INTELLISTRIPE 65
COMMAND REFERENCE MANUAL

Manual Part Number 99875161 Rev 16

JULY 2010

MAGTEK®
REGISTERED TO ISO 9001:2008
1710 Apollo Court
Seal Beach, CA 90740
Phone: (562) 546-6400
FAX: (562) 546-6301
Technical Support: (651) 415-6800
www.magtek.com

REVISIONS

Rev Number	Date	Notes
1	24 Mar 00	Initial Release
2	15 May 00	Section 5: ATR Map Structure: to number 7 added 0x00, added number 46 and increased subsequent numbers. Notification Model Request: change Hex value from 00 to 02.
3	23 Oct 00	Section 5. Updated to current engineering software. Added Appendix B, Memory Cards.
4	13 Nov 00	Appendix B. Added second Memory Card to SLE 4442/4432; new card is SLE 4428/4418.
5	9 Aug 01	Section 5, Card Types: Changed card type property from 0x02 to 0x01. Under Properties, deleted card type references and replaced with “See Card Types, this section, for currently supported values…”
6	23 Oct 01	Section 5. Editorial Corrections throughout. Appendix B. Added support for the SLE4404 Memory Card and I^{2}C™ Cards families.
7	2 May 02	Section 4: Added Decode JIS Type 2 Property; Changed Decode Status Definitions in CMD ID 81 and 82, Added JIS to Card Encode Type in CMD 81 and 82. Appendix B: Completely revised SLE 4404 Memory Card Control.
8	24 Jun 02	Removed all references to Fraud Detection Command.
9	21 May 03	Front Matter: added ISO line to logo, changed Tech Support phone number, replaced 90-day warranty with generic software license agreement.
10	20 Aug 04	Added support for Gemplus GAM326 memory card in Appendix B.
11	23 Mar 06	Added Section 8, Buzzer Application; Sec 3: added reset time of 5 seconds to SoftwareReset Command; Sec 4, Added MagnePrint,
12	10 Apr 07	Add Host Application property numbers 3 thru 8. Added OTI SCI 1000 Contact-less Smartcard Application.
13	Feb 08	Add description on sending commands to contactless module; Add Mifare support
14	10 Feb 09	Added description of Warm Reset command for Smart Cards. The Warm Reset feature has existed for a long time but it had not been documented in the manual.
15	13 Jul 09	Added support for Gemplus GAM375 memory card in Appendix B.
16	23 July 10	Added data security option.

SOFTWARE LICENSE AGREEMENT

IMPORTANT: YOU SHOULD CAREFULLY READ ALL THE TERMS, CONDITIONS AND RESTRICTIONS OF THIS LICENSE AGREEMENT BEFORE INSTALLING THE SOFTWARE PACKAGE. YOUR INSTALLATION OF THE SOFTWARE PACKAGE PRESUMES YOUR ACCEPTANCE OF THE TERMS, CONDITIONS, AND RESTRICTIONS CONTAINED IN THIS AGREEMENT. IF YOU DO NOT AGREE WITH THESE TERMS, CONDITIONS, AND RESTRICTIONS, PROMPTLY RETURN THE SOFTWARE PACKAGE AND ASSOCIATED DOCUMENTATION TO THE ABOVE ADDRESS, ATTENTION: CUSTOMER SUPPORT.

TERMS, CONDITIONS, AND RESTRICTIONS
MagTek, Incorporated (the "Licensor") owns and has the right to distribute the described software and documentation, collectively referred to as the "Software".

LICENSE: Licensor grants you (the "Licensee") the right to use the Software in conjunction with MagTek products. LICENSEE MAY NOT COPY, MODIFY, OR TRANSFER THE SOFTWARE IN WHOLE OR IN PART EXCEPT AS EXPRESSLY PROVIDED IN THIS AGREEMENT. Licensee may not decompile, disassemble, or in any other manner attempt to reverse engineer the Software. Licensee shall not tamper with, bypass, or alter any security features of the software or attempt to do so.

TRANSFER: Licensee may not transfer the Software or license to the Software to another party without the prior written authorization of the Licensor. If Licensee transfers the Software without authorization, all rights granted under this Agreement are automatically terminated.

COPYRIGHT: The Software is copyrighted. Licensee may not copy the Software except for archival purposes or to load for execution purposes. All other copies of the Software are in violation of this Agreement.

TERM: This Agreement is in effect as long as Licensee continues the use of the Software. The Licensor also reserves the right to terminate this Agreement if Licensee fails to comply with any of the terms, conditions, or restrictions contained herein. Should Licensor terminate this Agreement due to Licensee's failure to comply, Licensee agrees to return the Software to Licensor. Receipt of returned Software by the Licensor shall mark the termination.

LIMITED WARRANTY: Licensor warrants to the Licensee that the disk(s) or other media on which the Software is recorded are free from defects in material or workmanship under normal use.

THE SOFTWARE IS PROVIDED AS IS. LICENSOR MAKES NO OTHER WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Because of the diversity of conditions and PC hardware under which the Software may be used, Licensor does not warrant that the Software will meet Licensee specifications or that the operation of the Software will be uninterrupted or free of errors.

IN NO EVENT WILL LICENSOR BE LIABLE FOR ANY DAMAGES, INCLUDING ANY LOST PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE, OR INABILITY TO USE, THE SOFTWARE. Licensee's sole remedy in the event of a defect in material or workmanship is expressly limited to replacement of the Software disk(s) if applicable.

GOVERNING LAW: If any provision of this Agreement is found to be unlawful, void, or unenforceable, that provision shall be removed from consideration under this Agreement and will not affect the enforceability of any of the remaining provisions. This Agreement shall be governed by the laws of the State of California and shall inure to the benefit of MagTek, Incorporated, its successors or assigns.

ACKNOWLEDGMENT: LICENSEE ACKNOWLEDGES THAT HE HAS READ THIS AGREEMENT, UNDERSTANDS ALL OF ITS TERMS, CONDITIONS, AND RESTRICTIONS, AND AGREES TO BE BOUND BY THEM. LICENSEE ALSO AGREES THAT THIS AGREEMENT SUPERSEDES ANY AND ALL VERBAL AND WRITTEN COMMUNICATIONS BETWEEN LICENSOR AND LICENSEE OR THEIR ASSIGNS RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.

QUESTIONS REGARDING THIS AGREEMENT SHOULD BE ADDRESSED IN WRITING TO MAGTEK, INCORPORATED, ATTENTION: CUSTOMER SUPPORT, AT THE ABOVE ADDRESS, OR E-MAILED TO support@magtek.com.

TABLE OF CONTENTS

SECTION 1. APPLICATION MESSAGES

This section describes the format of application messages and defines the protocol for using these messages.

OVERVIEW

Application messages are the blocks of information exchanged between two applications. They consist of a header and data. The message format and contents are independent of the transport mechanism by which the messages are exchanged. Each message has a length that is provided by the transport mechanism for messages being received or is supplied to the transport mechanism for messages being transmitted. The transport mechanism is outside the scope of this document.

DATA SECURITY

Some devices support an optional mechanism for securing message data using encryption/decryption. The message elements involved in data security are the encryption flag and pad length bits of the message type field and the message data field. Data security is controlled with the system data encryption command and the key injection command in the device application.

There is no special data decryption command. When the host sends a application message command request to the device, if the message header's message type byte's encryption flag bit is set, the device will regard the following data as encrypted, the device will decrypt the data automatically and send it to the correct application.

MESSAGE FORMAT

The following diagram illustrates the message format:

MTYP	APPL	CMND	RC	DATA

Message Header

The message header contains four one-byte fields: Message Type, Application ID, Command ID and Result Code. The header is followed by zero or more bytes of data. The existence and format of the data depends on the Application ID and Command ID. The following sections describe each field of the header.

Message Type

This field specifies the message type: request, response or notification. The table below defines the encoding of the message type:

Bits	Value	Definition
7-6	00	Request message type.
	01	Response message type.
	10	Notification message type.
	11	Reserved for future use.
5-4	0	Reserved for future use.
3	0-1	Encryption flag. Zero when data field is not encrypted. One when data field is encrypted. When encrypted, the data field must be padded at the end so that its length is a multiple of 8 bytes.
2-0	0-7	Pad length. The number of pad bytes at the end of the data field when the data field is encrypted. If the data field is not encrypted, the pad length is zero.

Requests are the messages sent by a host application to a device application. The device performs the requested operation and sends a response message to the host application. The device can service only one request at a time. If a request is pending, no further requests should be sent to the device until a response is received.
Responses are the messages sent as a reply to a previously sent request. The response contains the result of the requested operation. The device application must send each response within a pre-determined finite amount of time from receiving the request.

Notifications are sent by a device application when it needs to notify the host application that the device's state has changed or that some external event has occurred (i.e., a magnetic card has been inserted). The device can send a notification at any time. The device does not expect a response or any specific action from the host application.

For device operations that take long or indefinite amount of time, the host application usually sends a request that initiates the operation. The device sends a response, indicating it has started the operation. When the operation completes, the device sends a notification message to the host application.

Application ID

This field specifies the application ID. The application ID identifies the device application that a message corresponds to. A device application is a functional subsystem in the device. For example, a device might contain a magnetic stripe, transport, and a smart card functional subsystem. Each application in a device has a unique application ID and a defined command set.

The following application ID values are defined:

App ID (Hex)	Definition
00-7F	This range is for generic applications. A generic application has a common command set for different device models.
80-FF	This range is for custom applications. A custom application has a unique command set for a particular device model.

Command ID

This field specifies the command ID. The Command ID has a different meaning for each of the three message types:

- For requests, the command ID defines the operation to be carried out by the device application.
- For responses, the command ID defines the operation that was carried out by the device application (always the same value as the Command ID from the request).
- For notifications, the command ID specifies the event that has occurred in the device application.

The following command ID values are defined:

Value (Hex)	Definition
00-7F	This range is for generic commands. A generic command has a common meaning for different device applications. The existence of generic commands allows standardizing on commonly used commands across multiple device applications. Device applications are not required to support all generic commands. Generic commands are defined further elsewhere in this document.
80-FF	This range is for custom commands. A custom command has a unique meaning for a particular device application. Custom commands are defined further elsewhere in this document.

Result Code

This field specifies the result code. The result code has different meaning for each of the three message types:

- For requests, the result code is currently undefined and should be set to 0.
- For responses, the result code defines the result of the operation that was carried out by the device application.
- For notifications, the result code specifies the result of the event that has occurred in the device application.

The following result codes are defined:

Value (Hex)	Definition
00-7F	This range is for generic result codes. A generic result code has a common meaning for different device applications. The following values are currently defined: 0 – success 1 – failure 2 – warning 3 – bad message header 4 – bad application id 5 – bad command id 6 – bad parameter 7 – timeout 8 – busy
80-FF	This range is for custom result codes. A custom result code has a unique meaning for a particular device application.

Data Field

If there is additional data associated with the application message, it is contained in this field. The length of this field is equal to the length of the message minus the length of the message header.

When encrypted, the data field must be padded at the end so that its length is a multiple of 8 bytes. Any pad character can be used for padding because the padding is discarded after decryption. The data field is encrypted using triple DES (TDEA) and the cipher-block chaining (CBC) mode with an initialization vector (IV) set to all zeros. Triple DES is used for 16-byte keys. Single DES is used for 8-byte keys.

SECTION 2. GENERIC COMMANDS

This section defines the commands that are generic to all device applications. Not all device applications support these commands.

GET PROPERTY COMMAND

Command ID: 00 (Hex)
Description:
This command is used to retrieve the value of an application property.

Request Message Data:

Byte	Field	Description
1	Property Type	Property Type definition follows
2	Property ID	Property ID identifies the property

Property Type definition:
The bits are identified by numbering the least significant bit 0 and the most significant bit 7.

Bits	Value	Definition
7-4	0	Reserved for future use.
3-0	0	Property type – None. The property type is unspecified.
	1	Property type Dword – 32-bit integer.
	2	Property type String – zero-terminated ASCII string.
	3	Property type Boolean – 8-bit integer (1 – TRUE, 0 – FALSE)
	4	Property type Binary – binary data.
	5–15	Property types reserved for future use.

If a property type other than NONE is used, the type will be checked with the type of the property ID being requested. If these types do not match then the command will fail.

Response Message Data:

Byte	Field	Description
1	Property Type	Property Type definition follows
2	Property ID	Property ID identifies the property
3	Property Value	Value of the property

The property type field is the same as defined in the request message data. Only valid property types are returned. The property type None is never returned. If the property type is Dword, the format of the property value is in the Intel LSB,MSB (Little Endian) layout. For example Dword value 0x12345678 is formatted as 0x78 0x56 0x34 0x12. Four bytes are always returned.

Response Message Result Codes:

Success = 00 (Hex)
The command completed successfully.

Failure = 01 (Hex)
The command failed.

SET PROPERTY COMMAND

Command ID: 01 (Hex)
Description:
This command is used to modify the value of an application property.

Request Message Data:

Byte	Field	Description
1	Property Type	Property Type definition follows
2	Property ID	Property ID identifies the property
3	Property Value	Value of the property

Property Type definition:
The bits are identified by numbering the least significant bit 0 and the most significant bit 7.

Bits	Value	Definition
7-4	0	Reserved for future use.
3-0	0	Reserved for Future use.
	1	Property type Dword – 32-bit integer.
	2	Property type String – zero-terminated ASCII string.
	3	Property type Boolean – 8-bit integer (1 – TRUE, 0 – FALSE)
	4	Property type Binary – binary data.
	5–15	Property types reserved for future use.

The property type None, defined in the get property command, is not allowed. Valid property types are required or the command will fail. If the property type is Dword, the format of the property value is in the Intel LSB,MSB (Little Endian) layout. For example Dword value 0x12345678 is formatted as 0x78 0x56 0x34 0x12. All four bytes are required.

Response Message Data: None

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

Failure = 01 (Hex)

The command failed.

SAVE PROPERTY COMMAND

Command ID: 02 (Hex)

Description:

This command is used to save the current property value as the power up / reset default value. This command can be used to modify the property default values so that the properties do not have to be initialized to other values after a power cycle or reset. Typically the set property command is sent first to set the property to the desired power up / reset value. After which, the save property command is sent to save the property in non-volatile memory so that it can then be used as the new default value after a power cycle or reset. Not all properties support this command. See the description of each property to see if this command is supported or not.

Request Message Data:

Byte	Field	Description
1	Property Type	Property Type definition follows
2	Property ID	Property ID identifies the property
3	Security Code 1	55 (hex)
4	Security Code 2	AA (hex)

Property Type definition:

The bits are identified by numbering the least significant bit 0 and the most significant bit 7.

Bits	Value	Definition
7-4	0	Reserved for future use.
3-0	0	Property type – None. The property type is unspecified.
	1	Property type Dword – 32-bit integer.
	2	Property type String – zero-terminated ASCII string.
	3	Property type Boolean – 8-bit integer (1 – TRUE, 0 – FALSE)
	4	Property type Binary – binary data.
	5–15	Property types reserved for future use.

Response Message Data: None

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

Failure = 01 (Hex)

The command failed.

SECTION 3. DEVICE APPLICATION

Application ID: 00 (Hex)

Description:
This application deals with device functionality.

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (Hex)
Save Property Command ID: 02 (Hex)
Description:
These commands are used to get, set and save the value of application properties. Details of these commands are described in the Generic command section of this document. The following describes each property this application supports.

MODEL NUMBER PROPERTY

Property ID: 00 (Hex)
Property Type: String
Maximum Length: 33 (including terminating zero)
Power Up/Reset Value: "IntelliStripe 65"
Get/Set/Save support: Get
Description:
This property is used to get the devices model number.

Values:

The value is fixed at "IntelliStripe 65".

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	00	00	00	02	00

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	00	00	00	02	00

Field	PVAL
Byte	7 – 23
Value (Hex)	49 6e 74 65 6c 6c 69 53 74 72 69 70 65 20 36 35 00 "IntelliStripe 65" (ASCII)

SOFTWARE ID PROPERTY

Property ID: 01 (Hex)
Property Type: String
Maximum Length: 33 (including terminating zero)
Power Up/Reset Value: Software dependent
Get/Set/Save support: Get
Description:
This property is used to get the devices software ID.

Values:
The value is fixed and is software dependent. For example, the software ID could be "21165812A06" where "21165812" is the software part number, "A" is the software revision and "06" is the software version.

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	00	00	00	02	01

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID	PVAL
Byte	1	2	3	4	5	6	7 - 18
Value (Hex)	40	00	00	00	02	01	32 31 31 36 35 38 31 32 41 30 36 00 "21165812A06" (ASCII)

SOFTWARE RESET COMMAND

Command ID: 80 (Hex)
Description:
This command is used to reset the device. This will put the device in its power up state. The device should be allowed time to power up before trying to communicate with the device. The maximum time it would take the device to reset would be 5 seconds.

Request Message Data: None
Response Message Data: None
Response Message Result Codes:
Success = 00 (Hex)

The command completed successfully.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	00	80	00

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	00	80	00

SYSTEM DATA ENCRYPTION COMMAND

Application ID: 00(Hex)
Command ID: 81 (Hex)
Data Length: 1
Default Value: 0 (No Encryption)
Data Value:
0: No Encryption
1: Selected Encrypt
2: Total Encrypt

The system data encryption command controls which application messages are encrypted before being sent to the host. The system data encryption value is non-volatile so it persists even when the device is power cycled or reset. Care must be taken before changing this value because once changed, it can never be changed back to a lower value. It can only be changed to a higher value.

If “No Encryption” is set, no application messages are encrypted before being sent to the host.

If “Selected Encrypt” is set, only certain application messages, that typically contain sensitive data, are encrypted before being sent to the host. The following application messages are encrypted. All magnetic stripe application command responses and notification messages that contain card data or MagnePrint data. All smart card application messages including command responses, get property responses and notification messages. The OTI module contactless smart card application contactless magstripe stripe notification message and OTI module transparent command response.

If “Total Encrypt” is set, all application messages are encrypted before being sent to the host.

This command will fail in following situations.

- If there is no key in the device, when you set the data value to 1 or 2, the device will send back BAD COMMAND response. That means, before running this command to set “Selected” or “TotalEnc”, there must be a key in the flash space, otherwise, you must run KEY INJECTION COMMAND first.
- If the data field length is not 1, the device will send back BAD PARAMETER response.
- If the data value is bigger than 2, the device will send back BAD PARAMETER response.
- If the data value is less than current system data encryption setting; the device will send back BAD PARAMETER response. That means, the system data encryption setting is one direction setting, it can only be changed from 0->1, 0->2 or 1->2.

Example Request :

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5
Value (Hex)	00	00	81	00	02

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	00	81	00

KEY INJECTION COMMAND

Application ID: 00(Hex)
Command ID: 82 (Hex)
Description:
This command is used to inject the key that is used to optionally encrypt/decrypt application messages. This key is non-volatile so it persists even when the device is power cycled or reset.

If the reader is equipped with an encrypting IntelliHead, then this command also initiates key paring between the reader and the head, and enables card data encryption between the two. The encryption between the reader and the head will persist even if the device is power cycled or reset. No encryption is performed between the reader and the head until this key injection command issued.

This command has two parts and the key is not usable until the second part is executed.

- The first part loads one of the components of the key, the second part loads the other.
- The first part must be entered first, the second part must be entered within two minutes of the first part.
- There must be no loss of power to the device between the entry of the first and second parts.
- Both parts must be the same length (either 8 bytes or 16 bytes) or an error is reported.
- The two parts are combined by XORing in the unit to create the final key.
- On receipt of the correctly formatted first part, all MSK structures (as described above) are cleared, and the new first part is stored in secure memory in anticipation of receipt of the second part.
- On receipt of the second part, the parts are combined, and the key is activated, returning to normal operations.

Request Message Data:

Byte	Usage
1	Key Part, 1 = first part, 2 = second part
2	RFU – Leave zero
3-10 or 3-18	Key value for component of new Master Key. It may be 8 bytes (single DES) or 16 bytes (triple DES).

Response Message Data: None

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

Bad Parameter = 06 (Hex)

The Key Part is not equal to 1 or 2

OR

The given key is not either 8 bytes or 16 bytes long.

Attempt to load second part with different key length than first part.

Attempt to load second part without previously loading first part (could also be due to lapse of more than two minutes or loss of power between loading parts).

Example Request: Load Key 0123456789ABCDEFFEDCBA9876543210

First Part = `0F0F0F0F0F0F0F0F 0F0F0F0F0F0F0F0F`

Second Part = `0E2C4A6886A4C2E0 F1D3B597795B3D1F`

Example Request (First Part):

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-22
Value (Hex)	00	00	82	00	0100 0F0F0F0F0F0F0F0F 0F0F0F0F0F0F0F0F

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	00	82	00

Example Request (Second Part):

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-22
Value (Hex)	00	00	82	00	0200 0E2C4A6886A4C2E0 F1D3B597795B3D1F

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	00	82	00

SECTION 4. MAGNETIC STRIPE APPLICATION

Application ID: 01 (Hex)

Description:
This application deals with decoding and presenting magnetic stripe card data. This application works closely with the transport application, which deals with card transportation and magnetic stripe data acquisition. The transport application must be used to acquire magnetic stripe data before any data is available to decode and present by the magnetic stripe application.

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)

Set Property Command ID: 01 (Hex)

Save Property Command ID: 02 (Hex)

Description:
These commands are used to get, set and save the value of application properties. Details of these commands are described in the Generic command section of this document. The following describes each property this application supports.

NOTIFY READ STATE PROPERTY

Property ID: 00 (Hex)
Property Type: Dword
Power Up/Reset Value: 0 (OFF)
Get/Set/Save support: All
Description:
If the notify read state is not set to OFF, then a notification message will be sent to the host when a card is read. The notification message will have the same syntax as the command response of the command that corresponds to the Notify Read State. The only difference is that the message header will contain a notification message type instead of a response message type. Note that when the Notify Read State is set to GET TRACK DECODE DATA, the track number is obtained from the Notify Read Track Property.

Values:

Value	Notify Read State
0	OFF
1	GET TRACK 123 DECODE DATA
2	GET TRACK DECODE DATA

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID	PVAL
Byte	1	2	3	4	5	6	7 - 10
Value (Hex)	00	01	01	00	01	00	02 00 00 00 (GET TRACK DECODE DATA)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	01	01	00

NOTIFY READ TRACK PROPERTY

Property ID: 01 (Hex)
Property Type: Dword
Power Up/Reset Value: 2 (Track 2)
Get/Set/Save support: All
Description:
This property contains the track number that is used when the Notify Read State property is set to the GET TRACK DECODE DATA state. This is the track that is sent in the notification message when a card is read.

Values:

Value	Notify Read Track
1	TRACK 1
2	TRACK 2
3	TRACK 3

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID	PVAL
Byte	1	2	3	4	5	6	7 - 10
Value (Hex)	00	01	01	00	01	01	03 00 00 00 (TRACK 3)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	01	01	00

DECODE JIS TYPE 2 PROPERTY

Property ID: 15 (Hex)
Property Type: Boolean
Power Up/Reset Value: 0 (False)
Get/Set/Save support: All
Description:
If this property is set to true then cards encoded to JIS X 6302 type 2 standards can be decoded.

Values:

Value	Decode JIS Type 2
0	False
1	True

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID	PVAL
Byte	1	2	3	4	5	6	7
Value (Hex)	00	01	01	00	03	15	01 (True)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	01	01	00

CLEAR DATA COMMAND

Command ID: 80 (Hex)
Description:
This command is used to clear all magnetic stripe data so that the data can no longer be acquired from the device. After this command is issued, no magnetic stripe data will be available until a card is read.

After a card is read, the magnetic stripe data is held in the device until:

1. Another card read occurs.
2. A Clear Data command is issued.
3. The device is power cycled or reset.

Request Message Data: None

Response Message Data: None

Response Message Result Codes:

Success = 00 (Hex)
The command completed successfully.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	01	80	00

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	01	80	00

GET TRACK 123 DECODE DATA COMMAND

Command ID: 81 (Hex)
Description:
This command is used to get decoded information related to a magnetic stripe read for tracks 1, 2 and 3. This command always returns data from the most current magnetic stripe data acquisition. This command is used to get decode status, card encode type, 3 track data lengths and decoded card data for 3 tracks. Each track of decoded data starts with a start sentinel and ends with an end sentinel and is converted to ASCII format. The ASCII representation of the start and end sentinels vary depending on the encode format.

Request Message Data: None

Response Message Data:

Byte	Field	Description
1	Decode Status	Decode status definition follows
2	Card Encode Type	Card Encode Type definition follows
3	Track 1 Data Length	Length of track 1 data
4	Track 2 Data Length	Length of track 2 data
5	Track 3 Data Length	Length of track 3 data
6 - ?	Track 1, 2, 3 Data	Track 1 followed by track 2 then track 3 decoded data

Decode Status definition:
The bits are identified by numbering the least significant bit 0 and the most significant bit 7.

Bit	Field	Description
0	Track 1 error	Indicates track 1 decode error if set to 1
1	Track 2 error	Indicates track 2 decode error if set to 1
2	Track 3 error	Indicates track 3 decode error if set to 1
3-7	RFU	Reserved for future use

A decode error is only indicated if a valid start sentinel exists on the track and the track cannot be decoded. If a track cannot be decoded and if it does not have a decode error it is considered blank.

Card Encode Type definition:

Value	Encode Type	Description
0	ISO/ABA	ISO/ABA encode format
1	AAMVA	AAMVA encode format
2	CADL	CADL encode format. Note that some versions of this reader can only read track 2 for this format. They can not read tracks 1 and 3. However, this format is obsolete. There should no longer be any cards in circulation that use this format. California is now using the AAMVA format.
3	Blank	The card is blank
4	Other	The card has a non-standard encode format. For example, ISO/ABA track 1 format on track 2.
5	Undetermined	The card encode type could not be determined because no tracks could be decoded.
6	None	No decode has occurred. This type occurs if no magnetic stripe data has been acquired since the data has been cleared or since the device was powered on.
7	JIS Type 2	JIS X 6302 Type 2 encode format

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	01	81	00

Example Response:

Field	MTYP	APPL	CMND	RC	DCD STAT	ENCD TYP
Byte	1	2	3	4	5	6
Value (Hex)	40	01	81	00	00	00

Field	TK1 LEN	TK2 LEN	TK3 LEN	TK123 DATA
Byte	7	8	9	10 - 153
Value (Hex)	37	22	37	Track1 Track2 Track3

GET TRACK DECODE DATA COMMAND

Command ID: 82 (Hex)

Description:

This command is used to get decoded information related to a magnetic stripe read for a single track. This command always returns data from the most current magnetic stripe data acquisition. This command is used to get decode status, card encode type and decoded card data for a given track. The track of decoded data starts with a start sentinel and ends with an end sentinel and is converted to ASCII format. The ASCII representation of the start and end sentinels vary depending on the encode format.

Request Message Data:

Byte	Field	Description
1	Track Number	Number of desired track. This value should be 1, 2 or 3.

Response Message Data:

Byte	Field	Description
1	Track Number	Number of retrieved track. Always the same as requested track.
2	Decode Status	Decode status definition follows
3	Card Encode Type	Card Encode Type definition follows
4 - ?	Track Data	Track decoded data

Decode Status definition:

The bits are identified by numbering the least significant bit 0 and the most significant bit 7.

Bit	Field	Description
0	Track 1 error	Indicates track 1 decode error if set to 1
1	Track 2 error	Indicates track 2 decode error if set to 1
2	Track 3 error	Indicates track 3 decode error if set to 1
3-7	RFU	Reserved for future use

A decode error is only indicated if a valid start sentinel exists on the track and the track cannot be decoded. If a track cannot be decoded and if it does not have a decode error it is considered blank.

Card Encode Type definition:

Value	Encode Type	Description
0	ISO/ABA	ISO/ABA encode format
1	AAMVA	AAMVA encode format
2	CADL	CADL encode format. Note that some versions of this reader can only read track 2 for this format. They can not read tracks 1 and 3. However, this format is obsolete. There should no longer be any cards in circulation that use this format. California is now using the AAMVA format.
3	Blank	The card is blank
4	Other	The card has a non-standard encode format. For example, ISO/ABA track 1 format on track 2.
5	Undetermined	The card encode type could not be determined because no tracks could be decoded.
6	None	No decode has occurred. This type occurs if no magnetic stripe data has been acquired since the data has been cleared or since the device was powered on.
7	JIS Type 2	JIS X 6302 Type 2 encode format

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

Bad Parameter = 06 (Hex)

The command failed due to a bad parameter in the request message data field

Example Request:

Field	MTYP	APPL	CMND	RC	TRACK
Byte	1	2	3	4	5
Value (Hex)	00	01	82	00	02

Example Response:

Field	MTYP	APPL	CMND	RC	TRACK	DCD STAT
Byte	1	2	3	4	5	6
Value (Hex)	40	01	82	00	02	00

Field	ENCD TYP	TK DATA
Byte	7	8 - 41
Value (Hex)	00	Track2

GET MAGNEPRINT DATA COMMAND

Command ID: 86 (Hex)

Description:

This command is used to get MagnePrint™ information for the most recently read card.

Note 1: This command available only on units with MagnePrint feature.

Note2: MagnePrint™ data obtained from a read-on-insert swipe (see MSR Direction Property in the Transport section) will not correlate as well as data obtained from a read on withdrawal. We recommend setting the MSR Direction Property to withdrawal when using the MagnePrint™ data.

Request Message Data: None

Response Message Data:

Byte	Field	Description
1-4	MagnePrint status	MagnePrint status. Byte 1 LSB is status bit 0. Byte 4 MSB is status bit 31 MagnePrint status is defined as follows: Bit 0 = This is a MagnePrint-capable product Bits 1 through 15 = Product revision & mode Bit 16 = STATUS-only state Bit 17 = Noise too high or “move me” away from the noise source (used only in STATUS-only) Bit 18 = Swipe too slow Bit 19 = Swipe too fast Bit 20 = Unassigned Bit 21 = Actual Card Swipe Direction (0 = Forward, 1 = Reverse) Bits 22 thru 31 = Unassigned (always set to Zero) Note that bits 16 and 17 will never be set because they would only be set if noise was too high during the swipe and in that case this command will fail setting the result code to 86 (Hex) to indicate the noisy condition. In this case, no response message data will be available.
5-58	MagnePrint™ data	MagnePrint™ data

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

MagnePrint™ Not Installed = 84 (Hex)

The MagnePrint™ feature is not installed on this device. Note that some versions of this reader will return result code 05 (Hex) if MagnePrint is not installed. Result code 05 (Hex) indicates that this command is not available.

Data cleared = 85 (Hex)

MagnePrint™ data was cleared by a Clear Read Data command.

Excess noise = 86 (Hex)

The MagnePrint™ circuit is sensing excessive electrical noise in the environment, no MagnePrint™ data is available. Moving the reader away from noise sources (monitors, speakers, etc.) may solve this problem. Note that this result indicates that the noisy condition was present at the time the card was swiped. It is not a real time indication that noise is present. For a real time indication that noise is present see the Indicators property in the Transport application.

Data not available = 87 (Hex)

There is no MagnePrint™ data available for this card format. This could occur when swiping a blank card.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	01	86	00

Example Response:

Field	MTYP	APPL	CMND	RC	STATUS	DATA
Byte	1	2	3	4	5-8	9-62
Value (Hex)	40	01	86	00	MagnePrint status	MagnePrint™ data

GET TRACK BINARY DATA COMMAND

Command ID: FF (Hex)

Description:

This command is used to get Binary information related to a magnetic stripe read for a single track. This command always returns data from the most current magnetic stripe data acquisition. This command is used to get track number, data length, and binary card data for a given track. The track of binary data is unprocessed and undecoded. The first bit read from the card goes in the least significant bit of the first byte, continuing on until that byte is full, then proceeding in the same manner to other bytes. The last byte will usually contain less than 8 valid bits; the valid bits start at the least significant bit. Readers that use MagTek's Delta, 3V or MagnePrint ASIC typically pad the binary data with two bytes of zeros at the beginning and at the end. This is because these ASICs don't start collecting data until a one bit is found and some decode routines require leading zeros prior to the first one bit.

Request Message Data:

Byte	Field	Description
1	Track Number	Number of desired track. This value should be 1,2 or 3.

Response Message Data:

Byte	Field	Description
1	Track Number	Number of retrieved track. Always the same as requested track.
2	Length in bytes	Number of bytes in Track Data
3	Bits in last byte	Number of valid bits in the last byte of data
4 - ?	Track Data	Track binary data

Response Message Result Codes:

Success = 00 (Hex), the command completed successfully.

Bad Parameter = 06 (Hex), the command failed due to a bad parameter in the request message data field

Example Request:

Field	MTYP	APPL	CMND	RC	TRACK
Byte	1	2	3	4	5
Value (Hex)	00	01	FF	00	02

Example Response:

Field	MTYP	APPL	CMND	RC	TRACK	Length (bytes)	Length (bits)	TK Data
Byte	1	2	3	4	5	6	7	8-41
Value (Hex)	40	01	FF	00	02	22	3	…data…

SECTION 5. SMART CARD APPLICATION

Application ID: 02 (Hex)
Description:
This application deals with smart card communications.

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (Hex)
Save Property Command ID: 02 (Hex)
Description:
These commands are used to get, set and save the value of application properties. Details of these commands are described in the Generic command section of this document. The following describes each property this application supports.

RESPONSE CODES USED THROUGHOUT THE APPLICATION

Two Response Codes are used throughout most of the application and have the same meaning in all cases.

> 80 (Hex) – This Response Code is used in conjunction with the Request/Response/Notification Model for all commands. It means command has not yet completed and that a Notification message will be sent later when the command has completed.
>
> 08 (Hex) – This Response Code may be used with any command and indicates either:
> - The Smart Card Application can not find hardware to execute on, or
> - The Smart Card Application is busy executing a previous command.

NOTIFICATIONS IN THE SMART CARD APPLICATION

The Smart Card Application attempts to respond to all requests in a timely manner. Because different Smart Cards may take longer than others to respond to a given request, it is not possible to predict how long it will take to complete a request. Because of this, we have adopted a Request/Response/Notification model for most commands to the Smart Card Application. The Get and Set Property commands are excluded from this model.

Request/Response/Notification Model

When the Smart Card Application receives a Request, it initiates the requested action. If the action completes in a timely manner, a Response message is sent informing the Host Application of completion. If the request does not complete in a timely manner, a Response message is sent indicating that the command has not yet completed (Response Code 80 (Hex)) and that a

Notification message will be sent later when the command has completed. The Notification message is identical in every way except for its Command Type to the Response message that would have been sent if the request had completed in a timely manner.

Example Request/Response/Notification Sequence

Request message (Power Up Command):

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	02	80	80

Response message (RC = 80 means Request Initiated, Notification to follow):

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	02	80	80

Notification message (Card Powered Up OK, Data is ATR):

Field	MTYP	APPL	CMND	RC	PVAL
Byte	1	2	3	4	5-8
Value (Hex)	80	02	80	00	3B 60 00 0A

THE CONDRPT PROPERTY

This property is used to give a detailed status of the Power Up Command, the TPDU, Card To IFD Command, the TPDU, IFD To Card Command, and the APDU Exchange Command. It consists of the following fields:

Bytes	Meaning
0	Ignore
1	Primary Status 0x80 = No Errors 0x01 = Parameter Errors, details follow in Secondary Status 0x01 = Header Block less than required length 0x04 = Length of Data not consistent with Lc 0x06 = Requested Data larger than available buffer space 0x82 = Card Handling Errors, details follow in Secondary Status 0x01 = Card Not In Connector 0x02 = Card Not Powered 0x05 = Card Not Responding (I^2C™ cards only) 0x03 = Abnormal Condition Report, details follow in Accumulator
2	Secondary Status, as defined above.
3 – 6	Accumulator is a bit map field documenting the abnormal conditions encountered during the process. The meaning of the individual bits depends on the command that produced the accumulator contents, see the definitions in the corresponding command descriptions.
7 – 10	Error Conditions Template – the actual Error Conditions Template used, in conjunction with the Warning Conditions Template, to formulate the Accumulator contents.
11-14	Warning Conditions Template – the actual Warnings Conditions Template used, in conjunction with the Error Conditions Template, to formulate the Accumulator contents.

ERROR/WARNING CONDITION TEMPLATES

In dealing with Smart Cards, there are numerous situations that can arise in operation that might be considered abnormal. We define three basic manners of handling these situations:

> Errors – These are situations that are considered disastrous. When they are encountered, the card is deactivated immediately and the Result Code Failure (01 Hex) is returned.
>
> Warnings – These are situations that are considered noteworthy, but may not require the card to be deactivated. The operation requested was able to complete. When these situations are encountered the Result Code Warning (02 Hex) is returned.
>
> Ignore – These are situations that are considered of no importance. The operation requested was able to complete. When these situations are encountered the Result Code Success (00 Hex) is returned.

Of course the operation requested may complete without encountering any abnormal situations at all. When this happens the Result Code Success (00 Hex) is returned.

Criteria for treating abnormal situations may change from application to application. One way of dealing with changing criteria is via Error/Warning Condition Templates. Each template is 4 bytes long. Each bit can be used to define a particular condition. If the bit is set in an Error Conditions Template and the condition is encountered, it is considered an Error and the card is deactivated with the Result Code Failure (01 Hex) being returned. Bits set in the Error Conditions Template override bits set in a corresponding Warning Conditions Template. If the bit is set in the Warning Conditions Template (and not in the corresponding Error Conditions Template) and the condition is encountered, it is considered a Warning and the operation is allowed to proceed, but a Result Code of Warning (02 Hex) will be returned when the operation completes. If the bit is not set in either the Error or Warning Conditions Template and the condition is encountered, the condition is Ignored, the operation is allowed to proceed, and a Response Code of Success (00 Hex) is returned.

As conditions are encountered, they are recorded in an Accumulator bit map. Each bit in the Accumulator bit map corresponds directly to the bits in the Error and Warning Conditions Template. The Accumulator bit map does not distinguish between Errors, Warnings, and Ignored conditions; it simply records the occurrence of the conditions. The Accumulator is part of the CondRpt property.

Different operations use different Error/Warning Conditions Templates.

CARD TYPES

Multiple card types are supported. The default card type (after power up) is 0x00, which supports microprocessor cards (T=0/T=1). The card type may be selected by the host application by setting the desired type in Property 01 (Card Type). The specified card type is used on all connectors until a new card type is set via the Set Property command.

There are two major categories of card types. The first are ISO 7816-3(1997) Microprocessor cards. Most Microprocessor cards available today comply with 7816-3 and this manual does not address their functionality in depth. The second category is Memory or Synchronous cards. There are numerous Memory cards available in the market. Some of them conform to ISO 7816-10, some don’t. There are no standards for command sets used to control these cards. The protocols and commands used to control these cards vary widely. In an effort to promote ease of use, we attempt to provide a command set similar in appearance and structure to the command set specified for the control of Microprocessor cards in ISO 7816-4. For the user this means there will be some consistency of principles of operation between the different Memory cards and Microprocessor cards.

Memory cards are controlled using three basic commands, Power Up, Power Down, and APDU Exchange. Some of the operating conditions can be specified via Properties documented in the Properties table later in this section and in *Appendix B. Memory Card Support.* The details of the Power Up and APDU Exchange commands vary somewhat depending on the type of

Memory card being used and are documented in a later section of this manual. The Power Down command functions identically for Microprocessor and Memory cards.

Control of many memory cards depends on the manufacturer’s specifications for the card, and it would be challenging/impossible to implement a well designed application without the manufacturer’s specifications. Many manufacturers release specifications only with Non Disclosure Agreements, so MagTek may not be able to provide such documentation to users. It is assumed that the user has access to the manufacturer’s specifications for the card type being used.

Currently supported values of the Card Type Property are:

Value	Card Type
0x00	ISO 7816-3(1997) Microprocessor Cards (T=0/T=1)
0x01	SLE 4428/4418 (or compatible) Memory Cards
0x02	SLE 4442/4432 (or compatible) Memory Cards
0x03	SLE 4404 family Memory Cards (includes many similar cards)
0x05	I^2C™ Cards
0x06	Gemplus GAM326 Memory Card
0x07	Gemplus GAM375 Memory Card

PROPERTIES

Reset Values marked with * are per ISO 7816-3(1997), or EMV 4.1. Changing these properties is not advisable unless the card you are operating is non-standard.

General Properties used by the whole application. In order for these properties to influence a card session, they must be set before the card session begins. Once set, these property values will persist for the current connector until a device reset, or until they are changed by further property settings.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set/ Save	Description
CondRpt	00	Binary	15	N/A	Get	Operation Condition Report. This property only has meaning if the most recent request was Power Up, Exchange APDU, or Exchange TPDU. This property gives detailed information on the operation and is most useful if the RC in the response was either Failure or Warning. See Error / Warning Condition Templates section for a detailed explanation of this property.
CardType	01	Binary	1	0x00	Get/Set	Card Type See Card Types in this section for values
Volts	02	Binary	1	0x03*	Get/Set	Card Voltages allowed: 0x01 = 5 volts only 0x02 = 3 volts only 0x03 = 5 and 3 volts
UART-ID	03	Binary	11	N/A	Get	ID of Smart Card UART

Microprocessor Card Setup Properties, used to control the Power Up sequence. In order for these properties to influence a card session, they must be set before the card session begins. Once set, these property values will persist for the current connector until a device reset, or until they are changed by further property settings.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set	Description
Initial ETU	15	Dword	2	372*	Get/Set	Initial value for ETU, in card clocks. Used to set bit time.
Initial CWT	16	Dword	2	449*	Get/Set	Maximum time, in 2 ms units, to wait between characters during the ATR. In EMV Mode this property is set to 472 yielding 10080 etus.
Initial BWT	18	Dword	2	6*	Get/Set	Maximum time, in 2 ms units, to wait for first character of ATR after Reset
Default EGT	19	Dword	1	0*	Get/Set	Default EGT, used if card sends no EGT info in TC1.
Default BGT	1A	Dword	1	22*	Get/Set	Default BGT, always used.
Power Up Error Conditions Template	1B	Binary	4	(Hex)* 0F 00 00 00	Get/Set	Used to define which conditions shall be considered errors when powering up the card. The options in this bit-map are defined in the description of the Power Up command.

Power Up Warning Conditions Template	1C	Binary	4	(Hex)* 70 d0 47 00	Get/Set	Used to define which conditions shall be considered warnings when powering up the card. The options in this bit-map are defined in the description of the Power Up command.
Power Up Warm Reset Conditions Filter Template	1D	Binary	4	(Hex)* 00 00 00 00	Get/Set	Used in combination with an EMV Cold / Warm Power Up sequence. When a warm reset is necessary, the Warning Conditions Template is ANDed with the inverse of the Warm Reset Conditions Filter Template, then the resulting Warning Conditions Template is copied to the Error Conditions Template just before the Warm Reset occurs. The effect is to allow conditions which were considered warnings during the Cold Reset to be ignored during the Warm Reset. See the description of the Power Up command for more information.
PPS Allowed	1E	Boolean	1	TRUE*	Get/Set	If a card sends an ATR allowing more than one protocol and/or speed, the value of this property indicates whether a PPS sequence should be performed. If this property is set to TRUE, a PPS will be performed, otherwise it will not be sent.
Protocol Selection Allowed	1F	Boolean	1	TRUE	Get/Set	This property indicates if a PPS sequence is allowed to select a different protocol. If it is set TRUE and a PPS sequence is performed, the sequence will not attempt to select a different protocol.
Speed Selection Allowed	20	Boolean	1	TRUE	Get/Set	This property indicates if a PPS sequence is allowed to select a different speed. If it is set to TRUE and a PPS sequence is performed, the sequence will not attempt to select a different speed.
Force PPS1 Transmission	21	Boolean	1	FALSE	Get/Set	This property is only referenced when performing a PPS sequence. If it is TRUE, the PPS sequence will always include a PPS1 byte. If it is FALSE a PPS1 byte would only be included if a speed change were required. This property exists to allow the user to tailor the PPS sequence for cards with special requirements. Unless your card is having problems with PPS, we recommend you leave this property FALSE.
Reset Delay	22	Dword	1	2*	Get/Set	Time to hold RST low in either a Cold or Warm Reset sequence. The units are increments of 200 card clocks. The default value of 2 yields 400 clocks per ISO 7816-3(1997). In EMV Mode this property is set to 205, yielding 41,000 clocks.
Abort on Parity Error in ATR	23	Boolean	1	FALSE*	Get/Set	If this property is TRUE and a parity error is detected in the ATR, the card is powered down immediately and the Power Up request fails. If it is FALSE and a parity error occurs in the ATR, the IFD will request the byte from the card again.

ATR Secondary Timeout	24	Dword	2	0*	Get/Set	If this property is zero, there is no limit on the total time an ATR takes to complete. If this property has a non-zero value, it specifies (in 2 ms intervals) the maximum time allowed from the start of the first byte of the ATR to the reception of the last byte in the ATR. If this time is exceeded, the card is deactivated and the Power Up request fails. In EMV Mode this property is set to 939, which yields 20160 etus.
EMV Reset Rules	25	Boolean	1	FALSE*	Get/Set	If this property is TRUE, EMV Reset Rules are followed in the Power Up sequence. If this property is FALSE, only a Cold Reset is performed. In EMV Mode this property is TRUE. See EMV specification for details of operation.
TC2 Minimum	26	Binary	1	0x01*	Get/Set	This property gives the minimum value allowed for TC2 in the ATR. A lower value will cause an Error or Warning depending on the contents of the Error and Warning Condition Templates.
TC2 Maximum	27	Binary	1	0xFF*	Get/Set	This property gives the maximum value allowed for TC2 in the ATR. A higher value w ill cause an Error or Warning depending on the contents of the Error and Warning Condition Templates. In EMV Mode this property has the value 10. TC2 is used to calculate the Work Waiting Time in T=0.
TA3 Minimum	28	Binary	1	1*	Get/Set	This property gives the minimum value allowed for TA3 in the ATR. A lower value w ill cause an Error or Warning depending on the contents of the Error and Warning Condition Templates. In EMV Mode this property has the value 16. TA3 gives IFSC, the maximum number of bytes in the INF field of a T=1 message.
TA3 Maximum	29	Binary	1	254*	Get/Set	This property gives the maximum value allowed for TA3 in the ATR. A higher value w ill cause an Error or Warning depending on the contents of the Error and Warning Condition Templates. TA3 gives IFSC, the maximum number of bytes in the INF field of a T=1 message.
BWI Maximum	2A	Binary	1	9*	Get/Set	This property gives the maximum value allowed for BWI in the ATR. A higher value w ill cause an Error or Warning depending on the contents of the Error and Warning Condition Templates. BWI is found in TB3 of the ATR and is used to calculate the Block Wait Time (BWT) for T=1. In EMV Mode this property is 4.
CWI Maximum	2B	Binary	1	15*	Get/Set	This property gives the maximum value allowed for CWI in the ATR. A higher value w ill cause an Error or Warning depending on the contents of the Error and Warning Condition Templates. CWI is found in TB3 of the ATR and is used to calculate the Character Wait Time (CWT) for T=1. In EMV Mode this property is 5.

EMV TD2 Rules	2C	Boolean	1	FALSE*	Get/Set	This property indicates whether the Power Up sequence should use EMV rules for screening TD2. If the property is TRUE, the EMV rules are used (protocol T=14 may be designated, though it will not be used). If the property is FALSE, TD2 with T=14 will cause an Error or Warning depending on the contents of the Error and Warning Condition Templates. This property is TRUE in the EMV Mode.
Parity Error Repeat Count	2D	Dword	1	5*	Get/Set	In T=0 there is a mechanism for requesting retransmission of a character with incorrect parity. This property sets the total number of receptions/transmissions allowed for a single character in T=0.
Operating Mode	2E	Binary	1	0	Get/Set	This property defines the Operating Mode for microprocessor cards. Changing this changes numerous other properties. **This property should only be changed when no cards are currently activated.** Values are as follows: 0 – ISO 7816-3(1997) Mode 1 – EMV 4.1 Mode Trying to set this property to any other value will get a RC of failure. Setting this property sets the Operating Mode of all connectors in the device.
PPS Delay	30	Dword	4	2	Get/Set	This property defines the delay between receiving an ATR from a card and transmitting a PPS request to a card. This delay is in 10ms units. So the default value of 2 is equivalent to 20ms. Valid values for this delay are 0 – 254 (0 – 2540ms).
Microprocessor Card Operating Properties, used to monitor Microprocessor Card operations. Settings of these properties only apply to the current card session.						
Name	ID (Hex)	Type	Length	Reset Value	Get/Set	Description
Protocol	38	Binary	1	N/A	Get	This property gives the protocol number in use for the current card session.
Proposed Protocol	39	Binary	1	N/A	Get	This property gives the protocol number the card proposed for use pending a PPS sequence.
Speed	3A	Dword	2	N/A	Get	This property gives the speed (in clocks/ETU) for the current card session.
Proposed Speed	3B	Dword	2	N/A	Get	This property gives the speed (in clocks/ETU) the card proposed for use pending a PPS sequence.
Proposed FI/DI	3C	Binary	1	N/A	Get	This property gives the FI and DI from TA1 which were used to calculate the Proposed Speed property.

Actual EGT	3D	Dword	1	N/A	Get	This property is the EGT value for the current session. If TC1 was transmitted in the ATR it was calculated from TC1, otherwise it was calculated from the Default EGT property.

Actual Class	3E	Binary	1	N/A	Get	This property gives the Class Indicator as reported by the card in the ATR, or in the absence of a Class Indicator from the card, the default (Class A only).
Actual Specific Mode	3F	Boolean	1	N/A	Get	This property gives the specific mode indication as reported by the card in the ATR, or in the absence of the Specific Mode byte (TA2), the default value: FALSE = Negotiable Mode (default) TRUE = Specific Mode
ATR Map	40	Binary	67	N/A	Get	This property gives a detailed analysis of the ATR as received from the card. For details see ATR Map Structure.
T=0 Microprocessor Card Setup Properties, used to control the T=0 protocol operations. In order for these properties to influence a card session, they must be set before the card session begins. Once set, these property values will persist for the current connector until a device reset, or until they are changed by further property settings.						
Name	ID (Hex)	Type	Length	Reset Value	Get/Set	Description
Initial T=0 Error Conditions Template	51	Binary	4	(Hex)* 0F 00 00 00	Get/Set	Used to define which conditions shall be considered errors when running with the T=0 protocol. See Error / Warning Condition Templates for further discussion. See T=0 Error / Warning Conditions for specific coding.
Initial T=0 Warning Conditions Template	52	Binary	4	(Hex) 00 00 00 00	Get/Set	Used to define which conditions shall be considered warnings when running with the T=0 protocol. See Error / Warning Condition Templates for further discussion. See T=0 Error / Warning Conditions for specific coding.
T=0 INS Mask	53	Binary	1	0xFE*	Get/Set	This property defines the mask that is ANDed with an INS byte returned from the card. If the INS byte, or its inverse, matches the result, the INS byte is considered to be a valid procedure byte. In the EMV Mode, this property is set to 0xFF, requiring all bits to match.
T=0 Microprocessor Card Operating Properties, used to control the T=0 protocol operations. Settings of these properties only apply to the current card session.						

Name	ID (Hex)	Type	Length	Reset Value	Get/Set	Description

T=0 Actual WWT	60	Dword	2	N/A	Get	This is the Work Wait Time, in units of 2 ms, being used for T=0 operations for this card.
T=0 Actual Error Conditions Template	61	Binary	4	(Hex)* 0F 00 00 00	Get/Set	Used to define which conditions shall be considered errors when running with the T=0 protocol. See Error / Warning Condition Templates for further discussion. See T=0 Error / Warning Conditions for specific coding.
T=0 Actual Warning Conditions Template	62	Binary	4	(Hex) 00 00 00 00	Get/Set	Used to define which conditions shall be considered warnings when running with the T=0 protocol. See Error / Warning Condition Templates for further discussion. See T=0 Error / Warning Conditions for specific coding.
T=0 Actual INS Mask	63	Binary	1	0xFE*	Get/Set	This property defines the mask that is ANDed with an INS byte returned from the card. If the INS byte, or its inverse, matches the result, the INS byte is considered to be a valid procedure byte. In the EMV Mode, this property is set to 0xFF, requiring all bits to match.
T=1 Microprocessor Card Setup Properties, used to control the T=1 protocol operations. In order for these properties to influence a card session, they must be set before the card session begins. Once set, these property values will persist for the current connector until a device reset, or until they are changed by further property settings.						
Initial NAD	6F	Binary	1	0x00*	Get/Set	This property is the NAD byte sent in a T=1 protocol frame. Leave it at 0x00 unless your card is special.
Initial IFSD Adjustment Allowed	70	Boolean	1	TRUE*	Get/Set	This property indicates whether an IFSD adjust S(IFSD Req, 254) message will be sent to the card after the ATR has been received (in T=1 Protocol). If TRUE, the message is sent, if FALSE, no message is sent. If the S(IFSD Req, 254) message is not sent, the card would assume the default IFSD of 32. This may be undesirable.
Initial Resynch Allowed	71	Boolean	1	TRUE*	Get/Set	This property indicates whether T=1 handling is allowed to send a RESYNCH message when it has certain protocol errors (see 7816-3(1997). If it is TRUE, the RESYNCH message is allowed. If it is FALSE, the message is not permitted. In EMV Mode this property is set to FALSE.
Initial Chaining Allowed	72	Boolean	1	TRUE*	Get/Set	This property indicates whether T=1 handling is allowed to chain messages longer than IFSC to the card. If TRUE, chaining is allowed. If FALSE, chaining is not allowed and any request longer than IFSC will fail.
Initial T=1 Error Conditions Template	73	Binary	4	(Hex)* C1 0B 60 03	Get/Set	Used to define which conditions shall be considered errors when running with the T=1 protocol. See Error / Warning Condition Templates for further discussion. See T=1 Error / Warning Conditions for specific coding.

Initial T=1 Warning Conditions Template	74	Binary	4	(Hex)* 3E F4 9E 00	Get/Set	Used to define which conditions shall be considered warnings when running with the T=1 protocol. See Error / Warning Condition Templates for further discussion. See T=1 Error / Warning Conditions for specific coding.
Initial Use EMV NAD Rules	75	Boolean	1	FALSE*	Get/Set	TRUE indicates to validate NAD according to EMV rules. FALSE indicates to ignore NAD received from cards. In EMV Mode this property is TRUE.
IFSD Allowed after ATR	76	Boolean	1	TRUE*	Get/Set	TRUE indicates an S(Req IFSD(254)) should be sent to the card after the ATR has been received. FALSE indicates no S(Req IFSD) should be sent. In all cases, the effective IFSD is 254 as the IFD always accepts full length messages.
T=1 Microprocessor Card Operating Properties, used to control the T=1 protocol operations. Settings of these properties only apply to the current card session.						
Actual NAD	88	Binary	1	0x00*	Get/Set	This is the NAD byte per ISO 7816-3(1997).
Actual CWT	89	Dword	2	N/A	Get	This is the current CWT, in units of 2 ms, calculated per ISO 7816-3(1997).
Actual BWT	8A	Dword	2	N/A	Get	This is the current BWT, in units of 2 ms, calculated per ISO 7816-3(1997).
Actual EDC	8B	Binary	1	N/A	Get	This is the EDC indicator per ISO 7816-3(1997).
Current Session Resynch Allowed	8C	Boolean	1	N/A	Get/Set	Similar to the Initial Resynch Allowed property, but applies to the current session only.
Current Chaining Allowed	8D	Boolean	1	N/A	Get/Set	Similar to the Initial Chaining Allowed proprety, but applies to the current session only.
Current IFSC	8E	Dword	1	N/A	Get	The IFSC in use for the current session.
Current Output Sequence Number	8F	Dword	1	N/A	Get	This is the sequence number of the next I-block that will be sent to the card.
Current Input Sequence Number	90	Dword	1	N/A	Get	This is the sequence number expected in the next I-block that will be received from the card.
Current T=1 Error Conditions Template	91	Binary	4	N/A*	Get/Set	Used to define which conditions shall be considered errors when running with the T=1 protocol. See Error / Warning Condition Templates for further discussion. See T=1 Error / Warning Conditions for specific coding. The values are initialized from the Initial T=1 Error Conditions Template property.

Current T=1 Warning Conditions Template	92	Binary	4	N/A*	Get/Set	Used to define which conditions shall be considered warnings when running with the T=1 protocol. See Error / Warning Condition Templates for further discussion. See T=1 Error / Warning Conditions for specific coding. The values are initialized from the Initial T=1 Warning Conditions Template property.
Current Use EMV NAD Rules	93	Boolean	1	N/A	Get/Set	TRUE indicates to validate NAD according to EMV rules. FALSE indicates to ignore NAD received from cards. In EMV Mode this property is TRUE. The value is initialized from the Initial Use EMV NAD Rules property.

Note: Other Property IDs are specified in *Appendix B. Memory Card Support.*

ATR MAP STRUCTURE – MICROPROCESSOR CARDS ONLY

The ATR Map structure is retrieved by reading property 40 (Hex). This property gives a detailed analysis of the ATR as received from the card. The structure is as follows:

Bytes	Meaning
0	TS character, as received from the card
1	T0 character, as received from the card
2	TA1 Present? 0 = no, 1 = yes
3	TA1 value If TA1 Present = 1, as received from the card If TA1 not present, set to 0x11
4	TB1 Present? 0 = no, 1 = yes
5	TB1 value If TB1 Present = 1, as received from the card If TB1 Present = 0, set to 0x25
6	TC1 Present? 0 = no, 1 = yes
7	TC1 value If TC1 Present = 1, as received from the card If TC1 Present = 0, set to 0X00
8	TD1 Present? 0 = no, 1 = yes
9	TD1 value If TD1 Present = 1, as received from the card If TD1 Present = 0, ignore the value
10	TA2 Present? 0 = no, 1 = yes
11	TA2 value If TA2 Present = 1, as received from the card If TA2 Present = 0, ignore the value
12	TB2 Present? 0 = no, 1 = yes
13	TB2 value If TB2 Present = 1, as received from the card If TB2 Present = 0, ignore the value
14	TC2 Present? 0 = no, 1 = yes
15	TC2 value If TC2 Present = 1, as received from the card If TC2 Present = 0, set to 0x0A
16	TD2 Present? 0 = no, 1 = yes
17	TD2 value If TD2 Present = 1, as received from the card If TD2 Present = 0, ignore the value
18	TCK Present, 0 = no, 1 = yes
19	TCK value If TCK Present = 1, as received from the card If TCK Present = 0, ignore the value
20	Number of history bytes received from the card
21 – 36	History bytes as received from the card. The bytes start at position 21 and continue for the count specified in location 20. Ignore additional bytes
37	T=0 Available? 0 = no, 1 = yes
38	T=1 Available? 0 = no, 1 = yes
39	TAi for T=1 Present? 0 = no, 1 = yes
40	TAi for T=1 value If TAi for T=1 Present = 1, as received from the card If TAi for T=1 Present = 0, 0x20

Bytes	Meaning
41	TBi for T=1 Present? 0 = no, 1 = yes
42	TBi for T=1 value If TBi for T=1 Present = 1, as received from the card If TBi for T=1 Present = 0, 0x4D
43	TCi for T=1 Present? 0 = no, 1 = yes
44	TCi for T=1 value If TCi for T=1 Present = 1, as received from the card If TCi for T=1 Present = 0, 0x00
45	T=15 Present? 0 = no, 1 = yes
46	TAi for T=15 Present? 0=no, 1=yes
47	TAi for T=15 value If T=15 Present = 1, as received from the card If T=15 Present = 0, 0x01
48	Convention, 0 = direct, 1 = inverse
49	FI, see ISO 7816-3(1997)
50	DI, see ISO 7816-3(1997)
51	II, see ISO 7816-3(1997)
52	PI1, see ISO 7816-3(1997)
53	N, see ISO 7816-3(1997)
54	Specific Mode Indicator, 0 = Negotiable, 1 = Specific
55	Specific Protocol
56	Specific Mode Implicit Parameters Requested? 0 = no, 1 = yes
57	Specific Mode Changeable, 0 = Changeable, 1 = Not Changeable
58	PI2 Present? 0 = no, 1 = yes
59	PI2 value If PI2 Present = 1, as received from the card If PI2 Present = 0, ignore the value
60	WI
61	Clock Stop Allowed Indication 0 = never 1 = stop clock low 2 = stop clock high 3 = stop clock high or low Note: This is an indication of what stop clock modes are possible for the card, the IFD **does not put a capable card into the clock stop mode.**
62	Classes supported by the card 1 = 5 volts only 2 = 3 volts only 3 = both 5 and 3 volts
63	IFSC being supported by the card
64	CWI
65	BWI
66	Error Detection Code 0 = LRC 1 = CRC

POWER UP COMMAND

Command ID: 80 (Hex)
Description:
This command is used to power up a smart card in the currently selected connector. If it is available, the ATR from the card is returned.

Request Message Data: None
Response Message Data:

Byte	Field	Description
1-n	ATR	Answer to Reset as received from the card.

Response Message Result Codes

Code (Hex)	Meaning
00	Success, the command completed without problems
01	Failure, the command was not able to complete, see CondRpt property for details. See below for Error/Warning Conditions Template construction.
02	Warning, the command completed, but some problems arose, see CondRpt property for details. See below for Error/Warning Conditions Template construction.

Error/Warning Conditions Template Construction – Microprocessor Cards Only

The Error/Warning Condition Templates (Power Up Error Conditions Template and Power Up Warning Conditions Template properties respectively) have the following structure:

Byte	Bit	Meaning
0	0	Unrecoverable Receive Error. This condition indicates a receive error that could not be corrected by the permitted number of parity correction retries.
0	1	Timeout On Receive. This condition indicates a receive timeout.
0	2	Specific f/d ratio required but not supported. This condition arises when a card sends the TA2 byte and the values of FI and DI in TA1 require an f/d ratio the IFD can not support. The IFD supports f/d ratios as low as 31, which is equivalent to an I/O rate of 129kbs.
0	3	Specific protocol required but not supported. The card has required a protocol which the IFD does not support. The IFD supports T=0 and T=1.
0	4	Requested f/d ratio not supported, may be negotiable. The card has requested an f/d ratio that is not supported, but it has indicated that it may negotiate via Warm Reset.
0	5	Requested protocol not supported, may be negotiable. The card has requested a protocol that is not supported, but it has indicated that it may negotiate via Warm Reset.
0	6	TCK not present or incorrect. This condition only appears when the ISO/EMV rules require a TCK and the card does not send it.
0	7	Card requests VPP. This condition occurs only if the card requests VPP. This IFD does not support VPP.

Byte	Bit	Meaning
1	0	Card started ATR early. This condition occurs when a card starts the ATR too earlier than allowed by ISO or EMV rules.
1	1	Invalid TS. This condition indicates that the TS character had a value other than 0x3B or 0x3F.
1	2	Unknown Interface Byte. This condition occurs when an interface byte not defined by ISO 7816-3 or EMV is sent as part of the ATR.
1	3	Invalid Protocol Sequence. This condition occurs when the ATR specifies protocols in a non-ascending order.
1	4	Specific Mode requesting Implicit Parameters. This condition happens when the card sends TA2 with bit b5 (see ISO 7816-3(1997)) set.
1	5	TC2 present with T=0. This condition indicates that TC2 was sent in an ATR that does not support T=0.
1	6	IFSC out of allowed range. This condition indicates that the card requested an IFSC out of the range permitted by current rules. The current range is specified by properties TA3 Minimum and TA3 Maximum.
1	7	Invalid EDC type requested. The card has requested an unsupported EDC type. At this time the IFD only supports LRC.
2	0	PPS sequence failed. The IFD was unable to complete a successful PPS sequence. In order to operate this card, it may be necessary to change one or more of the following properties: PPS Allowed Protocol Selection Allowed Speed Selection Allowed Force PPS1 Transmission See your card's documentation for more details.
2	1	Parity Error in ATR.
2	2	EGT out of range. The EGT calculated from N in the ATR is larger than 254 ETUs.
2	3	TD2 not according to EMV rules. Indicates that TD2 does not conform to EMV rules. The condition is only detected if the property EMV TD2 Rules is set to TRUE.
2	4	T=15 received. This condition only indicates that the card sent a byte specifying T=15.
2	5	TB2 received. This condition only indicates that the card sent a TB2 byte.
2	6	TC2 outside of allowed range. The card sent a TC2 byte out of the allowed range. The allowed range is specified in properties TC2 Minimum and TC2 Maximum.
2	7	EMV TB3 error. This condition indicates that for a card requesting T=1 either TB3 was absent or the values specified for BWI or CWI were out of range. See EMV, ICC, I-4.3.3.10
3	0-7	Reserved for Future Use

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	02	80	00

Example Response:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-8
Value (Hex)	40	02	80	00	3B 60 00 00

POWER DOWN COMMAND

Command ID: 81 (Hex)
Description:
This command is used to power down a smart card in the currently selected connector.

Request Message Data: None
Response Message Data: None
Response Message Result Codes

Code	Meaning
00	Success.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	02	81	00

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	02	81	00

T=0 ERROR / WARNING CONDITION TEMPLATES

Byte	Bit	Meaning
0	0	Timeout On Receive. This condition indicates a receive timeout.
0	1	Unrecoverable Receive Error. This condition indicates a receive error that could not be corrected by the permitted number of parity correction retries.
0	2	Invalid Procedure Byte received from card.
0	3	Unrecoverable Transmit Error. This condition indicates a transmit error that could not be corrected by the permitted number of parity correction retries.
0	4-7	Reserved for Future Use
1-3	All	Reserved for Future Use

T=1 ERROR / WARNING CONDITION TEMPLATES

Byte	Bit	Meaning
0	0	Timeout On Receive. This condition indicates a receive timeout.
0	1	Parity Error in received data. This condition indicates that there was a parity error in a message received from the card.
0	2	I/O errors in a message received from the card. These errors could be Length, EDC, or an invalid PCB.
0	3	The card rejected a message with the indication: Error Free
0	4	The card rejected a message with the indication: EDC Error
0	5	The card rejected a message with the indication: Other Error
0	6	Reserved for Future Use.
0	7	Card Aborted a chain being sent by the IFD.
1	0	Card Aborted a chain being sent by the Card.
1	1	Input Buffer Overflow. The card tried to send a chain of messages that is larger than the available buffer space.
1	2	The card sent an I-Block message (or chain) with no content in the INF field.
1	3	The card was sending chained data when a Resynch sequence was required; the chained data from the card was lost.
1	4	The IFD was not sending a chained message to the card and the card sent an R-Block requesting a message other than the one we are trying to send. This is a protocol error on the part of the card.
1	5	Invalid Sequence Number in an I-Block received from a card.
1	6	The IFD was sending a chained message to the card and the card sent an R-Block requesting a message other than the one we are trying to send.
1	7	Reserved for Future Use
2	0	Invalid S-Block Response Type from the card. An S-Block was sent to the card and it responded with an S-Block of the wrong type.
2	1	Invalid S-Block Response Data from the card. An S-Block was sent to the card and it responded with an S-Block with an invalid INF field.
2	2	Invalid S-Block Request Type from card. The card sent an S-Block Request with an invalid Type.
2	3	Invalid S-Block Request Data from card. The card sent an S-Block Request with an invalid INF field.
2	4	Resynch needed, completed successfully.
2	5	Resynch needed, but not performed because it is disables (property Initial Resynch Allowed or Current Session Resynch Allowed).
2	6	Resynch not attempted, protocol had not yet started.
2	7	Reserved for Future Use
3	0	Resynch failed.
3	1	S-Block exchange failed, no more retries.
3	2-7	Reserved for Future Use

WARM RESET COMMAND

Command ID: 82 (Hex)
Description:
This command is used to perform a Warm Reset on a smart card in the currently selected connector. If it is available, the ATR from the card is returned.

Request Message Data: None
Response Message Data:

Byte	Field	Description
1-n	ATR	Answer to Reset as received from the card.

Response Message Result Codes

Code (Hex)	Meaning
00	Success, the command completed without problems
01	Failure, the command was not able to complete, see CondRpt property for details. See below for Error/Warning Conditions Template construction.
02	Warning, the command completed, but some problems arose, see CondRpt property for details. See below for Error/Warning Conditions Template construction.

Error/Warning Conditions Template Construction – Microprocessor Cards Only

See Error/Warning Conditions Template Construction for the Power Up command.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	02	82	00

Example Response:

Field	MTYP	APPL	CMND	RC	PVAL
Byte	1	2	3	4	5-8
Value (Hex)	40	02	82	00	3B 60 00 00

TPDU, CARD TO IFD COMMAND

Command ID: 83 (Hex)
Description:

Note

Unless your card can not operate in the APDU mode, it is recommended not to use this command*, as it is not compatible with EMV Rules nor the intended ISO 7816-4 application usage.*

This command is used to send a TPDU to a card requesting data from the card. It is the user's responsibility to assure that the INS byte specifies a card command which will result in data being returned to the IFD. The TPDU is sent to the card and the reply is collected according to ISO 7816-3(1997) rules.

Request Message Data:

Byte	Field	Description
1	CLA	CLA byte per card's specification
2	INS	INS byte per card's specification
3	P1	P1 byte per card's specification
4	P2	P2 byte per card's specification
5	P3	P3, number of bytes expected in response from the card, per ISO 7816-3(1997)

Response Message Data:

Byte	Field	Description
1-n	TPDU Response	Response TPDU received from the card. SW1/SW2 are at the end.

Response Message Result Codes

Code (Hex)	Meaning
00	Success, the command completed without problems
01	Failure, the command was not able to complete, see CondRpt property for details. See T=0 or T=1 Error/Warning Condition Templates for Template construction.
02	Warning, the command completed, but some problems arose, see CondRpt property for details. See T=0 or T=1 Error/Warning Condition Templates for Template construction.

Example Request:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-9
Value (Hex)	00	02	83	00	00 B0 81 00 02

Example Response:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-8
Value (Hex)	40	02	83	00	0B 03 00 00

TPDU, IFD TO CARD COMMAND

Command ID: 84 (Hex)
Description:

> ***Note***
> *Unless your card can not operate in the APDU mode, it is recommended* not to use this command*, as it is not compatible with EMV Rules nor the intended ISO 7816-4 application usage.*

This command is used to send a TPDU to a card with data for the card, or with no data for the card and no data expected in response from the card. It is the user's responsibility to assure that the INS byte specifies a card command which will not result in data being returned to the IFD. The TPDU is sent to the card and the SW1/SW2 reply is collected according to ISO 7816-3(1997) rules.

Request Message Data:

Byte	Field	Description
1	CLA	CLA byte per card's specification
2	INS	INS byte per card's specification
3	P1	P1 byte per card's specification
4	P2	P2 byte per card's specification
5	P3	P3, number of Data bytes to send to the card
6-n	Data	Data to be sent to the card

Response Message Data:

Byte	Field	Description
1-2	TPDU Response	SW1/SW2.

Response Message Result Codes:

Code (Hex)	Meaning
00	Success, the command completed without problems
01	Failure, the command was not able to complete, see CondRpt property for details. See T=0 or T=1 Error/Warning Condition Templates for Template construction.
03	Warning, the command completed, but some problems arose, see CondRpt property for details. See T=0 or T=1 Error/Warning Condition Templates for Template construction.

Example Request:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-11
Value (Hex)	00	02	84	00	00 A4 02 0C 02 00 01

Example Response:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-6
Value (Hex)	40	02	84	00	90 00

APDU EXCHANGE COMMAND

Command ID: 85 (Hex)

Description:

This command is used to exchange an APDU with a card. See ISO 7816-4 and EMV for information on APDU structure. This command functions differently for T=0 and T=1 protocols:

For T=0, the Command APDU is mapped onto TPDU's and the exchange is managed according to EMV rules, returning a Response APDU.

For T=1, the Command APDU is sent directly to the card and the Response APDU is returned.

For Memory cards, the Command APDU is interpreted and the appropriate Control Sequence is sent to the card. A Response APDU is returned.

Request Message Data:

Byte	Field	Description
1-n	Command APDU	Command APDU per ISO 7816-4 and EMV

Response Message Data:

Byte	Field	Description
1-n	Response APDU	Response APDU per ISO 7816-4 and EMV

Response Message Result Codes:

Code (Hex)	Meaning
00	Success, the command completed without problems
01	Failure, the command was not able to complete, see CondRpt property for details. See T=0 or T=1 Error/Warning Condition Templates for Template construction.
02	Warning, the command completed, but some problems arose, see CondRpt property for details. See T=0 or T=1 Error/Warning Condition Templates for Template construction.

Example Request:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-11
Value (Hex)	00	02	85	00	00 A4 02 0C 02 00 01

Example Response:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5-6
Value (Hex)	40	02	85	00	90 00

SELECT CONNECTOR COMMAND

Command ID: 86 (Hex)
Description:
The IFD supports up to 8 connectors numbered 0-7. Connector 0 is the User Connector, and is used by default when the system starts. Connectors 1-7 may be used for Secure Access Modules (SAMs) as part of application infrastructure.

This command allows the application to select the connector to be used in commands that follow. The connector assignment stays in effect until it is changed by another Select Connector Command or the device is power cycled.

Request Message Data:

Byte	Field	Description
1	Connector Number	The number, in Hex, to be selected

Response Message Data: None
Response Message Result Codes

Code	Meaning
00	Success

Example Request:

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5
Value (Hex)	00	02	86	00	02

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	02	86	00

SECTION 6. LED APPLICATION

Application ID: 81 (Hex)
Description:
This application deals with the host controlled LED.

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (Hex)
Save Property Command ID: 02 (Hex)
Description:
These commands are used to get, set and save the value of application properties. Details of these commands are described in the Generic command section of this document. The following describes each property this application supports.

LED STATE PROPERTY

Property ID: 00 (Hex)
Property Type: Dword
Power Up/Reset Value: 0 (OFF)
Get/Set/Save support: All
Description:
Changing this property changes the LED state. The LED state can be retrieved by getting this property.

Values:
The lowest order byte governs the color of the LED as follows:

Value	LED State
0	OFF
1	RED
2	GREEN

The next byte signals whether the LED will blink and the frequency of the blink. Blinking will only be active if the specified color is Red or Green. A value of zero specifies solid color with no blinking. A value in the range of 1 – 254 specifies, at 10 milliseconds per count, how long both the on and off period of the LED should be. A value of 255 is not allowed and will result in an error.

Example Set Property Request: Green LED Blinking, 100 msec on, 100 msec off

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	81	01	00	01	00

Field	PVAL
Byte	7 – 10
Value (Hex)	02 64 00 00 (GREEN)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	81	01	00

SECTION 7. TRANSPORT APPLICATION

Application ID: 82 (Hex)
Description:
This application deals with card transportation and magnetic stripe data acquisition. This application works closely with the Magnetic Stripe application. The Magnetic Stripe application deals with decoding and presenting magnetic stripe card data.

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (Hex)
Save Property Command ID: 02 (Hex)
Description:
These commands are used to get, set and save the value of application properties. Details of these commands are described in the Generic command section of this document. The following describes each property this application supports.

INDICATORS PROPERTY

Property ID: 00 (Hex)
Property Type: Dword
Power Up/Reset Value: Dependent on indicator states
Get/Set/Save support: Get
Description:
This property is used to get the transport indicators. These indicators are used to determine the state of the transport and the location of any cards present in the transport.

Values:

The bits are identified by numbering the least significant bit 0 and the most significant bit 31.

Bits	Indicator	Description
0	Card Present	Set to 1 when the card seated indicator is set during card insertion. Set to 0 sometime after the card seated indicator is cleared during withdrawal. This would occur as soon as one of the following two events occurs. The magnetic stripe read ASIC indicates that card data is ready or two seconds has past since the card seated indicator is cleared. The two second timeout is only used to handle blank or wrongly inserted cards. This indicator can be used to indicate the end of a card insertion or the end or a card withdrawal. Note that if the MSR direction property is set to non-directional then the behavior of this indicator changes such that it will behave the same as the card seated indicator.
1	Card Seated	Set to 1 if a card is seated in the fully inserted position in the reader else cleared to 0. This bit can be used to indicate the end of a card insertion or the beginning of a card withdrawal. If this bit is set then the card is in the correct location for smart card communications and card latching.
2	Card Latched	Set to 0 at power up or reset because the card is automatically unlatched at this time. Set to 1 after the host issues the latch card command. Set to 0 after the host issues the unlatch card command.
3 – 6	RFU	Reserved for future use
7	*MagnePrint Noise	Set to 1 if the MagnePrint™ circuit is sensing excessive electrical noise in the environment, else cleared to 0. If this bit is set MagnePrint™ data may not be available. Moving the reader away from noise sources (monitors, speakers, etc.) may solve this problem. Note that the reader will not check for noise until after a card swipe occurs. So a card has to be swiped to initiate noise detection. If noise is detected after the swipe, the reader will continue to check for noise until the noise is no longer present. If no noise is detected after the swipe, the reader will not check for noise again until after the next swipe.
8-31	RFU	Reserved for future use

** Available only on units with MagnePrint feature*

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	82	00	00	01	00

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	82	00	00	01	00

Field	PVAL
Byte	7 - 10
Value (Hex)	07 00 00 00 (Card is present, seated and latched)

NOTIFY INDICATOR CHANGE 0 TO 1 PROPERTY

Property ID: 01 (Hex)
Property Type: Dword
Power Up/Reset Value: 0
Get/Set/Save support: All
Description:
This property is used control notification messages sent in response to a transport indicator changing from a 0 to a 1 state. If the bit corresponding to an indicator is set then, a notification message will be sent from the device to the host in response to that indicator changing from a 0 to a 1 state. The notification message will have the same syntax as the get property command response of the indicator property. The only difference is that the message header will contain a notification message type instead of a response message type. If more than one indicator changes state at the same time, only one notification message will be sent that contains the state of all the indicators.

Values:
The bits are identified by numbering the least significant bit 0 and the most significant bit 31.

Bits	Indicator	Description
0	Card Present	If set to 1 a notification message will be sent in response to a 0 to 1 transition of the card present indicator. If cleared to 0, no notification will be sent.
1	Card Seated	If set to 1 a notification message will be sent in response to a 0 to 1 transition of the card seated indicator. If cleared to 0, no notification will be sent.
2 – 6	RFU	Reserved for future use
7	*MagnePrint Noise	If set to 1 a notification message will be sent in response to a 0 to 1 transition of the MagnePrint™ noise indicator. If cleared to 0, no notification will be sent.
8-31	RFU	Reserved for future use

** Available only on units with MagnePrint feature*

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	82	01	00	01	01

Field	PVAL
Byte	7 - 10
Value (Hex)	02 00 00 00 (Card Seated)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	82	01	00

NOTIFY INDICATOR CHANGE 1 TO 0 PROPERTY

Property ID: 02 (Hex)
Property Type: Dword
Power Up/Reset Value: 0
Get/Set/Save support: All
Description:
This property is used control notification messages sent in response to a transport indicator changing from a 1 to a 0 state. If the bit corresponding to an indicator is set then, a notification message will be sent from the device to the host in response to that indicator changing from a 1 to a 0 state. The notification message will have the same syntax as the get property command response of the indicator property. The only difference is that the message header will contain a notification message type instead of a response message type. If more than one indicator changes state at the same time, only one notification message will be sent that contains the state of all the indicators.

Values:
The bits are identified by numbering the least significant bit 0 and the most significant bit 31.

Bits	Indicator	Description
0	Card Present	If set to 1 a notification message will be sent in response to a 1 to 0 transition of the card present indicator. If cleared to 0, no notification will be sent.
1	Card Seated	If set to 1 a notification message will be sent in response to a 1 to 0 transition of the card seated indicator. If cleared to 0, no notification will be sent.
2 – 6	RFU	Reserved for future use
7	*MagnePrint Noise	If set to 1 a notification message will be sent in response to a 1 to 0 transition of the MagnePrint™ noise indicator. If cleared to 0, no notification will be sent.
8-31	RFU	Reserved for future use

**Available only on units with MagnePrint feature*

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	82	01	00	01	02

Field	PVAL
Byte	7 - 10
Value (Hex)	02 00 00 00 (Card Seated)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	82	01	00

MSR ARM STATE PROPERTY

Property ID: 03 (Hex)
Property Type: Dword
Power Up/Reset Value: 0 (Unarmed)
Get/Set/Save support: All
Description:
This property is used control the magnetic stripe read arm state. This state must be set properly in order to read magnetic stripe cards. This property should be used with the MSR Direction property.

Values:

Value	State	Description
0	Unarmed	A read will not occur when a card is swiped.
1	Armed to Read Once	A read will occur on the next card swipe. The read direction depends on how the MSR Direction Property is set. Once the read occurs, the MSR Arm State Property will change to Unarmed State. After which a read will no longer occur when a card is swiped until the host re-arms the reader by changing the MSR Arm State.
2	Armed to Read Multiple Times	A read will occur on every card swipe. The read direction depends on how the MSR Direction Property is set. If the read data is not sent to the host by the time the next read occurs, it will be lost.

Caution:

Be careful when setting the MSR Arm State to Armed to Read Multiple times while the MSR Direction is set to both or non-directional. The reason for this is the user is in control of how fast the card is swiped. If the time period between an insert and a withdrawal is too short, a new read may occur before the magnetic stripe data can be processed and sent to the host. The same situation could occur if the time period between a withdrawal and an insert is too short. This time period depends on how soon the host requests data after a card is read. If the read data is being sent to the host using a notification message then this time period is 500ms maximum.

Example Set Property Request:

Field	**MTYP**	**APPL**	**CMND**	**RC**	**PTYP**	**PID**
Byte	1	2	3	4	5	6
Value (Hex)	00	82	01	00	01	03

Field	**PVAL**
Byte	7 - 10
Value (Hex)	01 00 00 00 (Armed to Read Once)

Example Set Property Response:

Field	**MTYP**	**APPL**	**CMND**	**RC**
Byte	1	2	3	4
Value (Hex)	40	82	01	00

MSR DIRECTION PROPERTY

Property ID: 04 (Hex)
Property Type: Dword
Power Up/Reset Value: 2 (Withdrawal)
Get/Set/Save support: All
Description:
This property is used to control the magnetic stripe read direction. This state must be set properly in order to read magnetic stripe cards. This property should be used with the MSR Arm State property.

Values:

Value	State	Description
0	Both	The reader can read during both card insertion and withdrawal.
1	Insert	The reader can only read during card insertion.
2	Withdrawal	The reader can only read during card withdrawal.
3	Non-directional	The reader can read any time a card with magnetic data is swiped in either direction. The card-present and card-seated indicators are not used for read detection during non-directional reading. Only the signal from the magnetic stripe read ASIC is used.

Caution:

Be careful when setting the MSR Arm State to Armed to Read Multiple times while the MSR Direction is set to both or non-directional. The reason for this is the user is in control of how fast the card is swiped. If the time period between an insert and a withdrawal is too short, a new read may occur before the magnetic stripe data can be processed and sent to the host. The same situation could occur if the time period between a withdrawal and an insert is too short. This time period depends on how soon the host requests data after a card is read. If the read data is being sent to the host using a notification message then this time period is 500ms maximum.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	82	01	00	01	04

Field	PVAL
Byte	7 - 10
Value (Hex)	02 00 00 00 (Withdrawal)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	82	01	00

LATCH CARD COMMAND

Command ID: 80 (Hex)
Description:
This command attempts to engage the card latch. To properly latch a card, this command should only be executed when the Card Seated indicator is set thus indicating that the card is fully inserted into the reader. The Card Latched indictor is set to 1 after this command is sent.

Request Message Data: None
Response Message Data: None
Response Message Result Codes:
Success = 00 (Hex)
The command completed successfully.

Example Request:

Field	**MTYP**	**APPL**	**CMND**	**RC**
Byte	1	2	3	4
Value (Hex)	00	82	80	00

Example Response:

Field	**MTYP**	**APPL**	**CMND**	**RC**
Byte	1	2	3	4
Value (Hex)	40	82	80	00

UNLATCH CARD COMMAND

Command ID: 81 (Hex)
Description:
This command attempts to disengage the card latch. The Card Latched indictor is set to 0 after this command is sent.
Request Message Data: None
Response Message Data: None
Response Message Result Codes:
Success = 00 (Hex)
The command completed successfully.
Example Request:

Field	**MTYP**	**APPL**	**CMND**	**RC**
Byte	1	2	3	4
Value (Hex)	00	82	81	00

Example Response:

Field	**MTYP**	**APPL**	**CMND**	**RC**
Byte	1	2	3	4
Value (Hex)	40	82	81	00

SECTION 8. BUZZER APPLICATION

Application ID: 07 (Hex)
Description:
This application deals with the buzzer.

> ***Note***
> *The buzzer is an optional feature intended to be used primarily with the contact-less smartcard option. It is not installed on all configurations.*

The buzzer is used two distinct ways. First, it is used as a stand-alone application giving the host the ability to generate tone sequences to guide users through operations on the device. Second, it is used by the Contactless Card Application to indicate a media being read.

When it is being used by the Contactless Card Application, it is unavailable to the host.

The following table shows the available tones by name, designator, and frequency.

TONE	DESIGNATOR	FREQUENCY
Silent	0	N/A
Low	1	1500 Hz
Medium	2	2000 Hz
High	3	2500 Hz

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (Hex)
Save Property Command ID: 02 (Hex)
Description:
These commands are used to get, set and save the value of application properties. Details of these commands are described in the Generic command section of this document. The following describes each property this application supports.

TONE SEQUENCE 1 PROPERTY

Property ID: 01 (Hex)
Property Type: Binary
Length: 20
Power Up/Reset Value: 0110 0220 0330 0000 0000 0000 0000 0000 0000 0000 (low tone for 160 ms, medium tone for 320 ms, high tone for 480 ms)
Get/Set/Save support: All
Description:
This property specifies an audible tone that may be played when the host chooses. The property is implemented as ten sets of Tone Information Bytes (TIB) as follows:

Byte	Usage
1	TIB0, designates tone 0 = Silent 1 = Low 2 = Medium 3 = High Other values may not result in a tone
2	TIB1, duration of tone in units of 10 milliseconds (e.g. 3 = 30 milliseconds). Setting this byte to zero terminates the Tone Sequence without processing further entries

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	07	00	00	04	01

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	07	00	00	04	01

PVAL
7-26
01 10 02 20 03 30 00 00 00 00 00 00 00 00 00 00 00 00 00 00

TONE SEQUENCE 2 PROPERTY

Property ID: 02 (Hex)
Property Type: Binary
Length: 20
Power Up/Reset Value: 0330 0220 0110 0000 0000 0000 0000 0000 0000 0000 (high tone for 480 ms, medium tone for 320 ms, low tone for 160 ms)
Get/Set/Save support: All
Description:
This property specifies an audible tone that may be played when the host chooses. The property is implemented as ten sets of Tone Information Bytes (TIB) as follows:

Byte	Usage
1	TIB0, designates tone 0 = Silent 1 = Low 2 = Medium 3 = High Other values may not result in a tone
2	TIB1, duration of tone in units of 10 milliseconds (e.g. 3 = 30 milliseconds). Setting this byte to zero terminates the Tone Sequence without processing further entries

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	07	00	00	04	02

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	07	00	00	04	02

PVAL
7-26
03 30 02 20 01 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00

TONE SEQUENCE 3 PROPERTY

Property ID: 03 (Hex)
Property Type: Binary
Length: 20
Power Up/Reset Value: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 (silent)
Get/Set/Save support: All
Description:
This property specifies an audible tone that may be played when the host chooses. The property is implemented as ten sets of Tone Information Bytes (TIB) as follows:

Byte	Usage
1	TIB0, designates tone 0 = Silent 1 = Low 2 = Medium 3 = High Other values may not result in a tone
2	TIB1, duration of tone in units of 10 milliseconds (e.g. 3 = 30 milliseconds). Setting this byte to zero terminates the Tone Sequence without processing further entries

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	07	00	00	04	03

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	07	00	00	04	03

PVAL
7-26
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

TONE SEQUENCE 4 PROPERTY

Property ID: 04 (Hex)
Property Type: Binary
Length: 20
Power Up/Reset Value: 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 (silent)
Get/Set/Save support: All
Description:
This property specifies an audible tone that may be played when the host chooses. The property is implemented as ten sets of Tone Information Bytes (TIB) as follows:

Byte	Usage
1	TIB0, designates tone 0 = Silent 1 = Low 2 = Medium 3 = High Other values may not result in a tone
2	TIB1, duration of tone in units of 10 milliseconds (e.g. 3 = 30 milliseconds). Setting this byte to zero terminates the Tone Sequence without processing further entries

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	07	00	00	04	04

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	07	00	00	04	04

PVAL
7-26
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

PLAYBACK TONE SEQUENCE COMMAND

Command ID: 80 (Hex)
Description:
This command is used playback a specific tone sequence. On receipt of the command, the playback sequence is started and success is reported.

Request Message Data:

Byte	Usage
1	Specifies which tone sequence to play back. Use the property number of the desired tone sequence.

Response Message Data: None

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

Example Request: (Playback Tone Sequence 2)

Field	MTYP	APPL	CMND	RC	DATA
Byte	1	2	3	4	5
Value (Hex)	00	07	80	00	02

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	07	80	00

SECTION 9. HOST COMMUNICATIONS APPLICATION

Application ID: 08 (Hex)
Description:
This application deals with host communications.

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (Hex)
Save Property Command ID: 02 (Hex)
Description:
These commands are used to get, set and save the value of application properties. Details of these commands are described in the Generic command section of this document. The following describes each property this application supports.

USB SERIAL NUMBER PROPERTY

Property ID: 00 (Hex)
Property Type: String
Maximum Length: 32 (including terminating zero)
Power Up/Reset Value: 00 (Null string “”)
Get/Set/Save support: All
Description:
This property is the USB serial number. The USB serial number is reported to the host in the USB serial number string descriptor during USB device enumeration. This serial number can be used to distinguish one USB device from another in a system. For example, if two IntelliStripe 65 devices are connected to the same host using USB ports, the USB serial numbers can be used to distinguish one device from the other as long as the serial numbers are not set to the same value.

MagTek’s MCP USB driver uses the USB serial number, if present, as part of the device instance name so that if multiple devices with the same model number are connected to the same system they can be distinguished from one another. For example, if a device that has its USB product set to “IntelliStripe 65” has no serial number its device instance name will be “IntelliStripe 65”. If it has serial number “123”, its device instance name will be “IntelliStripe 65.123”. See the host communications application’s USB product property for more information.

After setting the USB serial number to a new value, the serial number needs to be saved with the save property command and then the device needs to be power cycled or reset before the new serial number value is used as the USB serial number string descriptor. If MagTek’s MCP USB driver is being used, it needs to be closed and then opened using the new device instance name before communicating further.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	02	00

Field	PVAL
Byte	7 - 10
Value (Hex)	31 32 33 00 (“123”)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

RS232 PLUG AND PLAY SUPPORT ENABLE PROPERTY

Property ID: 01 (Hex)
Property Type: Boolean
Power Up/Reset Value: 00 (hex) (False)
Get/Set/Save support: All
Description:
This property enables or disables RS232 plug and play support. This support is enabled if this property is set to 1 (true) or disabled if this property is set to 0 (false).

This property is typically only enabled if the RS232 interface of this device is being used to communicate to a PCSC driver on the Windows operating system.

If this property is enabled, the device will transmit a string on the RS232 port when the CTS signal with respect to the device is asserted (goes from negative to positive) by the host according to the Plug and Play External COM Device Specification Version 1.00.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	03	01

Field	PVAL
Byte	7
Value (Hex)	00 (false)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

USB PRODUCT PROPERTY

Property ID: 02 (Hex)
Property Type: String
Maximum Length: 64 (including terminating zero)
Power Up/Reset Value: 49 6E 74 65 6C 6C 69 53 74 72 69 70 65 20 36 35 00 ("IntelliStripe 65")
Get/Set/Save support: All
Description:
This property is the USB product. The USB product is reported to the host in the USB product string descriptor during USB device enumeration.

MagTek's MCP USB driver uses the USB product as part of the device instance name. For example, if a device has no serial number and it's USB product is set to "IntelliStripe 65" then it's device instance name will be "IntelliStripe 65". If it has serial number "123", its device instance name will be "IntelliStripe 65.123". See the host communications application's USB serial number property for more information.

After setting the USB product to a new value, the USB product needs to be saved with the save property command and then the device needs to be power cycled or reset before the new USB product value is used as the USB product string descriptor. If MagTek's MCP USB driver is being used, it needs to be closed and then opened using the new device instance name before communicating further.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	02	02

Field	PVAL
Byte	7 - 23
Value (Hex)	49 6E 74 65 6C 6C 69 53 74 72 69 70 65 20 36 35 00 ("IntelliStripe 65")

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

MCP ERROR RECOVERY ENABLE PROPERTY

Property ID: 03 (Hex)
Property Type: Boolean
Power Up/Reset Value: 01 (hex) (True)
Get/Set/Save support: All
Description:
This property enables or disables the MCP error recovery feature. This feature is enabled if this property is set to 1 (true) or disabled if this property is set to 0 (false).

Disabling this property will greatly reduce the development effort required to implement the MCP protocol. In fact, disabling this property will reduce the protocol to simply sending and receiving I-frames with the exception of the optional auto baud synchronization S-frame sequence. However, disabling this property will also disable the error recovery feature of the protocol. The error recovery feature of the protocol allows the protocol to automatically recover from minor communication channel problems without requiring intervention from the application layer. Users that have reliable communications channels do not need the error recovery feature, so most users do not require this feature. Users that are writing their own MCP USB drivers are encouraged to disable this property to help simplify the driver development effort. If this property is disabled and it is not made non-volatile by using the Save property command, then this property should be set as the first I-frame sent to the device after power up to avoid having to implement any of the error recovery aspects of the protocol.

Disabling this property will change the following aspects of the MCP protocol from the devices perspective.

- The protocol will no longer automatically recover from minor communications channel problems.
- The device will no longer check the send and receive sequence numbers received from the host. The host is no longer required to maintain these sequence numbers. They can always be set to zero.
- The device will no longer require the host to acknowledge I-frames that it sent. The device will assume that the host received the I-frame correctly. The host is no longer required to send R-frames. The device will never send the host a polling R-frame. The block wait timeout (BWT) is no longer used. The device will no longer send the host an R-frame if takes longer than 250ms (BWT) to begin sending a response I-frame due to a request I-frame (command) that takes some time to complete.
- The device will no longer require the host to establish a connection by sending a RESYNC S-frame request. The device will be ready to accept I-frames as soon as baud synchronization is completed.

The device will still enforce the protocols character wait timeout (CWT) and so should the host. This is required to reset the nodes receive handler in case of an error. The host should still enforce a 5 second message response timeout so that it can detect and handle errors.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	03	03

Field	PVAL
Byte	7
Value (Hex)	01 (true)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

MCP EDC CHECK ENABLE PROPERTY

Property ID: 04 (Hex)
Property Type: Boolean
Power Up/Reset Value: 01 (hex) (True)
Get/Set/Save support: All
Description:
This property enables or disables MCP EDC (error detection code) checking. This feature is enabled if this property is set to 1 (true) or disabled if this property is set to 0 (false). Disabling this property will cause the device to no longer check the header EDC and the frame EDC for R-frames and S-frames. This property has no affect on the I-frame EDC checking because this is controlled by the I-frames PCB field. When this property is disabled, the EDC fields are still required, however they can be set to any value.

Disabling this property will simplify the protocol a bit, but it will also reduce the error detection/recovery feature of the protocol. This property can be useful when sending commands to the device manually so that the EDC does not have to be calculated manually. This property can also be useful during MCP protocol driver development.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	03	04

Field	PVAL
Byte	7
Value (Hex)	01 (true)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

MCP AUTO BAUD ENABLE PROPERTY

Property ID: 05 (Hex)
Property Type: Boolean
Power Up/Reset Value: 01 (hex) (True)
Get/Set/Save support: All
Description:
This property enables or disables MCP auto baud synchronization. This feature is enabled if this property is set to 1 (true) or disabled if this property is set to 0 (false). When this property is disabled, the device will no longer require the host to perform the auto baud synchronization sequence. Instead, the device will use the fixed baud rate specified by the host communication applications fixed baud rate property. After changing this property, the property will also need to be saved in non-volatile memory (Save property command) and the device will need to be power cycled or reset before the change will take affect.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	03	05

Field	PVAL
Byte	7
Value (Hex)	01 (true)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

FIXED BAUD RATE PROPERTY

Property ID: 06 (Hex)
Property Type: Dword
Power Up/Reset Value: 0a (hex) (57600 baud)
Get/Set/Save support: All
Description:
This property contains the value of the RS232 interfaces fixed baud rate. The MCP protocol does not use this property unless its host communications applications MCP auto baud enable property is set to false. After changing this property, the property will also need to be saved in non-volatile memory (Save property command) and the device will need to be power cycled or reset before the change will take affect.

Values:

Value (hex)	Baud Rate
05	9600
06	14400
07	19200
08	28800
09	38400
0A	57600
0B	115200

Example Set Property Request:

Field	**MTYP**	**APPL**	**CMND**	**RC**	**PTYP**	**PID**
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	01	06

Field	**PVAL**
Byte	7 - 10
Value (Hex)	0a 00 00 00 (57600 baud)

Example Set Property Response:

Field	**MTYP**	**APPL**	**CMND**	**RC**
Byte	1	2	3	4
Value (Hex)	40	08	01	00

RESET DETECTED PROPERTY

Property ID: 07 (Hex)
Property Type: Boolean
Power Up/Reset Value: 01 (hex) (True)
Get/Set/Save support: Get/Set
Description:
This property can be used to determine if the device has been reset. To use this property, first set it to false, then monitor it periodically, if its value changes to true, then the device has been reset.

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	08	01	00	03	07

Field	PVAL
Byte	7
Value (Hex)	00 (false)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

PROTOCOL PROPERTY

Property ID: 08 (Hex)
Property Type: Dword
Power Up/Reset Value: 00 (hex) (MCP)
Get/Set/Save support: All
Description:
This property contains the value of the communications protocol used by this device. The communications protocol is independent of the communications interface that is used. For example, both the RS232 and USB interfaces can use any protocol. After changing this property, the property will also need to be saved in non-volatile memory (Save property command) and the device will need to be power cycled or reset before the change will take affect.

Values:

Value (hex)	Protocol
00	MCP
01	ASCII Hex

MCP Protocol:

This binary protocol is described in MagTek's MCP Serial Transport Protocol reference manual, part number 99875163. This is the default protocol used by this device. MagTek's MCP Windows Driver for the RS232 and USB interfaces uses this protocol. This protocol is rather complicated, however, it can be extremely simplified by adjusting some of the other properties found in this application such as the Error Recovery Enable property.

ASCII Hex Protocol:

This is a very simple ASCII based protocol.   This protocol is very simple to implement. For the RS232 interface, a terminal emulation application such as Window's HyperTerminal can be used to communicate to the device using this protocol. USB drivers can be designed to use this protocol if they follow the pipe structure described in the USB section of MagTek's MCP Serial Transport Protocol reference manual, part number 99875163. Since this is an ASCII based protocol used to send Binary data, it is twice as slow as the binary MCP protocol because it requires twice as many bytes to be sent.

Only the following ASCII characters are valid for this protocol. All other characters will be ignored by the device.

Value (Hex)	Code/Char
0D	Carriage Return <CR>
18	Cancel <CAN>
30-39	1-9
41-46	A-F
a-f	a-f

Command requests should be sent to the device as they are defined in this manual followed by a carriage return. Each binary byte should be sent as two ASCII hex characters. For example, the binary byte 5A (hex) should be sent as the two ASCII characters 5 (35 hex) and A (41 hex). The device will send Command responses and notification messages to the host as they are defined in this manual followed by a carriage return. The device will only send upper case letters A thru F. The host is free to also send lower case a thru f in addition to upper case A thru F.

The device is also free to send the cancel character (18 hex) at anytime to cancel anything that may be in the devices receive buffer prior to sending the carriage return. However, this cannot be used to cancel a command that is in progress. The host should wait at least 50ms before sending anything else to the device after sending this cancel character. The device will not respond to the cancel character. The cancel character could be useful when the host application first starts or after a communications error occurs to make sure that the device is in a known state.

If the device is using the ASCII protocol, the host would send the following sequence of bytes (in hex) to send the following example requrest:
30 30 30 38 30 31 30 30 30 31 30 38 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 0D

If the device is using the ASCII protocol the device would send the following sequence of bytes (in hex) to send the following example response:
34 30 30 38 30 31 30 30 0D

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID	PVAL
Byte	1	2	3	4	5	6	7 - 10
Value (Hex)	00	08	01	00	01	08	00 00 00 00 (MCP)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	08	01	00

SECTION 10. OTI SCI 1000 CONTACTLESS SMART CARD APPLICATION

Application ID: 0E (Hex)
Description:
This application deals with communications with the OTI module. This module supports MasterCard-PayPass™, Visa-Wave™ , AMEX-ExpressPay™, Discover and Mifare cards.

GET/SET/SAVE PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (Hex)
Save Property Command ID: 02 (Hex)
Description:
These commands are used to retrieve and set the value of application properties. Details of these commands are described in the Generic Command Section of this document. The following describes each property this application supports.

NOTIFY CONTACTLESS MAGSTRIPE PROPERTY

Property ID: 00 (Hex)
Property Type: Boolean
Power Up/Reset Value: 1 (ON)
Get/Set support: Both
Description:
If the Notify Contactless Magstripe state is set to ON, then a notification message will be sent to the host when a Contactless Smart Card enters the operating field.

Values:

Value	Notify Answer To Request State
0	OFF
1	ON

Example Set Property Request:

Field	**MTYP**	**APPL**	**CMND**	**RC**	**PTYP**	**PID**
Byte	1	2	3	4	5	6
Value (Hex)	00	0E	01	00	03	00
Field	**PVAL**					
Byte	7					
Value (Hex)	01 (NOTIFY ANSWER TO REQUEST STATE - ON)					

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	0E	01	00

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	0E	00	00	03	00

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	0E	00	00	03	00

Field	PVAL
Byte	7
Value (Hex)	01 (NOTIFY ANSWER TO REQUEST STATE - ON)

CONTACTLESS AUTOMATIC RF CONTROL PROPERTY

Property ID: 01 (Hex)
Property Type: Boolean
Power Up/Reset Value: 0 (OFF)
Get/Set support: Both
Description:
Due to close proximity of the antenna to the magnetic stripe read head, the RF field can prevent magnetic stripe data from being read correctly. The RF can be turned off automatically by the reader when reading magnetic stripe data in the withdrawal direction. This only works when the transport application MSR direction property is not set to non-directional. When the Contactless Automatic RF Control state is set to ON, the RF is automatically turned OFF just prior to withdrawing a card, after the card withdrawal completes, the RF will be turned back ON. If a card is partially inserted and pulled back out of the reader, the RF will turn OFF and after a timeout, will turn back ON.

The reader is not capable of automatically turning the RF off prior to a card insertion. However, this can be done manually by the host by using the RF OFF and RF ON commands to make sure the RF does interfere with a read on insertion.

If a command is sent to the reader to turn RF OFF, then the automatic control of the RF will also be disabled until a command turning ON the RF is received.

Values:

Value	AUTOMATIC RF CONTROL PROPERTY
0	OFF
1	ON

Example Set Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	0E	01	00	03	01

Field	PVAL
Byte	7
Value (Hex)	01 (AUTOMATIC RF CONTROL - ON)

Example Set Property Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	0E	01	00

Example Get Property Request:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	00	0E	00	00	03	01

Example Get Property Response:

Field	MTYP	APPL	CMND	RC	PTYP	PID
Byte	1	2	3	4	5	6
Value (Hex)	40	0E	00	00	03	01

Field	PVAL
Byte	7
Value (Hex)	01 (AUTOMATIC RF CONTROL – ON)

Auto Mode

The contactless module is always in Auto Mode and will automatically read the card’s equivalent magnetic stripe data when it comes within reading range. The reader will also sound a buzzer when the contactless card data is successfully read. This data is also sent to the host automatically.

Example Notification Sequence

On the assumption that the Notify Contactless Magstripe Property is ON, the application will send the message when a contactless card comes into the operating field.

Contactless Magstripe Notification message for Paypass, Visa Wave, ExpressPay and Discover:

On a successful read:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	80	0E	00	00

Field	DATA
Byte	5 – up to 124 (Variable Length)
Value (Hex)	DF 23 (LEN) (Track 1 Data) DF 22 (LEN) (Track 2 Data)

Contactless Magstripe Notification message for Mifare:

On a successful read:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	80	0E	00	00

Field	DATA
Byte	5 – up to 124 (Variable Length)
Value (Hex)	DF 0D (LEN) (UID)

Notes:

(1) DF 23 is the header for Track 1

(2) DF 22 is the header for Track 2

(3) LEN number of bytes available for Track 1 or Track 2.

(4) For Track 2 data, 2 characters are sent using one byte, the application software will have to convert each character into two bytes.

(5) DF 0D id the header for a Mifare Card’s UID. See appendix D to read Mifare cards.

RF ON Command

Command ID: 84 (Hex)

Description:

This command is used to turn the RF field ON.

Request Message Data: None

Response Message Data: None

Response Message Result Codes:

Success = 00 (Hex)

The command completed successfully.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	0E	84	00

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	0E	84	00

RF OFF COMMAND

Command ID: 85 (Hex)
Description:
This command is used to turn the RF field OFF. When this command is accepted by the reader, the automatic control of the RF will also be disabled
Request Message Data: None
Response Message Data: None
Response Message Result Codes:
Success = 00 (Hex)
The command completed successfully.

Example Request:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	0E	85	00

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	0E	85	00

SENDING OTI MODULE TRANSPARENT COMMANDS

Command ID: 81 (Hex)
Description:
This command is used to send additional commands to the OTI module through the IS65 interface.
Request Message Data: None
Response Message Data: None
Response Message Result Codes:
Success = 00 (Hex)

The command completed successfully.

Example Request:

This example sends the command to the OTI module to POLL and activate once.

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	00	0E	81	00

Field	DATA
Byte	5 – 10
Value (Hex)	00 3E DF 7E 01 03

Example Response:

Field	MTYP	APPL	CMND	RC
Byte	1	2	3	4
Value (Hex)	40	0E	81	00

Field	DATA
Byte	5 – 7
Value (Hex)	3E 00 00

APPENDIX A. EXAMPLE HOST APPLICATION

The following is an example of a very simple host application that uses the IntelliStripe 65 command set to read a magnetic stripe and communicate to a smart card. This example would not be used as a real application because it has no error handling. It is only intended to be used as a demonstration of how the command set can be used to build an application.

Perform the following steps:

1. Initialize the device by setting MSR Direction transport property to Withdrawal and setting MSR Arm State transport property to Armed to Read Multiple Times.

2. Wait for a card to be inserted completely into the reader. Check Card Seated indicator inside a loop until it is set. The Card Seated indicator is obtained by issuing the Get Indicators transport property.

3. Issue the Card Latch transport command to latch the card.

4. Perform any required smart card communications.

5. Issue the Card Unlatch transport command to unlatch the card.

6. Wait for the card to be completely removed from the reader. Check Present indicator inside a loop until it is clear. The Card Present indicator is obtained by issuing the Get Indicators transport property.

7. Issue the Get Track 123 Decode Data magnetic stripe command to get the magnetic stripe data.

8. Go to step 2 to read another card.

APPENDIX B. MEMORY CARD SUPPORT

This appendix defines the commands needed to control supported Memory Cards

SLE 4442/4432 MEMORY CARD CONTROL

This section defines the commands needed to control the SLE 4442 and 4432 card types. It is assumed that the user has access to the manufacturer's specifications for this type of card.

According to Siemens, this card is divided into "Memories" which each have different uses. The word Memory as used in this section (always starting with uppercase M) should be interpreted as one of the Memories defined by Siemens. For command purposes we equate each Memory with a Short File ID (read about this in ISO 7816-4 if you like). The Short File IDs are:

00 – Main Memory
01 – Security Memory (4442 only)
04 – Protection Memory

Properties

With the addition of the SLE 4442/4432 support come a few properties the user *may* need access to. If the cards you are using are true SLE 4442/4432 cards, you should not need to manipulate these properties. If you are using a clone with a different mapping, you *may* need to change these properties.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set /Save	Description
4442-ByteSizeMain	A0	Binary	2	256	Get/Set	Gives the size of Main Memory in bytes. Reads and Writes to Main Memory will be screened to assure they stay within these limits.
4442-ByteSizeSec	A1	Binary	1	4	Get/Set	Gives the size of Security Memory (4442 only). All commands directed to this Memory will be screened to assure they stay within these limits.
4442-ByteSizeProt	A2	Binary	1	32	Get/Set	Gives the size of Protection Memory. Reads and Writes to Main Memory will be screened to assure they stay within these limits.

Power Up

The Power Up command is issued as with Microprocessor cards and can get same Result Codes. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

APDU Exchange Command

The APDU Exchange Command is used to control logical operations of the card between Power Up and Power Down. The APDU is issued as with Microprocessor cards. The Result Codes are the same as for Microprocessor cards. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

Two SW1/SW2 Status Codes may apply to any APDU:

SW1/SW2	**Meaning**
6E00	CLA Not Supported
6D00	INS Not Supported

Read Binary APDU

Command APDU:

CLA	**INS**	**P1**	**P2**	**Le**
00	B0	Adr1	Adr0	Len

CLA is always 00
INS is always B0
P1 is upper byte of address as follows:
80 reads Main Memory (Short File ID 0)
81 reads Security Memory (Short File ID 1) (SLE 4442 only)
84 reads Protection Memory (Short File ID 4)
P2 is lower byte of address within the requested Memory
Le is the number of bytes to read, 0-256, 0 indicates 256 bytes

Response APDU:

Data Field (from card)	SW1	SW2

When reading from the Protection Memory, the high order bit of the first data byte returned represents the protection state of byte 0 in the Main Memory. The low order bit of the first data byte returned represents the protection state of byte 7, etc.

Possible SW1/SW2 combinations:

SW1/SW2	**Meaning**
9000	Normal completion, no errors
6A82	Memory Not Found (P1 is invalid)
6B00	Offset outside of the Memory (Combination of P2 and Le is invalid)

Update Binary APDU

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
00	D6	Adr1	Adr0	Len	Data to be written

CLA is always 00
INS is always D6
P1 is upper byte of address as follows:
80 updates Main Memory (Short File ID 0)
81 updates Security Memory (Short File ID 1) (SLE 4442 only)
84 updates Protection Memory (Short File ID 4)
P2 is lower byte of address, address within the requested Memory
Lc is the number of bytes to write, 0-256, 0 indicates 256 bytes
Data Field contains the bytes to write. If the Memory being written is the Protection Memory, only bytes that are different from the values currently on the card are written, then protected. If a byte to be written is already protected, a status 6581 will be returned.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6581	Memory Failure, some bytes were not written
6A82	File Not Found (P1 is invalid)
6B00	Offset outside of the Memory (Combination of P2 and Lc is invalid)

Verify APDU (Programmable Security Code)

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
00	20	00	00	Len	Data to be verified

CLA is always 00
INS is always 20
P1 is 00
P2 is 00
Lc is the number of bytes for verification:
If the value is 00, the status 63Cx will be returned where x indicates the number of further retries allowed. This mode does not use up an attempt!
If the value is the one less than the value of Property A1, the PSC contained in the Data Field will be presented for verification.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors (PSC verified OK)
63Cx	Verification failed, x indicates number of further retries allowed
6A86	P1 and/or P2 not zero
6B00	Offset outside of the Memory (Lc is invalid)

SLE 4428/4418 MEMORY CARD CONTROL.

This section defines the commands needed to control the SLE 4428 and 4418 card types. It is assumed that the user has access to the manufacturer's specifications for this type of card.

This section will address the memory available on the card as a file. For compatibility of terminology with ISO 7816-4, we will assume that the memory is in an "implicitly selected" file after we power up the card.

Properties

With the addition of the SLE 4428/4418 support come a few properties the user *may* need access to. If the cards you are using are true SLE 4428/4418 cards, you should not need to manipulate these properties. If you are using a clone with a different mapping, you *may* need to change these properties.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set /Save	Description
4428-AddressSecurity	A6	Binary	2	1021	Get/Set	Gives Byte Address on card of Security Area. The Security starts with a one byte Error Counter and continues with a Programmable Security Code (PSC). The PSC occupies all remaining bytes of the Security Area.
4428-LenSecurity	A7	Binary	1	3	Get/Set	Gives the length (in bytes) of the Security Area

Power Up

The Power Up command is issued as with Microprocessor cards and can get same Result Codes. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

APDU Exchange Command

The APDU Exchange Command is used to control logical operations of the card between Power Up and Power Down. The APDU is issued as with Microprocessor cards. The Result Codes are the same as for Microprocessor cards. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

Two SW1/SW2 Status Codes may apply to any APDU:

SW1/SW2	Meaning
6E00	CLA Not Supported
6D00	INS Not Supported

Read Binary APDU

Command APDU:

CLA	INS	P1	P2	Le
00	B0	Adr1	Adr0	Len

CLA is always 00
INS is always B0
P1 is upper byte of address to read and must not be greater than ‘03’
P2 is lower byte of address to read.
Le is the number of bytes to read

Response APDU:

Data Field (from card)	SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6B00	Offset outside of the Memory (Combination of P2 and Le is invalid)

Read Protection Status APDU

This APDU will not be found in ISO 7816-4. That is why it has the CLA of D0 (Proprietary).

Command APDU:

CLA	INS	P1	P2	Le
D0	B8	Adr1	Adr0	Len

CLA is always D0
INS is always B8
P1 is upper byte of address to read protection status and must not be greater than ‘03’
P2 is lower byte of address to read protection status.
Le is the number of bytes of protection status to read

Response APDU:

Data Field (from card)	SW1	SW2

Data indicates the protection status for the byte located at the corresponding address on the card. The value '00' indicates the location is protected and cannot be updated. A value '01' indicates the location is not protected and can be updated.

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6B00	Offset outside of the Memory (Combination of P2 and Le is invalid)

Update Binary APDU

Because of card functionality, this APDU should not be sent until at least one Read Binary APDU has been sent first.

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
00	D6	Adr1	Adr0	Len	Data to be written

CLA is always 00
INS is always D6
P1 is upper byte of address to write and must not be greater than '03'
P2 is lower byte of address to write.
Lc is the number of bytes to write, 0-1024
Data Field contains the bytes to write. If a byte to be written is already protected, a status 6581 will be returned.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6581	Memory Failure, some bytes were not written
6B00	Offset outside of the Memory (Combination of P2 and Lc is invalid)

Update Binary With Protection APDU

Because of card functionality, this APDU should not be sent until at least one Read Binary APDU has been sent first.

This APDU will not be found in ISO 7816-4. That is why it has the CLA of D0 (Proprietary).

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
D0	D8	Adr1	Adr0	Len	Data to be written

CLA is always D0
INS is always D8
P1 is upper byte of address to write and must not be greater than '03'
P2 is lower byte of address to write.
Lc is the number of bytes to write, 0-1024
Data Field contains the bytes to write. If a byte to be written is already protected, a status 6581 will be returned. Data is written first, then the Protect bit is written with comparison, assuring that the data being protected is the data you wanted to protect.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6581	Memory Failure, some bytes were not written
6B00	Offset outside of the Memory (Combination of P2 and Lc is invalid)

Verify APDU (Programmable Security Code)

Because of card functionality, this APDU should not be sent until at least one Read Binary APDU has been sent first.

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
00	20	00	00	Len	Data to be verified

CLA is always 00
INS is always 20
P1 is 00
P2 is 00
Lc is the number of bytes for verification:

If the value is 00, the status 63Cx will be returned where x indicates the number of further retries allowed. This mode does not use up an attempt!

If the value is the one less than the value of Property A7, the PSC contained in the Data Field will be presented for verification.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors (PSC verified OK)
63Cx	Verification failed, x indicates number of further retries allowed
6A86	P1 and/or P2 not zero
6B00	Offset outside of the Memory (Lc is invalid)

SLE 4404 MEMORY CARD CONTROL.

This section defines the commands needed to control the SLE 4404 card type. There are many cards available that are similar to the SLE4404, and this device will support many of them. It is assumed that the user has access to the manufacturer's specifications for this type of card.

This section will address the memory available on the card as a file. For compatibility of terminology with ISO 7816-4, we will assume that the memory is in an "implicitly selected" file after we power up the card.

Properties

With the addition of the SLE 4404 support comes a property the user *may* need access to.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set/ Save	Description
4404-BitOrder	AE	Boolean	1	0	Get/Set	A value of 0 causes the least significant bit of each byte to be sent or received first. A value of 1 causes the most significant bit of each byte to be sent or received first. This property DOES NOT APPLY to the Power Up command, which always reads the bytes least significant bit first.

Data is sent and received as bytes. When the 4404-BitOrder property is set to 0, the least significant bit of each byte represents the lowest addressed bit on the card. If we write the bytes 33 22 11 to a card at address 73, the card will end up with the following bit pattern on the card:

```
Address
73……….81……….89…….96
11001100 01000100 10001000
```

When the 4404-BitOrder property is set to 1, the most significant bit of each byte represents the lowest addressed bit on the card. If we write the bytes 33 22 11 to a card at address 73, the card will end up with the following bit pattern on the card:

Address
73……….81……….89…….96
00110011 00100010 00010001

Note: This property is new. Older versions of IntelliStripe products may not have this. The addition of this property is backwards compatible with such older products. Compatibility is achieved by having the Reset Value provide the same functionality as the older products. If you wish to use the msb first option, you will have to set this property to 1.

Power Up

The Power Up command is issued as with Microprocessor cards and can get same Result Codes. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

APDU Exchange command

The APDU Exchange Command is used to control logical operations of the card between Power Up and Power Down. The APDU is issued as with Microprocessor cards. The Result Codes are the same as for Microprocessor cards. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

Two SW1/SW2 Status Codes may apply to any APDU:

SW1/SW2	Meaning
6E00	CLA Not Supported
6D00	INS Not Supported

Special handling for C4 and RST

Some of the cards that are controllable with as this card type use Contact 4 (C4) and the Contact 2 (RST) for special functions. For some cards, C4 may be used to put the card into a special “test” mode where it simulates an already personalized card, even if it has not completed personalization. For some cards, RST may be used during write/erase operations to stop the card internal address from incrementing.

The user is given control of C4 for all APDUs. This is accomplished by manipulating the lowest bit of the CLA byte. If the bit is 0, C4 will be maintained Low for the duration of the operation. If the bit is 1, C4 will be set High at the start of the operation and then returned to Low after the operation is completed.

Some of the APDUs give the user control over the RST contact. For these APDUs a value of 0 will force Contact 2 low, a value of 1 will force it high during selected portions of the operation.

It is expected that the user has access to the technical specifications of the card he is dealing with and thus knowledge of how to use these contacts.

Read Binary APDU

Command APDU:

CLA	INS	P1	P2	Le
D0/D1	B0	Xx	Adr	Len

CLA is always D0 or D1 (the lowest bit specifies the state of C4)
INS is always B0
P1 is ignored
P2 is the address where the read will start
Le is the number of bytes to read, 0 will read 256 bytes

Response APDU:

Data Field (from card)	SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors

Write Binary APDU

The Write Binary command performs a logical AND of the bits already present in the card with the bits given in the command APDU. For each bit that is 0 in the command APDU, if the corresponding bit on the card is 1, it will be written to 0. Bits in the command APDU that are set to 1 will be ignored, the state of the corresponding bit on the card will not be checked. If there is a need to set a bit on the card to the value of 1, it must be done using the Erase Binary command or the Verify command. These commands will cause erasure of more than one bit on the card, so be careful!

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
D0/D1	D0	See below	Adr	Len	Data to be written

CLA is always D0 or D1 (the lowest bit specifies the state of C4)
INS is always D0
P1 specifies RST and allows bit addressing within the specified starting byte. The bits shown with x are ignored.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RST	x	x	x	x	Bit address		

The bit address is added to the byte address to form the address of the first bit to be affected by the command. For every count over zero, the card internal address in incremented one. This allows effective addressing of a specific bit and may be useful for blowing fuses in some cards.

P2 is the address where the write will start
Lc is the number of bytes to write.
Data Field contains the bytes to write. If a bit to be written cannot be changed, a status 6581 will be returned.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6581	Memory Failure, some bits were not written

Erase Binary APDU

The Erase Binary command is used to erase parts of the card. For this type of card erased means that each bit = 1. Cards of this type are organized as 16 bit "frames". Typically, if an erase operation is performed on any bit in the frame, all bits in the frame are erased. In order to erase successfully, security conditions may have to be met.

This command provides one of two methods for erasing data on the card. The other method is with the Verify command. When it is used to "verify" the erase password of one of the application zones, it causes that application zone to be erased (if security requirements have been satisfied).

Command APDU:

CLA	INS	P1	P2	Lc
D0/D1	0E	RST	Adr	0

CLA is always D0 or D1 (the lowest bit specifies the state of C4)
INS is always 0E
P1 specifies RST, all other bits are ignored

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RST	x	x	x	x	x	x	x

P2 is the address where the erase will occur
Lc should be zero, but is ignored (0 is used) if it is not.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6581	Memory Failure, some bits were not erased

Verify / Erase Application Zone APDU

This command is used to do two types of security operations on the card. First, it is used to present the User Code which unlocks further functionality. Second, it is used to erase/reload application zones by presenting the Memory Code.

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
D0/D1	20/22	Tries	Adr	Len	Data to be verified

CLA is always D0 or D1 (the lowest bit specifies the state of C4)

INS is either 20(Verify) or 22(Erase Application Zone)

P1 specifies Tries, the maximum number bits which will be searched for a ‘1’ value to write to ‘0’ in preparation for the erase sub-operation that completes the Verify/Erase command. For application zones that do not require writing of a bit to cause erasure, use a 00 in Tries. This will cause the erase operation to be performed on the next bit after presentation of the Memory Code.

P2 is the byte address of the code to be verified (usually called User Code or Memory Code). *NOTE: WHEN USING Lc=0, THIS ADDRESS MUST BE THE ADDRESS OF THE COUNTER TO BE EVALUATED (I.E. IN A 4404 THE USER CODE IS AT ADDRESS 08H AND THE CORRESPONDING ERROR COUNTER IS AT ADDRESS 0AH. TO EVALUATE THIS COUNTER THE COMMAND APDU WOULD BE:*
D0 20 04 0A 00).

Lc is the number of bytes for verification:

If the value is 00, the status 63Cx will be returned where x indicates the number of further retries allowed, if x is F, there may be even more than 15 retries remaining. This mode does not use up any retries!

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion for Verify (INS 20). Verification was successful.
6200	Normal completion for Erase Application Zone (INS 22), no errors, but user must verify erasure. This status could be returned even though erasure did not occur.
63Cx	Verification failed, x indicates number of further retries allowed

I^2C™ MEMORY CARD CONTROL

This section defines the commands needed to control basic I^2C™ types of cards. There are two types of cards addressed, dependent on the amount of memory they contain.

We call the first type the 3 byte card because a 3 byte command frame is used to control it. These cards must contain 2048 bytes or less of data (byte address must fit in 11 bits). At this time we do not know of any cards containing 2048 or less bytes of data that use a 4 byte command frame, but if such a card is developed later, it would be classed as a 4 byte card.

We call the second type the 4 byte card because a 4 byte command frame is used to control it. These cards usually contain more than 2048 bytes of data and thus need an additional byte of address in the control frame.

If you are not sure which type of card you have, check the manufacturer's specifications to see how many bytes are used in the control frame for Byte Write, it will be either 3 or 4.

Properties

Support of I^2C™ cards requires one property to specify the frame size. The default value for this property is to support the 3 byte cards. If you wish to work with the 4 byte cards, you should change this property.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set/ Save	Description
I^2C-3ByteCard	AA	Boolean	1	TRUE	Get/Set	TRUE specifies a 3 byte control frame, FALSE specifies a 4 byte control frame.

Power Up

The Power Up command is issued as with Microprocessor cards and can get same Result Codes. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

Note

When the Primary Status in the CondRpt Property is 0x82, a Secondary Status of 0x05 indicates that the card is not responding. This Secondary Status applies only to I^2C™ card handling.

A 4 byte ATR is returned when the power up command is successful. Many I^{2}C™ cards do not deliver an ATR on power up, but some do. If the value FFFFFFFFH is returned, the card did not deliver an ATR. The presence or absence of an ATR may not positively identify the card in the reader. If the ATR is different from FFFFFFFFH it probably does identify the card, see specifications for the various cards to determine which ATR identifies which card. If FFFFFFFFH is returned, the card may be an I^{2}C™ card that does not deliver an ATR. The only way to be sure is to send a Read Binary to retrieve known data from a card you are familiar with.

APDU Exchange Command

The APDU Exchange Command is used to control logical operations of the card between Power Up and Power Down. The APDU is issued as with Microprocessor cards. The Result Codes are the same as for Microprocessor cards. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

Two SW1/SW2 Status Codes may apply to any APDU:

SW1/SW2	Meaning
6E00	CLA Not Supported
6D00	INS Not Supported
9000	Normal completion, no errors

Read Binary APDU

Command APDU:

CLA	INS	P1	P2	Le
00	B0	Adr1	Adr0	Len

CLA is always 00
INS is always B0
P1 is upper byte of address to read.
P2 is lower byte of address to read.
Le is the number of bytes to read

Note that if the address given is outside of the bounds of the card in the reader you may not get the expected results. If the sum of Le and the address is outside of the bounds of the card, some of the data will probably represent a wrap-around to the beginning of card memory.

Response APDU:

Data Field (from card)	SW1	SW2

Update Binary APDU

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
00	D6	Adr1	Adr0	Len	Data to be written

CLA is always 00
INS is always D6
P1 is upper byte of address to write.
P2 is lower byte of address to write.
Lc is the number of bytes to write.
Data Field contains the bytes to write.

Note that if the address given is outside of the bounds of the card in the reader you may not get the expected results. If the sum of Le and the address is outside of the bounds of the card, some of the data will probably represent a wrap-around to the beginning of card memory.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
6581	Memory Failure, some bytes were not written

GEMPLUS GAM326 MEMORY CARD CONTROL.

This section defines the commands needed to control the Gemplus GAM326 card type. There may be other cards that are similar to the GAM326, and this device may support some of them. It is assumed that the user has access to the manufacturer's specifications for this type of card.

This section will address the memory available on the card as a file. For compatibility of terminology with ISO 7816-4, we will assume that the memory is in an "implicitly selected" file after we power up the card.

Properties

With the addition of the GAM326 support come properties the user *may* need access to.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set/ Save	Description
GAM326-BitOrder	B2	Boolean	1	1	Get/Set	A value of 0 causes the least significant bit of each byte to be sent or received first. A value of 1 causes the most significant bit of each byte to be sent or received first. This bit order applies to all operations, including the ATR returned from the Power Up command.
GAM326 value of M	B3	Dword	1	160	Get/Set	This is the value of M to be used in the Internal Authenticate command, see the card manufacturer's documentation for more information. A value of 0 is interpreted as 256.
GAM326 Counter Lowest Address	B4	Dword	1	8	Get/Set	This is the lowest address that will be accepted in an Erase Counter command.
GAM326 Counter Highest Address	B5	Dword	1	11	Get/Set	This is the highest address that will be accepted in an Erase Counter command.

Data is sent and received as bytes. When the GAM326-BitOrder property is set to 0, the least significant bit of each byte represents the lowest addressed bit on the card. If we write the bytes 33 22 11 to a card at address 72, the card will end up with the following bit pattern on the card:

```
Address
72……….80……….88…….95
11001100 01000100 10001000
```

When the GAM326-BitOrder property is set to 1, the most significant bit of each byte represents the lowest addressed bit on the card. If we write the bytes 33 22 11 to a card at address 72, the card will end up with the following bit pattern on the card:

```
Address
72……….80……….88…….95
00110011 00100010 00010001
```

Power Up

The Power Up command is issued as with Microprocessor cards and can get same Result Codes. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

APDU Exchange command

The APDU Exchange Command is used to control logical operations of the card between Power Up and Power Down. The APDU is issued as with Microprocessor cards. The Result Codes are the same as for Microprocessor cards. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

Two SW1/SW2 Status Codes may apply to any APDU:

SW1/SW2	Meaning
6E00	CLA Not Supported
6D00	INS Not Supported

Read Binary APDU

Command APDU:

CLA	INS	P1	P2	Le
00	B0	xx	Adr	Len

CLA is always 00
INS is always B0
P1 is ignored
P2 is the address where the read will start
Le is the number of bytes to read, 0 will read 256 bytes

The entire card can be read with one command. Bits that are “unreadable” will be returned with the electrical value the card imposes on the I/O line at the time of read sampling.

Response APDU:

Data Field (from card)	SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors

Write Binary APDU

The Write Binary command performs a logical AND of the bits already present in the card with the bits given in the command APDU. For each bit that is 0 in the command APDU, if the corresponding bit on the card is 1, it will be written to 0. Bits in the command APDU that are set to 1 will be ignored, the state of the corresponding bit on the card will not be checked. Even though each specified bit will be written, the result of the operation is NOT verified and no errors are reported. The user may desire to Read the data to assure all intended bits have been written.

If there is a need to set a bit on the card to the value of 1, it must be done using the Erase Counter command. These commands may cause erasure of more than one bit on the card, so be careful!

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
00	D0	xx	Adr	Len	Data to be written

CLA is always 00
INS is always D0
P1 is ignored
P2 is the address where the write will start
Lc is the number of bytes to write.
Data Field contains the bytes to write.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors

Erase Counter APDU

The Erase Counter command is used to erase counter bytes of the card. For this type of card erased means that each bit = 1.

If the specified address is not in the range specified by the Counter Lowest Address and the

Counter Highest Address properties, the command will not be processed.

This command writes one bit of the addressed byte from 1 to 0 (if there are any 1 bits available) under control of the Erasure Mask. If a bit was actually written to 0 in the first step, the byte at the next address on the card is erased.

The Erasure Mask is formulated to indicate one or more bits that can be written to zero to initiate the erasure. Erasure will only be initiated if at least one zero bit in the Erasure Mask corresponds to a one bit on the card. The first correspondence will initiate erasure and subsequent correspondences will be written to zero until the entire Erasure Mask is processed.

The GAM326-BitOrder property dictates the order of bit processing.

If there is a need to erase multiple bytes of the Counter, multiple Erase Counter commands must be sent in the proper order. Determining the order is the user's responsibility.

Command APDU:

CLA	INS	P1	P2	Lc	Data
D0	0A	xx	Adr	01	Erasure Mask

CLA is always D0
INS is always 0A
P1 is ignored
P2 is the address where the erase will occur
Lc **must** be 01
Data contains one byte giving the Erasure Mask.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6300	No bits to write, couldn't erase
6700	Lc gives incorrect length
6B00	P2 is incorrect

Internal Authenticate APDU

The Internal Authenticate is used to authenticate the card to the hosting terminal.
If the Internal Authenticate is performed with CBC enabled, the CBC mode remains in effect until the card is powered down.

The specified address is a *bit* address. It specifies the address of the bit where a Dummy Write is performed to start the Authentication operation. For more details, see the Gemplus documentation.

In the Authentication operation a value called M is used. This value is specified by agreement with Gemplus. The user has control over this value via the GAM326 value of M property

Command APDU:

CLA	INS	P1	P2	Lc	Data	Le
D0	88	CBC / Address	Address	06	xx xx xx xx xx xx	02

CLA is always D0
INS is always 88
P1 gives an indication of Cipher Block Chaining and the upper bits of the *bit* address to be used to start the Authentication operation:

Bits	Usage
0	CBC indication, 0 = no CBC, 1 = CBC
1-7	Address bits

P2 is the low order byte of the *bit* address used to start the Authentication operation.
Lc gives the length of the data to be used as the Random Number in the Authentication operation, its value must always be 6.
Data is the Random Number to be used in the Authentication operation.
Le gives the length of the expected reply, which is always 2.

Response APDU:

MAC (2 bytes)	SW1 = 90	SW2 = 00

OR

SW1 = 67	SW2 = 00

If SW1/SW2 indicates normal completion (9000), the MAC field is 2 bytes long and is the result of the Authentication. If SW1/SW2 indicates a wrong length, the MAC is not included in the message

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6700	Wrong length, Lc does not equal 6 or Le does not equal 2

GEMPLUS GAM375 MEMORY CARD CONTROL.

This section defines the commands needed to control the Gemplus GAM375 card type. There may be other cards that are similar to the GAM375, and this device may support some of them. It is assumed that the user has access to the manufacturer’s specifications for this type of card.

This section will address the memory available on the card as a file. For compatibility of terminology with ISO 7816-4, we will assume that the memory is in an “implicitly selected” file after we power up the card.

Properties

With the addition of the GAM375 support come properties the user *may* need access to.

Name	ID (Hex)	Type	Length	Reset Value	Get/Set/ Save	Description
GAM375-BitOrder	B8	Boolean	1	1	Get/Set	A value of 0 causes the least significant bit of each byte to be sent or received first. A value of 1 causes the most significant bit of each byte to be sent or received first. This bit order applies to all operations, including the ATR returned from the Power Up command.
GAM375 Counter Lowest Address	B9	Dword	1	8	Get/Set	This is the lowest address that will be accepted in an Erase Counter command.
GAM375 Counter Highest Address	BA	Dword	1	11	Get/Set	This is the highest address that will be accepted in an Erase Counter command.

Data is sent and received as bytes. When the GAM375-BitOrder property is set to 0, the least significant bit of each byte represents the lowest addressed bit on the card. If we write the bytes 33 22 11 to a card at address 72, the card will end up with the following bit pattern on the card:

```
Address
72……….80……….88…….95
11001100 01000100 10001000
```

When the GAM375-BitOrder property is set to 1, the most significant bit of each byte represents the lowest addressed bit on the card. If we write the bytes 33 22 11 to a card at address 72, the card will end up with the following bit pattern on the card:

```
Address
72……….80……….88…….95
00110011 00100010 00010001
```

Power Up

The Power Up command is issued as with Microprocessor cards and can get same Result Codes. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

APDU Exchange command

The APDU Exchange Command is used to control logical operations of the card between Power Up and Power Down. The APDU is issued as with Microprocessor cards. The Result Codes are the same as for Microprocessor cards. When the Result Code indicates an Error or Warning, the Primary Status of the CondRpt Property will never be 0x03.

Two SW1/SW2 Status Codes may apply to any APDU:

SW1/SW2	Meaning
6E00	CLA Not Supported
6D00	INS Not Supported

Read Binary APDU

Command APDU:

CLA	INS	P1	P2	Le
00	B0	xx	Adr	Len

CLA is always 00
INS is always B0
P1 is ignored
P2 is the address where the read will start
Le is the number of bytes to read, 0 will read 256 bytes

The entire card can be read with one command. Bits that are “unreadable” will be returned with the electrical value the card imposes on the I/O line at the time of read sampling.

Response APDU:

Data Field (from card)	SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors

Write Binary APDU

The Write Binary command performs a logical AND of the bits already present in the card with the bits given in the command APDU. For each bit that is 0 in the command APDU, if the corresponding bit on the card is 1, it will be written to 0. Bits in the command APDU that are set to 1 will be ignored, the state of the corresponding bit on the card will not be checked. Even though each specified bit will be written, the result of the operation is NOT verified and no errors are reported. The user may desire to Read the data to assure all intended bits have been written.

If there is a need to set a bit on the card to the value of 1, it must be done using the Erase Counter or Erase Data command. These commands may cause erasure of more than one bit on the card, so be careful!

Command APDU:

CLA	INS	P1	P2	Lc	Data Field
00	D0	00	Adr	Len	Data to be written

CLA is always 00
INS is always D0
P1 is ignored
P2 is the address where the write will start
Lc is the number of bytes to write.
Data Field contains the bytes to write.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors

Erase Counter APDU

The Erase Counter command is used to erase counter bytes of the card. For this type of card erased means that each bit = 1.

If the specified address is not in the range specified by the Counter Lowest Address and the Counter Highest Address properties, the command will not be processed.

This command writes bits of the addressed byte to 0 under control of the Erasure Mask. The Erasure Mask is formulated to indicate one or more bits that can be written to zero and then erasure initiated. If there are multiple zero bits in the Erasure Mask, multiple write to zero followed by erasure operations will occur (all to the same byte).

The GAM375-BitOrder property dictates the order of bit processing.

If there is a need to erase multiple bytes of the Counter, multiple Erase Counter commands must be sent in the proper order. Determining the order is the user's responsibility.

Command APDU:

CLA	INS	P1	P2	Lc	Data
D0	0A	xx	Adr	01	Erasure Mask

CLA is always D0
INS is always 0A
P1 is ignored
P2 is the address where the erase will occur
Lc **must** be 01
Data contains one byte giving the Erasure Mask.

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6300	No bits to write, couldn't erase
6700	Lc gives incorrect length
6B00	P2 is incorrect

Erase Binary APDU

The Erase Binary command is used to erase specific bytes in the card. For this type of card erased means that each bit = 1.

If the bytes specified by Adr and Cnt is in the range specified by the Counter Lowest Address and the Counter Highest Address properties, the command will not be processed. **Warning: the GAM375 address can "wrap" around. If you specify Adr = 0x28 and Cnt = 0x28 you will wrap around to the start of the card and end up erasing the counter bytes!** There are many possible combinations of Adr and Cnt that could result in a wrap around erasing the counter bytes, so be careful.

This command erases each byte starting at the specified address for as many bytes as specified by Cnt.

Command APDU:

CLA	INS	P1	P2	Lc	Data
D0	0E	Cnt	Adr	00	

CLA is always D0
INS is always 0A
P1 gives Cnt, which is the count of bytes to be erased
P2 gives Adr, which is the address where the erase will occur
Lc **must** be 00
Data is empty

Response APDU:

SW1	SW2

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6300	No bits to write, couldn't erase (P1/Cnt was zero)
6700	Lc gives incorrect length
6B00	P1 or P2 is incorrect

Internal Authenticate APDU

The Internal Authenticate is used to authenticate the card to the hosting terminal.

Command APDU:

CLA	INS	P1	P2	Lc	Data	Le
D0	88	00	00	04	xx xx xx xx	01

CLA is always D0
INS is always 88
P1 is always 00
P2 is always 00
Lc gives the length of the data to be used as the Random Number in the Authentication operation, its value must always be 4.
Data is the Random Number to be used in the Authentication operation.
Le gives the length of the expected reply, which is always 1.

Response APDU:

MAC (1 byte)	SW1 = 90	SW2 = 00

OR

SW1 = 67	SW2 = 00

If SW1/SW2 indicates normal completion (9000), the MAC field is 1 byte long and is the result of the Authentication. If SW1/SW2 indicates a wrong length, the MAC is not included in the message

Possible SW1/SW2 combinations:

SW1/SW2	Meaning
9000	Normal completion, no errors
6700	Wrong length, Lc does not equal 4 or Le does not equal 1

APPENDIX C. MIFARE COMMANDS

Tag: DF 78
Len: VAR
Scope: DO
Object description: This object is used in order to send MIFARE command to the PICC. The Data consists of a 1-byte command Number (Op Code) and optional data bytes. Response will be in the same format including the Op Code before optional data.

Please note: When referring to “Mode” in the Table below, please use the scheme represented here:
Mode (binary) 0 0 0 0 0 AB KS1 KS2
AB – 0 authentication with key A, AB – 1 authentication with key B
KS1, KS2 key set: 00-set 0, 01-set1, 10-set2

Table 11 Mifare OPCODEs

OpCode	
A5	Name: Read
	Description: Read block command
	Data structure: Op Code (one byte), Block # (one byte)
	Data Response structure: Op Code (one byte), 16 bytes of Read block Data
A6	Name: Write
	Description: Write block (16 bytes) command
	Data structure: Op Code (one byte), Block # (one byte), 16 bytes of Write to block Data
	Data Response structure: Op Code (one byte)
A7	Name: Write Ultra Light
	Description: Write 4 bytes into selected Ultra Light block
	Data structure: Op Code (one byte), Block # (one byte), 4 bytes of Write to block Data
	Data Response structure: Op Code (one byte)
A9	Name: Load Key
	Description: Replace and load new key
	Data structure: Op Code (one byte), Mode (one byte), Sector # (one byte), Transport key (6 bytes), New key (6 bytes)
	Data Response structure: Op Code (one byte)
AA	Name: Increment
	Description: Increment value of block in RAM
	Data structure: Op Code (one byte), Block # (one byte), Value (4 bytes)
	Data Response structure: Op Code (one byte)

OpCode	
AB	Name: Decrement
	Description: Decrement value of block in RAM
	Data structure: Op Code (one byte), Block # (one byte), Value (4 bytes)
	Data Response structure: Op Code (one byte)
AC	Name: Transfer
	Description: Move RAM value to EE
	Data structure: Op Code (one byte), Block # (one byte)
	Data Response structure: Op Code (one byte)
AD	Name: HALTA
	Description: ISO 14443-3 Type A HALTA command in short frame.
	Data structure: Op Code (one byte)
	Data Response structure: Op Code (one byte)
AE	Name: Restore
	Description: Restore recent value of block
	Data structure: Op Code (one byte), Block # (one byte)
	Data Response structure: Op Code (one byte)
AF	Name: Decrement & Transfer
	Description: Implement the Decrement & Transfer functions together
	Data structure: Op Code (one byte), Block # (one byte), Value (4 bytes)
	Data Response structure: Op Code (one byte)
B0	Name: Authentication2
	Description: Authentication versus a PICC
	Data structure: Op Code (one byte), Mode (one byte), Key # (one byte), Sector # (one byte)
	Data Response structure: Op Code (one byte)
C0	Name: Read Macro
	Description: Enables Read operation of Multiple Blocks of one Sector
	Data structure: Op Code (one byte), Address (one byte) Bits 0-3 for Sector Number, Bits 4-7 for Block Mask (Bit 4 for Block 0 up to Bit 7 for Block 3) Auth2KeySel (one byte) 0 – Use Key A 1 – Use Key B
	Data Response structure: Op Code (one byte) Data Sequential Data of all requested Blocks, from Lowest to Highest - Each Block 16 bytes length.

OpCode	
C1	Name: Write Macro
	Description: Enables Write operation of Multiple Blocks of one Sector
	Data structure: Op Code (one byte), Address (one byte) Bits 0-3 for Sector Number, Bits 4-7 for Block Mask (Bit 4 for Block 0 up to Bit 7 for Block 3) Auth2KeySel (one byte) 0 – Use Key A 1 – Use Key B Data (16,32,48,64 bytes) Sequential Data to Write of all requested Blocks, from Lowest to Highest - Each Block must be 16 bytes length.
	Data Response structure: Op Code (one byte) Approve completion of Write operation
Other	RFU

EXAMPLE:

To use a Mifare 1K Classic Card

(1) Load the key for the sector that will be used into the OTI module, this needs to be performed only once, it gets saved in the contactless reader's flash memory.

(2) Send a POLL command in the ACTIVATE ONCE only mode. Note that this will disable automatic reading of cards that have built-in applications (PayPass, VisaWave, ExpressPay, Discover)

(3) After a card is detected, the UID will be sent back by the reader.

(4) Send the AUTHENTICATE A command for the sector you want to read.

(5) Read BLOCK.

Follow the same steps above for writing a block.

Examples:
Sending Load Transport key "BD DE 6F 37 83 83" and New key "FF FF FF FF FF FF" for Sector 0:
00 0E 81 00 3E DF 78 0F A9 00 00 BD DE 6F 37 83 83 FF FF FF FF FF FF

Send POLL – Activate Once:
00 0E 81 00 3E DF 7E 01 03

After a Mifare Card is detected, the reader will respond with the Mifare card's Unique ID in the DATA field:
3E FF 01 0D FC 0B DF 16 01 4D DF 0D 04 XX XX XX XX

UID Tag = DF 0D
Len = 04

Send the Authenticate A command for sector 0:
00 0E 81 00 3E DF 78 04 B0 00 00 00

Send the Read Block 0 command:
00 0E 81 00 3E DF 78 02 A5 00

Send the Read Block 2 command:
00 0E 81 00 3E DF 78 02 A5 02

Send the Write Block 2 command:
00 0E 81 00 3E DF 78 12 A6 02 00 00 00 00 00 00 00 00 11 11 11 11 11 11 11 11

INDEX