# MSR120

Magnetic Stripe Card Reader

Universal Serial Bus (USB) Interface

# Programmer's Manual

**Document PM010**

**Revision 1.1**

**09 Feb. 2001**

# NOTICE

The issuer of this manual has made every effort to provide accurate information contained in this manual. The issuer shall not be held liable for any technical and editorial omissions or errors made herein; nor for incidental consequential damages resulting from the furnishing, performance or use of this material.

 The information provided in this manual is subject to change without notice.

# AGENCY APPROVED

This Equipment, MSR120, had been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment.

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. Operation of this equipment in a residential area is also likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

This MSR120 also had been tested and found to comply with the agency requirements of specification for CE mark Class A, UL, and cUL.

# WARNING

You are cautioned that changes or modifications not expressly approved by the party responsible for compliance could void your authority to operate the equipment.

# WARRANTY

This product is served under one-year warranty of defects in material and functionality to the original purchasers. Within the warranty period, if the product found to be defective will be repaired or replaced. This warranty applies to the products only under the normal use of the original purchasers, and in no circumstances covers incidental or consequential damages through consumers' misuse or modification of the product.

# PREFACE

This manual provides detailed information relating to the overall operational, electrical, mechanical, environmental and functional aspects of the MSR120. This document should be read and understood prior to initial operation of the product.

For ease of installation and programming use, we have addressed everything from its attractive features to its various configurations.

When designing the MSR120, we selected what we feel are the most useful features and functions. If in some cases you find that your specific needs differ from our existing products, we welcome your comments and suggestions.

Custom-designed models are also available.

If further questions do arise, please call for technical support, our FAE will assist you in any way we can.

# Table of Contents

# Section 1 General Description

This section presents general information about the basic characters of the MSR120.

## ➢ Features

The MSR120 provides the following features:

| | |
|---|---|
| 1 | Light weight: 120g |
| 2 | Compact size: 100L*34W*28H (mm) |
| 3 | LED and Buzzer indicators indicate card status |
| 4 | Low power consumption |
| 5 | Single, dual, or triple track versions allow for reading all types of magnetic cards, including credit cards and drivers licenses. |

## ➢ Application

This Magnetic Stripe Reader is design to read high or low coercive magnetic cards. It can decode/verify up to 3 tracks of data simultaneously. This product communicates with a host computer or other terminal using a standard USB interface. Because of the transmitting protocol of MSR120 is more precise, it is suitable for using in financial industry.

## ➢ Function

### Self Test

Whenever the reader experiences a reset cycle, a self-test is performed. The reader will respond with ":" and the LED will turn green if the entire test is successful. Otherwise, the LED will turn orange and no response will be generated.

Table 1-1. Self Test

| Indication | | Cause | Post Condition |
|---|---|---|---|
| LED | Buzzer | | |
| Green | 1 Beep | Test Success | Responds with ":" |
| Orange | | EEPROM Failed | Hang |
| Orange | | Internal ROM Failed | Hang |

### Transmission Reset by Hardware

When the reader cannot communicate with the host, user can set the switch S1-1 on and power on the reader. The reader will beep twice and response the baud rate, parity, CTS and RTS default setting, but the data inside EEPROM will not be changed.

## Reading

The reader can read magnetic data form any available track encoded per ISO 7810, 7811, AAMVA, CA old DMV, JIS. The host can request the read data from the reader with commands. For details and examples of commands and responses, refer to section 4.

## Reading Customized Data

The interface can read customized encoded magnetic data. Data integrity is not verified when reading customized data. Customized data is not formatted into ASCII characters prior to transmission to the host.

## Self –Arm Mode

The default reader configuration is the "Self-Arm Mode", which allows the magstripe functions to run automatically, reporting magstripe activity to the host without instruction from the host. In the Self-Arm mode, the reader also can accept commands from host. However, the reader can be configured to only "Host Polled Mode" by disabling Self-Arm mode. The "Host Polled Mode" allows the magstripe functions to run by commands. The conditions of indicators when the reader is in Self-Arm mode are shown below.

Table 1-2. Self-Arm Mode

| Indication | | Cause |
|---|---|---|
| LED | Buzzer | |
| Off | | Card Swipe |
| Orange | 2 Beeps | 1 or 2 Tracks Read Miss |
| Red | 3 Beeps | Read Error |
| Green | 1 Beep | Read Success |

When the reader is in the Self-Arm mode, the green LED indicates the reader is waiting for accepting card. While card is moving, the LED will turn off. When one or two tracks read miss, the LED shows orange for 1 second and the buzzer beeps twice. If tracks read error, the LED turns red and beeps thrice. When the reader is in the Host Polled mode, the green LED keeps flashing.

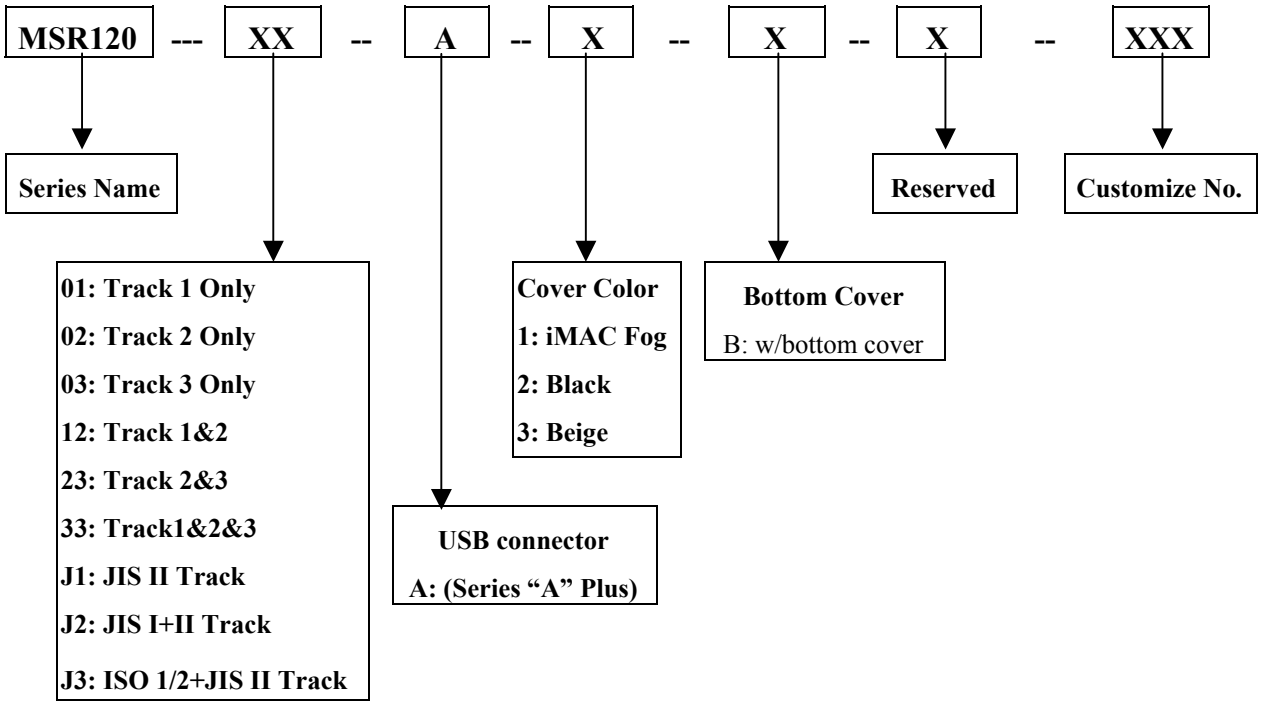
## Transmitting Data Block of Tracks

Each track could divide into 1-3 channels (blocks) for transmission. For example, user could set "Channel A" of track1 format as: ISO standard card track1 $15^{th}$-$30^{th}$ characters. If the set data is different from the data after swiping, the reader will transmit completed data without limitation. The commands of setting channels are shown in section 5.

# ➢ Model Description

The definitions of MSR120 model number are shown in Figure 1-1.

Figure 1-1. Model Description

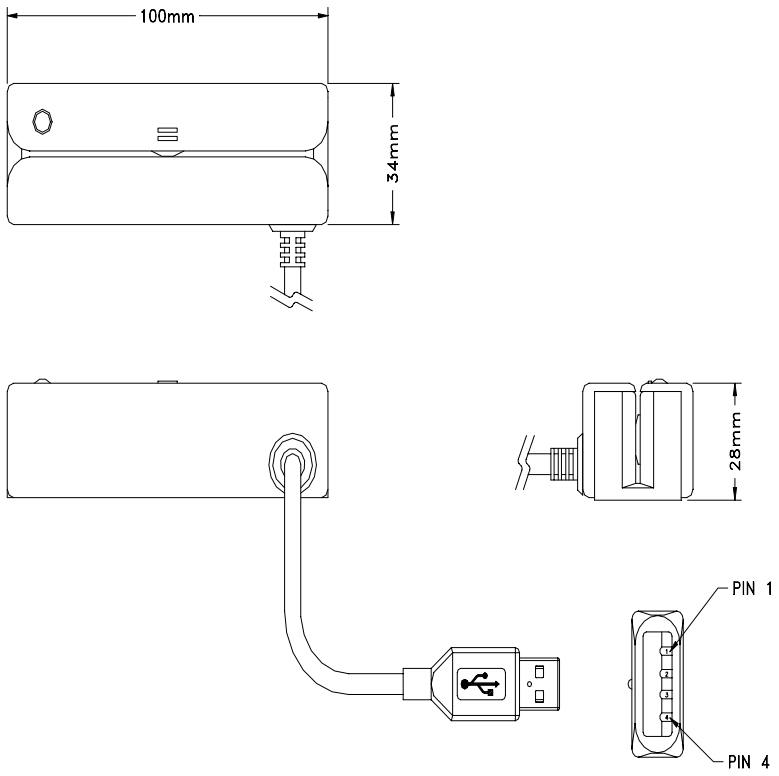


# Section 2 Configurations

This section shows the dimensions, accessories and setup for the MSR120.

## ➢ Dimensions of MSR120

Figure 2-1 Dimensions of MSR120




## ➢ Accessories of MSR120

The following accessories should be supplied along with MSR120. Make sure all the following accessories are contained in your package.

1. Signal cable (USB Connector, 1.7M)
2. Programmer's manual

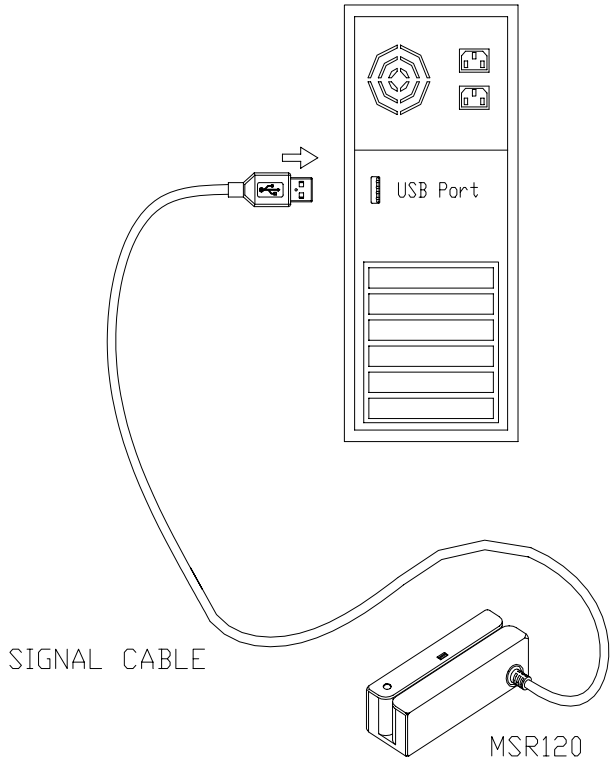
## ➢ Installation

1. Power off your PC system.
2. Connect PC and MSR120 as below.

3. Connect USB connector of signal cable to a free USB port. Connect the connector to MSR120.

4. Power on your PC system.

Figure 2-2 Installing MSR120



# Section 3 Technical Specifications

## ➢ Card Specifications

### Card Type

ISO standard card

CA old DMV

AAMVA

JIS

Read high or low coercive magnetic stripes (300-4000oe)

Read triple track 7 BPC & 210 BPI cards

### Thickness

0.76 mm ±0.08 mm

### Card Format

Track 1 & 3:210 bpi

Track 2: 75/210 bpi

JIS II 210 bpi

*Note: The card data output sequence for Model J2 is ISO track 2 prior to JIS II.*

*The card data output sequence for Model J3 is ISO track 1 & 2 prior to JIS II.*

### Card Operation Speed

Table 3-1. Card Operation Speed

| Test Card | Speed (IPS) |
|---|---|
| ISO standard card | 5-55 |
| * Jitter | 5-50 |
| ** Low Amplitude | 5-50 |

*Note: *Jitter card: Reliable reading of magnetic stripes encoded with bit cell length variations within +/-12% of normal as defined by ISO 7811.*

***Low amplitude: Reliable reading of magnetic stripes encoded at 60% or more of the encoding amplitude as defined by ISO 7811.*

## ➢ Mechanical Specifications

### Body Material

ABS 94V-0

### Dimension

Length: 100mm

Width: 34mm

Height: 28mm

### Weight

120g

### Magnetic Head Life

500K swipes Min. 1M option

## ➢ Electrical Specifications

### Power Required

DC 5V±10%

### Power Consumption

120mA Max. in normal condition

### Communication

Standard USB signal levels (comply with USB specification v 1.1)

### Dielectric Strength

250VDC for 1 minute

### Insulation Resistance

10M Ohms min. at 250VDC

## ➢ Environmental Specifications

### Temperature

Operating: -10-50℃

Storage: -30-70℃

### Humidity

Operating: 10-85% (non condensing)

Storage: 10-90% (non condensing)

# ➢ USB Connector Termination Assignment

Table 3-2. Connector Assignment

| Contact Number | Signal Name | Typical Wiring Assignment |
|---|---|---|
| 1 | Vbus | Red |
| 2 | D- | White |
| 3 | D+ | Green |
| 4 | GND | Black |
| Shell | Shield | Drain Wire |

# ➢ Communication

The interface receives and transmits serial bit stream at voltage levels compatible with the USB specification.

## Transmission Protocol

The user may select from three different protocols: Protocol 0, 1, and 2.

Upon reset, the reader sends the power-on response ":", depending upon the configuration setting. The reader then configures itself to the protocol of the first command from the host. From this point on, the protocol is unchangeable until a reset occurs.

### Protocol 0

In Protocol 0, all characters are transmitted and received using exactly the characters listed in section 4. There are no headers and Block Check Characters (BCC). Protocol 0 presumes no transmission errors. If the host detects an error, it may request a retransmission.

Table 3-3. Example for Protocol 0

| Host Command | Reader Response | Comment |
|---|---|---|
| P | | Ready to read |
| | ^ | Reader ACK |

### Protocol 1

In Protocol 1, all messages are preceded by the ASCII character <STX> and terminated with the ASCII character <ETX>, followed by a one byte <BCC>. <BCC> is an XOR of the 7 data bits, excluding parity, of each character in the entire message, including <STX>.

Format: <STX><MESSAGE><ETX><BCC>

where STX=02Hex and ETX=03Hex.

Table 3-4. Example for Protocol 1

| Host Command | Reader Response | Comment |
|---|---|---|
| 02h 50h 03h 51h | | Ready to read |
| | 02h 5Eh 03h 5Fh | Reader ACK |

**Protocol 2**

In Protocol 2, all messages are preceded by the ASCII character <SOH>, followed by a one byte reader address, one byte character count and terminated with a one byte <BCC>. The <BCC> is an XOR of the characters (8 bits) in the entire message, including <SOH>.

Format: <SOH><ADDRESS><00Hex><COUNT><MESSAGE><BCC>

or

<SOH><ADDRESS><00Hex><00Hex><MESSAGE><EOT><BCC>

where STX=02Hex and ETX=03Hex.

Table 3-5. Example for Protocol 2

| Host Command | Reader Response | Comment |
|---|---|---|
| 01h 00h 00h 01h 50h 50h | | Ready to read |
| | 01h 00h 00h 01h 5Eh 5Eh | Reader ACK |

The <ADDRESS> field is for a multi-reader system. This function is not currently supported. The recommended value for this field is NULL (00Hex), however, any value will work.

If the value of <COUNT> fields are zero, an <EOT>, followed by the <BCC>, completes the message. The reader may, at its option, use NULL for COUNT when transmitting.

For Protocols 1 and 2, if the reader detects an error in an incoming transmission, it will respond with a "Communications Error" message. If the host detects a transmission error, it may request a retransmission. Both protocols enforce a 100mSec timeout between characters.

For all Protocols, the host may, at any time, stop/start the reader transmission by using software "handshake" (DC3/DC1) or hardware "handshake" (if enabled in EEPROM configuration) by controlling the CTS line.

## Transmission Format

**Data output format (Self-ARM mode)**

| | | | | | | |
|---|---|---|---|---|---|---|
| Protocol code | Tk1 prefix | Tk1 Data | Tk1 suffix | EOT | BCC | |
| Protocol code | Separator | Tk2 prefix | Tk2 Data | Tk2 suffix | EOT | BCC |
| Protocol code | Separator | Tk3 prefix | TK3 Data | Tk3 suffix | EOT | BCC |

**Read data for command**

Read tk1 data for command

| | | | | | |
|---|---|---|---|---|---|
| Protocol code | Tk1 prefix | Tk1 Data | Tk1 suffix | EOT | BCC |

Read tk2 data for command

| Protocol code | Tk2 prefix | Tk2 Data | Tk2 suffix | EOT | BCC |
|---|---|---|---|---|---|

Read tk3 data for command

| Protocol code | Tk3 prefix | Tk3 Data | Tk3 suffix | EOT | BCC |
|---|---|---|---|---|---|

*Note: Tk x data: SS (option) track data ES (option) LRC (option)*

# Section 4 COMMANDS and RESPONSES

This section describes the commands and responses available for the MSR120 series. Each item includes the ASCII, hexadecimal codes and comments paragraph. The comments paragraph provides an explanation of the command. The letter 'x' indicates a variable and the letter 'h' is an abbreviation of 'hexadecimal'.

All readers are capable of communicating in 3 protocols: 0, 1 and 2. If protocol 0 is selected, all commands and responses are as listed in this section. If protocol 1 is selected, the characters STX, ETX and BCC must be added to all transmissions. If protocol 2 is selected, the characters SOH, ADDRESS, COUNT and BCC must be added to all transmissions.

## ➢ Command — Host to Reader

### P (50h) — Ready to Read

Comments

| | |
|---|---|
| 1 | Clear buffers |
| 2 | Transmit "ACK" |
| 3 | Expect for card swipe |
| 4 | Transmit "ACK" after card swipe |

After an "Ready to Read" command is received and acknowledged, the only valid commands that will be accepted for execution are: "Abort" <ESC>, "Status" <$> and LED control commands. When the reader is ready for "Read on Insert", the "ACK" after card swipe will be transmitted after the rear sensor is activated. For read on reverse swipe, the "ACK" after card swipe will be transmitted after the front sensor is de-activated.

### p (70h) — Ready to Read

Comments

Same as 'P' command, except an extra response " ( " is reported when a media is detected through the read head. A " ) " response is reported when media detect goes inactive, and response " > " when no magstripe.

### Q (51h) — Transmit Standard Data, Track 1
### R (52h) — Transmit Standard Data, Track 2
### S (53h) — Transmit Standard Data, Track 3

Comments

| | |
|---|---|
| 1 | Process data in the read buffer for the specified track according to ISO, AAMVA and DVM format. |
| 2 | Transmit data in ASCII. |
| 3 | If error is detected, transmit proper error response "*". If no data transmit "+". (Refer to **Responses— Reader to Host** later in this section). |

### U (55h) — Transmit Customized Data, Track 1
### V (56h) — Transmit Customized Data, Track 2
### W (57h) — Transmit Customized Data, Track 3

Comments

| | |
|---|---|
| 1 | To request customized data with no "nulls" allowed, use the two bytes command: "transmit customized data" command, followed by an ASCII number (3-8), which specifies the number of bits per customized character. |
| 2 | Process data to the read buffer for the specified track, based upon the "number of bits" character. |

### u (75h) — Transmit Reverse Customized Data, Track 1
### v (76h) — Transmit Reverse Customized Data, Track 2
### w (77h) — Transmit Reverse Customized Data, Track 3

Comments

| | |
|---|---|
| 1 | To request customized data with no "nulls" allowed, use the two bytes command: "transmit customized data" command, followed by an ASCII number (3-8), which specifies the number of bits per customized character. |
| 2 | Process reversed-bit data (result of reverse swipe) to the read buffer for the specified track, based upon the "number of bits" character. |

### % (25h) — Retransmit

Comments

Request to retransmit the previous valid command except "P" and "p" commands.
*Note: Z (or z) won't be retransmitted.*

### # (23h) — Configuration Request

Comments

Transmit a byte, which represent configuration of the interface as follows:

Table 4-2. Interface Configuration

| | |
|---|---|
| Bit 0 | Track 1 read capability |
| Bit 1 | Track 2 read capability |
| Bit 2 | Track 3 read capability |
| Bit 3 | 0 |
| Bit 4 | 0 |
| Bit 5 | 1 |
| Bit 6 | 1 |
| Bit 7 | Parity (protocol 0 and 1 only), 0 (protocol 2) |

Where "1" bit means "capable of…" and "0" bit means "not capable of…".

### L (4Ch) — Green LED On
### l (6Ch) — Green LED Off
### M (4Dh) — Red LED On
### m (6Dh) — Red LED Off

Comments

| | |
|---|---|
| 1 | Turn green/red LED on/off, as specified. |
| 2 | Transmit "ACK". |

### ( (28h) — Green LED Flash
### ) (29h) — Red LED Flash

Comments

| | |
|---|---|
| 1 | Begin flashing the specified LED on and off (approximately 250mSec On and 250mSec Off). Continues flashing until changed by another LED command or by continued interrupt mode operation. |
| 2 | Transmit "ACK". |

### Z (5Ah) — One Long Beep
### z (7Ah) — One Short Beep

Comments

| | |
|---|---|
| 1 | The buzzer beeps a long (approx. 0.5mSec)/short (approx. 0.2mSec) beep. |
| 2 | Transmit "ACK". |

### DC3 (13h) — Pause Transmit

Comments

Stop transmitting data.

### DC1 (11h) — Resume Transmit

Comments

Resume transmission of data.

### T (54h) — Card Type Report

Comments

Transmit a byte, which represent the swiping card type.

| | |
|---|---|
| 1 (31H) | CA old DMV |
| 2 (32H) | AAMVA |
| 3 (33H) | ISO |
| 4 (34H) | JIS |
| 6 (36H) | Trade Show Card |
| 0 (30H) | No data (sending T command just right after power on) |

### ESC (1Bh) — Abort

Comments

| | |
|---|---|
| 1 | Abort command is issued after the first character of double character "read" command has been issued. |
| 2 | Abort command is issued after the reader has responded to command "P" or "p". (Reader would be waiting for card swipe to read). |
| 3 | No response to "Abort" command if the reader just power on and no command is received before. |

### 9 (39h) — Version Report

Comments

Transmit a string that includes the version number (8-digit) and its date (mmddyy).

### DEL (7Fh) — Warm Reset

Comments

Abort all current actions and cause the device to execute all initialization functions (device will respond exactly as it would for a "power on" cycle).

*Note: This command byte is not recognized as a command within data strings.*

### $ (24h) — Reader Status Request

Comments

Transmit one byte representing the reader status as follows:

Table 4-3. Reader Status

| Bit | 0 | 1 |
|---|---|---|
| 1 & 0 | 00:Green LED off<br>01:Green LED on<br>10:Green LED flash | |
| 3 & 2 | 00:Red LED off<br>01:Red LED on<br>10:Red LED flash | |
| 4 | Buzzer disable | Buzzer enable |
| 5 | No magstripe data | Magstripe data available |
| 6 | Not ready to read | Ready to read |
| 7 | Parity for protocols 0 &1 | Not used for protocol 2 |

## ➢ Response— Reader to Host

### ^ (5Eh) — ACK

Comments

Last command has been completed without an error condition, and ready for the next command.

### + (2Bh) — No Data

Comments

If host gets the response from reader, it may mean one of following:

| | |
|---|---|
| 1 | In case of a "standard format" read command, this response "+" means "no start sentinel (SS) found". |
| 2 | In case of a "read error" command, this response "+" means "no history of a previous read error". |

### * (2Ah) — Error

Comments

If host gets the response from reader, it may mean one of following:

| | |
|---|---|
| 1 | No end sentinel (ES). |
| 2 | Parity error. |
| 3 | LRC error. |

### ? (3Fh) — Communication Error

Comments

If host gets the response from reader, it may mean one of following:

| | |
|---|---|
| 1 | Bad parity. |
| 2 | Wrong BCC. |
| 3 | Receive character time-out (approximately 2mSec). |
| 4 | Message more than maximum character allowed. |

### ! (21h) — Invalid Command

Comments

Command issued by the host was not recognized or won't accept.

### : (3Ah) — Power On Report

Comments

The interface has completed its initialization cycle.

### ~ (7Eh) — Cannot Execute

Comments

Read or encode command cannot be executed due to lack of hardware in the device.

# Section 5 CONFIGURATION COMMANDS

This section describes the internal configuration commands available for the MSR120. Each item provides the ASCII, hexadecimal code and an explanation of the command. The letter "x" indicates a variable and the letter "h" is an abbreviation of "hexadecimal". The hardware related configuration commands need to do warm reset before becoming effective.

## ➢ Command Form

**<09Hex><ADDRESS><command counter Hex><COMMAND><BCC>**

*Note: Command Length is in hex value.*

### ESx (45h 53h x) ─ ES & SS Send Enable/Disable

x = E (45h enable) or D (44h disable)

Table 5-1. ES & SS Send Enable/Disable

| Command Form (Hex) | **ES&SS Sending** |
|---|---|
| 09h 00h 03h 45h 53h 44h 58h | Disable |
| 09h 00h 03h 45h 53h 45h 59h | Enable (default) |

### LCx (4Ch 43h x) ─ LRC Send Enable/Disable

x = E(45h enable) or D(44h disable)

Table 5-2. LRC Send Enable/Disable

| Command Form (Hex) | **LRC Sending** |
|---|---|
| 09h 00h 03h 4Ch 43h 44h 41h | Disable (default) |
| 09h 00h 03h 4Ch 43h 45h 40h | Enable |

### PCx (50h 43h x) ─ Set Protocol of Power On Report

x is an ASCII number (1-3)

Table 5-3. Protocol Setting

| **ASCII** | Command Form (Hex) | **Protocol** |
|---|---|---|
| 1 | 09h 00h 03h 50h 43h 31h 28h | Protocol 0 (default) |
| 2 | 09h 00h 03h 50h 43h 32h 2Bh | Protocol 1 |
| 3 | 09h 00h 03h 50h 43h 33h 2Ah | Protocol 2 |

### BZx (42h 5Ah x) ─ Buzzer Enable/Disable

x = E(45h enable) or D(44h disable)

*Note: No matter buzzer is enable or not, the reader still can accept Z or z commands to let it beep.*

Table 5-4. Buzzer Setting

| Command Form (Hex) | **Buzzer** |
|---|---|

| | |
|---|---|
| 09h 00h 03h 42h 5Ah 44h 56h | Disable |
| 09h 00h 03h 42h 5Ah 45h 57h | Enable (default) |

## TKx (54h 4Bh x) — Set Transmitting Data Tracks

x is an ASCII number (1-7)

Table 5-5. Transmit Tracks Setting

| **ASCII** | Command Form (Hex) | **Transmit Tracks** |
|---|---|---|
| 1 | 09h 00h 03h 54h 4Bh 31h 24h | Track 1 |
| 2 | 09h 00h 03h 54h 4Bh 32h 27h | Track 2 |
| 3 | 09h 00h 03h 54h 4Bh 33h 26h | Track 1 & 2 |
| 4 | 09h 00h 03h 54h 4Bh 34h 21h | Track 3 |
| 5 | 09h 00h 03h 54h 4Bh 35h 20h | Track 1 & 3 |
| 6 | 09h 00h 03h 54h 4Bh 36h 23h | Track 2 & 3 |
| 7 | 09h 00h 03h 54h 4Bh 37h 22h | Track 1, 2 & 3 (default) |

## AAx (41h 41h x) — Set Address

| | |
|---|---|
| 1 | x is a binary byte (00h-0Fh) |
| 2 | Set address 00h: <09h 00h 03h 41h 41h 00h 0Ah> |

## SAx (53h 41h x) — Self -Arm Mode Enable/Disable

x = E(45h enable) or D(44h disable)

Table 5-6. Self-Arm Mode Setting

| Command Form (Hex) | **Self-Arm Mode** |
|---|---|
| 09h 00h 03h 53h 41h 44h 5Ch | Disable |
| 09h 00h 03h 53h 41h 45h 5Dh | Enable (default) |

## SPx (53h 50h x) — Set Track Separator

| | |
|---|---|
| 1 | x = Hex Code |
| 2 | x = 00h means do not send separator code |

Table 5-7. Track Separator Setting

| Command Form (Hex) | **Track Separator** |
|---|---|
| 09h 00h 03h 53h 50h 00h 09h | Disable (default) |
| Example of Track Separator Setting | |
| 09h 00h 03h 53h 50h 0Dh 04h | 0Dh, <CR> |
| 09h 00h 03h 53h 50h 2Bh 22h | 2Bh, <+> |
| 09h 00h 03h 53h 50h 3Bh 32h | 3Bh, <;> |

## JHx (4Ah 48h x) — Set JIS Read Head

x = E(45h enable) or D(44h disable)

Table 5-8. JIS Read Head Setting

| Command Form (Hex) | **JIS Read Head** |
|---|---|
| 09h 00h 03h 4Ah 48h 44h 4Ch | Without (default) |
| 09h 00h 03h 4Ah 48h 45h 4Dh | With |

## p1 *data* (70h 31h *data*) — Set Track 1 Prefix Code

| | |
|---|---|
| 1 | Prefix can be set as 1-6 characters |
| 2 | If first character = 00h, prefix code of track 1 will not be send. It means this function is disabling. |

Table 5-9. Set Track 1 Prefix Code

| Command Form (Hex) | **Track 1 Prefix Code** |
|---|---|
| 09h 00h 03h 70h 31h 00h 4Bh | Disable (default) |
| Example of Track 1 Prefix Code Setting | |
| 09h 00h 00h 46h 41h 0Ah 04h | One character: 0Ah |
| 09h 00h 05h 70h 31h 54h 4Bh 01h 53h | Three characters: 54h 4Bh 01h |

## p2 *data* (70h 32h *data*) — Set Track 2 Prefix Code

| | |
|---|---|
| 1 | Prefix can be set as 1-6 characters |
| 2 | If first character = 00h, prefix code of track 2 will not be send. It means this function is disabling. |

Table 5-10. Set Track 2 Prefix Code

| Command Form (Hex) | **Track 2 Prefix Code** |
|---|---|
| 09h 00h 03h 70h 32h 00h 48h | Disable (default) |
| Example of Track 2 Prefix Code Setting | |
| 09h 00h 00h 46h 42h 0Bh 06h | One character: 0Bh |
| 09h 00h 05h 70h 32h 54h 4Bh 02h 53h | Three characters: 54h 4Bh 02h |

## p3 *data* (70h 33h *data*) — Set Track 3 Prefix Code

| | |
|---|---|
| 1 | Prefix can be set as 1-6 characters |
| 2 | If first character = 00h, prefix code of track 3 will not be send. It means this function is disabling. |

Table 5-11. Set Track 3 Prefix Code

| Command Form (Hex) | **Track 3 Prefix Code** |
|---|---|
| 09h 00h 03h 70h 33h 00h 49h | Disable (default) |
| Example of Track 3 Prefix Code Setting | |
| 09h 00h 00h 46h 43h 0Ch 00h | One character: 0Ch |
| 09h 00h 05h 70h 33h 54h 4Bh 03h 53h | Three characters: 54h 4Bh 03h |

## s1 (46h 61h x) – Set Track 1 Suffix Code

| | |
|---|---|
| 1 | Suffix can be set as 1-6 characters |
| 2 | If first character = 00h, suffix code of track 1 will not be send. It means this function is disabling. |

Table 5-12. Set Track 1 Suffix Code

| Command Form (Hex) | **Track 1 Suffix Code** |
|---|---|
| 09h 00h 03h 73h 31h 00h 48h | Disable (default) |
| Example of Track 1 Suffix Code Setting | |
| 09h 00h 03h 73h 31h 06h 4Eh | One character: 06h |

## s2 *data* (73h 32h *data*) – Set Track 2 Suffix Code

| | |
|---|---|
| 1 | Suffix can be set as 1-6 characters |
| 2 | If first character = 00h, suffix code of track 2 will not be send. It means this function is disabling. |

Table 5-13. Set Track 2 Suffix Code

| Command Form (Hex) | **Track 2 Suffix Code** |
|---|---|
| 09h 00h 03h 73h 32h 00h 4Bh | Disable (default) |
| Example of Track 2 Suffix Code Setting | |
| 09h 00h 03h 73h 32h 06h 4Dh | One character: 06h |

## s3 *data* (73h 33h *data*) – Set Track 3 Suffix Code

| | |
|---|---|
| 1 | Suffix can be set as 1-6 characters |
| 2 | If first character = 00h, suffix code of track 3 will not be send. It means this function is disabling. |

Table 5-14. Set Track 3 Suffix Code

| Command Form (Hex) | **Track 3 Suffix Code** |
|---|---|
| 09h 00h 03h 73h 33h 00h 4Ah | Disable (default) |
| Example of Track 3 Suffix Code Setting | |
| 09h 00h 03h 73h 33h 06h 4Ch | One character: 06h |

## K1A *type start end* (4Bh 31h 41h *type start end*)

– Set Transmitting Data Block, Channel A of Track 1

where ***type*** means card type, ***start*** means start address of transmitting data, and ***end*** means end address of transmitting data.

| | |
|---|---|
| 1 (31H) | CA old DMV |
| 2 (32H) | AAMVA |
| 3 (33H) | ISO |
| 4 (34H) | JIS |
| 6 (36h) | Trade Show |

| | |
|---|---|
| O (4FH) | any type |

Table 5-15. Data Block Channel A, Tk1

| Command Form (Hex) | **Data Block Channel A, Tk1** |
|---|---|
| 09h 00h 06h 4Bh 31h 41h 00h 00h 00h 34h | Disable (default) |
| Example of Track 1 Data Block Channel A Setting | |
| 09h 00h 06h 4Bh 31h 41h 33h 06h 20h 21h | ISO 6-32 |

## K1B *type start end* (4Bh 31h 42h *type start end*)

– Set Transmitting Data Block, Channel B of Track 1

Table 5-16. Data Block Channel B, Tk1

| Command Form (Hex) | **Data Block Channel B, Tk1** |
|---|---|
| 09h 00h 06h 4Bh 31h 42h 00h 00h 00h 37h | Disable (default) |
| Example of Track 1 Data Block Channel B Setting | |
| 09h 00h 06h 4Bh 31h 42h 33h 26h 30h 12h | ISO 38-48 |

## K1C type start end (4Bh 31h 43h type start end)

– Set Transmitting Data Block, Channel C of Track 1

Table 5-17. Data Block Channel C, Tk1

| Command Form (Hex) | **Data Block Channel C, Tk1** |
|---|---|
| 09h 00h 06h 4Bh 31h 43h 00h 00h 00h 36h | Disable (default) |
| Example of Track 1 Data Block Channel C Setting | |
| 09h 00h 06h 4Bh 31h 43h 33h 36h 40h 73h | ISO 54-64 |

## K2A type start end (4Bh 32h 41h type start end)

– Set Transmitting Data Block, Channel A of Track 2

Table 5-18. Data Block Channel A, Tk2

| Command Form (Hex) | **Data Block Channel A, Tk2** |
|---|---|
| 09h 00h 06h 4Bh 32h 41h 00h 00h 00h 37h | Disable (default) |
| Example of Track 2 Data Block Channel A Setting | |
| 09h 00h 06h 4Bh 32h 41h 32h 0Ah 16h 19h | AAMVA 10-22 |

## K2B type start end (4Bh 32h 42h type start end)

– Set Transmitting Data Block, Channel B of Track 2

Table 5-19. Data Block Channel B, Tk2

| Command Form (Hex) | **Data Block Channel B, Tk2** |
|---|---|
| 09h 00h 06h 4Bh 32h 42h 00h 00h 00h 34h | Disable (default) |
| Example of Track 2 Data Block Channel B Setting | |

| | |
|---|---|
| 09h 00h 06h 4Bh 32h 42h 33h 0Ah 1Ah 55h | ISO 10-26 |

## K2C type start end (4Bh 32h 43h type start end)

– Set Transmitting Data Block, Channel C of Track 2

Table 5-20. Data Block Channel C, Tk2

| Command Form (Hex) | **Data Block Channel C, Tk2** |
|---|---|
| 09h 00h 06h 4Bh 32h 43h 00h 00h 00h 35h | Disable (default) |
| Example of Track 2 Data Block Channel C Setting | |
| 09h 00h 06h 4Bh 32h 43h 31h 06h 20h 22h | CA old DMV 06-32 |

## K3A type start end (4Bh 33h 41h type start end)

– Set Transmitting Data Block, Channel A of Track 3

Table 5-21. Data Block Channel A, Tk3

| Command Form (Hex) | **Data Block Channel A, Tk3** |
|---|---|
| 09h 00h 06h 4Bh 33h 41h 00h 00h 00h 36h | Disable (default) |
| Example of Track 3 Data Block Channel A Setting | |
| 09h 00h 06h 4Bh 33h 41h 4Fh 10h 30h 59h | Any type 16-48 |

## K3B type start end (4Bh 33h 42h type start end)

– Set Transmitting Data Block, Channel B of Track 3

Table 5-22. Data Block Channel B, Tk3

| Command Form (Hex) | **Data Block Channel B, Tk3** |
|---|---|
| 09h 00h 06h 4Bh 33h 42h 00h 00h 00h 35h | Disable (default) |
| Example of Track 3 Data Block Channel B Setting | |
| 09h 00h 06h 4Bh 33h 42h 33h 11h 34h 23h | ISO 17-52 |

## K3C type start end (4Bh 33h 43h type start end)

– Set Transmitting Data Block, Channel C of Track 3

Table 5-23. Data Block Channel C, Tk3

| Command Form (Hex) | **Data Block Channel C, Tk3** |
|---|---|
| 09h 00h 06h 4Bh 33h 43h 00h 00h 00h 34h | Disable (default) |
| Example of Track 3 Data Block Channel C Setting | |
| 09h 00h 06h 4Bh 33h 43h 32h 06h 40h 40h | AAMVA 06-64 |

## DF0 (44h 46h 00h) – Default Setting

Command Form: <09h 00h 03h 44h 46h 00h 08h>

Default setting is as below:

| | | | |
|---|---|---|---|
| 1 | 9600 baud rate | 2 | 8 bits non parity |

| | | | |
|---|---|---|---|
| 3 | Send SS/ES | 4 | LRC not send |
| 5 | CTS disable | 6 | RTS disable |
| 7 | Buzzer enable | 8 | without JIS read head |
| 9 | Power on in protocol 0 | 10 | Transmit TK1, TK2 and TK3 |
| 11 | Self-Arm Mode enable | 12 | Read address = 00h |

## RE0 (52h 45h 00h) – Read EEPROM Data

Command Form: <09h 00h 03h 52h 45h 00h 1Dh>

| | |
|---|---|
| 1 | The reader's setting status is recorded in EEPROM. 77 bytes total. |
| 2 | The symbol "*" in this paragraph indicates "do not care". |

**Byte1 and 2 in EEPROM** are 00h, 13h separately. They are identical characters.

**Byte 3 in EEPROM**

Table 5-24. Byte 3 of EEPROM Status

| Bit 7 | SS&ES Status | Bit 6 | LRC Status | Bit 5 | Bit 4 | Bit 3 | Bit & Parity | Bit 2 | Bit 1 | Bit 0 | Baud Rate Status |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Not Send | 0 | Send | 0 | 0 | 0 | 7 even | 0 | 0 | 0 | 1200 |
| 1 | Send | 1 | Not send | 0 | 0 | 1 | 7 odd | 0 | 0 | 1 | 2400 |
| * | * | * | * | 0 | 1 | 0 | 7 mark | 0 | 1 | 0 | 4800 |
| * | * | * | * | 0 | 1 | 1 | 7 space | 0 | 1 | 1 | 9600 |
| * | * | * | * | 1 | 0 | 0 | 8 none | 1 | 0 | 0 | 19200 |

*Note: Bit 0-5 are reserved.*

**Byte 4 in EEPROM**

Table 5-25. Byte 4 of EEPROM Status

| Bit 7 | Bit 6 | Protocol | Bit 5 | CTS Status | Bit 4 | RTS Status | Bit 3 (always 0) | Bit 2 | Bit 1 | Bit 0 | Transmitting Data Track |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | Ignore | 0 | Always low | 0 | * | * | 0 | Not Transmit Tk1 |
| 1 | 0 | 1 | 1 | Consider | 1 | Low when transmit data | 0 | * | * | 1 | Transmit Tk1 |
| 1 | 1 | 2 | * | * | * | * | 0 | * | 0 | * | Not Transmit Tk2 |
| * | * | * | * | * | * | * | 0 | * | 1 | * | Transmit Tk2 |
| * | * | * | * | * | * | * | 0 | 0 | * | * | Not Transmit Tk3 |
| * | * | * | * | * | * | * | 0 | 1 | * | * | Transmit Tk3 |

*Note: If never set "PC" command then bit 6 and bit 7 are 00, it means protocol 0.*

*Note: Bit 4 & 5 are reserved.*

**Byte 5 in EEPROM**

Table 5-26. Byte 5 of EEPROM Status

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Address (Hex code) | Bit 3 | JIS Read Head | Bit 2 | Self-Arm Mode | Bit 1 | Bit 0 | Buzzer |
|---|---|---|---|---|---|---|---|---|---|---|---|
| * | * | * | * | * | 0 | Without | 0 | Disable | * | 0 | Disable |
| * | * | * | * | * | 1 | With | 1 | Enable | * | 1 | Enable |
| * | * | * | * | * | * | * | * | * | * | * | * |
| * | * | * | * | * | * | * | * | * | * | * | * |
| * | * | * | * | * | * | * | * | * | * | * | * |
| * | * | * | * | * | * | * | * | * | * | * | * |

**Byte 6 in EEPROM: Track Separator Setting**

Hex code "00h" means do not transmit this byte.

**Byte 7-12 in EEPROM:** Track 1 Prefix Code Setting

**Byte 13-18 in EEPROM:** Track 2 Prefix Code Setting

**Byte 19-24 in EEPROM:** Track 3 Prefix Code Setting

**Byte 25-30 in EEPROM:** Track 1 Suffix Code Setting

**Byte 31-36 in EEPROM:** Track 2 Suffix Code Setting

**Byte 37-42 in EEPROM:** Track 3 Suffix Code Setting

*Note: Each byte in byte 7-42 is Hex code. When one of the byte is "00h" the device do not transmit data from this byte to the end byte of the block.*

**Byte 43-45 in EEPROM:** Channel A of Track 1 Setting

**Byte 47-49 in EEPROM:** Channel B of Track 1 Setting

**Byte 51-53 in EEPROM:** Channel C of Track 1 Setting

**Byte 55-57 in EEPROM:** Channel A of Track 2 Setting

**Byte 59-61 in EEPROM:** Channel B of Track 2 Setting

**Byte 63-65 in EEPROM:** Channel C of Track 2 Setting

**Byte 67-69 in EEPROM:** Channel A of Track 3 Setting

**Byte 71-73 in EEPROM:** Channel B of Track 3 Setting

**Byte 75-77 in EEPROM:** Channel C of Track 3 Setting

*Note: Each byte in byte 43-77 is Hex code. The first byte of each block represents card type. The second byte of each block represents the start address of transmitting data. The third byte of each block represents the end address of transmitting data.*

*Note: Byte 46, 50, 54, 58, 62, 66, 70, 74 are not used.*

## RE1 (52h 45h 31h) — Read Configuration Status

Command Form: <09h 00h 03h 52h 45h 31h 2Ch>

This is used to send byte 3-6 data of EEPROM to host.

## RE2 (52h 45h 32h) － Read Prefix & Suffix Setting Status of Tracks

Command Form: <09h 00h 03h 52h 45h 32h 2Fh >

This is used to send byte 7-42 data of EEPROM to host.

## RE3 (52h 45h 33h) － Read Channel Setting Status of Tracks

Command Form: <09h 00h 03h 52h 45h 33h 2Eh>

This is used to send byte 743-77 data of EEPROM to host.

*Note: In the case of RE1, RE2, RE3 command, the device transforms the EEPROM data from hexadecimal code into ASCII code, and send it out. For example, if the EEPROM data is "2Ah" the host will receive "32h 41h". Hence, executing these 3 commands, the host will serially transform each 2 bits of received date into hexadecimal code.*

# Appendix A USB Driver Installation

This section illustrates instructions in **the USB driver installation guide for Windows 95/98**.

## ➢ System Requirements

Before you use the MSR120, make sure your computer is an IBM PC-compatible with the following minimum system requirements:

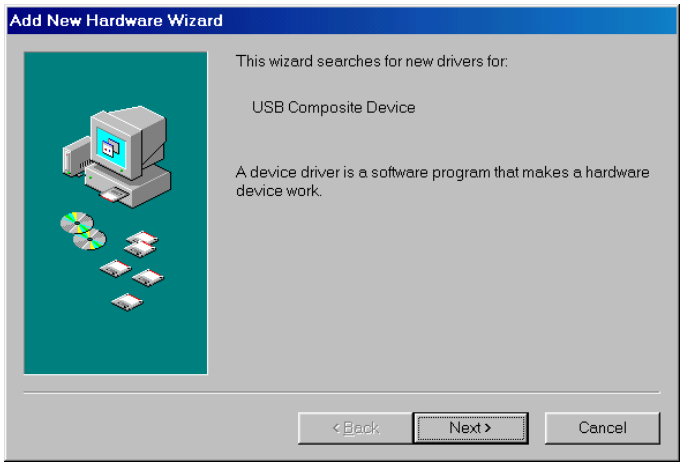
1. Intel-compatible 486DX-66 MHz CPU or higher
2. One standard USB port (4-pin) and has the symbol
3. Either the following operating systems:

   Windows 95 OSR2.1 (B version) with USB version 1214, Windows 98, Windows 98 SE, or Windows 2000.

## ➢ Driver Installation

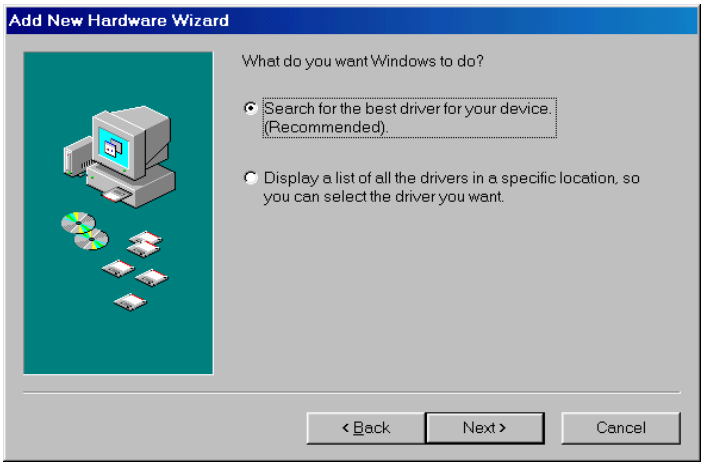
Follow the steps below on installing the MSR120 for the first time:

1. Power on computer where you will connect the MSR120 and make sure that the USB port is enabled and working properly.
2. Plug in the USB cable into the USB port and Windows will detect an USB Device and run the **Add New Hardware Wizard** to assist you in setting up the new USB Composite Device.

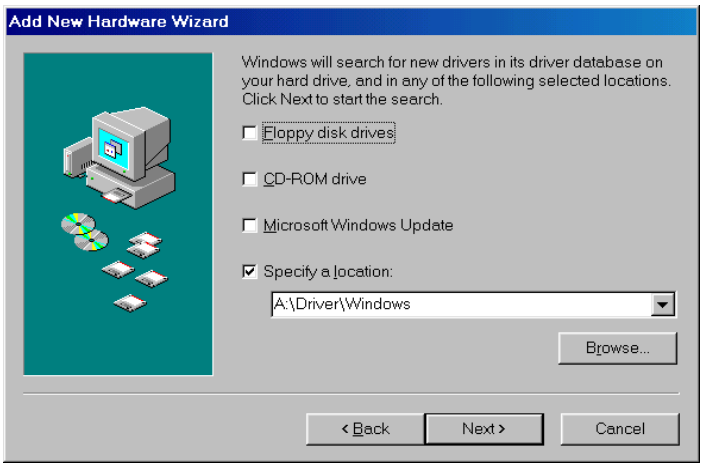



3. Insert the USB driver diskette (DISK 2) into the floppy drive and click **Next** to continue:

   a. Select **Search for the best driver for your device** and click **Next**.




b. Select **Specify a location** and click **Browse**. Change the folder of your floppy drive (i.e.: **A:\Driver\Windows**) and click **OK**.



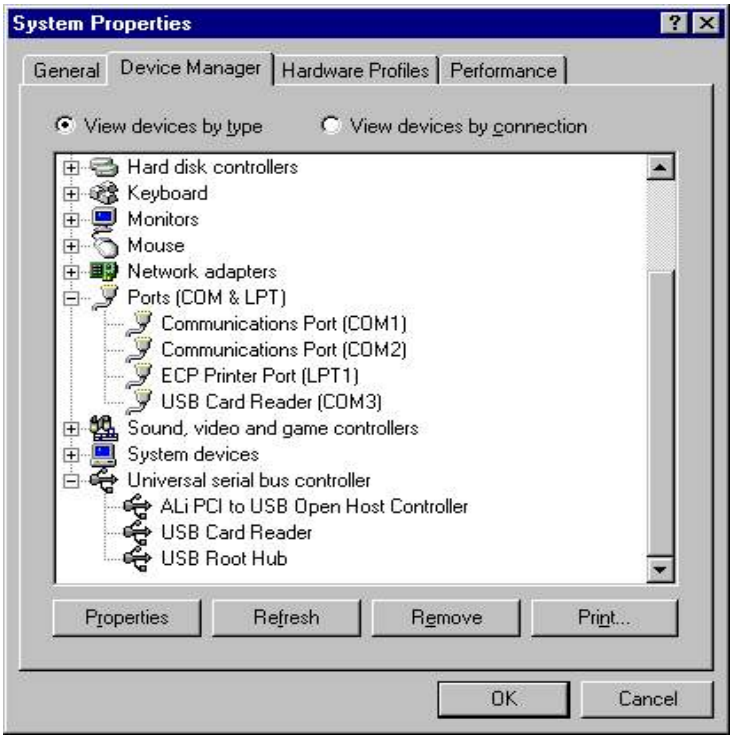

c. Double-check the directory that Windows prompts. Click **Next**.


d. Windows will detect the driver (**SER9PL.inf**) and shows the USB Card Reader. Click **Next** to continue until installation is complete.


e. Click Next to continue and let Windows copy the needed files to your hard disk.

Click **Finish** while installation is complete.


f. After installing, the System will generate an additional COM Port, USB Card Reader (e.g.: COM3) for the connection to RS232 Serial Device.




## ➢ Driver Un-installation

You can uninstall the MSR120 driver by executing **Uninstall Program: Dremover98.exe** in the driver diskette and click the **Remove** button to accomplish the un-installation.