IBM 4778 Encrypting PIN pad
Product Attachment Information
Version 1.0

IBM Corporation

Preface

Notice

The solution described by this document may undergo revision in function and/or scope at any time. The information contained herein does not imply a commitment on IBM's part to produce or deliver any of the functionality described.

Contents

Pin Pad Model Configuration

These devices are Encrypting Pin Pads, with all models incorporating a 12 key keyboard for data entry and a 16 by 1 LCD for displaying messages.

3 models are available:

MODEL 1 PART #07H5065 ENCRYPTING PIN PAD W/TK 1 & 2 CARD READER

The magnetic stripe reader supports read of ISO tracks 1 and 2.

Read track 2 at 75 BPI.
Read track 1 at 210 BPI.

MODEL 2 PART #07H5066 ENCRYPTING PIN PAD WITHOUT CARD READER

MODEL 3 PART #07H5067 ENCRYPTING PIN PAD W/TK 1,2 & 3 CARD READER

The magnetic stripe reader supports read of ISO tracks 1,2 and 3.

Read track 2 at 75 BPI.
Read track 1 and 3 at 210 BPI.

Chapter 1. Communications Interface

Dual electrical interfaces are included, to support both the Serial RS232 or Mouse interfaces.

These devices will attach to members of PS/2, PS/VP or equivalent computers via the RS232 Serial port at 9600 Baud, and PS/2 equivalent auxilliary(mouse)port.

For all communications modes, the data will be one start bit, 8 data bits, 1 parity bit (odd) and 1 stop bit.Each byte transmission consists of 11 bits transmitted serially on the data line.

1st bit:	Start bit
2nd bit:	Data bit 0 (Least Significant Bit)
3rd bit:	Data bit 1
4th bit:	Data bit 2
5th bit:	Data bit 3
6th bit:	Data bit 4
7th bit:	Data bit 5
8th bit:	Data bit 6
9th bit:	Data bit 7 (Most Significant Bit)
10th bit:	Parity bit (odd parity)
11th bit:	Stop bit

The Serial and Mouse Interfaces allow the Magnetics Device to co-exist with a 4717/4777 Magnetic Stripe Reader/Encoder(MSR/E) device on a single host port. The interconnection cabling, together with logic within the assemblies, uses the busy line (Mouse) or RTS/CTS lines (Serial) to control which device has ownership of the interface.

The 4778 communication mechanism allows sharing a common port. Each device requires a unique Activate ID, and must be deactivated when not in use to allow the sharing device(4777) to gain access to the communication link.

The host issues a command selecting one device for communication or an I/O device that requires attention may request control of the interface, then issue an attention, requesting service from the host.

Each I/O device connected to a single port requires the unique Activate Command, this distinguishs the device to the Host from the other devices. Each I/O device type is assigned a permanent Activate ID. Once activated, an I/O device will assert the control line(-Busy/RTS) indicating to other devices sharing the link to remain inactive.

Host Communication Interface Cable

The Host attachment cable consists of an 8 wire interface, including six signals, 5 V and ground. The cable is 150mm long (approximately 6") and terminates in an 8 pin locking female mini-DIN plug. The host personality cable plugs into this cable, as does the multidrop configurations.

Unique cables for RS232 and Auxilliary(mouse) port connection are necessary.

Serial Communication Interface

The communications implemented in the 4778 devices include the RS232 Serial interface. The output and inputs are defined by RS232C. The 'ON' (logic high) is +3V to +15V while the 'OFF' (logic Low) is defined as -3V to -15V. The RS 232 serial port will only run at 9600 Baud in 11-bit start stop mode.

RXD and TXD Data Signals

The 'RXD' line is a uni-directional signal used to transfer data between the Host and the 4778 and other I/O device on a common communication link. The 'TXD' line is a uni-directional signal used to transfer data between the 4778 or other I/O device on a common link to the Host. To allow more than one device on the interface the 'TXD' signal is active only when the I/O device has been activated and will switch 'TXD' in the connection state only during transmission sequences. This 'TXD' switch interface allows the 4778 Encrypting PIN pad to co-exist with a I/O device (4777 MSR/E) on a single RS-232C port.

Device Contention

The 'CTS' and 'RTS' signals are used to allow the 4778 or other I/O device to gain control of the interface and to indicate when the interface is in use. This function is necessary to allow inactive devices to differentiate between communications between the Host and the active I/O Device.

When both the 4778 device and another I/O device are connected to one serial port special connector cables are required, allowing the communications interfaced to be shared at a single serial port. This cabling supports connection of the 'CTS' signal of the first device to the 'RTS' of the second device and the 'CTS' signal of the second device to the 'RTS' of the first device allowing for a "Busy" interface state management between the two devices.

```
 ____________________                     __________________
|          'CTS'   |7 <----------------- 8| 'RTS'          |
|DEVICE 1          |                    |       DEVICE 2 |
|          'RTS'   |8 -----------------> 7| 'CTS'         |
|____________________|                    |__________________|
                   Control Line Configuration
```

Mouse Communications Interface

The Mouse interface supports 3 interface signal lines, as described below, 5 volts and ground. Please refer to the PS/2 Keyboard/Auxilliary controller port specification for further connector definition.

+Data

Pin 1

The '+Data' line is a bi-directional signal used to transfer data between the device and the Host. A logical '1' is defined as a high level on the interface. This signal is driven and received by the Host and all devices attached to the Host. Contention over which I/O device in a multi-device configuration can drive this line is resolved by the '-Busy' line.

+Clock

Pin 5

The '+Clock' line is a bi-directional signal used to synchronize the data transfer between I/O devices and Host. This signal is driven and received by the Host all I/O devices attached to the Host. The Host can hold this line low to inhibit all I/O devices from communicating. For data transfers (both transmit and receive) this line is driven by the I/O device controlling the interface. Contention over which I/O device in a multi-device configuration can drive this line is resolved by the '-Busy' line.

-Busy

Pin 2

The '-Busy' line is a bi-directional signal that allows the I/O device to gain control of the interface and to indicate when the interface is in use. This function is necessary to allow inactive devices to differentiate between communications between the Host and the active device and the Host transmission of 'ACTIVATE' commands that all I/O devices must receive. A low voltage level on this line indicates a request is being made or that the interface is busy. A high level indicates that the interface is not busy and all I/O devices should monitor interface activity. This line can be driven and received by all I/O devices attached to the Host. The I/O device can drive this line to request the interface or when it has control.

Device Contention

The definition of the '+Data' and '+Clock' signals requires that one I/O device always has control of the interface and will drive the '+Clock' line for communications. An I/O device must first gain control of the interface before a communication sequence can occur. The '-Busy' line allows for an I/O device to request and gain control of the interface. The request sequence can be initiated in two ways:

1. Device Initiated

The I/O device requesting service will initiate a request for control of the interface by sending an 'ATTENTION' command to the Host. After the device has acquired control of the interface, set the busy mode and sent the 'ATTENTION', the device will return to the interface mode (monitor or busy) that it was in prior to sending the 'ATTENTION'. If a host transmission is started ('Request to Send' on the clock and data lines) after the device has set the '-Busy' line to send an 'ATTENTION', the device will return to the monitor mode and clock in data bits from the host. If the host aborts the transmission of the 'ATTENTION' by holding the clock line low, the device will return to the monitor mode and prepare to clock in a transmission from the host. The device will also initiate a request for the interface when it has completed power-on initialization.

2. Host Initiated

When the Host has a command to issue to a device, it must be assured control of the interface. The Host will issue an 'ACTIVATE' command identifying the device. All I/O devices monitor 'ACTIVATE' commands. The device will respond to receiving its ID by requesting and gaining control of the interface and sending an 'ACTIVE' command identifying itself back to the Host.

Once the device has been selected by the 'ACTIVATE' command and has gained control of the interface, it will remain in control of the interface with the '-Busy' line active (busy mode) until receiving a 'DEACTIVATE' command from the Host. When this command is received the device will deactivate the '-Busy' line (monitor mode) to allow other devices to gain control of the interface.

Chapter 2. Functional Description

Encrypting PIN pad (EPP)

4778 includes 3 models of Encrypting PIN Pad (EPP) devices. All models incorporate a 12 key keyboard for PIN entry and a 16 by 1 LCD for displaying output messages. The models vary with respect to the type of Magnetic Stripe Reader(MSR) capability included. The Model 1 includes credit card track 1 and 2 read capability, the model 3 has credit card track 1,2 and 3 read capability and the model 2 has no credit card capability.

Overall Function

The 4778 Encrypting PIN Pad (EPP) is designed to operate with all members of the PS/2, ValuePoint or equivalent product families. It accepts a personal identification number (PIN) at its keyboard, combines it cryptographically in accordance with applicable specifications and sends the appropriate message to the Host for further processing.

The EPP contains a Master Cryptographic Key, which can be changed either automatically via the Host interface or manually via the key pad. Additional keys are available as variants of the Master Key, encrypted under the Master Key, or as plain text keys. Keys cannot be read out of the security module once placed therein.

The EPP provides both an audio and visible response to data entry via the keyboard.

Attachment

The 4778 Encrypting PIN Pad (EPP) is comparable to the 4718, providing plug compatible function. The 4778 devices offer a full range of attachment capability for both serial (RS232) and Mouse port connection.

Magnetic Capability

The 4778 with MSR, when Activated as a magnetic stripe reader, operates similar to a 4777 MSR device. In this mode, the 4778 MSR is selected with a unique ACTIVATE and uses similar commands and protocols that a 4777 MSR uses. When the EPP is selected as a PIN Pad device it uses the commands and protocols defined in this document.

The 4778 EPP with Magnetic Stripe Reader capability is a single device that allows data from multiple tracks to be read and stored on a single pass of a credit card. When the MSR has detected that a card has been slotted, the MSR will request attention from the attached Host. The attached Host can issue commands to read the stored data and re-enable the MSR. Once a card has been slotted, the MSR will remain disabled until re-enabled by the attached Host.

Read Functions

Track 1

Track 1 characters comprise 6 bits + parity.

1. Is capable of reading Track-1 at 210 BPI, in accordance to ISO Standards 7811/2 and /4.
2. All data read will be stored until requested, or until cleard by command.
3. Storage may be cleared on command.

Track 2

Track 2 characters comprise 4 bits + parity.

1. Is capable of reading Track-2 at 75 BPI with no conditioning, in accordance with ISO Standards 7811/2 and /4.
2. All data read will be stored until requested, or until cleard by command.
3. Storage may be cleared on command.

Track 3

Track 3 characters comprise 4 bits + parity.

1. Is capable of reading Track 3 at 210 BPI, in accordance to ISO Standards 7811/2, /4, and /5.
2. All data read will be stored until requested, or until cleard by command.
3. Storage may be cleared on command.

Indicators

The 4778 device uses "↑" and "↓" for the Indicator function. The 4778 uses three position of the 16 X 1 LCD to provide the indicator function. The indicator function is provided by either 'UP ARROW' ↑ or a 'DOWN ARROW' ↓ to point to the appropriate GREEN, YELLOW, or RED indicator condition. A template is provide with each 4778. This template also has the GREEN, YELLOW or RED indication on it. There is a GREEN, YELLOW and RED area above the LCD that represent the indicator functions of the Magnetic stripe reader and a GREEN area just below the LCD that represents the PIN Pad indicator functions.

MSR indicator functions

The three indicator functions for the magnetic stripe reader are as follows:

Ready (Green)

1. Turned on by command from the Host, 4778 MSR is enabled. Mag command.
2. Held on steady until the read pass has been completed.

In Process (Amber)

1. Turned on and off by commands through the external interface.

Check (RED)

Turned on and off by commands through the external interface.

EPP indicator functions

The only indicator function for the PIN Pad is as follows:

Ready (Green)

1. On steady to indicate the PIN Pad has been enabled for keypad entry (PIN or Key Loading).
2. Turned off by EPP when command is completed.

Chapter 3. 4778 EPP and MSR device selection Commands

Issuing Commands

A transaction between the Host and the 4778 consists of three parts;

1) Device Selection(ACTIVATE)
2) Execution of a EPP or MSR command sequence or sequences
3) Termination(DEACTIVATE)

The Host initiates a transaction by first selecting the 4778 using the 'ACTIVATE' command and waiting for it to be acknowledged by the 'ACTIVE' status. The Host can then send a MSR or EPP command.

MSR command sequences are completed when the 4778 sends the 'OPCOMP' command to the Host. Upon receiving the 'OPCOMP' the Host can issue another magnetics command or terminate the transaction by issuing the 'DEACTIVATE' command.

EPP command sequence are completed when the command response to the last frame is received, some EPP commands are multiframe commands, the command sequence is not complete until the response to the last frame is received. The Host must issue the 'DEACTIVATE' command after receiving the last response available for the command that was issued.

After receiving the 'DEACTIVATE' command the 4778 will release the control line and return to the monitor mode.

If the 4778 requires service from the Host it must first gain control of the interface then send an 'ATTENTION' to the Host. After sending the 'ATTENTION', the 4778 will return to the interface mode (busy or monitor) that it was in prior to the 'ATTENTION' transmission. Upon receiving the 'ATTENTION' the Host will activate the 4778 (if not active). When the 'ATTENTION' was from the MSR the Host will issue a 'SENSE' command to determine the reason for the 'ATTENTION'. At the completion of the 'SENSE' command sequence the Host my issue another MSR command or terminate the transaction with the 'DEACTIVATE' command. When the 'ATTENTION' was from the EPP the HOST will issue the 'READ DATA' command, this is only issued for asyncronous commands.

MSR command sequences consists of a command issued by the Host and concluded by the 'OPCOMP' response from the 4778. If execution of the command requires additional data be sent from the Host, the 4778 will send the 'NEXT' status when it is ready for each data byte. Upon receiving the 'NEXT' status the Host will send one additional byte of data. This byte-by-byte handshaking will continue until all data has been transferred. The Host will indicate to the 4778 that all data has been transferred by sending an 'EOP' in response to the 'NEXT' status. For all MSR command sequences transferring data to the 4778, the 4778 will calculate an LRC for the data field and send it to the Host in response to the 'EOP'. The sequence will be completed with another 'EOP' from the Host and the 'OPCOMP' response. Data transferred to the 4778 can have 1 to 7 valid bits per character. The high order bit (bit 7) must always be '0' for data bytes and '1' for command bytes sent to the 4778.

If execution of the MSR command sequence requires data to be transferred to the Host, the 4778 will respond to the command with a 1-byte header indicating that data follows. The 4778 will wait for the 'NEXT' status from the Host before sending each data byte. The Host will indicate when all data has been received by sending an 'EOP' instead of the 'NEXT'. The 4778 will then send a one byte LRC character. The sequence is completed with another 'EOP' from the Host and the 'OPCOMP' response.

Device Selection Commands

The following commands are used to control which I/O device is in control of the interface. A brief description of each command is given in this section.

Activate 4778 EPP:

ACTIVATE EPP - Hex Code X'B1'

This command is issued by the host to select the 4778 EPP for execution of a communication sequence. The host terminal will not issue this command if there are any commands in progress (I/O devices must be in the monitor mode). After receiving this command, the 4778 EPP will gain control of the interface, set the busy mode and respond with the status 'EPP ACTIVE'. After sending the 'EPP ACTIVE' the 4778 will then be ready to accept any EPP commands from the host.

There is not a retry command defined for the 'ACTIVATE EPP' command or 'EPP ACTIVE' response. Any errors will result in the host re-initiating the sequence from the beginning. The 4778 or other I/O device will not return the 'EPP ACTIVE' status if a parity error was detected on the transmission of the 'ACTIVATE EPP' command. If no I/O device detects its valid ID, the status of the interface will not change and the 'EPP ACTIVE' status will not be returned by any device. After a 25 milli-second timeout the host may reissue the 'ACTIVATE EPP' command. If the host detects an error in the 'EPP ACTIVE' response or receives and invalid ID in the response, the host will send the 'DEACTIVATE COMMAND' to force all devices to the monitor mode and reissue the 'ACTIVATE EPP' command.

Activate 4778 MSR:

ACTIVATE MSR - Hex Code X'B2'

This command is issued by the host to select the 4778 MSR for execution of a communication sequence. The host terminal will not issue this command if there are any commands in progress (I/O devices must be in the monitor mode). After receiving this command, the 4778 MSR will gain control of the interface, set the busy mode and respond with the status 'MSR ACTIVE'. After sending the 'MSR ACTIVE' the 4778 will then be ready to accept any MSR commands from the host. The 4778 MSR will remain active until receiving 'DEACTIVATE' from the host.

There is not a retry command defined for the 'ACTIVATE MSR' command or 'MSR ACTIVE' response. Any errors will result in the host re-initiating the sequence from the beginning. The 4778 or other I/O device will not return the 'MSR ACTIVE' status if a parity error was detected on the transmission of the 'ACTIVATE MSR' command. If no I/O device detects its valid ID, the status of the interface will not change and the 'MSR

ACTIVE' status will not be returned by any device. After a 25 milli-second timeout the host may reissue the 'ACTIVATE MSR' command. If the host detects an error in the 'MSR ACTIVE' response or receives and invalid ID in the response, the host will send the 'DEACTIVATE COMMAND' to force all devices to the monitor mode and reissue the 'ACTIVATE MSR' command.

Activate Reserved:

Hex Code X'B0' and Hex Code x'B3' to Hex Code x'BF'

These commands are reserved for activate functions of other I/O devices. These commands could be received by the 4778 when in the monitor mode.

4778 EPP Attention:

ATTENTION EPP - Hex Code X'C1'

The 'EPP ATTENTION' command is issued by the 4778 when service is required. The 4778 must first gain control of the interface and assert the control line before issuing this command. This command identifies the 4778 EPP as requiring service. Following transmission of the 'ATTENTION' the 4778 will return to interface mode (busy or monitor) that it was in prior to the cause of the 'ATTENTION'.

4778 MSR Attention:

ATTENTION MSR - Hex Code X'C2'

The 'MSR ATTENTION' command is issued by the 4778 when service is required. The 4778 must first gain control of the interface and assert the control line before issuing this command. This command identifies the 4778 MSR as requiring service. Following transmission of the 'ATTENTION' the 4778 will return to interface mode (busy or monitor) that it was in prior to the cause of the 'ATTENTION'. The 4778 will then wait to be activated (if not active) and receive the 'SENSE'command.

Reserved Attention:

Hex Code X'C0' and Hex Code X'C3' to Hex Code x'CF'

These 'ATTENTION' commands are reserved for other I/O devices.

Deactivate:

Hex Code X'FF'

This command is issued by the Host to cause the active I/O device to halt communications and return to the monitor mode. This command is used for normal command sequence completion. In addition, multiple X'FF' bytes (20 in number) must be sent to ensure reset of the PIN Pad in the event of a PC Software reset. No response is defined of the 'DEACTIVATE' command.

Chapter 4. 4778 Encrypting PIN Pad(EPP) Supported Functions

The supported functions fall within two general areas, Cryptographic Key Management and Encryption Functions. In addition a set of utility functions are defined for error recovery and support of asynchronous operation. The supported functions are listed below.

4778 EPP Command Protocol

During the Active Communication State, the PIN Pad communicates with the PC using a block protocol. Both requests by the PC and responses from the PIN Pad utilize the same general protocol. A request contains all the information necessary for the receiving party to perform the desired action. The response contains all the information resulting from that action.

Each request or response consists of one or more transmission blocks or Frames. Each Frame consists of three (3) components as follows:

PROLOG	DATA	EPILOG

The Prolog contains the CONTROL Byte, a COMMAND Byte, and an OPTION Byte. The Data consists of from 1 to 16 data bytes. The Epilog is a single byte formed by XORing all the preceding bytes in the frame. The choice of Control byte limits the amount of data that can be sent in one Frame to 16 bytes (4-bit length field) due to the limited amount of data RAM available in the Security Processor. Longer messages can be handled by use of the chain flag. A zero (0) value for the Chain Flag indicates this is the last or only Frame to be transmitted or received. A one (1) value for the Chain Flag indicates there is another Frame following this one.

CONTROL	COMMAND	OPTION	data0	-//-	dataN	CKSUM

```
CONTROL
     ----> Bit 7 (MSB)      = Chain Flag
     --------> Chain Flag  = 1 "Another Frame Follows this one"**
     --------> Chain Flag  = 0 "This is the Last (or only) Frame"
     ----> Bit 6            = Request/Response Indicator
     --------> Request      = 0
     --------> Response     = 1
     ----> Next two bits    = Reserved future Use
     ----> Remaining bits   = Data Length
                            = 0 - 15 (1-16 Data Bytes)
**Note: Once a request or response has been sent with the chain flag
        set, any change in the Command is rejected as an error.
CKSUM
     ----> CONTROL xor COMMAND xor OPTION xor (data0) xor...(dataN)
```

4778 EPP Command Summary

Command	Description	Command Type
X'00'	Reserved	NA
X'01'	Read Status	Syncronous
X'02'	Read Nonencrypted Key pad Data	Asyncronous
X'03'	Read Data	Syncronous
X'04'	Read Serial Number	Syncronous
X'05'	Abort	Syncronous
X'06'	Resend	Syncronous
X'07'	Set PIN Keypad Mode	Syncronous
X'08'	Display a String	Syncronous
X'10'	Enter Master Key	Asyncronous
X'11'	Load Master Key	Syncronous
X'12'	Load Key	Syncronous
X'13'	Load ICV	Syncronous
X'15'	Load PIN Verification Parameters	Syncronous
X'2X'	Create PIN Block	Asyncronous
X'3X'	Verify PIN/Create PIN Offset Data	Asyncronous
X'6X'	Generate Message Authentication Code	Syncronous
X'7X'	Verify Message Authentication Code	Syncronous

4778 EPP Command Detail

Read Status

The Read Status command is used to get hardware status information from the 4778. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'01'	X'0a'	X'00'	X'cc'

```
Where
             a = Option
               =   (Return BAT Results)
               = 1 (Rerun BAT Tests and return Results)
               = 2 - Run Keyboard Test - The device will scan the
                   key pad until twelve keys have been depressed,
                   pad out to sixteen digits, encrypt the results
                   using a fixed cryptographic key                ,
                   and return the 8 byte result (FEB7B9253F35EB D).
               = 3 - Return PIN Microcode Version Information
            cc = XOR of preceding bytes
```

The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	-//- '	' dataL	Cksum
X'4L'	X'01'	X'rr'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
             L = Data Length
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            dd = from 1-14 Response Bytes
               = for options  ,1 - one byte with bit significance
                 bit  = 1 if one or more keys are closed
                 bit1 = 1 if EEPROM Checksum Verification failed
                 bit2 = 1 if RAM test failed
          bit3 - bit6 = reserved future use
                 bit7 =   for Clear Mode, = 1 for Encrypted Mode
               = for option 2 - 8 Bytes (FEB7B9253FE5EB DH if correct)
               = for option 3 - 14 Byte Ascii String "v.vv, mm/dd/yy"
                 (v.vv - version, mm - month, dd - day, yy - year)
            cc = XOR of preceding bytes
```

Read Nonencrypted (Clear) Keypad Data

This command is used to read data entered into the key pad without encryption. This may be used in support of applications currently written for the Nonencrypted PIN Pad. A maximum of 32 digits can be read. The command is executed asynchronously. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'02'	X'00'	X'00'	X'02'

If the command is received correctly, a positive return code is sent back to the Host, the 4778 key pad is enabled, and the key pad is will be scanned for input data. If an error occurs during receipt of the command, the command is aborted and a negative response is returned. The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'02'	X'rr'	X'00'	X'cc'

```
Where
             rr = Return Code (see “4778 EPP Command Return Codes” on page 4
             cc = XOR of previous bytes
```

After the data has been read from the keypad (indicated by the receipt of an END scan code) an ATTENTION byte is sent to the Host. The Data is then read using a Read Data Command, with data returned in a packed BCD format. See “Set PIN Keypad Mode” on page 4-5 for setting the keypad input mode.

Read Data

The Read Data command is used to get data from the 4778 which is known to be pending as a result of the receipt of an ATTENTION Byte following an Asynchronous Command Request. The Host Driver is responsible for keeping track of the pending asynchronous command. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'03'	X'00'	X'00'	X'03'

The response to the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	' dataL '	Cksum
X'4L'	X'03'	X'rr'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
              L = data length
             rr = Return Code (see “4778 EPP Command Return Codes” on page
          d -dL = data returned
             cc = XOR of preceding bytes
```

Read Serial Number

This command is used to read the 4778 Serial Number from the nonvolatile memory. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'04'	X'00'	X'00'	X'04'

The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' data1'	'data7'	Cksum
X'47'	X'04'	X'rr'	X'd0'	X'd1'	X'd7'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes” on page
         d -d7 = 8 Bytes representing the serial number
                in packed decimal as follows:
              = 477841 sssssssFF
              where    4778 = Machine type
                        41 = Plant of Control (Charlotte)
                    sssssss = Serial Number
                        FF = Serial # set Flag
            cc = XOR of preceding bytes
```

Abort

The Abort command is used to prematurely end an Asynchronous Command request. The 4778 will stop its current processing, return all process flags to their initial states, and return a positive return code to the Host. It will then be ready to execute the next command request. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'05'	X'00'	X'00'	X'05'

The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'05'	X'rr'	X'dd'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            dd = Command code of command aborted
            cc = XOR of preceding bytes
```

Resend

The Resend command is used to request a resend of the last data block in the case of a communications error. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'06'	X'00'	X'00'	X'06'

The response to this command is a resend of the response to the last command. If the last command was a resend request, no action is taken, since it is assumed that the communications link has a non-correctable fault. This command can be initiated by the host or 4778.

Set PIN Keypad Mode

This function is used to set the PIN Keypad Mode for either encrypted or non-encrypted operation. When the encrypted mode is set, all keys must be reloaded, as all keys previously loaded are rendered invalid by the process. When in non-encrypting mode a Create PIN Block request will not be honored but will result in an error return code. Similarly when in encrypting mode, a Read Nonencrypted Keypad Data request will result in an error return code. All other cryptographic operations will work in both modes as long as valid keys are loaded in the Keypad. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'07'	X'0m'	X'00'	X'cc'

```
Where
             m = Mode to be set
              =   - Set non-encrypted mode
              = 1 - Set Encrypted Mode
            cc = XOR of preceding bytes
```

The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'07'	X'rr'	X'00'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes” on page 4
            cc = XOR of preceding bytes
```

Display A String

This function is used to display a string of up to 16 characters on the LCD display on the 4778. The message shares the LCD with the 4778 MSR Indicator command, as such the message will be erased when ever the RED or. Yellow indicators are turn ON or OFF. The 16th position is also shared with the "GREEN" indicator which is turned ON/OFF by the hardware. When ever the "GREEN" indicator is turn on only the 16 position is overlaid. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	' dataN '	Cksum
X'0L'	X'08'	X'00'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
             L = length of the data string
              =   - 15 (1 - 16 data bytes)
         d -dL = data string to be displayed ( Max 16 character
              = see appendix - B for display characters
            cc = XOR of previous bytes
```

The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'08'	X'rr'	X'00'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes” on page 4
            cc = XOR of preceding bytes
```

Enter Master Key

The Enter Master Key Function will read entered keystrokes from the key pad, convert them to hexadecimal, and place the resulting Master Key in the storage location indicated by the Command Parameters. This is an asynchronous command.The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'10'	X'tm'	X'00'	X'cc'

```
Where
            t = type of key
              =      if 16 bytes (48 key strokes)
              =    1 if 8 bytes (24 key strokes)
              =   1  - 1111 reserved
            m = method of entry
              =      if single entry
              =    1 if dual entry (two pieces to be XOR'ed)
              =   1  - 1111 reserved
           cc = X'1 ' xor X'tm'

Note: If an 8 byte key is entered, it will be duplicated as the second 8
     bytes of the double length Master Key in the Security Processor.
```

On successful receipt of command, the 4778 returns a positive response and starts to execute the command. If an error occurs during the receipt of the command, a negative response is returned to the Host and the command is aborted. The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'00'	X'10'	X'rr'	X'00'	X'cc'

```
Where
           rr = Return Code (see “4778 EPP Command Return Codes” on page
           cc = Xor of previous bytes
```

As the key is entered into the key pad, the control logic converts the entries into Hex characters and stores the key in a buffer. After the last key is read from the key pad, the EPP checks the key for correct parity. If the parity is correct, the old key is replaced by the new key. An "ATTENTION" is sent to the Host indicating completion of the command and a response is formatted for sending on the receipt of a read data command. If the command was successfully completed the response includes as data the triple encryption of the device 8 byte serial number (see Read Serial Number Command) under the new Master Key as a check on the key. If an error occurred during the command execution, an error response code is returned.

Load Master Key

The Load Master Key command will load a new Master Key into the 4778 via the Host Interface. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	'dataN '	Cksum
X'0L'	X'11'	X'tm'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
            L = length of data (7 or F)
            t = type of key
              =      if 16 bytes
              =    1 if 8 bytes
              =   1  - 1111 reserved
            m = method of entry
              =      if clear
              =    1 if encrypted under previous key
              =   1  - 1111 reserved
         d -dL = key bytes (clear or encrypted)
           cc = XOR of previous bytes

Note: If an 8 byte key is entered, it will be duplicated as the second 8
     bytes of the double length Master Key in the Security Processor.
```

When the new key is received, it is first decrypted with the old key (if required) and then checked for correct parity. If the parity is correct, the old key is replaced by the new key and a positive response is returned to the Host. Otherwise a negative response is returned. The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	'data8 '	Cksum
X'4L'	X'11'	X'rr'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
            L = data length
              =   for negative response
              = 7 for positive response
           rr = Return Code (see “4778 EPP Command Return Codes” on page
           d  = X'  ' for negative response
         d -dL = triple encryption of the 4778 Serial# with the
                new Key for positive response
           cc = XOR of preceding bytes
```

Load Key

The Load Key command is used to load keys into the 4778 Security Processor non-volatile storage. The keys are passed from the Host to the 4778 triple encrypted under an appropriate variant of the Master Key. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	'data8 '	Cksum
X'08'	X'12'	X'vv'	X'kk'	X'd1'	X'd8'	X'cc'

```
Where
               vv = Variant Table Descriptor (index to correct variant)
                  =    if variants not used
                  =  3- 6 to indicate use of the given variant
               kk = key identifier (index number)
                  =    to 1F for Secret Storage
                  = 2  to FF for non-secret storage
                              (keys remain encrypted until used)
            d1-d8 = key bytes
               cc = XOR of previous bytes
```

When the new key is received, it is first decrypted and then checked for correct parity. If the parity is correct, the key is placed in storage and a positive response is returned to the Host. Otherwise a negative response is returned. The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	'dataN '	Cksum
X'4L'	X'12'	X'rr'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
                L = data length
                  =   for negative response
                  = 7 for positive response
               rr = Return Code (see “4778 EPP Command Return Codes” on page
               d  = X'  ' for negative response
            d -dL = triple encryption of the 4778 Serial# with the
                    new Key for positive response
               cc = XOR of preceding bytes
```

Load ICV

The Load ICV Command is used to store an initial chaining vector internal to the 4778 Security Processor. The ICV, which may be used in the Generate MAC Command, is an 8 byte quantity. It is passed from the Host encrypted under a variant of the Master Key, as described below.

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	'data7 '	Cksum
X'07'	X'13'	X'vv'	X'd0'	X'dd'	X'd7'	X'cc'

```
Where
               vv = Variant Table Descriptor
                  =    if variants not used
                  =  2 for fixed variant  2
            d -d7 = data bytes (encrypted ICV)
               cc = XOR of previous bytes
```

When the ICV is received it is decrypted and stored in the secret storage area of the Security processor for later use. An appropriate response frame is returned to the Host as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'13'	X'rr'	X'00'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes” on page 4
            cc = XOR of preceding bytes
```

Load PIN Verification Parameters

This command is used to load the parameters needed by the 4778 to verify PINs or to Create PIN Offset Data. The parameters loaded are stored in non-volatile storage and are used until the command is repeated. The format of the command is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	'dataF '	Cksum
X'0F'	X'15'	X'0p'	X'd0'	X'dd'	X'dF'	X'cc'

```
Where
             p = PINMINL (length of PIN to be checked)
              = X' ' - X'F' (1-16)
         d -dF = Decimalization Table
            cc = XOR of previous bytes
```

The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'15'	X'rr'	X'00'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes” on page 4
            cc = XOR of previous bytes
```

Create PIN Block

In response to the Create PIN Block Command the 4778 control code reads a Personal Identification Number entered into the key pad and creates an encrypted PIN Block of the format indicated. This command is executed asynchronously as described below.

In order to receive the information required to perform the command, a two frame chained command sequence is required. The first command frame sets the required encryption key. The required frame is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//- '	'dataN '	Cksum
X'8L'	X'2k'	X'vv'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
             L = data length (dependent on key format)
             k = Key format
              =      for Master Key (1st 8 Bytes)
              =    1 for 1 Byte Internal Key Short Pointer
              =   11 for 8 byte Key data field
              =   1 ,   1 -1111 Reserved
            vv = Variant Table Descriptor
              =    if variants not used
              =  3 for fixed variant  3
         d -dL = data
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'2k'	X'rr'	X'00'	X'cc'

```
Where
             k = same as above
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            cc = XOR of previous bytes
```

Given a positive response frame, the Host shall output the second command frame as follows.

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataN '	Cksum
X'0L'	X'2p'	X'fa'	X'd0'	X'dd'	X'dL'	X'cc'

```
Where
             L = data length (dependent on format)
             p = PAN required or not
               =      PAN not required
               =    1 PAN required (present in data)
               =   1  - 1111 reserved
             f = PIN Block Format
               =      IBM 47 4
               =    1 ANSI X9.8
               =   1  IBM 3624
               =   11-1111 reserved
             a = Pad Character ( -F)
            d  = X'  ' if PAN not required
         d -d5 = 12 digit (BCD) PAN (if required)
            cc = Xor of previous bytes
```

On successful receipt of the PAN/format data, the 4778 returns a positive response and starts to execute the command. If an error occurs during the receipt of the PAN/format data, a negative response is returned to the Host and the command is aborted. The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'2p'	X'rr'	X'00'	X'cc'

```
Where
             p = same as above
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            cc = Xor of previous bytes
```

After the PIN has been read from the key pad and correctly formatted, an ATTENTION byte is sent to the Host. The PIN Block is then read using a Read Data Command.

Verify PIN / Create PIN Offset Data

This command can be used to verify PIN entries utilizing PIN offset and validation data read from a magnetic stripe card. or to Create PIN Offset Data for encoding on a magnetic stripe card. The PIN Pad encrypts and compares the data passed with the command according to the 3624 algorithm. As with the Create PIN Block Command, this command is executed asynchronously. In order to receive the information required to perform the function, a two frame chained command sequence is required as described below.

The first command frame sets the required encryption key. The required frame is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataN '	Cksum
X'8L'	X'3K'	X'vv'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             L = data length (dependent on key format)
             k = Key format
               =      for Master Key (1st 8 Bytes)
               =    1 for 1 Byte Internal Key Short Pointer
               =   11 for 8 byte Key data field
               =   1 ,  1  -1111 Reserved
            vv = Variant Table Descriptor
               =    if variants not used
               =  4 for fixed variant  4
            dd = data (1 or 8 bytes as per k)
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'3k'	X'rr'	X'00'	X'cc'

```
Where
             k = Same as above
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            cc = Xor of previous bytes
```

Given a positive response frame, the Host shall output the second command frame as follows.

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataN '	Cksum
X'0L'	X'30'	X'0m'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             L = data length
               = X'7' for just validation data
               = X'F' for validation and offset data
             m = Mode to be set
               =   - verify PIN
               = 1 - Create PIN Offset Data
            dd = data (8 Byte validation data plus offset data
                       if required)
            cc = Xor of previous bytes
```

On successful receipt of the validation data, the 4778 returns a positive response and starts to execute the command. If an error occurs during the receipt of the validation data, a negative response is returned to the Host and the command is aborted. The format of the response is as follows:

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'30'	X'rr'	X'00'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            dd = data
            cc = Xor of previous bytes
```

After the PIN has been read from the keypad and correctly formatted, an ATTENTION byte is output to the Host. The response is then read using a Read Data Command. If the PIN is verified, a single data byte of X'00' is returned to the Host, if not verified a data byte of X'FF' is returned. If in Create PIN Offset Data Mode, 8 data bytes are returned (16 BCD digits, padded with X'F's if less than 16 digits based on PINML(see “Load PIN Verification Parameters” on page 4-10) are to be used for verification.

Generate Message Authentication Code

A message authentication code on a message of arbitrary length can be generated by the 4778 using chained command sequences. The first two command frames set the encryption key and the ICV. These are followed by one or more 8 byte data sequences. The response to the last data sequence (chain flag = 0) contains an 8 Byte MAC (for systems requiring only four bytes, the application will select the leftmost 4 bytes to be retained).

The first command frame sets the required encryption key. The required frame is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataN '	Cksum
X'8L'	X'6k'	X'vv'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             L = data length (dependent on key format)
             k = Key format
               =      for Master Key
               =    1 for 1 Byte Internal Key Short Pointer
               =   11 for 8 byte Key data field
               =   1 ,  1  -1111 reserved
            vv = Variant Table Descriptor
               =    if variants not used
               =  5 for fixed variant  5
            dd = data
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'6k'	X'rr'	X'00'	X'cc'

```
Where
             k = Same as above
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            cc = Xor of previous bytes
```

Given a positive response frame, the Host shall output a second command frame to set the ICV as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataL '	Cksum
X'8L'	X'6I'	X'vv'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             L = data length (  or 7)
             I =  1   if internal ICV to be used
               =  11  if ICV in data
               =     -  11, 111-1111 reserved
            vv = Variant Table Descriptor
               =    if variants not used
               =  2 for fixed variant  2
            dd = 8 Byte ICV (if included) encrypted under
                  variant of Master Key
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'6I'	X'rr'	X'00'	X'cc'

```
Where
             I = Same as above
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            cc = Xor of previous bytes
```

Given a positive response frame, the Host shall output one or more data frames as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' data7 '	Cksum
X'C7'	X'60'	X'00'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             C = chain flag
            dd = 8 Bytes of data to be included in MAC
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataL '	Cksum
X'4L'	X'60'	X'rr'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             L = Length Byte (  or 7)
            rr = Return Code (see “4778 EPP Command Return Codes” on page
            dd = 8 Byte MAC if response to last data frame
              = X'  ' if response to chained data frame
            cc = Xor of previous bytes
```

Verify Message Authentication Code

A message authentication code on a message of arbitrary length can be verified by the 4778 using chained command sequences. The first two command frames set the encryption key and the ICV. These are followed by one or more 8 byte data sequences. The response to the last data sequence (chain flag = 0) contains the result of the MAC Verify process. The last data sequence contains either a 4 byte or 8 byte quantity which is compared with the MAC calculated on the preceding data.

The first command frame sets the required encryption key. The required frame is as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataN '	Cksum
X'8L'	X'7k'	X'vv'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             L = data length (dependent on key format)
             k = Key format
              =      for Master Key
              =    1 for 1 Byte Internal Key Short Pointer
              =   11 for 8 byte Key data field
              =   1 ,  1  -1111 reserved
            vv = Variant Table Descriptor
              =    if variants not used
              =  6 for fixed variant  6
            dd = data
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'7k'	X'rr'	X'00'	X'cc'

```
Where
             k = Same as above
            rr = Return Code (see “4778 EPP Command Return Codes” on page 4
            cc = Xor of previous bytes
```

Given a positive response frame, the Host shall output a second command frame to set the ICV as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' dataL '	Cksum
X'8L'	X'7I'	X'vv'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             L = data length (  or 7)
             I =  1   if internal ICV to be used
               =  11  if ICV in data
               =     -  11, 111-1111 reserved
            vv = Variant Table Descriptor
               =    if variants not used
               =  2 for fixed variant  2
            dd = 8 Byte ICV (if included) encrypted under
                  variant of Master Key
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'7I'	X'rr'	X'00'	X'cc'

```
Where
             I = Same as above
            rr = Return Code (see “4778 EPP Command Return Codes” on page 4
            cc = Xor of previous bytes
```

Given a positive response frame, the Host shall output one or more data frames as follows:

CONTROL	COMMAND	OPTION	' data0 '	' -//-	' data7 '	Cksum
X'C7'	X'70'	X'00'	X'dd'	X'dd'	X'dd'	X'cc'

```
Where
             C = chain flag
            dd = 8 Bytes of data to be included in MAC
               = or MAC to be checked if last data frame
            cc = Xor of previous bytes
```

The 4778 will output the following response frame.

CONTROL	COMMAND	OPTION	' data0 '	Cksum
X'40'	X'70'	X'rr'	X'dd'	X'cc'

```
Where
            rr = Return Code (see “4778 EPP Command Return Codes”)
            dd =    in response to data frames
               = result of MAC Check on last frame
               =    - MAC Checks, = FF - MAC Invalid
            cc = Xor of previous bytes
```

4778 EPP Command Return Codes

'Return Code'	Description
00H	Complete (No Error)
01H	Invalid Command
02H	Check Sum Mismatch
03H	Data not available
04H	More Data Required to Process Function
05H	Too much Data passed
06H	Invalid Key Pressed
07H	DES Key has Bad Parity
08H	4778 Write ERROR
09H	Tried to Abort with no Function Pending
0AH	Bad Parity on EEPROM Stored Key
0BH	Command Invalid in Clear Mode
0CH	Command Invalid in Encrypted Mode
0DH	Too many data frames received
0EH	Too few data frames received
0FH	Incorrect Variant for Command
10H	Variant has invalid Parity
11H	Invalid field for the command
12H	Invalid Data Length of MAC Function

Chapter 5. 4778 Magnetic Stripe Reader(MSR) Commands

The following commands are unique 4778 commands required to implement the magnetics functions. A brief description of each command is given in this section.

Commands: Auxilliary(Mouse) versus Serial(RS232C)

The command sequence is different when the 4778 is connected to the Auxilliary port then when connected to the Serial port. The 4778 will request a command to be resent for verification when connected to the auxilliary port. The Serial port requires the command be sent only once.

Auxilliary(Mouse)port

```
Host:   <Mag Cmd>        <Mag Cmd>      .....
4778:         <RESEND>         <NEXT> ......
```

This representation for the auxilliary port connected 4778 is abbreviated in all the command descriptions as '<COMMAND>'.

Serial(RS232C)port

```
Host:   <Mag Cmd>   .....
4778:           <NEXT> ......
```

This representation for the serial port connected 4778 is abbreviated in all the command descriptions as '<COMMAND>'.

Disable Magnetics:

DISMAG - Hex Code X'F1'

This command causes the 4778 to turn off the 'Ready' indicator. The command is acknowledged by sending an 'OPCOMP' to the Host.

If a card is swiped while the slot is disabled, the 4778 will ignore the action and the 'ATTENTION' status will not be issued.

```
Host:   <COMMAND>
4778:         <OPCOMP>
```

Echo Data:

ECHO - Hex Code X'E8'

The 'ECHO' command causes the 4778 to echo data bytes back to the Host. Upon receiving 'ECHO' the 4778 sends a header, 'DAV', to the Host and then waits for the Host to send a data byte. The data byte is sent back to the Host as soon as it is received. If the data was received in error, the byte will be echoed as it was received (parity regenerated). The Host must wait until it receives the previous byte before sending another. The 4778 echoes every data byte until the Host terminates the command by sending an 'EOP' (the data byte cannot equal a command). Upon receiving the 'EOP' the 4778 then responds with an 'OPCOMP').

Host: <COMMAND> <DATA> ...<DATA> <EOP>
4778: <HEADER> <DATA> <DATA> <OPCOMP>

Enable Magnetics:

ENMAG - Hex Code X'F0'

This command 'ENABLE MAGNETICS' causes the 4778 to enable all available track read heads for reading magnetic media. The command is acknowledged by sending an 'OPCOMP' to the Host.

The 4778 will turn the 'Ready' indicator on. At this time swiping a card will load the internal storage with the data read from the magnetic media.

After acknowledging the command the 4778 waits for a card to be swiped. When a card is detected the 4778 waits for the card. Then takes control of the internal storage, then turns the 'Ready' indicator off. After taking control of the internal storage the 4778 sends an 'ATTENTION' to the Host. The Host then issues a 'SENSE' command and the 4778 responds with the 'MDR' status bit set to indicate that magnetic data is available.

Host: <COMMAND>
4778: <OPCOMP>

End of Operation:

EOP - Hex Code X'F2'

This command should be used to terminate the 'RDRBX', 'WRRDX' 'ECHO', 'SENSE', and 'WRIND' commands. If no command is pending when the 'EOP' is received it is acknowledged by the 4778 with an 'OPCOMP'.

Host: <COMMAND>
4778: <OPCOMP>

Read Magnetics Read Buffer 1:

RDRB1A - Hex Code X'F6'

This command causes the 4778 to take control of the internal storage associated with the track 1 read hardware. It will then start assembling data from internal storage (track 1 read buffer) into 8-bit characters starting with the first bit in the storage. Data in the storage is assumed to be arranged so that the least significant bit of a character is the first bit read.

Before sending any data, the 4778 will respond with a header, 'DAV'. The Host will then send a 'NEXT'. Upon receiving the 'NEXT', the 4778 will start assembling characters from storage. When a byte is assembled, the 4778 will send it to the Host. The Host will then respond with either a 'NEXT', 'RESEND' or an 'EOP'. The 4778 will assemble data from the internal storage associated with track 1 read hardware until the Host sends an 'EOP'. Upon receiving the 'EOP', the 4778 will send an LRC to the Host. The 4778 calculates this LRC from the data that it sent to the Host. This LRC is added by the 4778 and is not necessarily the same as the LRC read from the stripe. After receiving the LRC, the Host should send an 'EOP' to which the 4778 will respond with an 'OPCOMP'.

> Note: If the 4778 is expecting a 'NEXT', 'RESEND' or an 'EOP' but the byte received is neither then the 4778 will treat it as a 'NEXT'.

Reading from internal storage does not clear the internal storage.

```
Host:  <COMMAND>      <NEXT>      ...<EOP>   <EOP>
4778:           <HEADER>    <DATA>       <LRC>   <OPCOMP>
```

Read Magnetics Read Buffer 2:

RDRB2A - Hex Code X'F7'

This command causes the 4778 to take control of the internal storage associated with the track 2 read hardware. It will then start assembling data from internal storage (track 2 read buffer) into 8-bit characters starting with the first bit in the storage. Data in the storage is assumed to be arranged so that the least significant bit of a character is the first bit read.

Before sending any data the 4778 will respond with a header, 'DAV'. The Host will then send a 'NEXT'. Upon receiving the 'NEXT', the 4778 will start assembling characters from storage. When a byte is assembled, the 4778 will send it to the Host. The Host will then respond with either a 'NEXT', 'RESEND' or an 'EOP'. The 4778 will assemble data from the internal storage associated with track 2 read hardware until the Host sends an 'EOP'. Upon receiving the 'EOP', the 4778 will send an LRC to the Host. The 4778 calculates this LRC from the data that it sent to the Host. This LRC is added by the 4778 and is not necessarily the same as the LRC read from the stripe. After receiving the LRC, the Host should send an 'EOP' to which the 4778 will respond with an 'OPCOMP'.

> Note: If the 4778 is expecting a 'NEXT', 'RESEND' or an 'EOP' but the byte received is neither then the 4778 will treat it as a 'NEXT'.

Reading from internal storage does not clear the internal storage.

```
Host:  <COMMAND>      <NEXT>      ...<EOP>   <EOP>
4778:           <HEADER>    <DATA>       <LRC>   <OPCOMP>
```

Read Magnetics Read Buffer 3:

RDRB3A - Hex Code X'F9'

This command causes the 4778 to take control of the internal storage associated with the track 3 read hardware. It will then start assembling data from internal storage (track 3 read buffer) into 8-bit characters starting with the first bit in the storage. Data in the storage is assumed to be arranged so that the least significant bit of a character is the first bit read.

Before sending any data, the 4778 will respond with a header, 'DAV'. The Host will then send a 'NEXT'. Upon receiving the 'NEXT' the 4778 will start assembling characters from storage. When a byte is assembled, the 4778 will send it to the Host. The Host will then respond with either a 'NEXT', 'RESEND' or an 'EOP'. The 4778 will assemble data from internal storage '3' until the Host sends an 'EOP'. Upon receiving the 'EOP', the 4778 will send an LRC to the Host. The 4778 calculates this LRC from the data that it sent to the Host. This LRC is added by the 4778 and is not necessarily the same as the LRC read from the stripe. After receiving the LRC, the Host should send an 'EOP' to which the 4778 will respond with an 'OPCOMP'.

> Note: If the 4778 is expecting a 'NEXT', 'RESEND' or an 'EOP' but the byte received is neither then the 4778 will treat it as a 'NEXT'.

Reading from internal storage does not clear the internal storage.

Host:	<COMMAND>		<NEXT>		...<EOP>		<EOP>	
4778:		<HEADER>		<DATA>		<LRC>		<OPCOMP>

Read Magnetics Read Buffer 1 SOM:

RDRB1S - Hex Code X'FA'

Upon receiving this command, the 4778 takes control of the internal storage associated with track 1 read hardware. The 4778 then searches the data in the (track 1) internal storage for the IATA 'SOM' character. The 4778 sends a header to the Host to indicate whether the 'SOM' was found or the end of the internal storage was reached,

If the 4778 reaches the end of the internal storage before finding the 'SOM' it will automatically terminate the command, indicated by the 'OPCOMP' being sent to the Host.

After the Host receives the 'DAV' header, it will then send a 'NEXT', 'EOP' or a 'RESEND'. Upon receiving the 'NEXT', the 4778 will start assembling and transmitting 7-bit characters (starting with the SOM). The 4778 will transmit one character at a time waiting for a 'NEXT' before sending another. The 4778 will continue to send data to the Host until the Host sends an 'EOP' (in place of the 'NEXT'). Upon receiving the 'EOP' the 4778 will send to the Host an LRC which the 4778 calculated from the data bytes sent to the Host. This LRC is added by the 4778 and is not necessarily the same as the LRC read from the stripe. After receiving the LRC the Host should send a 'EOP' to which the 4778 will respond with an 'OPCOMP'.

> Note: If the 4778 is expecting a 'NEXT', 'RESEND' or an 'EOP' but the byte received is neither then the 4778 will treat it as a 'NEXT'.

Reading from internal storage does not clear the internal storage.

Figure 5-1. Track 1 with SOM

Track	Valid SOM(s)	Data Type
1	100 0101	IATA

Host: <COMMAND> <NEXT> ...<EOP> <EOP>
4778: <HEADER> <DATA> <LRC> <OPCOMP>

Read Magnetics Read Buffer 2 SOM:

RDRB2S - Hex Code X'FB'

Upon receiving this command the 4778 takes control of the internal storage associated with track 2 read hardware. The 4778 then searches the data in (track 2) internal storage for the ABA 'SOM' character. The 4778 sends a header to the Host to indicate whether the 'SOM' was found or the end of the internal storage was reached,

If the 4778 reaches the end of the internal storage before finding the 'SOM' it will automatically terminate the command, indicated by the 'OPCOMP' being sent to the Host.

After the Host receives the 'DAV' header it will then send a 'NEXT', 'EOP' or a 'RESEND'. Upon receiving the 'NEXT', the 4778 will start assembling and transmitting 5-bit characters (starting with the SOM). The 4778 will transmit one character at a time waiting for a 'NEXT' before sending another. The 4778 will continue to send data to the Host until the Host sends an 'EOP' (in place of the 'NEXT'). Upon receiving the 'EOP' the 4778 will send, to the Host, an LRC which the 4778 calculated from the data bytes sent to the Host. This LRC is added by the 4778 and is not necessarily the same as the LRC read from the stripe. After receiving the LRC the Host should send an 'EOP' to which the 4778 will respond with an 'OPCOMP'.

> Note: If the 4778 is expecting a 'NEXT', 'RESEND' or an 'EOP' but the byte received is neither then the 4778 will treat it as a 'NEXT'.

Reading from internal storage does not clear the internal storage.

Figure 5-2. Track 2 with SOM

Track	Valid SOM(s)	Data Type
2	0 1011	ABA

Host: <COMMAND> <NEXT> ...<EOP> <EOP>
4778: <HEADER> <DATA> <LRC> <OPCOMP>

Read Magnetics Read Buffer 3 SOM:

RDRB3S - Hex Code X'FD'

Upon receiving this command the 4778 takes control of the internal storage associated with the track 3 read hardware. The 4778 then searches the data in (track 3) internal storage for an ABA or DIN 'SOM' character. The 4778 sends a header to the Host to indicate whether a 'SOM' was found or the end of the internal storage was reached,

If the 4778 reaches the end of the internal storage before finding the 'SOM' it will automatically terminate the command, indicated by the 'OPCOMP' being sent to the Host.

After the Host receives the 'DAV' header it will then send a 'NEXT', 'EOP' or a 'RESEND'. Upon receiving the 'NEXT' the 4778 will start assembling and transmitting 5-bit characters (starting with the SOM). The 4778 will transmit one character at a time waiting for a 'NEXT' before sending another. The 4778 will continue to send data to the Host until the Host sends an 'EOP' (in place of the 'NEXT'). Upon receiving the 'EOP' the 4778 will send to the Host an LRC which the 4778 calculated from the data bytes sent to the Host. This LRC is added by the 4778 and is not necessarily the same as The LRC read from the stripe. After receiving the LRC the Host should send an 'EOP' to which the 4778 will respond with an 'OPCOMP'.

> Note: If the 4778 is expecting a 'NEXT', 'RESEND' or an 'EOP' but the byte received is neither then the 4778 will treat it as a 'NEXT'.

Reading from internal storage does not clear the internal storage.

Figure 5-3. Track 3 with SOM

Track	Valid SOM(s)	Data Type
3	0 1101	Din
3	0 1011	Thrift

```
Host:  <COMMAND>      <NEXT>     ...<EOP>   <EOP>
4778:            <HEADER>    <DATA>       <LRC>   <OPCOMP>
```

Resend:

RESEND - Hex Code X'F5'

'RESEND' can be used in place of any transmission, by the Host or the 4778, except a data byte sent from the 4778. It will cause the receiver to repeat the last byte that is transmitted. Data bytes resent in response to this command are not used in the LRC calculation.

```
Host:  ...<REQUEST>      ... or ...        <DATA> ...
4778:              <DATA>           <REQUEST>
```

Restart Magnetics:

RESTART - Hex Code X'EE'

The 'RESTART' command causes the 4778 to return to it's initial state. The interface is re-initialized using the same procedure as a power-on reset.

'RESTART' is similar to 'STEST' in that it is a software reset but 'RESTART' does not perform a self-test. The test results of the previous self-test are retained.

```
Host:  <COMMAND>  ... <DATA(I)>       ... <EOP>
4778:                  <DATA(I)>       <OPCOMP>
```

Sense Attention:

SENSE - Hex Code X'E9'

The 'SENSE' command is used to determine the reason for an 'ATTENTION' or to read the results of a self-test. The 4778 will respond with a header, 'DAV'. The Host then sends a 'NEXT' and the 4778 responds with the status byte. Upon receiving the status byte the Host sends an 'EOP'. The Host then sends an 'EOP' and the 4778 returns an 'OPCOMP'.

If instead of the first 'EOP' the Host sends a 'NEXT' then the 4778 will respond with the extended status byte.

Figure 5-4. Sense Status Byte

BIT7	**BIT6**	**BIT5**	**BIT4**	**BIT3**	**BIT2**	**BIT1**	**BIT0**
ID Bit 7	ID Bit 6	ID Bit 5	Test Results Bit 4	Test Results Bit 3	Illegal (ILLGL)	Magnetic Data Read (MDR)	Request (REQ)

REQ = 1 An 'ATTENTION' was sent since the last 'SENSE' command was received.
Reset by 'SENSE'

MDR = 1 Valid card data is in the internal storage. Reset by a power-on
reset, 'STEST', 'RESTART', 'ENMAG', or 'DISMAG'.

ILLGL = 1 Card swipe was started before the slot was enabled. Reset by 'SENSE'.

TEST RESULTS

01 = Test Passed
10 = Test Failed

Figure 5-5. Identification Bit Definitions

ID BITS	Functions
0 0 0	Extended Status Indicator
0 0 1	Track 1 Reader
	Track 2 Reader
0 1 0	Track 2 Reader
	Track 3 Reader
	Track 2 210 BPI Encoder
	Track 3 210 BPI Encoder
0 1 1	Track 1 Reader
	Track 2 Reader
	Track 1 210 BPI Encoder
	Track 2 75 BPI Encoder
1 0 0	Track 2 Reader
	Track 3 Reader

Note: ID = 000 is reserved as an extended status indicator. The extended status is used to provide additional information about the device capabilities. The extended status is in the form of an additional byte of data sent in response to the 'SENSE' command. Refer to the following figure for bit definitions of the extended status byte.

Figure 5-6. Extended Status Byte

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
Not Defined	Read Track 1 (R1)	Read Track 2 (R2)	Read Track 3 (R3)	Not Defined	Encode Track 1 (E1)	Encode Track 2 (E2)	Encode Track 3 (E3)

A '1' indicates the device has the ability perform the corresponding function.

```
Host:  <COMMAND>      <NEXT>      <EOP>   <EOP>
4778:            <HEADER>    <STATUS>   <LRC>   <OPCOMP>
```

. . . OR . . .

```
Host:  <COMMAND>      <NEXT>      <NEXT>      <EOP>    <EOP>
4778:          <HEADER>    <STATUS>    <X-STAT>    <LRC>    <OPCOMP>
```

Self Test:

STEST - Hex Code X'EF'

This command is equivalent to a software reset. Upon receiving the 'STEST' command the 4778 then goes into a self-test routine. After the completion of this command The 4778 enters the 'ECHO' mode, see section 'ECHO' command. If the 4778 is capable of responding it will be in 'ECHO' mode within 0.5 seconds after the 'STEST' command is received.

The Host initializes the magnetics interface and the 4778 by using the same procedure as in the power-on reset.

```
Host:  <COMMAND> ...  <DATA>        ... <EOP>
4778:                   <DATA>          <OPCOMP>
```

Write Indicators:

WRIND - Hex Code X'DF'

This command causes the 4778 to take bits 6 and 5 of the data byte sent by the Host and output them to the indicator drivers. The 4778 will request the byte of the command by sending a 'NEXT' to the Host. The 4778 continues to request data and output it to the indicators until an 'EOP' is received. The 4778 responds to the 'EOP' with an LRC (calculated from the data). Upon receiving the LRC the Host sends an 'EOP' and the 4778 returns an 'OPCOMP'.

The form of the second byte is 0AB0 0000, where A and B are the levels to be output to the drivers (a 1 turns the indicator on).

Bit 6 is used to set or reset the In Process indicator and bit 5 is used to set or reset the Check indicator.

```
Host:  <COMMAND>    <DATA>    <EOP>    <EOP>
4778:           <NEXT>    <NEXT>    <LRC>    <OPCOMP>
```

Write Magnetics Read Buffer 1:

WRRB1 - Hex Code X'E5'

This command is used to write data to the internal storage associated with the track 1 read hardware. The Host must provide a second byte which specifies the number of bits per character (0000 0101 ==> 5 bits per character). The number of bits per character must be between 1 and 7, inclusive. Bit 7 of data transmitted to the 4778 must always be '0'. The Host must provide all leading zeros in addition to the data. The data is to be right justified, with bit 0 containing the least significant bit.

Upon receiving this command the 4778 will take control of the internal storage. It will then start writing the data provided by the Host to the internal storage (track 1 read buffer).

After all data has been accepted by the 4778 the Host must terminate the operation by sending an 'EOP' to the 4778.

The 4778 will request all bytes of data from the Host (after the command) by sending a 'NEXT' to the Host. The first byte after the command will be the number of bits per character. The command is terminated when the Host sends an 'EOP' to the 4778. When the 'EOP' is received the 4778 adds the trailing zeros to fill the shift register and then acknowledges the command. The command is acknowledged by sending an LRC which the 4778 calculated from the data (including the byte which specified the number of bits per character). After receiving the LRC the Host should respond with an 'EOP'. upon receiving the 'EOP' the 4778 will send an 'OPCOMP' to the Host.

Note: Data which is in error when the 4778 receives it, i.e., the 4778 responds with a 'RESEND', is not used in calculating the LRC.

Note: The buffer data cannot be read after issuing the "ENABLE MAGNETICS" command.

```
Host:  <COMMAND>    <DATA>      ... <EOP>   <EOP>
4778:          <NEXT>    <NEXT>         <LRC>   <OPCOMP>
```

Write Magnetics Read Buffer 2:

WRRB2 - Hex Code X'E6'

This command is used to write data to the internal storage associated with track 2 read hardware. The Host must provide a second byte which specifies the number of bits per character (0000 0101 ==> 5 bits per character). The number of bits per character must be between 1 and 7, inclusive. Bit 7 of data transmitted to the 4778 must always be '0'. The Host must provide all leading zeros in addition to the data. The data is to be right justified, with bit 0 containing the least significant bit.

Upon receiving this command the 4778 will take control of the internal storage. It will then start writing the data provided by the Host to the internal storage (track 2 read buffer).

After all data has been accepted by the 4778 the Host must terminate the operation by sending an 'EOP' to the 4778.

The 4778 will request all bytes of data from the Host (after the command) by sending a 'NEXT' to the Host. The first byte after the command will be the number of bits per character. The command is terminated when the Host sends an 'EOP' to the 4778. When the 'EOP' is received the 4778 adds the trailing zeros to fill the shift register and then acknowledges the command. The command is acknowledged by sending an LRC which the 4778 calculated from the data (including the byte which specified the number of bits per character). After receiving the LRC the Host should respond with an 'EOP'. Upon receiving the 'EOP' the 4778 will send an 'OPCOMP' to the Host.

Note: Data which is in error when the 4778 receives it, i.e., the 4778 responds with a 'RESEND', is not used in calculating the LRC.

Note: The buffer data cannot be read after issuing the "ENABLE MAGNETICS" command.

```
Host:  <COMMAND>    <DATA>      ... <EOP>   <EOP>
4778:        <NEXT>    <NEXT>        <LRC>   <OPCOMP>
```

Write Magnetics Read Buffer 3:

WRRB3 - Hex Code X'E7'

This command is used to write data to the internal storage associated with the track 3 read hardware. The Host must provide a second byte which specifies the number of bits per character (0000 0101 ==> 5 bits per character). The number of bits per character must be between 1 and 7, inclusive. Bit 7 of data transmitted to the 4778 must always be '0'. The Host must provide all leading zeros in addition to the data. The data is to be right justified, with bit 0 containing the least significant bit.

Upon receiving this command the 4778 will take control of the internal storage. It will then start writing the data provided by the Host to the internal storage (track 3 read buffer).

After all data has been accepted by the 4778 the Host must terminate the operation by sending an 'EOP' to the 4778.

The 4778 will request all bytes of data from the Host (after the command) by sending a 'NEXT' to the Host. The first byte after the command will be the number of bits per character. The command is terminated when the Host sends an 'EOP' to the 4778. When the 'EOP' is received the 4778 adds the trailing zeros to fill the shift register and then acknowledges the command. The command is acknowledged by sending an LRC which the 4778 calculated from the data (including the byte which specified the number of bits per character). After receiving the LRC the Host should respond with an 'EOP'. Upon receiving the 'EOP' the 4778 will send an 'OPCOMP' to the Host.

Note: Data which is in error when the 4778 receives it, i.e., the 4778 responds with a 'RESEND', is not used in calculating the LRC.

```
Host:  <COMMAND>    <DATA>      ... <EOP>   <EOP>
4778:        <NEXT>    <NEXT>        <LRC>   <OPCOMP>
```

Chapter 6. 4778 Device Selection Command Reference

Figure 6-1. 4778 Device Selection Commands

Reference	Hex Code	Type	Originator
“ ACTIVATE EPP - Hex Code X'B1'” on page 3-2	X'B1'	COMMAND	Host
“ ACTIVATE MSR - Hex Code X'B2'” on page 3-2	X'B2'	COMMAND	Host
“ Hex Code X'B0' and Hex Code x'B3' to Hex Code x'BF'” on page 3-3	X'B0' and X'B3'-X'BF'	COMMAND	Host
EPP ACTIVE	X'B1'	STATUS	4778
MSR ACTIVE	X'B2'	STATUS	4778
“ Hex Code X'B0' and Hex Code x'B3' to Hex Code x'BF'” on page 3-3	X'B0' & X'B3'-X'BF'	STATUS	I/O DEVICE
“ ATTENTION EPP - Hex Code X'C1'” on page 3-3	X'C1'	COMMAND	4778
“ ATTENTION MSR - Hex Code X'C2'” on page 3-3	X'C2'	COMMAND	4778
“ Hex Code X'C0' and Hex Code X'C3' to Hex Code x'CF'” on page 3-3	X'C0' and X'C3'-'CF'	COMMAND	I/O DEVICE
“ Hex Code X'FF'” on page 3-3	X'FF'	COMMAND	Host

Chapter 7. 4778 EPP Device Command Reference

Figure 7-1. 4778 EPP Device Commands

Reference	Hex Code	Type	Originator
"Read Status" on page 4-2	X'01'	Syncronous	Host
"Read Nonencrypted (Clear) Keypad Data" on page 4-3	X'02'	Asyncronous	Host
"Read Data" on page 4-4	X'03'	Syncronous	Host
"Read Serial Number" on page 4-4	X'04'	Syncronous	Host
"Abort" on page 4-5	X'05'	Syncronous	Host
"Resend" on page 4-5	X'06'	Syncronous	Host/4778
"Set PIN Keypad Mode" on page 4-5	X'07'	Syncronous	Host
"Display A String" on page 4-6	X'08'	Syncronous	Host
"Enter Master Key" on page 4-7	X'10'	Asyncronous	Host
"Load Master Key" on page 4-7	X'11'	Syncronous	Host
"Load Key" on page 4-8	X'12'	Syncronous	Host
"Load ICV" on page 4-9	X'13'	Syncronous	Host
"Load PIN Verification Parameters" on page 4-10	X'15'	Syncronous	Host
"Create PIN Block" on page 4-10	X'2x'	Asyncronous	Host
"Verify PIN / Create PIN Offset Data" on page 4-12	X'3x'	Asyncronous	Host
"Generate Message Authentication Code" on page 4-13	X'6x'	Syncronous	Host
"Verify Message Authentication Code" on page 4-15	X'7x'	Syncronous	Host

Chapter 8. 4778 MSR Device Command Reference

Figure 8-1. 4778 MSR Device Commands, Status, Headers

Reference	Hex Code	Type	Originator
DAV	X'85'	HEADER	4778
RESERVED	X'8E'	HEADER	4778
OPCOMP	X'E0'	STATUS	4778
NEXT	X'FE'	STATUS	Host/4778
“ RESEND - Hex Code X'F5'” on page 5-6	X'F5'	COMMAND	Host/4778
“ DISMAG - Hex Code X'F1'” on page 5-1	X'F1'	COMMAND	Host
“ ECHO - Hex Code X'E8'” on page 5-2	X'E8'	COMMAND	Host
“ ENMAG - Hex Code X'F0'” on page 5-2	X'F0'	COMMAND	Host
“ EOP - Hex Code X'F2'” on page 5-2	X'F2'	COMMAND	Host
“ RESTART - Hex Code X'EE'” on page 5-7	X'EE'	COMMAND	Host
“ SENSE - Hex Code X'E9'” on page 5-7	X'E9'	COMMAND	Host
“ STEST - Hex Code X'EF'” on page 5-9	X'EF'	COMMAND	Host
“ WRIND - Hex Code X'DF'” on page 5-9	X'DF'	COMMAND	Host
“ RDRB1A - Hex Code X'F6'” on page 5-2	X'F6'	COMMAND	Host
“ RDRB2A - Hex Code X'F7'” on page 5-3	X'F7'	COMMAND	Host
“ RDRB3A - Hex Code X'F9'” on page 5-4	X'F9'	COMMAND	Host
“ RDRB1S - Hex Code X'FA'” on page 5-4	X'FA'	COMMAND	Host
“ RDRB2S - Hex Code X'FB'” on page 5-5	X'FB'	COMMAND	Host
“ RDRB3S - Hex Code X'FD'” on page 5-6	X'FD'	COMMAND	Host
“ WRRB1 - Hex Code X'E5'” on page 5-9	X'E5'	COMMAND	Host
“ WRRB2 - Hex Code X'E6'” on page 5-10	X'E6'	COMMAND	Host
“ WRRB3 - Hex Code X'E7'” on page 5-11	X'E7'	COMMAND	Host
RESERVED	X'FC'	COMMAND	Host