INTELLISTRIPE 310
COMMAND REFERENCE MANUAL

Manual Part Number 99875218 Rev 3

JUNE 2003

MAGTEK®

REGISTERED TO ISO 9001:2000

20725 South Annalee Avenue
Carson, CA 90746
Phone: (310) 631-8602
FAX: (310) 631-3956
Technical Support: (651) 415-6800
www.magtek.com

Copyright© 2003
MagTek®, Inc.
Printed in the United States of America

Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of MagTek, Inc.

MagTek is a registered trademark of MagTek, Inc.

IntelliStripe is a registered trademark of MagTek, Inc.

REVISIONS

Rev Number	Date	Notes
1	4 Mar 02	Initial Release
2	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.
3	06 Jun 03	Front Matter: added ISO line to logo, changed Tech Support phone number, and replaced warranty with generic license.

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 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 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 to be free from defects in material or workmanship under normal use. THE SOFTWARE IS PROVIDED AS IS WITHOUT 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 insure 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

TABLE OF FIGURES

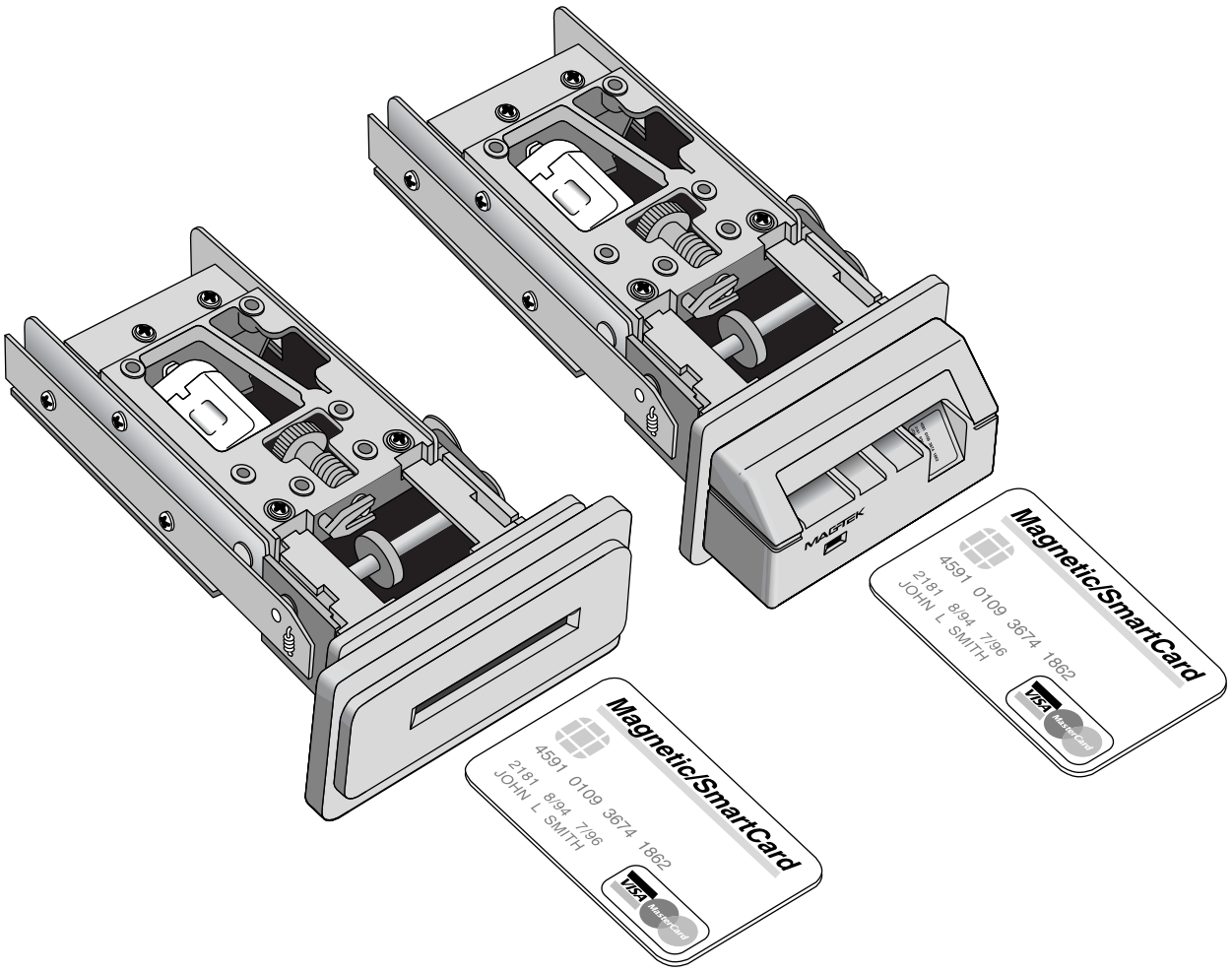

Figure 1. IntelliStripe 310 with “D” and “C” Bezels

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 (for further information related to the transport mechanism, please refer to MagTek document part number 99875163).

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.
5-0	–	Reserved for future use.

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 (e.g., 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 a 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.

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.

SECTION 3. DEVICE APPLICATION

Application ID: 00 (Hex)
Description:

This application deals with device functionality.

GET/SET PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)

Set Property Command ID: 01 (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.

MODEL NUMBER PROPERTY

Property ID: 00 (Hex)
Property Type: String
Maximum Length: 33 (including terminating zero)
Power Up/Reset Value: "IntelliStripe 310"
Get/Set support: Get
Description:

This property is used to get the device's model number.

Values:

The value is fixed at "IntelliStripe 310".

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 33 31 30 00 "IntelliStripe 310" (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 support: Get
Description:

This property is used to get the device's software ID.

Values:

The value is fixed and is software dependent. For example, the software ID could be "16051315A02" where "16051315" is the software part number, "A" is the software revision and "02" 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
Byte	1	2	3	4	5	6
Value (Hex)	40	00	00	00	02	01

Field	PVAL
Byte	7 - 18
Value (Hex)	31 36 30 35 31 33 31 35 41 30 31 00 "16051315A01" (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 on state. The device should be allowed time to power up (we recommend three seconds) before trying to communicate with the device.

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

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 PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (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 READ STATE PROPERTY

Property ID: 00 (Hex)
Property Type: Dword
Power Up/Reset Value: 0 (OFF)
Get/Set support: Both
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
Byte	1	2	3	4	5	6
Value (Hex)	00	01	01	00	01	00

Field	PVAL
Byte	7 - 10
Value (Hex)	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 support: Both
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
Byte	1	2	3	4	5	6
Value (Hex)	00	01	01	00	01	01

Field	PVAL
Byte	7 - 10
Value (Hex)	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 support: Both
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
Byte	1	2	3	4	5	6
Value (Hex)	00	01	01	00	03	15

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	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
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)	00	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
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)	00	01	82	00	02	00

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

SECTION 5. LED APPLICATION

Application ID: 81 (Hex)
Description:

This application deals with the host controlled LED.

GET/SET PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (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.

LED STATE PROPERTY

Property ID: 00 (Hex)
Property Type: Dword
Power Up/Reset Value: 0 (OFF)
Get/Set support: Both
Description:

Changing this property changes the LED state. The LED state can be retrieved by getting this property.

Values:

Value	LED State
0	OFF
1	RED
2	GREEN

Example Set Property Request:

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 00 00 00 (GREEN)

Example Set Property Response:

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

SECTION 6. 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 PROPERTY COMMANDS

Get Property Command ID: 00 (Hex)
Set Property Command ID: 01 (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.

INDICATORS PROPERTY

Property ID: 00 (Hex)
Property Type: Dword
Power Up/Reset Value: Dependent on indicator states
Get/Set 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	Front Card Present	Set to 1 if a card is present anywhere in the front (insertion) end of the card transport, or else cleared to 0. This bit can be used to indicate the beginning of a card insertion or the end of a card withdrawal. Note that once a standard sized card is transported completely to the rear of the reader, this indicator is no longer set.
1	Middle Card Present	Set to 1 if a card is present anywhere in the middle of the card transport, or else cleared to 0. This bit can be used to indicate that a card is in a position from which it can be ejected. If this bit is set, then a standard sized card should be in contact with the transport rollers and thus be in a position from which it can be ejected. Note that this bit is still set once a standard sized card is transported completely to the rear of the reader.
2	Rear Card Present	Set to 1 if a card is present at the rear of the card transport else cleared to 0. This bit can be used to indicate that the transport has completely transported the card into the reader.
3	Auto Transporting	Set to 1 if the transport is transporting a card automatically, or else cleared to 0. The transport is automatically transporting when it transports a card due to an auto transport property being set such as the Auto Consume property. The Auto Transporting bit would not be set if the transport is consuming a card in response to a consume card command. This bit can be used to verify that the transport is idle before taking action on the card position indicators.
4	Transport Cooling	Set to 1 if the transport is cooling else cleared to 0. When this bit is set, the transport will temporarily not be functional. This cooling mechanism is to ensure that the transport is not damaged by excessive stalling of the motor by the card user or by the host software. A stall occurs when a card is held in position while the transport tries to move the card. The transport will give up trying to move the card in 0.5 to 2 seconds. If 2 stalls occur within a 5-second period, then the transport goes into a cooling mode in which it will not function for 5 seconds during which time this bit is set.
5-31	RFU	Reserved for future use.

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)	06 00 00 00 (Middle and Rear Card Present)

NOTIFY INDICATOR CHANGE 0 TO 1 PROPERTY

Property ID: 01 (Hex)
Property Type: Dword
Power Up/Reset Value: 0
Get/Set support: Both
Description:

This property is used to 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	Front Card Present	If set to 1, a notification message will be sent in response to a 0 to 1 transition of the front card present indicator. If cleared to 0, no notification will be sent.
1-2	RFU (0)	Reserved for future use (Must be zero)
3	Auto Transporting	If set to 1, a notification message will be sent in response to a 0 to 1 transition of the auto transporting indicator. If cleared to 0, no notification will be sent.
4-31	RFU (0)	Reserved for future use (Must be zero)

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)	01 00 00 00 (Front Card Present)

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 support: Both
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	Front Card Present	If set to 1, a notification message will be sent in response to a 1 to 0 transition of the front card present indicator. If cleared to 0, no notification will be sent.
1-2	RFU (0)	Reserved for future use (Must be zero)
3	Auto Transporting	If set to 1, a notification message will be sent in response to a 1 to 0 transition of the auto transporting indicator. If cleared to 0, no notification will be sent.
4-31	RFU	Reserved for future use (Must be zero)

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)	01 00 00 00 (Front Card Present)

Example Set Property Response:

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

AUTO CONSUME PROPERTY

Property ID: 03 (Hex)
Property Type: Boolean
Power Up/Reset Value: 0 (Disabled)
Get/Set support: Both
Description:

This property is used to enable and disable card auto consuming. When auto consuming is enabled, the reader will attempt to consume a card when the card is being inserted into the mouth of the reader. The transport attempts to consume a card into the transport to a position where magnetic stripe reading is complete. The reader will not attempt to consume a card if a card is already present in the transport. When the reader is attempting to automatically consume a card, the Auto Transporting indicator is set. If the consume attempt is successful and the card is standard size, the Rear Card Present indicator and the Middle Card Present indicator will be set to show that the card is completely consumed. If the Middle Card Present indicator is set and the consume attempt is unsuccessful, then the card should be ejected. After a card is ejected or a consume fails, the card needs to be completely removed from the reader before the reader attempts to automatically consume a card again.

Values:

Value	State	Description
0	Disabled	Cards will not be automatically consumed.
1	Enabled	Cards will be automatically consumed.

Example Set Property Request:

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

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

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: 1 (Consume)
Get/Set support: Both

Description:

This property is used to control the magnetic stripe read direction.

Values:

Value	State	Description
0	Both	The reader can read during both card consuming and ejecting.
1	Consume	The reader can only read during card consuming.
2	Eject	The reader can only read during card ejection.

Note: If the front-gate option is installed, the IntelliStripe 310 Reader will be incapable of reliably reading mag-stripe cards during the card ejection cycle. Reliable mag-stripe card reading will only be possible during the card consumption cycle. If mag-stripe reading is required during the card ejection cycle, then the product must be ordered without the front-gate option.

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)	01 00 00 00 (Consume)

Example Set Property Response:

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

CONSUME CARD COMMAND

Command ID: 80 (Hex)
Description:

This command attempts to consume a card into the transport to a position where smart card communications can occur and magnetic stripe reading is complete. A card must be present at the front (insertion) end of the transport when this command is executed in order for it to succeed. The reader will not attempt to consume a card if a card is already present in the transport. If the consume attempt is successful and the card is standard sized, the Rear Card Present indicator and the Middle Card Present indicator will be set to show that the card is completely consumed. If the Middle Card Present indicator is set and the consume attempt is unsuccessful then the card should be ejected.

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

Success = 00 (Hex)
: The command completed successfully.

Transporting Failed = 80 (Hex)
: The transport operation failed to complete successfully.

Transport Busy = 81 (Hex)
: The command failed because the transport is busy transporting a card. This could occur if the transport is set to Auto Consume.

Transport Cooling = 82 (Hex)
: The command failed because the transport is cooling. This occurs if the transport is temporarily disabled due to too many stalls. See the transport indicators property for more details.

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

EJECT CARD COMMAND

Command ID: 81 (Hex)

Description:

This command attempts to eject a card from inside the transport to a position where the card can be removed by the user and where magnetic stripe reading is complete. A card must be present at the middle or rear of the transport when this command is executed in order for it to succeed. If the eject attempt is successful and the card is standard sized, the Rear Card Present indicator and the Middle Card Present indicator will be clear to show that the card is completely ejected from inside the transport. Also the Front Card Present indicator will be set until the user removes the card from the reader. If the reader is set to auto consume, it will not auto consume a card until the user removes the ejected card from the reader.

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

Success = 00 (Hex)
: The command completed successfully.

Transporting Failed = 80 (Hex)
: The transport operation failed to complete successfully.

Transport Busy = 81 (Hex)
: The command failed because the transport is busy transporting a card. This could occur if the transport is set to Auto Consume.

Transport Cooling = 82 (Hex)
: The command failed because the transport is cooling. This occurs if the transport is temporarily disabled due to too many stalls. See the transport indicators property for more details.

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

APPENDIX A. EXAMPLE HOST APPLICATION

The following is an example of a very simple host application that uses the IntelliStripe 310 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 intended only to be used as a demonstration of how the command set can be used to build an application.

The steps are as follows:

1. Initialize device by setting MSR Direction transport property to Consume and setting Auto Consume transport property to Enabled.

2. Wait for a card to be consumed completely into the reader. Check Rear Card Present indicator inside a loop until it is set. The Rear Card Present indicator is obtained by issuing the Get Indicators transport property. For error handling, also check for the following case inside the loop. If the Auto Transporting indicator is clear and the Middle Card Present Indicator is set but the Rear Card Present indicator is not set then a card is stuck in the transport so go to step 5.

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

4. Issue the Card Eject transport command to eject the card.

5. Go to step 2 to read another card.

INDEX