEP Addr(max.1FH), Value	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	00H	01H	Value at specified address in RPkt after write op.	...	03H

17. Set Output

RPktC:

Stx	Address	Length	Data	CSum	Etx
02H	...	03h	'o' (6FH), IONo, OnT	...	03H

ByteNo	Data	Description / Assigned values		
		Bit	Value	Definition
2	IONo	7,6,5,4	0000	Permanently on
			0001	Blink (100 ms on, 100 ms off)
		3,2,1,0	0H..5H	Digital I/O #
3	OnT	On (or blink) time x 100ms. =0 means permanently on		

Table 18:

Using this command the host activates (switches on or blinks) desired digital output. The addressed reader ***does not respond*** to this command (**RPktC**).

Sample Set Output RPktC

Set Digital IO RPktC	Description
02H, 05H, 03H, 6FH, 12H, 14H, 6FH, 03H	Set Output 2 of 5th reader to blink for 2.0 sec

Sample Set Output RPktC

Set Digital IO RPktC	Description
02H, 05H, 03H, 6FH, 05H, 0H, 6CH, 03H	Set permanently Output 5 of 5th reader

Sample Set Output RPktC

Set Digital IO RPktC	Description
02H, 05H, 03H, 6FH, 04H, 0FH, 62H, 03H	Set Output 4 of 5th reader on for 1.5 sec

18. Fill LCD

RPktC:

Stx	Add.	Length	Data	CSum	Etx
02H	...	22h	'>' , L1_01,L1_02,...L1_16, L2_01,L2_02,...L2_16, ScrT	...	03H

ByteNo	Name	Description
2..17	L1_01 ... L1_16	Text to be shown on LCD Line 1
18..33	L2_01 ... L2_16	Text to be shown on LCD Line 2
34	ScrT	The time (in 100 msec) during which the text will be shown

Table 19:

19. Read Terminal Setup

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	01h	'd' (64H)	...	03H

APkt:

Stx	Address	Length	Data	CSum	Etx
02H	40H	01H	B00, B01, ... , B63	...	03H

#	Pos	Parameter Code	Parameter Description	Values	Dfl
01	0	TermStpIntl	Setup table initialized. If not set to 85, defaults will be assumed.	85 -initialized	85
02	1	TermAddr	Terminal address.	1..254	1
03	2	CommSpeed	99H - 4800bps, 88H - 9600bps, 77H - 19200bps, 66H -38400bps, 55H - 57600bps	99H, 88H, 77H, 66H, 55H	77H
04	3	CommParity	99H - No, 88H - Odd, 77H - Even, 66H - Mark, 55H – Space	99H, 88H, 77H, 66H, 55H	99H
05	4	CommType	99H – Rs485 4 wires, 88H – Rs485 2 wires, 77H – Rs422	99H, 88H, 77H	99H
06	5	CardReadMode	99H – Standard mode; 88H – Send UID; 77H – Send block; 66H – Send sector; 55H – Send UID ASCII; 44H – Send block ASCII;	99H, 88H, 77H, 66H, 55H, 44H	99H
07	6	KeyNoAndType	Master key # and key type (A or B), used for sector login.	0..31, 128..159	0
08	7	BuzzerBeepTime	Buzzer beep time on successful card read	in 100 mili seconds	0
09	8	BlockNo	Block # to be read, if any of a block read modes is selected.	0..63	0
10	9	SectorNo	Sector # to be read, if any of a sector read modes is selected.	0..15	0
11	10	HWType	Terminal type.	0..99	0
12	11	HWModel	Terminal model.	0..99	0
13	12	HWIssue	Terminal production issue.	0..99	0
14	13	FWRevision	Terminal firmware revision.	0..99	0
15	14	FWIssue	Terminal firmware issue.	0..99	0
16	15	RFU1	Reserved for future use		0
17	30	StartBlinkPosLCD	Default LCD text – first blinking character position.	1..32, if disabled = 0	0
18	31	EndBlinkPosLCD	Default LCD text – last blinking character position.	1..32, if disabled = 0	0
19	32	LCDTextWhenIdle	Default text, shown on the LCD when terminal is idle.	32...255	32

Table 20:

Par # 02 ... Par # 05 - communication parameters. Altering any of these parameters affects the related behavior just after reader rebooting (either powering on or receiving "reset" command)

Par # 06: CardReadMode –

Standard mode — the reader only executes commands, received via the serial port

Send UID — along with executing serial commands, when the reader is in idle state (no any activity on the ser. channel for more than a second), continuously performs "select card" command, and on valid card read a packet of data is send to the host, just like the response to a "select card" command (ref. V.1)

Send block — in idle state the reader scans for a valid card. On detection of such, tries to login to the sector, to which block #, specified in parameter # 9 belongs. Login is done like in "Login Sector with Master Key" command (ref. V.3.g or V.3.h), using key #, specified in parameter # 7 (KeyNoAndType). If login succeeds, follows reading of the above mentioned block and in case of correct completion, the answer is constructed and put on the serial channel just like if a "Read Block" command is executed.

Send sector — the reader works the same manner like in "send block" mode, except that it reads the first 3 blocks of the sector, specified in parameter # 10. Data length of the newly constructed packet is 30H (3 blocks each 16 bytes).

Send UID ASCII — along with executing serial commands, when the reader is in idle state (no any activity on the ser. channel for more than a second), continuously performs "select card" command, and on valid card read a packet, consisting of 2 bytes long reader address + 8 byte long card unique identifier (UID) + carriage return character + line feed character. Example: if reader # 2 reads card with UID = 08 AB 19 6E, then the following packet will be sent: 30 32 30 38 41 42 31 39 36 45 0D 0A

Send block ASCII — internally the reader performs the same operations as when it is in "send block" mode. In this mode the response packet is similar to that of the "Send UID ASCII" packet, but instead o 8 bytes, relating to 4 bytes long UID, 32 bytes, relating to 16 bytes block data are sent: 2 bytes long reader address + 32 byte long block data + carriage return character + line feed character.

Par # 07: KeyNoAndType – least significant 7 bits (b0...b6) = master key #, used for login when reader is in any of "send block", "send sector" or "send block ASCII" modes. If the most significant bit (b7) is set, login will be done with key B, if cleared - with key B. Prior to use, specified master key must be written with the appropriate values (corresponding 6 bytes long sector key of the cards, subject of use with the system).

20. Write Terminal Setup

RPkt:

Stx	Address	Length	Data	CSum	Etx
02H	...	41h	'c' (64H), B00, B01, ... , B63	...	03H

APkt:

Stx	Address	Length	Data		CSum	Etx
02H	00H	01H	'B'	Successful write	...	03H
			'F'	Fail		
			'E'	Erroneous Command form		

21. Reset

RPktC:

Stx	Address	Length	Data	CSum	Etx
02H	...	01h	'x'	...	03H

This command reboots the reader.