



Advanced Card Systems Ltd.
Card & Reader Technologies

ACR83 PINeasy



Application Programming Interface V1.06



Table of Contents

1.0.	Introduction	3
2.0.	Features	4
3.0.	Device Control	5
3.1.	Operation Flow (PC/SC 2.0 Part 10)	5
3.2.	Specific ScardControl	6
3.3.	Smart Card Device IOCTLs	6
3.3.1.	CM_IOCTL_GET_FEATURE_REQUEST	6
3.3.2.	FEATURE_VERIFY_PIN_DIRECT	7
3.3.3.	FEATURE_MODIFY_PIN_DIRECT	8
3.3.4.	FEATURE_IFD_PIN_PROP	10
3.3.5.	IOCTL_SMARTCARD_GET_FIRMWARE_VERSION	11
3.3.6.	IOCTL_SMARTCARD_DISPLAY_LCD_MESSAGE	12
3.3.7.	IOCTL_SMARTCARD_READ_KEY	12
Appendix A.	Set <i>bKeyReturnCondition</i>	14
Appendix B.	Response Error Codes	15
Appendix C.	<i>bmFormatString</i> Description	16
Appendix D.	<i>bmPINBlockString</i> Description	17
Appendix E.	<i>bmPINLength</i> Format	18

List of Figures

Figure 1 :	Operation Flowchart	5
-------------------	----------------------------------	----------



1.0. Introduction

ACR83 PINeasy is a USB reader featuring a keypad and display that supports Secure PIN Entry (SPE) where the PIN (Personal Identification Number) is entered securely using the device keypad. Cardholder authentication is performed within the device and since the PIN is entered in the secure ACR83 PINeasy rather than the PC or workstation, the possibility of middle-man attack is addressed.

This API document shows how the PC/SC APDU commands are implemented for the device peripherals. For contact card operations, please refer to the related card documentation and the PC/SC specifications.



2.0. Features

- 14-key keypad
- 2 rows x 16 characters dot matrix LCD, each character has 5x8 dots
- Supports ISO 7816 Microprocessor Smart Cards with the following features:
 - Class A, B, C (5 V, 3 V and 1.8 V respectively)
 - T=0 and/or T=1 protocol
- Supports Secure PIN Entry (SPE)
- EMV Level 1 Certified
- Full speed USB interface (12 Mbps)
- Compliant to the following standards:
 - PC/SC
 - WHQL
 - CCID
 - CE/FCC
 - RoHS

3.0. Device Control

This section describes the system smart card device IOCTLs.

3.1. Operation Flow (PC/SC 2.0 Part 10)

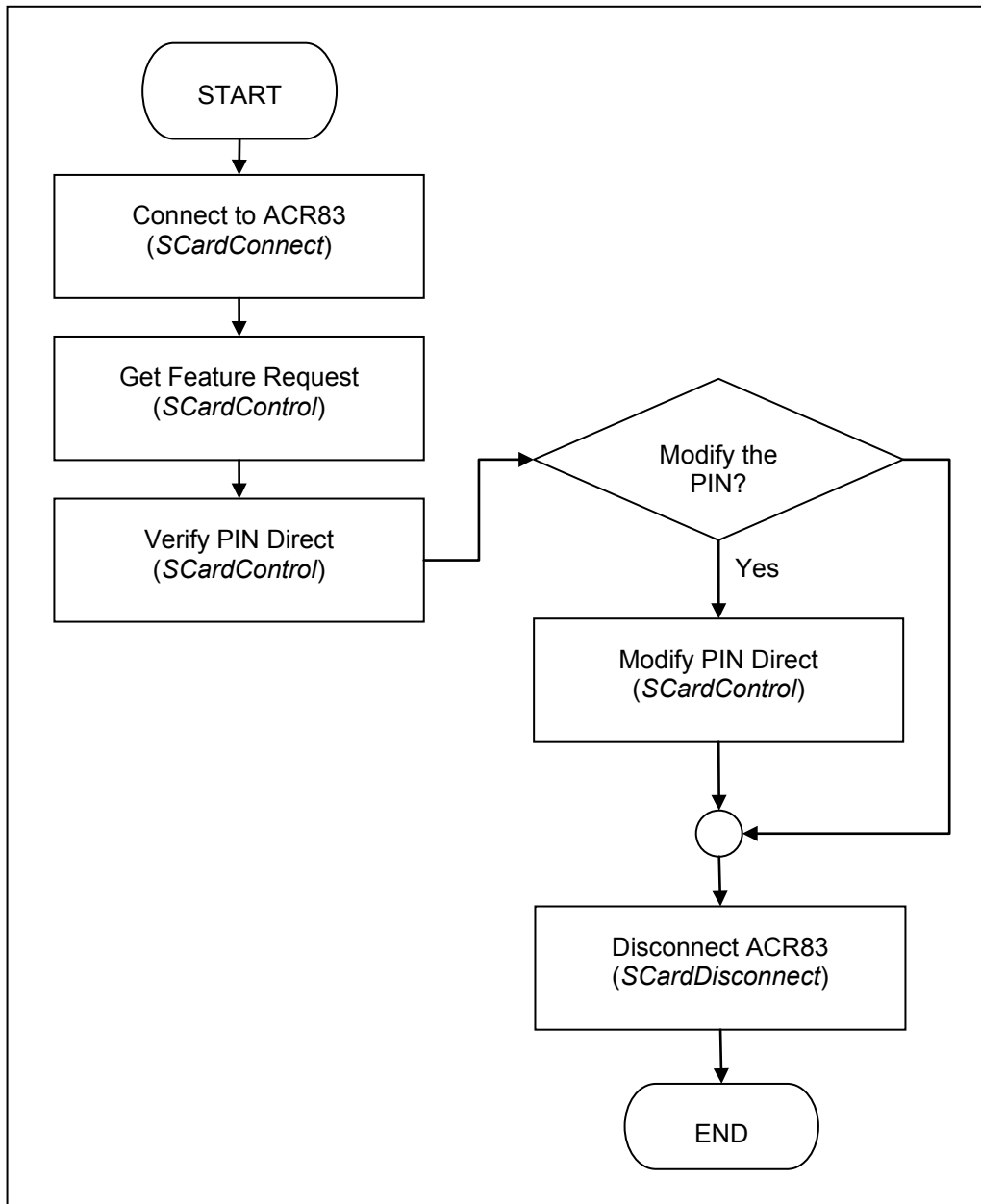


Figure 1: Operation Flowchart

In order to use PIN verification and modification, the *SCardControl* API must be called with Get Feature Request control code and this API will return a list of supported features from the reader.

In ACR83, only *Verify PIN Direct*, *Modify PIN Direct* and *IFD PIN Properties* are supported. To use these features, you can get the control codes from the list. For more information, please refer to PC/SC 2.0 Specification Part 10.



3.2. Specific SCardControl

```
LONG SCardControl(
    SCARDHANDLE hCard,
    DWORD dwControlCode,
    LPCVOID lpInBuffer,
    DWORD nInBufferSize,
    LPVOID lpOutBuffer,
    DWORD nOutBufferSize,
    LPDWORD lpBytesReturned
);
#define IOCTL_SMARTCARD_GET_FIRMWARE_VERSION SCARD_CTL_CODE(2078)
#define IOCTL_SMARTCARD_DISPLAY_LCD_MESSAGE SCARD_CTL_CODE(2079)
#define IOCTL_SMARTCARD_READ_KEY SCARD_CTL_CODE(2080)
// PC/SC 2.0 Part 10
#define CM_IOCTL_GET_FEATURE_REQUEST SCARD_CTL_CODE(3400)
```

Note: Data is stored in little-endian form in which the LSB (Least Significant Byte) is first. Furthermore, SCardControl command must be declared in the source code.

3.3. Smart Card Device IOCTLs

3.3.1. CM_IOCTL_GET_FEATURE_REQUEST

The CM_IOCTL_GET_FEATURE_REQUEST returns a list of supported features from the reader.

hCard	Reference value returned from <i>SCardConnect</i>
dwControlCode	CM_IOCTL_GET_FEATURE_REQUEST
lpInBuffer	NULL
nInBufferSize	Must be the sizeof(ULONG) of <i>lpInBuffer</i>
lpOutBuffer	According to PC/SC 2.0 Specification Part 10, the following features had been defined:

```
#define FEATURE_VERIFY_PIN_START 0x01
#define FEATURE_VERIFY_PIN_FINISH 0x02
#define FEATURE_MODIFY_PIN_START 0x03
#define FEATURE_MODIFY_PIN_FINISH 0x04
#define FEATURE_GET_KEY_PRESSED 0x05
#define FEATURE_VERIFY_PIN_DIRECT 0x06
#define FEATURE_MODIFY_PIN_DIRECT 0x07
#define FEATURE_MCT_READERDIRECT 0x08
#define FEATURE_MCT_UNIVERSAL 0x09
#define FEATURE_IFD_PIN_PROP 0x0A
#define FEATURE_ABORT 0x0B
```

In ACR83, the following features are supported:

```
#define FEATURE_VERIFY_PIN_DIRECT 0x06
#define FEATURE_MODIFY_PIN_DIRECT 0x07
#define FEATURE_IFD_PIN_PROP 0x0A
```



If the ACR83 reader used supports PC/SC 2.0 Part 10, you will get the following data:

06 04 XX XX XX XX 07 04 XX XX XX XX 0A 04 XX XX XX XXh

where, XX XX XX XXh is the control code for the feature.

nOutBufferSize `sizeof(ULONG)` of *IpOutBuffer*

IpBytesReturned pointer to a *DWORD* that receives the size, in bytes, of the data stored into the buffer pointed to by *IpOutBuffer*

3.3.2. FEATURE_VERIFY_PIN_DIRECT

hCard Reference value returned from *SCardConnect*

dwControlCode `CM_IOCTL_GET_FEATURE_REQUEST`

IpInBuffer

Offset	Field	Size	Value	Description
0	<i>bTimeOut</i>	1	-	Number of seconds. If the value is equal to 00h, then default value is used.
1	<i>bTimeOut2</i>	1	00h	Not supported. Number of seconds after first key stroke.
2	<i>bmFormatString</i>	1	-	Several parameters for the PIN format options. <i>For more information, please refer to Appendix A.</i>
3	<i>bmPINBlockString</i>	1	-	Defines the length in bytes of the PIN block to present in the APDU command. <i>For more information, please refer to Appendix B.</i>
4	<i>bmPINLengthFormat</i>	1	-	Allows the insertion of the PIN length in the APDU command. <i>For more information, please refer to Appendix C.</i>
5	<i>wPINMaxExtraDigit</i>	2	XXYYh	XXh: Maximum PIN size in digit YYh: Minimum PIN size in digit
7	<i>bEntryValidationCondition</i>	1	-	The value is a bit wise OR operation. 01h = Maximum size reached 02h = Validation key pressed 04h = Timeout occurred
8	<i>bNumberMessage</i>	1	FFh	Number of messages to display for PIN verification
9	<i>wLangId</i>	2	0409h	Language for messages
11	<i>bMsgIndex</i>	1	00h	Message index (should be 00h)
12	<i>bTeoPrologue</i>	3	000000h	T=1 I-block prologue field to use (fill with 00h)



Offset	Field	Size	Value	Description
15	<i>ulDataLength</i>	4	-	Length of data to be sent to the ICC
19	<i>abData</i>	-	-	Data to send to the ICC

nInBufferSize 19 + *ulDataLength*

lpOutBuffer

Offset	Field	Size	Value	Description
0	<i>abStatus</i>	2	-	6400h: SPE operation timed out 6401h: SPE operation was cancelled by the "Cancel" button 6402h: Modify PIN operation failed because two "New PIN" entries do not match 6403h: User entered too short or too long PIN regarding MIN/MAX PIN Length. Note: ACR83 will not return this status because it checks the PIN length during input. 6B80h: Invalid parameter in passed structure SW1SW2: Result from the card

nOutBufferSize 2

lpBytesReturned Pointer to a *DWORD* that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*.

3.3.3. FEATURE_MODIFY_PIN_DIRECT

hCard Reference value returned from *SCardConnect*

dwControlCode CM_IOCTL_GET_FEATURE_REQUEST

lpInBuffer

Offset	Field	Size	Value	Description
0	<i>bTimeOut</i>	1	-	Number of seconds. If value is equal to 00h, then default value is used.
1	<i>bTimeOut2</i>	1	00h	Not supported. Number of seconds after first key stroke.
2	<i>bmFormatString</i>	1	-	Several parameters for the PIN format options. For more information, please refer to Appendix A .



Offset	Field	Size	Value	Description
3	<i>bmPINBlockString</i>	1	-	Defines the length in bytes of the PIN block to present in the APDU command. <i>For more information, please refer to Appendix B.</i>
4	<i>bmPINLengthFormat</i>	1	-	Allows the insertion of the PIN length in the APDU command. <i>For more information, please refer to Appendix C.</i>
5	<i>bInsertionOffsetOld</i>	1	-	Insertion position offset in byte for the current PIN
6	<i>bInsertionOffsetNew</i>	1	-	Insertion position offset in byte for the new PIN
7	<i>wPINMaxExtraDigit</i>	2	XXYYh	XXh: Maximum PIN size in digit YYh: Minimum PIN size in digit
9	<i>bConfirmPIN</i>	1	00h, 01h, 02h, 03h	Indicates if a confirmation is requested before acceptance of a new PIN (meaning that the user has to enter this new PIN twice before it is accepted) Indicates if the current PIN must be entered and set in the same APDU field of not. b0: (0/1) If 0 = No confirmation requested If 1 = Confirmation requested b1: (0/1) If 0 = No current PIN entry requested. (In this case, the <i>bInsertinoOffsetOld</i> value must not be taken into account.) If 1 = Current PIN entry requested b2 – b7: RFU
10	<i>bEntryValidationCondition</i>	1	-	The value is a bit wise OR operation. 01h = Maximum size reached 02h = Validation key pressed 04h = Timeout occurred
11	<i>bNumberMessage</i>	1	FFh	Number of messages to display for PIN verification
12	<i>wLangId</i>	2	0409h	Language for message
14	<i>bMsgIndex1</i>	1	00h	Index of 1st prompting message
15	<i>bMsgIndex2</i>	1	01h	Index of 2nd prompting message
16	<i>bMsgIndex3</i>	1	02h	Index of 3rd prompting message
17	<i>bTeoPrologue</i>	3	000000h	T=1 I-block prologue field to use (fill with 00h).



Offset	Field	Size	Value	Description
20	<i>ulDataLength</i>	4	-	Length of Data to be sent to the ICC
24	<i>abData</i>	-	-	Data to send to the ICC

nInBufferSize 24 + *ulDataLength*

lpOutBuffer

Offset	Field	Size	Value	Description
0	<i>abStatus</i>	2	-	6400h: SPE operation timed out 6401h: SPE operation was cancelled using the "Cancel" button 6402h: Modify PIN operation failed because two "New PIN" entries do not match 6403h: User entered too short or too long PIN regarding MIN/MAX PIN Length. Note: ACR83 will not return this status because it checks the PIN length during input. 6B80h: Invalid parameter in passed structure SW1SW2: Result from the card

nOutBufferSize 2

lpBytesReturned Pointer to a *DWORD* that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*

3.3.4. FEATURE_IFD_PIN_PROP

hCard Reference value returned from *SCardConnect*.

dwControlCode Return from *CM_IOCTL_GET_FEATURE_REQUEST*.

lpInBuffer NULL

LpOutBuffer

Offset	Field	Size	Value	Description
0	<i>wLcdLayout</i>	2	0210h	Display characteristics: 2 lines, 16 characters per line
2	<i>bEntryValidationCondition</i>	1	07h	Support timeout reached, maximum PIN size reached, validation key pressed
3	<i>bTimeOut2</i>	1	00h	0 = IFD does not distinguish <i>bTimeOut</i> from <i>bTimeOut2</i> 1 = IFD distinguishes <i>bTimeOut</i> from <i>bTimeOut2</i>



nOutBufferSize 4

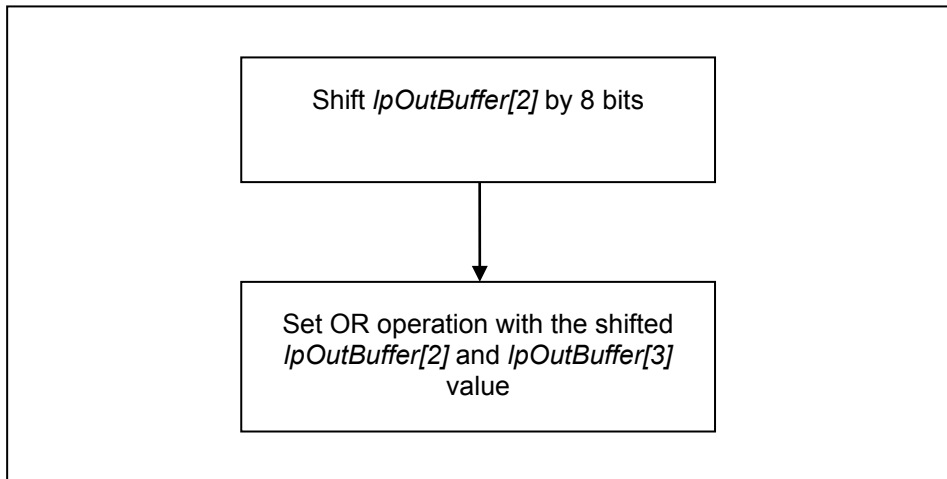
lpBytesReturned Pointer to a *DWORD* that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*

3.3.5. IOCTL_SMARTCARD_GET_FIRMWARE_VERSION

The `IOCTL_SMARTCARD_GET_FIRMWARE_VERSION` enables *Get Firmware Version* command.

3.3.5.1. Firmware Version

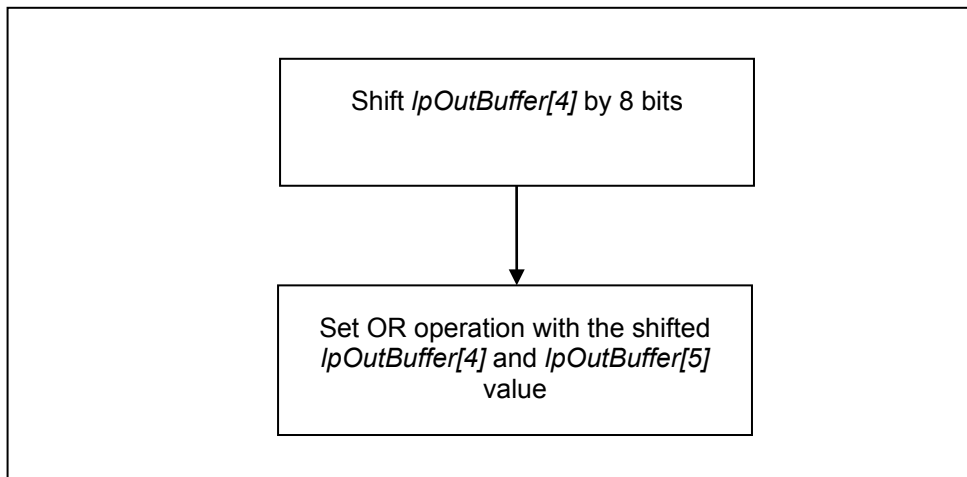
To acquire device firmware version, take the third element of the received buffer and shift it by 8 bits. Set the result to an OR operation together with the fourth element of the received buffer.



Example: `Firmware_Version = (Common.RecvBuff[2] << 8) | Common.RecvBuff[3];`

3.3.5.2. LCD

To acquire the device's LCD, take the fifth element of the received buffer and shift it by 8 bits. Set the result to an OR operation together with the sixth element of the received buffer.



Input:

hCard Reference value returned from `SCardConnect`

dwControlCode `IOCTL_SMARTCARD_GET_FIRMWARE_VERSION`



Output:

lpOutBuffer Output value of command
nOutBufferSize **sizeof(ULONG)** of *lpOutBuffer*
lpBytesReturned Pointer to a *DWORD* that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*

Offset	Field	Size	Value	Description
0	<i>abStatus</i>	2	0000h	SUCCESS
2	<i>wACR83Firmware</i>	2	-	-
4	<i>LCD</i>	2	-	-

3.3.6. IOCTL_SMARTCARD_DISPLAY_LCD_MESSAGE

The IOCTL_SMARTCARD_DISPLAY_LCD_MESSAGE enables display LCD message command.

hCard Reference value returned from *SCardConnect*
dwControlCode IOCTL_SMARTCARD_DISPLAY_LCD_MESSAGE
lpInBuffer Set value for Display LCD message option
nInBufferSize **sizeof(ULONG)** of *lpInBuffer*

Offset	Field	Size	Value	Description
0	<i>abLCDmessage</i>	0-32	-	LCD message (maximum 32 characters)

Output:

lpOutBuffer Output value of command
nOutBufferSize **sizeof(ULONG)** of *lpOutBuffer*
lpBytesReturned Pointer to a *DWORD* that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*

Offset	Field	Size	Value	Description
0	<i>abStatus</i>	2	0000h 0001h	SUCCESS BAD_PARAMETER

3.3.7. IOCTL_SMARTCARD_READ_KEY

The IOCTL_SMARTCARD_READ_KEY enables *Read Key* command.

Input:

hCard Reference value returned from *SCardConnect*



dwControlCode IOCTL_SMARTCARD_READ_KEY
lpInBuffer Set value for Display LCD message option
nInBufferSize sizeof(ULONG) of *lpInBuffer*

Offset	Field	Size	Value	Description
0	<i>bTimeOut</i>	1	-	Number of seconds. If value is equal to 00h, then the default value is used.
1	<i>wPINMaxExtraDigit</i>	2	XXYYh	XXh: Maximum PIN size in digit YYh: Minimum PIN size in digit
3	<i>bKeyReturnCondition</i>	1	-	The value is a bit wise OR operation. 01h: Maximum size reached 02h: Key [E] pressed 04h: Timeout occurred 08h: Key [C] pressed
4	<i>bEchoLCDStartPosition</i>	1	-	Starting position (0 – 31)
5	<i>bEchoLCDMode</i>	1	-	00h: Echo key press data ASCII representation to LCD 01h: Display all key presses as asterisks "*" on LCD

Output:

lpOutBuffer Output value of command
nOutBufferSize sizeof(ULONG) of *lpOutBuffer*
lpBytesReturned Pointer to a *DWORD* that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*.

Offset	Field	Size	Value	Description
0	<i>abStatus</i>	2	0000h 0001h	SUCCESS BAD_PARAMETER
2	<i>bKeyReturnCondition</i>	1	31h 32h 33h 34h	Maximum size reached Key [E] pressed Timeout occurred Key [C] pressed
3	<i>abNumericInputKeys</i>	0-32	-	-



Appendix A. Set *bKeyReturnCondition*

<i>bKeyReturnCondition</i>	OR Operand
if Maximum PIN size is reached	0x01h
if ACR83 device KEY_E is pressed	0x02
if ACR83 session TIMEOUT has reached	0x04h
if ACR83 device KEY_C is pressed	0x08h
if ACR83 device KEY_BACK is pressed	0x10h
if ACR83 device KEY_FN is pressed	0x20h

Note: Set value to an OR Operation again the specific OR Operand.



Appendix B. Response Error Codes

The following table summarizes the possible error code returned by the ACR83 (CCID).

Error Code	Status
0001h	BAD_PARAMETER
0083h	SLOTERROR_LCDCOMMANDERROR
0084h	SLOTERROR_WRONGCONFIRMPIN
0085h	SLOTERROR_UNKNOWN_LCD
0086h	SLOTERROR_MAXPIN_SIZE_EQUAL_ZERO
00EFh	SLOTERROR_PIN_CANCELLED
00F0h	SLOTERROR_PIN_TIMEOUT



Appendix C. *bmFormatString* Description

Bit Number	Description
Bit 7	The system units' type indicator: If 0h: the system units are bits If 1h: the system units are bytes This bit quantifies the next parameter (unit moving).
Bit 6 – 3	Define the PIN position after format in the APDU command (relative to the first data after <i>Lc</i>). The position is based on the system units' type indicator (maximum 1111 for 15 system units).
Bit 2	Bit mask for the PIN justification: If 0h: Left justify data If 1h: Right justify data
Bit 1-0	Bit wise for the PIN format type: 00h: binary 01h: BCD 10h: ASCII



Appendix D. *bmPINBlockString* Description

Bit Number	Description
Bit 7 - 4	Size in bits of the PIN length inserted in the APDU command. (If value is equal to 0h, then the effective PIN length is not inserted in the APDU command)
Bit 3 - 0	PIN length information: PIN block size in bytes after justification and formatting



Appendix E. *bmPINLength* Format

Bit Number	Description
Bit 7-5	RFU
Bit 4	The system units' type indicator: If 0h: the system units are bits If 1h: the system units are bytes
Bit 3 - 0	Indicate the PIN length position in the APDU command according to the previous parameters (maximum 1111 for 15 system units).