advanced card systems ltd

ACR38

Reference Manual

ACR38 Smart Card Reader/Writer

rev 1.2, June 2003

Advanced Card Systems Ltd.
Room 302, 3/F., Shun Fat Industrial building,
17 Wang Hoi Road, Kowloon Bay, Kowloon, Hong Kong.

Tel: 852-2796 7873
Website: www.acs.com.hk

Fax: 852-2796 1286
E-mail : info@acs.com.hk

Table of contents

1. INTRODUCTION

The ACS Smart Card Reader/Writer ACR38 is an interface for the communication between a computer (for example, a PC) and a smart card. Different types of smart cards have different commands and different communication protocols. This prevents in most cases the direct communication between a smart card and a computer. The ACR38 Reader/Writer establishes a uniform interface from the computer to the smart card for a wide variety of cards. By taking care of the card specific particulars, it releases the computer software programmer of getting involved with the technical details of the smart card operation, which are in many cases not relevant for the implementation of a smart card system.

The ACR38 Smart Card Reader/Writer is connected to the computer through USB interface. The reader accepts commands from the computer, carries out the specified function at the smart card and returns the requested data or status information.

NOTE - Although the ACR38 is a true *card reader/writer* as it can read and write smart cards, the terms *card reader* or *reader* will be used indifferently to refer to the ACR38, for the sake of readability and because these designations are commonly in use for this kind of devices.

2. FEATURES

- ISO7816-1/2/3 compatible smart card interface
- Support CPU-based cards with T=0 and/or T=1 protocol
- Support smart card with 5V, 3V and 1.8V voltage
- Support PPS (Protocol and Parameters Selection) with 9600 – 307200 bps in reading and writing smart cards
- Full speed USB 2.0 interface (12 Mbps) to PC with simple command structure
- Support most common memory-based smart cards

3. SUPPORTED CARD TYPES

The ACR38 can operate MCU card with T=0 and T=1 protocol in its main card slot.

3.1 Microcontroller-based smart cards (asynchronous interface)

The ACR38 supports EEPROM microcontroller-based cards with internal programming voltage (VPP) generation and the following programming parameters transmitted in the ATR:

PI1= 0 or 5

I = 25 or 50

The ACR38 performs the Protocol and Parameters Selection (PPS) procedure as specified *in ISO7816-3: 1997*.

When the card ATR indicates the specific operation mode (TA_2 present; bit b5 of TA_2 must be 0) and that particular mode is not supported by the ACR38, the reader will reset the card to set it to negotiable mode. If the card cannot be set to negotiable mode, the reader will reject the card.

When the card ATR indicates the negotiable mode (TA_2 not present) and communication parameters other than the default parameters, the ACR38 will execute the PPS and try to use the communication parameters that the card suggested in its ATR. If the card does not accept the PPS, the reader will use the default parameters (F=372, D=1).

For the meaning of the aforementioned parameters, please refer to *ISO7816, part 3*.

3.2 Memory-based smart cards (synchronous interface)

- Cards following the I2Cbus protocol (free memory cards) with maximum 128 bytes page with capability, including:
  Atmel AT24C01/02/04/08/16/32/64/128/256/512/1024
- Cards with secure memory IC with password and authentication, including:
  Atmel AT88SC153, AT88SC1608
- Cards with intelligent 1k bytes EEPROM with write-protect function, including:
  Infineon SLE4418, SLE4428
- Cards with intelligent 256 bytes EEPROM with write-protect function, including:
  Infineon SLE4432, SLE4442

4. SMART CARD INTERFACE

The interface between the ACR38 and the inserted smart card follows the specifications of *ISO7816-3* with certain restrictions or enhancements to increase the practical functionality of the ACR38.

4.1 Smart Card Power Supply VCC (C1)

The current consumption of the inserted card must not be higher than 50 **mA**.

4.2 Programming Voltage VPP (C6)

According to ISO 7816-3, the smart card contact C6 (VPP) supplies the programming voltage to the smart card. Since all common smart cards in the market are EEPROM based and do not require the provision of an external programming voltage, the contact C6 (VPP) has been implemented as a normal control signal in the ACR38. The electrical specifications of this contact are identical to those of the signal RST (at contact C2).

4.3 Card Type Selection

The controlling PC has to always select the card type through the proper command sent to the ACR38 prior to activating the inserted card. This includes both the memory cards and MCU-based cards.

For MCU-based cards the reader allows to select the preferred protocol, T=0 or T=1. However, this selection is only accepted and carried out by the reader through the PPS when the card inserted in the reader supports both protocol types. Whenever an MCU-based card supports only one protocol type, T=0 or T=1, the reader automatically uses that protocol type, regardless of the protocol type selected by the application.

4.4 Interface for Microcontroller-based Cards

For microcontroller-based smart cards only the contacts C1 (VCC), C2 (RST), C3 (CLK), C5 (GND) and C7 (I/O) are used. A frequency of 4 MHz is applied to the CLK signal (C3).

4.5 Card Tearing Protection

The ACR38 provides a mechanism to protect the inserted card when it is suddenly withdrawn while it is powered up. The power supply to the card and the signal lines between the ACR38 and the card are immediately deactivated when the card is being removed. As a general rule, however, to avoid any electrical damage, **a card should only be removed from the reader while it is powered down.**

NOTE - The ACR38 does never by itself switch on the power supply to the inserted card. This must explicitly be done by the controlling computer through the proper command sent to the reader.

5 POWER SUPPLY

The ACR38 requires a voltage of 5V DC, 100mA, regulated, power supply. The ACR38 gets the power supply from PC (through the cable supplied along with each type of reader).

5.1 Status LED

Green LED on the front of the reader indicate the activation status of the smart card interface:

Flashing slowly (turns on 200ms for every 2 seconds)

Indicates ACR38 is powered up and in the standby state. Either the smart card has not been inserted or the smart card has not been powered up (if it is inserted).

Lighting up

Indicates power supply to the smart card is switched on, i.e., the smart card is activated.

Flashing quickly

Indicates there are communications between ACR38 and smart card.

6. USB INTERFACE

The ACR38 is connected to a computer through a USB following the USB standard.

6.1 Communication Parameters

The ACR38 is connected to a computer through USB as specified in the USB Specification 2.0. The ACR38 is working in full speed mode, i.e. 12 Mbps.

USB Interface Wiring

Pin	Signal	Function
1	**V_{BUS}**	+5V power supply for the reader
2	**D-**	Differential signal transmits data between ACR30 and PC.
3	**D+**	Differential signal transmits data between ACR30 and PC.
4	**GND**	Reference voltage level for power supply

NOTE - In order for the ACR38 to function properly through USB interface, either **ACS proprietary device driver** or **ACS PC/SC device driver** has to be installed. Please refer to the *Device Driver Installation Guide* for more detail.

6.2 Endpoints

The ACR38 uses the following endpoints to communicate with the host computer:

Control Endpoint For setup and control purpose

Bulk OUT For command to sent from host to ACR38 (data packet size is 64 bytes)

Bulk IN For response to sent from ACR38 to host (data packet size is 64 bytes)

Interrupt IN For card status message to sent from ACR38 to host (data packet size is 8 bytes)

7. COMMUNICATION PROTOCOL

In the normal operation, the ACR38 acts as a slave device with regard to the communication between a computer and the reader. The communication is carried out in the form of successive command-response exchanges. The computer transmits a command to the reader and receives a response from the reader after the command has been executed. A new command can be transmitted to the ACR38 only after the response to the previous command has been received.

There are two cases where the reader transmits data without having received a command from the computer, namely, the Reset Message of the reader and the Card Status Message.

7.1 Command to ACR38

A command consists of six protocol bytes and a variable number of data bytes and has the following structure:

Byte	1	2	3 4	5 ... N+4 (N>0)
	Header	Instruction	Data Length = N	Data
	01_H		Data Length N	

Header Always 01_H to indicate the start of a command.

Instruction The instruction code of the command to be carried out by the ACR38.

Data Length Number of subsequent data bytes, and is encoded in 2 bytes. The first byte (MSB) and second byte (LSB) represent data length N.

Data Data contents of the command.

For a READ command, for example, the data bytes would specify the start address and the number of bytes to be read. For a WRITE command, the data bytes would specify the start address and the data to be written to the card.

The data bytes can represent values to be written to a card and/or command parameters such as an address, a counter, etc.

Note: Commands are sent from host computer to ACR38 through the BULK OUT endpoint.

7.2 Response from ACR38

The response from the ACR38 to any command depends on whether the command has been received by the reader without error (e.g., checksum error).

The response by the ACR38 to a correctly received command consists of three protocol bytes, two status bytes and a variable number of data bytes and has the following structure:

Byte	1	2	3	4	5 ... N+4 (N>0)
	Header	Status	Data Length = N		Data
	01_H		Data Length N		

Header Always 01_H to indicate the start of the response.

Status Indicates the command execution status:

00_H = command successfully executed

Otherwise = error in command data, or command cannot be executed

A table listing the possible values of the status byte and the corresponding meaning is given in Appendix B.

Data Length Number of subsequent data bytes, and is encoded in 2 bytes. The first byte (MSB) and second byte (LSB) represent data length N.

Data Data contents of the command.

For a *READ_DATA* command, for example, the data bytes would contain the contents of the memory addresses read from the card. The data bytes can represent values read from the card and/or status information.

Note: Responses are sent from ACR38 to the host computer through BULK IN endpoint.

7.3 Card Status Message

When a card is being inserted into the reader or an inserted card is being removed from the reader while the reader is idle, i.e., not executing a command, the reader transmits a Card Status Message to notify the host computer of the change in the card insertion status.
The Card Status Messages have the following structure and contents:

Card Status Message for Card Insertion

Byte	1	2	3	4
	Header	Status	Data Length	
	01_H	$C1_H$	00_H	00_H

Card Status Message for Card Removal

Byte	1	2	3	4
	Header	Status	Data Length	
	01_H	$C0_H$	00_H	00_H

A card status message is transmitted only **once** for every card insertion or removal event. The reader does not expect an acknowledge signal from the computer. After transmitting a status message, the reader waits for the next command from the computer.

Note: Card status messages are sent from ACR38 to the host computer through INTERRUPT IN endpoint.

8. Commands

The commands executed by the ACR38 can generally be divided into two categories, namely, Control Commands and Card Commands.

Control Commands control the internal operation of the ACR38. They do not directly affect the card inserted in the reader and are therefore independent of the selected card type.

Card Commands are directed toward the card inserted in the ACR38. The structure of these commands and the data transmitted in the commands and responses depend on the selected card type.

8.1 Control Commands

8.1.1 GET_ACR_STAT

This command returns relevant information about the particular ACR38 model and the current operating status, such as, the firmware revision number, the maximum data length of a command and response, the supported card types, and whether a card is inserted and powered up.

Command format

Header	Instruction	Data length	
01_H	01_H	00_H	00_H

Response data format

Header	Status	Data length LEN		INTERNAL										MAX_C	MAX_R	C_TYPE		C_SEL	C_STAT
01_H																			

INTERNAL 10 bytes data for internal use only

MAX_C The maximum number of command data bytes.

MAX_R The maximum number of data bytes that can be requested to be transmitted in a response.

C_TYPE The card types supported by the ACR38. This data field is a bitmap with each bit representing a particular card type. A bit set to '1' means the corresponding card type is supported by the reader and can be selected with the *SELECT_CARD_TYPE* command. The bit assignment is as follows:

Byte	1								2							
card type	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

See Appendix A for the correspondence between these bits and the respective card types.

C_SEL The currently selected card type as specified in a previous *SELECT_CARD_TYPE* command. A value of 00_H means that no card type has been selected.

C_STAT Indicates whether a card is physically inserted in the reader and whether the card is powered up:

00_H: no card inserted

01_H: card inserted, not powered up

03_H: card powered up

8.1.2 SELECT_CARD_TYPE

This command sets the required card type. The firmware in the ACR38 adjusts the communication protocol between reader and the inserted card according to the selected card type.

Command format

Header	Instruction	Data length		Data
		LEN		TYPE
01_H	02_H	00_H	01_H	

TYPE See Appendix A for the value to be specified in this command for a particular card to be used.

Response data format

Header	Status	Data length	
		LEN	
01_H			

8.1.3 SET_OPTION

This command selects the options for the reader.

Command format

Header	Instruction	Data length		Data
		LEN		Option
01_H	07_H	00_H	01_H	

Option Bit 4 : Select for EMV mode
Specifies whether the reader is in EMV mode
0 : reader not in EMV mode (default)
1 : reader in EMV mode
Bit 5 : Select for memory card mode
Specifies whether the reader is in EMV mode
0 : reader not in memory card mode (default)
1 : reader in memory card mode
Bit 0, 1, 2, 3, 6 and 7
Reserved

Response data format

Header	Status	Data length LEN	
01_H			

8.1.4 SET_CARD_PPS

This command sends PPS Request to the smart card. This command should work in pair with SET_READER_PPS.

Command format

Header	Instruction	Data length LEN		Data PPS Request
01_H	$0A_H$	MSB	LSB	

LEN Length of PPS request. Typical value is "4"

PPS Request PPS Request to send to the card (Please refer to ISO/IEC 7816-3:1997 Section 7 for details of PPS request)
A typical PPS request to select T=1 protocol and FD=0x94 (62500 baud at 4MHz) is: 0xFF 0x11 0x94 0x7A

Response data format

Header	Status	Data length LEN		Data				
01_H							…	

8.1.5 SET_READER_PPS

This command sends PPS Response to the reader, and asks the reader to switch its protocol and/or speed to communication with the smart card. This command should work in pair with SET_CARD_PPS.

Command format

Header	Instruction	Data length LEN		Data PPS Response
01_H	$0B_H$	MSB	LSB	

LEN Length of PPS response. Typical value is "4".

PPS Response PPS Response received from the card (Please refer to ISO/IEC 7816-3:1997 Section 7 for details of PPS response). After the driver or the application validates the PPS Response, it should send the PPS Response to the reader. The reader can then switch the protocol and/or speed.
A typical PPS response should be the same as PPS Request.

Response data format

Header	Status	Data length LEN	
01_H			

8.2 Card Commands

The available commands and the parameters specified in the card commands as well as the data transmitted in the response from the ACR38 depend on the selected card type.

8.2.1 MCU-based Card

8.2.1.1 RESET_WITH_5_VOLTS_DEFAULT

This command powers up the card inserted in the card reader and performs a card reset. If the card is powered up when the command is being issued, only a reset of the card is carried out. The power supply to the card is not switched off.

Command format

Header	Instruction	Data length LEN	
01_H	80_H	00_H	00_H

Response data format

Header	Status	Data length LEN		ATR				
01_H							.....	

ATR Answer-To-Reset as transmitted by the card according to ISO7816-3.

NOTE - The ATR is only returned in the ACR38 response if the communication protocol of the card is compatible with the reader, i.e., if the card can be processed by the ACR38. Otherwise, the ACR38 returns an error status and deactivates the smart card interface.

8.2.1.2 RESET_WITH_SPECIFIC_VOLTAGE

This command powers up the card inserted in the card reader and performs a card reset. If the card is powered up when the command is being issued, only a reset of the card is carried out. The power supply to the card is not switched off.

Command format

Header	Instruction	Data length LEN		Data
01_H	80_H	00_H	01_H	

Data = 00_H for automatic voltage detection
= 01_H for 5-volt card
= 02_H for 3-volt card
= 03_H for 1.8-volt card

Response data format

Header	Status	Data length LEN		ATR				
01_H							.....	

ATR Answer-To-Reset as transmitted by the card according to ISO7816-3.

NOTE - The ATR is only returned in the ACR38 response if the communication protocol of the card is compatible with the reader, i.e., if the card can be processed by the ACR38. Otherwise, the ACR38 returns an error status and deactivates the smart card interface.

8.2.1.3 POWER_OFF

This command powers off the card inserted in the card reader.

Command format

Header	Instruction	Data length LEN	
01_H	81_H	00_H	00_H

Response data format

Header	Status	Data length LEN	
01_H			

8.2.1.4 EXCHANGE_TPDU_T0

To exchange an APDU (Application Protocol Data Unit) command/response pair between the MCU card inserted in the ACR38 and the host computer.

Command format

Header	Instruction	Data length LEN		Data				
		MSB	LSB	T0 TPDU				
01_H	$A0_H$						.....	

LEN Length of APDU command data, N

Data T0 TPDU to be sent to the card

Case 1: CLA INS P1 P2 00

Case 2: CLA INS P1 P2 Le

Case 3: CLA INS P1 P2 Lc Data

Case 4: Not supported. The driver/application should break case 4 command into case 3 + case 2 commands.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01_H									

BYTE x Response data from card (if any).

SW1, SW2 Status code returned by the card.

8.2.1.5 EXCHANGE_TPDU_T1

To exchange an APDU (Application Protocol Data Unit) command/response pair between the MCU card inserted in the ACR38 and the host computer using T1 protocol.

Command format

Header	Instruction	Data length LEN		Data				
		MSB	LSB	T1 TPDU Frame				
01_H	$A1_H$	MSB	LSB				……	

LEN Length of APDU command data, N

Data T1 TPDU frame to be sent to the card. It should include NAD, PCB, LEN, INF and EDC fields. Please refer to ISO/IEC 7816:3:1997(E) Section 9.4 for detailed information.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N
01_H							

BYTE x Response T1 Block from card (if any). The response should include NAD, PCB, LEN, INF and EDC fields. Please refer to ISO/IEC 7816:3:1997(E) Section 9.4 for detailed information.

8.2.2 Memory Card – 1,2,4,8,16 kbit I2C card

8.2.2.1 SELECT_CARD_TYPE

This command powers down and up the selected card inserted in the card reader and performs a card reset.

Notes: This command must be the first command after card insertion!

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Card Type
01 $_H$	A0 $_H$	00 $_H$	06 $_H$	FF $_H$	A4 $_H$	00 $_H$	00 $_H$	01 $_H$	01 $_H$

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01 $_H$									

BYTE x ATR from card (if any)
SW1, SW2 = 90 $_H$ 00 $_H$ if no error

8.2.2.2 SELECT_PAGE_SIZE

This command will choose the page size to read the smart card. The default value is 8-byte page write. It will reset to default value whenever the card is removed or the reader is powered off.

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Page size
01 $_H$	A0 $_H$	00 $_H$	06 $_H$	FF $_H$	01 $_H$	00 $_H$	00 $_H$	01 $_H$	

Page size = 03 $_H$ for 8-byte page write
= 04 $_H$ for 16-byte page write
= 05 $_H$ for 32-byte page write
= 06 $_H$ for 64-byte page write
= 07 $_H$ for 128-byte page write

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.2.3 READ_MEMORY_CARD

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	Byte Address		MEM_L
						MSB	LSB	
01_H	$A0_H$	00_H	05_H	FF_H	$B0_H$			

Byte Address Memory address location of the memory card.

MEM_L Length of data to be read from the memory card.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01_H									

BYTE x Data read from memory card

SW1, SW2 = 90_H 00_H if no error

8.2.2.4 WRITE_MEMORY_CARD

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte n
		MSB	LSB			MSB	LSB					
01_H	$A0_H$			FF_H	$D0_H$							

Byte Address Memory address location of the memory card.

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

Byte x Data to be written to the memory card.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.3 Memory Card – 32,64,128,256,512,1024 kbit I2C card

8.2.3.1 SELECT_CARD_TYPE

This command powers down and up the selected card inserted in the card reader and performs a card reset.

Notes: This command must be the first command after card insertion!

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Card Type
01 $_H$	A0 $_H$	00 $_H$	06 $_H$	FF $_H$	A4 $_H$	00 $_H$	00 $_H$	01 $_H$	02 $_H$

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01 $_H$									

BYTE x ATR from card (if any)

SW1, SW2 = 90 $_H$ 00 $_H$ if no error

8.2.3.2 SELECT_PAGE_SIZE

This command will choose the page size to read the smart card. The default value is 8-byte page write. It will reset to default value whenever the card is removed or the reader is powered off.

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Page size
01 $_H$	A0 $_H$	00 $_H$	06 $_H$	FF $_H$	01 $_H$	00 $_H$	00 $_H$	01 $_H$	

Data TPDU to be sent to the card

Page size = 03 $_H$ for 8-byte page write

= 04 $_H$ for 16-byte page write

= 05 $_H$ for 32-byte page write

= 06 $_H$ for 64-byte page write

= 07 $_H$ for 128-byte page write

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.3.3 READ_MEMORY_CARD

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	Byte Address		MEM_L
						MSB	LSB	
01_H	$A0_H$	00_H	05_H	FF_H				

INS = $B0_H$ for 32,64,128,256,512kbit iic card

= 1011 000* $_b$ for 1024kbit iic card, where * is the MSB of the 17 bit addressing

Byte Address Memory address location of the memory card.

MEM_L Length of data to be read from the memory card.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01_H									

BYTE x Data read from memory card

SW1, SW2 = 90_H 00_H if no error

8.2.3.4 WRITE_MEMORY_CARD

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte n
		MSB	LSB			MSB	LSB					
01_H	$A0_H$			FF_H								

INS = $D0_H$ for 32,64,128,256,512kbit iic card

= 1101 000* $_b$ for 1024kbit iic card, where * is the MSB of the 17 bit addressing

Byte Address Memory address location of the memory card.

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

Byte x Data to be written to the memory card.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.4 Memory Card – ATMEL AT88SC153

8.2.4.1 SELECT_CARD_TYPE

This command powers down and up the selected card inserted in the card reader and performs a card reset. It will also select the page size to be 8-byte page write.

Notes: This command must be the first command after card insertion!

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Card Type
01_H	$A0_H$	00_H	06_H	FF_H	$A4_H$	00_H	00_H	01_H	03_H

Response data format

Header	Status	Data length LEN		BYTE 1	BYTE 2	BYTE 3	BYTE 4	SW1	SW2
01_H									

BYTE x ATR from card

SW1, SW2 = 90_H 00_H if no error

8.2.4.2 READ_MEMORY_CARD

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	P1	Byte Address	MEM_L
01_H	$A0_H$	00_H	05_H	FF_H		00_H		

INS = $B0_H$ for reading zone 00_b

= $B1_H$ for reading zone 01_b

= $B2_H$ for reading zone 10_b

= $B3_H$ for reading zone 11_b

= $B4_H$ for reading fuse

Byte Address Memory address location of the memory card.

MEM_L Length of data to be read from the memory card.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01_H									

BYTE x Data read from memory card

SW1, SW2 = 90_H 00_H if no error

8.2.4.3 WRITE_MEMORY_CARD

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	Bye Address	MEM_L	Byte 1	....	....	Byte n
		MSB	LSB									
01_H	$A0_H$			FF_H		00_H						

INS = $D0_H$ for writing zone 00_b

= $D1_H$ for writing zone 01_b

= $D2_H$ for writing zone 10_b

= $D3_H$ for writing zone 11_b

= $D4_H$ for writing fuse

Byte Address Memory address location of the memory card.

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

MEM_D Data to be written to the memory card.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.4.4 VERIFTY_PASSWORD

Command format

Header	Instruction	Data length		Data							
		LEN		CLA	INS	P1	P2	Lc	Pw(0)	Pw(1)	Pw(2)
01_H	$A0_H$	00_H	08_H	FF_H	20_H	00_H		03_H			

Pw(0),Pw(1),Pw(2) Passwords to be sent to memory card.

P2 = $0000\ 00rp_b$

where the two bits “rp” indicate the password to compare

r = 0 : Write password,

r = 1: Read password,

p: Password set number,

rp = 01 for the secure code.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.4.5 INITIALIZE_AUTHENTICATION

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	P2	Lc	Q(0)	Q(1)	…	Q(7)
01_H	$A0_H$	00_H	$0D_H$	FF_H	84_H	00_H	00_H	08_H				

Q(0),Q(1)…Q(7) Host random number, 8 bytes.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.4.6 VERIFY_AUTHENTICATION

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	P2	Lc	Ch(0)	Ch(1)	…	Ch(7)
01_H	$A0_H$	00_H	$0D_H$	FF_H	82_H	00_H	00_H	08_H				

Ch(0),Ch(1)…Ch(7) Host challenge, 8 bytes.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.5 Memory Card – ATMEL AT88SC1608

8.2.5.1 SELECT_CARD_TYPE

This command powers down and up the selected card inserted in the card reader and performs a card reset. It will also select the page size to be 16-byte page write.

Notes: This command must be the first command after card insertion!

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Card Type
01 H	A0 H	00 H	06 H	FF H	A4 H	00 H	00 H	01 H	04 H

Response data format

Header	Status	Data length LEN		BYTE 1	BYTE 2	BYTE 3	BYTE 4	SW1	SW2
01 H									

BYTE x ATR from card

SW1, SW2 = 90 H 00 H if no error

8.2.5.2 READ_MEMORY_CARD

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	Zone Address	Byte Address	MEM_L
01 H	A0 H	00 H	05 H	FF H				

INS = B0 H for reading user zone

= B1 H for reading configuration zone or reading fuse

Zone Address = 0000 $0A_{10}A_9A_8$ b, where A_{10} is the MSB of zone address

= don't care for reading fuse

Byte Address = $A_7A_6A_5A_4$ $A_3A_2A_1A_0$ b is the memory address location of the memory card.

= 1000 0000 b for reading fuse

MEM_L Length of data to be read from the memory card.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01_H									

BYTE x Data read from memory card

SW1, SW2 = 90_H 00_H if no error

8.2.5.3 WRITE_MEMORY_CARD

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	Zone Address	Byte Address	MEM_L	Byte 1	....	....	Byte n
		MSB	LSB									
01_H	$A0_H$			FF_H								

INS = $D0_H$ for writing user zone

= $D1_H$ for writing configuration zone or writing fuse

Zone Address = $0000\ 0A_{10}A_9A_{8\ b}$, where A_{10} is the MSB of zone address

= don't care for writing fuse

Byte Address = $A_7A_6A_5A_4\ A_3A_2A_1A_{0\ b}$ is the memory address location of the memory card.

= $1000\ 0000_b$ for writing fuse

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

Byte x Data to be written to the memory card.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.5.4 VERIFTY_PASSWORD

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	P2	Lc	Data			
01_H	$A0_H$	00_H	08_H	FF_H	20_H	00_H	00_H	04_H	RP	Pw(0)	Pw(1)	Pw(2)

Pw(0),Pw(1),Pw(2) Passwords to be sent to memory card.

RP = $0000\ rp_2p_1p_{0\ b}$

where the four bits "$rp_2p_1p_0$" indicate the password to compare:

r = 0: Write password,

r = 1: Read password,

$p_2p_1p_0$: Password set number.

($rp_2p_1p_0$ = 0111 for the secure code).

Response data format

Header	Status	Data length LEN		SW1	SW2
01 H					

SW1, SW2 = 90 H 00 H if no error

8.2.5.5 INITIALIZE_AUTHENTICATION

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	P2	Lc	Q(0)	Q(1)	…	Q(7)
01 H	A0 H	00 H	0D H	FF H	84 H	00 H	00 H	08 H				

Byte Address Memory address location of the memory card.

Q(0),Q(1)…Q(7) Host random number, 8 bytes.

Response data format

Header	Status	Data length LEN		SW1	SW2
01 H					

SW1, SW2 = 90 H 00 H if no error

8.2.5.6 VERIFY_AUTHENTICATION

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	P2	Lc	Q1(0)	Q1(1)	…	Q1(7)
01 H	A0 H	00 H	0D H	FF H	82 H	00 H	00 H	08 H				

Byte Address Memory address location of the memory card.

Q1(0),Q1(1)…Q1(7) Host challenge, 8 bytes.

Response data format

Header	Status	Data length LEN		SW1	SW2
01 H					

SW1, SW2 = 90 H 00 H if no error

8.2.6 Memory Card – SLE4418/4428

8.2.6.1 SELECT_CARD_TYPE

This command powers down and up the selected card inserted in the card reader and performs a card reset.

Notes: This command must be the first command after card insertion!

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Card Type
01_H	$A0_H$	00_H	06_H	FF_H	$A4_H$	00_H	00_H	01_H	05_H

Response data format

Header	Status	Data length LEN		BYTE 1	BYTE 2	BYTE 3	BYTE 4	SW1	SW2
01_H									

BYTE x ATR from card

SW1,SW2 = 90_H 00_H if no error

8.2.6.2 READ_MEMORY_WITH_PROTECT_BIT_CARD

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	Byte Address		MEM_L
						MSB	LSB	
01_H	$A0_H$	00_H	05_H	FF_H	$B0_H$			

MSB Byte Address = $0000\ 00A_9A_8{}_b$ is the memory address location of the memory card.

LSB Byte Address = $A_7A_6A_5A_4\ A_3A_2A_1A_0{}_b$ is the memory address location of the memory card.

MEM_L Length of data to be read from the memory card. (Max. allowable size is EC_H.)

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	PROT 1	…	…	PROT L	SW1	SW2
01_H													

BYTE x Data read from memory card

PROT y Bytes containing the protection bits of the data bytes read

SW1,SW2 = 90_H 00_H if no error

The number L of protection bytes returned in the response is determined by the number N of data bytes read from the card as follows:

L = 1 + INT(N/8)

The arrangement of the protection bits in the PROT bytes is as follows:

PROT 1								PROT 2								…							
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9	..	..	..	..	..	..	P18	P17

Px is the protection bit of BYTE x in the response data

'0' byte is write protected

'1' byte can be written

8.2.6.3 READ_MEMORY_WITHOUT_PROTECT_BIT_CARD

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	Byte Address		MEM_L
						MSB	LSB	
01_H	$A0_H$	00_H	05_H	FF_H	$B2_H$			

MSB Byte Address = 0000 $00A_9A_8{}_b$ is the memory address location of the memory card.

LSB Byte Address = $A_7A_6A_5A_4$ $A_3A_2A_1A_0{}_b$ is the memory address location of the memory card.

MEM_L Length of data to be read from the memory card.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	SW1	SW2
01_H									

BYTE x Data read from memory card

SW1, SW2 = 90_H 00_H if no error

8.2.6.4 WRITE_MEMORY_CARD

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte N
		MSB	LSB			MSB	LSB					
01_H	$A0_H$			FF_H	$D0_H$							

MSB Byte Address = 0000 $00A_9A_{8\ b}$ is the memory address location of the memory card.

LSB Byte Address = $A_7A_6A_5A_4\ A_3A_2A_1A_{0\ b}$ is the memory address location of the memory card.

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

Byte x Data to be written to the memory card.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.6.5 WRITE_PROTECTION_MEMORY_CARD

Each of the bytes specified in the command is internally in the card compared with the byte stored at the specified address and if the data match, the corresponding protection bit is irreversibly programmed to '0'.

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte N
		MSB	LSB			MSB	LSB					
01_H	$A0_H$			FF_H	$D1_H$							

MSB Byte Address = 0000 $00A_9A_{8\ b}$ is the memory address location of the memory card.

LSB Byte Address = $A_7A_6A_5A_4\ A_3A_2A_1A_{0\ b}$ is the memory address location of the memory card.

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

Byte x Byte values to be compared with the data in the card starting at Byte Address. BYTE 1 is compared with the data at Byte Address; BYTE N is compared with the data at (Byte Address+N-1).

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.6.6 PRESENT_CODE_MEMORY_CARD (only SLE 4428)

To submit the secret code to the memory card to enable the write operation with the SLE4428 card. The following actions are executed:

- search a '1' bit in the presentation error counter and write the bit to '0'
- present the specified code to the card
- try to erase the presentation error counter

Command format

Header	Instruction	Data length		Data						
		LEN		CLA	INS	P1	P2	MEM_L	CODE	
		MSB	LSB						Byte 1	Byte 2
01_H	$A0_H$	00_H	07_H	FF_H	20_H	00_H	00_H	02_H		

CODE Two bytes secret code (PIN)

Response data format

Header	Status	Data length LEN		ERRCNT	CODE		SW1	SW2
					Byte 1	Byte 2		
01_H								

ERRCNT The value of the presentation error counter after the code presentation.

CODE The two bytes secret code read from the card.

SW1, SW2 = 90_H 00_H if no error

If the correct code has been presented to the card, the value of ERRCNT is FF_H and the value of CODE is identical to the code data specified in the command.

8.2.6.7 READ_PRESENTATION_ERROR_COUNTER_MEMORY_CARD (only SLE 4428)

To read the presentation error counter for the secret code.

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	P1	P2	MEM_L
		MSB	LSB					
01_H	$A0_H$	00_H	05_H	FF_H	$B1_H$	00_H	00_H	00_H

Response data format

Header	Status	Data length LEN		ERRCNT	DUMMY 1	DUMMY 2	SW1	SW2
01_H								

ERRCNT The value of the presentation error counter.

DUMMY Three bytes dummy data read from the card.

SW1, SW2 = 90_H 00_H if no error

8.2.7 Memory Card – SLE4432/4442

8.2.7.1 SELECT_CARD_TYPE

This command powers down and up the selected card inserted in the card reader and performs a card reset.

Notes: This command must be the first command after card insertion!

Command format

Header	Instruction	Data length		Data					
		LEN		CLA	INS	P1	P2	Lc	Card Type
01_H	$A0_H$	00_H	06_H	FF_H	$A4_H$	00_H	00_H	01_H	06_H

Response data format

Header	Status	Data length LEN		BYTE 1	BYTE 2	BYTE 3	BYTE 4	SW1	SW2
01_H									

BYTE x ATR from card

SW1, SW2 = 90_H 00_H if no error

8.2.7.2 READ_MEMORY_CARD

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	P1	Byte Address	MEM_L
01_H	$A0_H$	00_H	05_H	FF_H	$B0_H$	00_H		

Byte Address = $A_7A_6A_5A_4\ A_3A_2A_1A_0\ _b$ is the memory address location of the memory card.

MEM_L Length of data to be read from the memory card.

Response data format

Header	Status	Data length LEN		BYTE 1	…	…	BYTE N	PROT 1	PROT 2	PROT3	PROT 4	SW1	SW2
01_H													

BYTE x Data read from memory card

PROT y Bytes containing the protection bits from protection memory

SW1, SW2 = 90_H 00_H if no error

The arrangement of the protection bits in the PROT bytes is as follows:

PROT 1								PROT 2								…							
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9	..	..	..	..	..	..	P18	P17

Px is the protection bit of BYTE x in the response data

'0' byte is write protected

'1' byte can be written

8.2.7.3 WRITE_MEMORY_CARD

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	Byte Address	MEM_L	Byte 1	....	....	Byte N
		MSB	LSB									
01_H	$A0_H$			FF_H	$D0_H$	00_H						

Byte Address = $A_7A_6A_5A_4\ A_3A_2A_1A_{0\ b}$ is the memory address location of the memory card.

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

Byte x Data to be written to the memory card.

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

8.2.7.4 WRITE_PROTECTION_MEMORY_CARD

Each of the bytes specified in the command is internally in the card compared with the byte stored at the specified address and if the data match, the corresponding protection bit is irreversibly programmed to '0'.

Command format

Header	Instruction	Data length		Data								
		LEN		CLA	INS	P1	Byte Address	MEM_L	Byte 1	....	....	Byte N
		MSB	LSB									
01_H	$A0_H$			FF_H	$D1_H$	00_H						

Byte Address = $000A_4\ A_3A_2A_1A_0\ _b$ ($00\ _H$ to $1F\ _H$) is the protection memory address location of the memory card.

LEN = 5 + MEM_L

MEM_L Length of data to be written to the memory card.

Byte x Byte values to be compared with the data in the card starting at Byte Address. BYTE 1 is compared with the data at Byte Address; BYTE N is compared with the data at (Byte Address+N-1).

Response data format

Header	Status	Data length LEN		SW1	SW2
$01\ _H$					

SW1, SW2 = $90\ _H\ 00\ _H$ if no error

8.2.7.5 PRESENT_CODE_MEMORY_CARD (only SLE 4442)

To submit the secret code to the memory card to enable the write operation with the SLE4442 card. The following actions are executed:

- search a '1' bit in the presentation error counter and write the bit to '0'
- present the specified code to the card
- try to erase the presentation error counter

Command format

Header	Instruction	Data length		Data							
		LEN		CLA	INS	P1	P2	MEM_L	CODE		
		MSB	LSB						Byte 1	Byte 2	Byte 3
$01\ _H$	$A0\ _H$	$00\ _H$	$08\ _H$	$FF\ _H$	$20\ _H$	$00\ _H$	$00\ _H$	$03\ _H$			

CODE Three bytes secret code (PIN)

Response data format

Header	Status	Data length LEN		ERRCNT	CODE			SW1	SW2
					Byte 1	Byte 2	Byte 3		
$01\ _H$									

ERRCNT The value of the presentation error counter after the code presentation.

CODE The three bytes secret code read from the card.

SW1, SW2 = $90\ _H\ 00\ _H$ if no error

If the correct code has been presented to the card, the value of ERRCNT is $07\ _H$ and the value of CODE is identical to the code data specified in the command.

8.2.7.6 READ_PRESENTATION_ERROR_COUNTER_MEMORY_CARD (only SLE 4442)

To read the presentation error counter for the secret code.

Command format

Header	Instruction	Data length		Data				
		LEN		CLA	INS	P1	P2	MEM_L
		MSB	LSB					
01_H	$A0_H$	00_H	08_H	FF_H	$B1_H$	00_H	00_H	00_H

Response data format

Header	Status	Data length LEN		ERRCNT	DUMMY 1	DUMMY 2	DUMMY 3	SW1	SW2
01_H									

ERRCNT The value of the presentation error counter.

DUMMY Three bytes dummy data read from the card.

SW1, SW2 = 90_H 00_H if no error

8.2.7.7 CHANGE_CODE_MEMORY_CARD (only SLE 4442)

To write the specified data as new secret code in the card.

The current secret code must have been presented to the card with the PRESENT_CODE command prior to the execution of this command!

Command format

Header	Instruction	Data length		Data							
		LEN		CLA	INS	P1	P2	MEM_L	CODE		
		MSB	LSB						Byte 1	Byte 2	Byte 3
01_H	$A0_H$	00_H	08_H	FF_H	$D2_H$	00_H	01_H	03_H			

Response data format

Header	Status	Data length LEN		SW1	SW2
01_H					

SW1, SW2 = 90_H 00_H if no error

9 APPENDIX A: SUPPORTED CARD TYPES

The following table summarizes which values must be specified in the *SET_CARD_TYPE* command for a particular card type to be used, and how the bits in the response to the *GET_ACR_STAT* command correspond with the respective card types.

Cyber-mouse card type code	Card Type
$\mathbf{00_H}$	Auto-select T=0 or T=1 communication protocol
$\mathbf{01_H}$	I2C memory card (1k, 2k, 4k, 8k and 16k bits)
$\mathbf{02_H}$	I2C memory card (32k, 64k, 128k, 256k, 512k and 1024k bits)
$\mathbf{03_H}$	Atmel AT88SC153 secure memory card
$\mathbf{04_H}$	Atmel AT88SC1608 secure memory card
$\mathbf{05_H}$	Infineon SLE4418 and SLE4428
$\mathbf{06_H}$	Infineon SLE4432 and SLE4442
$\mathbf{0C_H}$	MCU-based cards with T=0 communication protocol
$\mathbf{0D_H}$	MCU-based cards with T=1 communication protocol

10 APPENDIX B: RESPONSE STATUS CODES

The following table summarizes the possible status code returned by the ACR38:

Status Cod	Status
00	OK – command successfully executed
F4	SLOTERROT_PROCEDURE_BYTE_CONFLICT
F6	SLOTERROR_BAD_LENGTH
F7	SLOTERROR_BAD_FIDI
F8	SLOTERROR_BAD_ATR_TS
F9	SLOTERROR_ICC_NOT_POWERED_UP
FA	SLOTERROR_ICC_NOT_INSERTED
FB	SLOTERROR_HW_ERROR
FC	SLOTERROR_XFE_OVERRUN
FD	SLOTERROR_XFE_PARITY_ERROR
FE	SLOTERROR_ICC_MUTE
FF	SLOTERROR_CMD_ABORTED

11 APPENDIX C: TECHNICAL SPECIFICATIONS

Device

ACR38 Smart Card Reader/Writer

Power supply

Supply voltage ..... Regulated 5V DC
Supply current..... < 100mA (without smart card)

Universal Serial Bus Interface

Type..... USB, four lines: +5V, GND, D+ and D-
Connector ..... supplied together with the reader
Speed ..... Full Speed Device, 12 Mbps

Smart Card Interface

Standard ..... ISO 7816 1/2/3, T=0 and T=1
Supply current..... max. 50mA
Smart card voltage..... 5V / 3V / 1.8V
Smart card read / write speed..... 9600 – 307200 bps
Short circuit protection ..... +5V / GND on all pins

The presence of the smart card power supply voltage is indicated through a green LED on the reader

CLK frequency ..... 4 MHz
Card connector ..... sliding contacts (8 contacts)
Card insertion cycles..... min. 100,000

Case

Dimensions ..... 77mm (L) x 68mm (B) x 20mm (H)
Color ..... bone gray
Weight..... 0.16 kg

Operating Conditions

Temperature ..... 0 - 50° C

NOTES