MICR Plus

RS-232 WITH OPTIONAL 3-TRACK MSR

TECHNICAL REFERENCE MANUAL

Manual Part Number: 99875081 Rev 5

APRIL 2003

MAGTEK®

REGISTERED TO ISO 9001:2000

1710 Apollo Court
Seal Beach, CA 90740
Phone: (562) 546-6400
FAX: (562) 546-6301
Technical Support: (651) 415-6800
www.magtek.com

REVISIONS

Rev Number	Date	Notes
1	10 Sep 97	Initial Release
2	1 Oct 97	Sec 1, p.2, Changed cable number from 30012526 to 30015126
3	5 Nov 97	Appendix F, Revised Pin Numbers on Figures F-1 and F-2.
4	2 Aug 01	Front Matter, Agency page: Changed FCC to Class B; editorial correction for UL/CUL.
5	24 Apr 03	Front Matter: added ISO line to logo, changed Tech Support phone number, added new warranty statement

LIMITED WARRANTY

MagTek warrants that the products sold to Reseller pursuant to this Agreement will perform in accordance with MagTek's published specifications. This warranty shall be provided only for a period of one year from the date of the shipment of the product from MagTek (the "Warranty Period"). This warranty shall apply only to the original purchaser unless the buyer is authorized by MagTek to resell the products, in which event, this warranty shall apply only to the first repurchase.

During the Warranty Period, should this product fail to conform to MagTek's specifications, MagTek will, at its option, repair or replace this product at no additional charge except as set forth below. Repair parts and replacement products will be furnished on an exchange basis and will be either reconditioned or new. All replaced parts and products become the property of MagTek. This limited warranty does not include service to repair damage to the product resulting from accident, disaster, unreasonable use, misuse, abuse, customer's negligence, Reseller's negligence, or non-MagTek modification of the product. MagTek reserves the right to examine the alleged defective goods to determine whether the warranty is applicable.

Without limiting the generality of the foregoing, MagTek specifically disclaims any liability or warranty for goods resold in other than MagTek's original packages, and for goods modified, altered, or treated by customers.

Service may be obtained by delivering the product during the warranty period to MagTek (1710 Apollo Court, Seal Beach, CA 90740). If this product is delivered by mail or by an equivalent shipping carrier, the customer agrees to insure the product or assume the risk of loss or damage in transit, to prepay shipping charges to the warranty service location and to use the original shipping container or equivalent. MagTek will return the product, prepaid, via a three (3) day shipping service. A Return Material Authorization (RMA) number must accompany all returns.

MAGTEK MAKES NO OTHER WARRANTY, EXPRESS OR IMPLIED, AND MAGTEK DISCLAIMS ANY WARRANTY OF ANY OTHER KIND, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

EACH PURCHASER UNDERSTANDS THAT THE MAGTEK PRODUCT IS OFFERED AS IS. IF THIS PRODUCT DOES NOT CONFORM TO MAGTEK'S SPECIFICATIONS, THE SOLE REMEDY SHALL BE REPAIR OR REPLACEMENT AS PROVIDED ABOVE. MAGTEK'S LIABILITY, IF ANY, TO RESELLER OR TO RESELLER'S CUSTOMERS, SHALL IN NO EVENT EXCEED THE TOTAL AMOUNT PAID TO MAGTEK BY RESELLER UNDER THIS AGREEMENT. IN NO EVENT WILL MAGTEK BE LIABLE TO THE RESELLER OR THE RESELLER'S CUSTOMER FOR ANY DAMAGES, INCLUDING ANY LOST PROFITS, LOST SAVINGS OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF OR INABILITY TO USE SUCH PRODUCT, EVEN IF MAGTEK HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY OTHER PARTY.

LIMITATION ON LIABILITY

EXCEPT AS PROVIDED IN THE SECTIONS RELATING TO MAGTEK'S LIMITED WARRANTY, MAGTEK'S LIABILITY UNDER THIS AGREEMENT IS LIMITED TO THE CONTRACT PRICE OF THE PRODUCTS.

MAGTEK MAKES NO OTHER WARRANTIES WITH RESPECT TO THE PRODUCTS, EXPRESSED OR IMPLIED, EXCEPT AS MAY BE STATED IN THIS AGREEMENT, AND MAGTEK DISCLAIMS ANY IMPLIED WARRANTY, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

MAGTEK SHALL NOT BE LIABLE FOR CONTINGENT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES TO PERSONS OR PROPERTY. MAGTEK FURTHER LIMITS ITS LIABILITY OF ANY KIND WITH RESPECT TO THE PRODUCTS, INCLUDING ANY NEGLIGENCE ON ITS PART, TO THE CONTRACT PRICE FOR THE GOODS.

MAGTEK'S SOLE LIABILITY AND BUYER'S EXCLUSIVE REMEDIES ARE STATED IN THIS SECTION AND IN THE SECTION RELATING TO MAGTEK'S LIMITED WARRANTY.

FCC WARNING STATEMENT

This equipment has been tested and found to comply with the limits for Class B digital device, pursuant to Part 15 of FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a residential environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

FCC COMPLIANCE STATEMENT

This device complies with Part 15 of the FCC Rules. Operation of this device is subject to the following two conditions: (1) This device may not cause harmful interference. And (2) this device must accept any interference received, including interference that may cause undesired operation.

CANADIAN DOC STATEMENT

This digital apparatus does not exceed the Class B limits for radio noise for digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de las classe B prescrites dans le Réglement sur le brouillage radioélectrique édicté par les ministère des Communications du Canada.

UL/CSA

This product is recognized per Underwriter Laboratories and Canadian Underwriter Laboratories 1950.

TABLE OF CONTENTS

FIGURES

TABLES

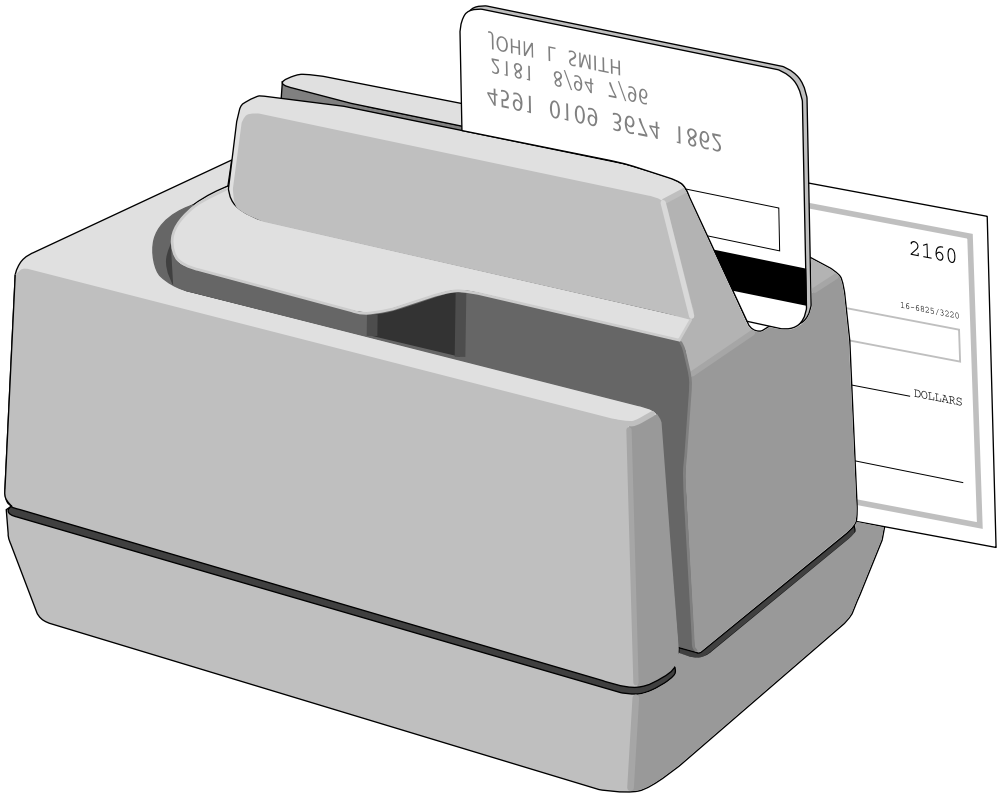


Figure 1-1. MICR Plus

SECTION 1. OVERVIEW

The MICR Plus is both a MICR (Magnetic Ink Character Recognition) Check Reader and an MSR (Magnetic Stripe Reader).

The MICR Plus, in a typical application, reads the magnetic data encoded on the bottom of checks or magnetic stripe cards and transmits this data to a Host device. The Host device then uses a specific authorization or verification process to validate a business transaction.

The use of the MICR Plus improves accuracy and speed because there is no manual data entry; therefore there are no keying errors or unwanted delays.

The MICR Plus will communicate with the Host system using a standard RS-232 interface. All data is transmitted as ASCII characters (See Appendix G). The MICR Plus has the capability of supporting some hardware handshaking signals. (See Section 4, Commands.)

The MICR Plus also has an auxiliary RS-232 port for connection to another RS-232 device, such as a PINPad or bar code reader.

FEATURES

- Available with MICR Reader only or with 3-Track or 2-Track MSR.
- Three track MSR - autodiscriminates different card formats: ISO (International Standards Organization), CDL (California Drivers License), or AAMVA (American Association of Motor Vehicle Administrators).
- Small footprint.
- Automatic parsing of MICR fields: transit, account, etc.
- Extensive list of formats to transmit MICR data.
- Optional error/status reporting for check reading.
- Reads E13-B or CMC-7 MICR fonts.
- Field Upgraded Changes– New programs or changes can be downloaded in the field (see Appendix E).
- A single hardware platform supports all the interfaces available: RS-232, OCIA, and IBM device emulations (MSR, Scanner, Feature Card) for an ECR (Electronic Cash Register).
- One auxiliary RS-232 port available. When connecting to an ECR Host, there are two RS-232 ports available.

ACCESSORIES

- Interface Cable, DB-25 female, Part Number 22617504
- Interface Cable, DB-9, female, Part Number 22617506
- AC Power Adapter with Cable, Part Number 64300050
- SET-MICR Demo Program, Part Number 22000020 (Rev P or higher)
- MICR Reader Cleaning Card, Part Number 96700006
- Sample Checks, Part Number 96530005
- Program Mode Insta-Change check for MICR Plus, Part Number 96530022
- Software Utility for Downloading , PROGMICR.EXE, Part Number 22596805
- IntelliPIN Plus, portable, RS-232 interface, Part Number 30015125
- IntelliPIN Plus, nonportable, RS-232 interface, Part Number 30015126

SPECIFICATIONS

Table 1-1 lists the specifications for the MICR Plus.

Table 1-1. Specifications

OPERATING	
Reference Standards	ISO/CDL/AAMVA
Power Input	120 VAC, 50/60 Hz
Output Signal Levels	12 VAC, 1 Amp
Check Read/Decode/Transit Time	1 second
MICR fonts supported	E13-B CMC-7
MSR supported	Tracks 1, 2, and 3; or Tracks 1 and 2
MECHANICAL	
Dimensions	Length 6.0”, Width 4.0”, Height 4.4”
Weight:	3.0 lbs. MSR and Adapter included
Cable length	6’
Connectors	9 pin DB female, 25 pin DB female
ENVIRONMENTAL	
Temperature	
Operating	0^{o}C to 50^{o}C (32^{o}F to 122^{o}F)
Storage	-30^{o}C to 70^{o}C (-22^{o}F to 158^{o}F)
Humidity	
Operating	10% to 90% noncondensing
Storage	Up to 100% noncondensing

SECTION 2. INSTALLATION

The installation for the MICR Plus is as follows:

REQUIREMENTS

The following is required for the Installation:

- MICR Plus
- Interface Cable, DB25 female, Part Number 22617504 or
  Interface Cable, DB9, female, Part Number 22617506
- AC Power Adapter with Cable, 120VAC to 12 VAC, 1 Amp, Part Number 64300050
- Interface Cable, if required, from RS-232 Auxiliary Port to external RS-232 device

PORTS

The MICR ports are shown in Figure 2-1.

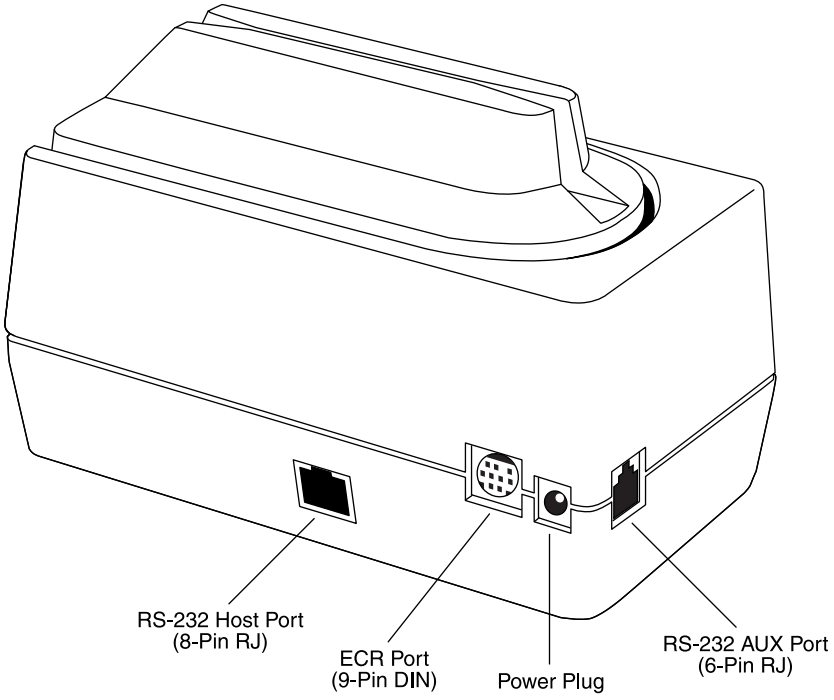


Figure 2-1. Ports on the MICR Plus

The ports are for the following connections:

- The RS-232 Host port is used when connecting to an RS-232 Host.
- The ECR port is used when connecting to an ECR Host.
- The RS-232 Auxiliary Port is used to attach an external RS-232 device to the MICR Plus. The MICR Plus in turn, acts as a communication bridge between the external device and the Host device.
- The Power plug connects to an adapter and wall power.

To Install the MICR Plus, perform the following steps:

1. On the interface cable connect the DB25/DB9 connector to the PC. The pin lists for these connectors are shown in Appendix F.

2. On the interface cable connect the 8-pin RJ plug to the RS-232 Host port.

3. If an external RS-232 device is to be used, connect the cable from the RS-232 device to the RS-232 auxiliary port. An example is shown in Figure 2-2.

Note

The RS-232 auxiliary port will be set with the same communication parameters as the RS-232 Host Port.

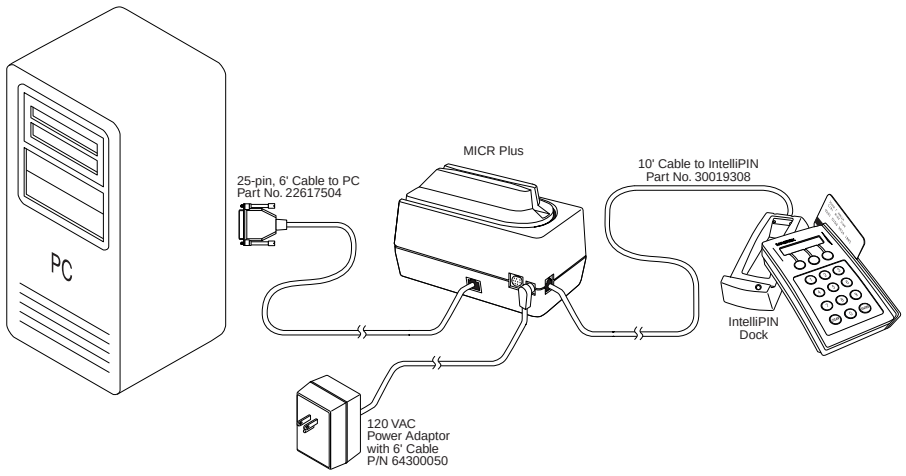


Figure 2-2. MICR Plus Connections

4. On the AC power adapter, connect the jack to the plug on the MICR Plus.

5. On the AC power adapter, connect the plug to the wall outlet.

6. The LED indicator on the MICR Plus should turn on to a steady green. The LED indicator is located below the slot where the check is first inserted for reading.

Caution

Do not place the MICR Plus within 6 inches of a computer monitor or power supply. These devices may cause undesirable interference with the check reading operation.

SECTION 3. OPERATION

This section contains check and card reading procedures and LED indicator states.

CHECK READING PROCEDURE

1. Orient the check so the MICR line is down and the printed side faces the center on the MICR Plus as shown in Figure 3-1.

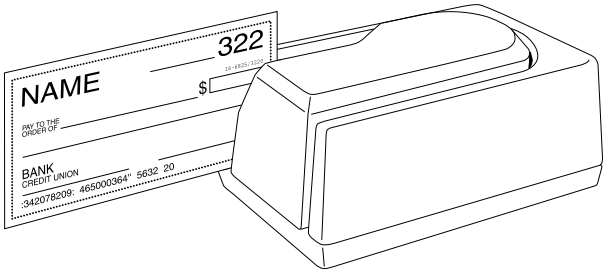


Figure 3-1. Check Orientation

2. Drop the check so the leading edge is in the open slot.

3. When the unit detects the presence of the check, the motor will turn on. At this time gently urge the check forward until the unit grabs the check. When this happens, release the check. The check will then be transported around the check path and will exit through the other side.

4. After the check is read, the Reader will transmit the data as specified by the parameters described in Section 4, “Commands”.

CARD SWIPE PROCEDURE

The card may be swiped through the Magnetic Stripe Reader (MSR) in either direction, but the magnetic stripe must be oriented in only one direction as shown in Figure 1-1. The reader will transmit raw card data (“as is” on the card) for all tracks that have been enabled using the HW (Hardware) command (Section 4, Commands).

The MSR is capable of reading ISO, AAMVA, and CDL encoded cards. The MSR will autodiscriminate all the card formats when the ID Card Decoding option is enabled using the HW command (Section 4, Commands).

LED INDICATORS

Table 3-1 describes the LED indicator conditions for check and card reading operations. The LED indicator is located below the slot where the check is first inserted for reading. The commands and all possible combinations of the LED indicator are listed in Section 4 under LE Command.

Table 3-1. LED indicators

LED INDICATOR	DESCRIPTION
OFF	Power off
SOLID GREEN	Ready to read check or card
OFF→ SOLID RED	Check or card read error
OFF→ SOLID GREEN	Good read
SOLID GREEN → FLASH RED/GREEN	EMF noise detected*
FLASH GREEN	Needs initialization*
FLASH RED/GREEN	Data sensor blocked (motor does not run)*
FLASH RED	Motor sensor blocked (motor does not run)*
FLASH GREEN FAST	Monitor mode (factory use only)*

*Refer to “Appendix C. Troubleshooting Guide”

SECTION 4. COMMANDS

This section describes the use of commands and programmable options available for the MICR Plus.

Note

All options described below can be factory set as specified by the user when ordering.

To execute the MICR Plus commands, either one of two methods is required: Insta-Change Checks or a PC with a program that provides access to a serial communications (COMM) port.

INSTA-CHANGE CHECKS

The first method is the use of Insta-Change checks, which is a more practical way of setting up the MICR Plus for most applications. The Insta-Change check is an E13-B MICR encoded document that contains commands and options used to reset the parameters of the MICR Plus. Multiple commands and options may be contained on one Insta-Change check. When used, the Insta-Change checks are run through the MICR Plus the same as a standard check, and the options to be used are automatically selected. To obtain Insta-Change checks, notify a MagTek representative and specify what options will be used. To operate Insta-Change checks, install the MICR Plus as described in Section 2, and watch the LED indicator. When the Insta-Change check is run through the MICR Plus and read successfully, the LED indicator will blink green. If the LED indicator turns red, the read is not successful. Try again or use a different Insta-Change check.

PC PROGRAM

The second method, the PC program, may be MagTek's SET-MICR Demo program (Rev P or higher), or any other PC program that provides access to the COMM port. With the SET-MICR program, commands may be selected from menus, but with any other program the command data is entered manually (typed) into the system. Whether SET-MICR, or any other program is used, the PC and the MICR Plus must be set to the same communication parameters (baud rate, parity, stop bits, etc.)

COMMAND FORMAT

When the commands are entered manually, they must use the following format:

[COMMAND][DATA]<CR>

Where:

- **[COMMAND]** is 2 or 3 alpha characters.
- **[Data]** is optional as described below for each command.
- **<CR>** is always required.
- All characters are ASCII
- No spaces, brackets, or angle brackets required.

SWA - SWITCH A COMMAND

The SWA command, shown in Table 4-1, controls the communication parameters for the RS-232 Host port. The data for this command consists of 8 ASCII bits ("0" = hex 30 and "1" = hex 31).

Note

The RS-232 auxiliary port will be set with the same communication parameters as the RS-232 Host Port.

To execute, send the SWA command as follows:

SWA 01010101<CR> (with data)
or
SWA <CR> (without data)

When sending data, all 8 bits must be provided. The MICR Plus will execute the command but it will not reply. To make this command permanent, use the SA command described below.

If no data is sent, the MICR Plus responds with the current settings for SWA.

Note

The new settings for the serial port will not become effective until the RS command is executed.

Table 4-1. SWA Command

BITS								FUNCTION
7	6	5	4	3	2	1	0	
					0	0	0	Reserved
					0	0	1	Baud Rate: 300
					0	1	0	Baud Rate: 600
					0	1	1	Baud Rate: 1200
					1	0	0	Baud Rate 2400
					1	0	1	Baud Rate: 4800
					1	1	0	Baud Rate: 9600
					1	1	1	Baud Rate: 19200
	0		0	0				Data, Stop Bits, Parity: 8, 1, None
	1		0	0				Data, Stop Bits, Parity: 8, 2, None
	0		0	1				Data, Stop Bits, Parity: 8, 1, Even
	1		0	1				Data, Stop Bits, Parity: 8, 1, Odd
	0		1	0				Data, Stop Bits, Parity: 7, 1, Even
	1		1	0				Data, Stop Bits, Parity: 7, 2, Even
	0		1	1				Data, Stop Bits, Parity: 7, 1, Odd
	1		1	1				Data, Stop Bits, Parity: 7, 2, Odd
		0						CTS/DSR: Use
		1						CTS/DSR: Ignore
0								Intercharacter Delay: No
1								Intercharacter Delay: Yes

SWA PARAMETERS

The SWA functions are listed in Table 4-1 and described below.

Baud Rate

The baud rate is one of seven speeds at which the MICR Plus communicates with the host. The lowest speed is 300 baud, and the highest is 19200.

Data, Stop Bits, and Parity

Data refers to the number of data bits used to transmit every character; the options available are 7 or 8. Stop Bits refer to the number of bits used to indicate the end of transmission for every character; the options available are 1 or 2. Parity refers to a means of detecting bit-level transmission errors for every character; the options available are None, Even or Odd.

CTS/DSR

When CTS/DSR (Clear to Send/Data Set Ready) is set to Ignore, the MICR Plus sends data to the host without waiting for the CTS and DSR signals to be active. When CTS/DTS is set to Use, the MICR Plus waits for the CTS and DSR signals to be active before sending data.

Intercharacter Delay

The intercharacter delay is used to increase the time between characters transmitted from the MICR Plus. The delay is equivalent to the transmission of one character. This parameter affects character rate but not baud rate (i.e., each character takes the same time to transmit but the time between characters is increased).

SWB - SWITCH B COMMAND

The SWB command controls the message format, shown in Table 4-2. The data for this command consists of 8 ASCII bits ("0" = hex 30 and "1" = hex 31).

To execute, send the SWB command as follows:

SWB 01010101<CR> (with data)
or
SWB <CR> (without data)

When sending data, all 8 bits must be provided. The MICR Plus will execute the command but it will not reply. The new settings become effective immediately.

If no data is sent, the MICR Plus responds with the current settings for SWB. To make this command permanent, use the SA command described below.

Table 4-2. SWB Command

BIT								FUNCTION
7	6	5	4	3	2	1	0	
							0	<LF>: No
							1	<LF>: Yes
						0		<CR>: No
						1		<CR>: Yes
					0			<ETX>: No
					1			<ETX>: Yes
				0				<ESC>: No
				1				<ESC>: Yes
			0					<STX>: No
			1					<STX>: Yes
		0						Send Data After Error?: No
		1						Send Data After Error?: Yes
	0							Send Status After Data?: No
	1							Send Status After Data?: Yes
0	0	0	0	0	0	0	0	Comm Mode: 0 - Data Only
1	0	0	0	0	0	0	0	Comm Mode: 1 - Data <CR>
0	0	0	0	0	0	0	1	Comm Mode: 2 - Data -<LF>
0	0	0	0	0	0	1	1	Comm Mode: 3 - Data -<CR><LF>
0	0	0	0	1	0	0	0	Comm Mode: 4 - <ESC> Data
0	0	0	0	1	0	1	0	Comm Mode: 5 - <ESC> Data<CR>
0	0	0	1	0	1	0	0	Comm Mode: 6 - <STX> Data<ETX>
1	0	0	0	0	0	0	1	Comm Mode: 7 - Packet Mode (<STX>Data<ETX><LRC>)

SWB PARAMETERS

The SWB functions are listed in Table 4-2 and described below.

Control Characters and MICR Data

Control Characters may be added to the MICR data message. The characters are always in the following locations:

<STX> <ESC> data <ETX> <CR> <LF>

The control characters, descriptions, and hex values are shown in Table 4-3.

Table 4-3. Control Characters

CONTROL CHARACTER	DESCRIPTION	HEX VALUE
<STX>	Start of Text	02
<ESC>	Escape	1B
<ETX>	End of Text	03
<CR>	Carriage Return	0D
<LF>	Line Feed	0A

For example, if <STX> and <CR> are set to YES, the message from the MICR Plus will look like this:

`MICR Data: <STX>data<CR>`

Control Characters and Card Data

The control characters are also available for card data but their position on the message is controlled by the Card Data Message parameter (see SWC Command, below). For example, if the <STX> and <ETX> options are set to YES, the message may be transmitted as follows:

If Multiple Message: <STX>[TK1]<ETX><STX>[TK2]<ETX><STX>[TK3]<ETX>

If Single Message: <STX>[TK1] [TK2] [TK3]<ETX>

Communication Modes

The selection of Comm Modes is a quick way of selecting multiple Control Characters. For instance, to send a carriage return/line feed pair after the data, you can specify Comm Mode 3.

Comm Mode 7, also known as Packet Mode, calculates an LRC (Longitudinal Redundancy Check), and appends it to the data message. Also, if a <NAK> (hex 15) character is received in this mode, the MICR Plus will resend the last message.

Send Data After Error

The request Send Data After Error specifies whether the MICR Plus will return data to the host after a read error. If YES is selected and the MICR Plus reads a check with an error, the MICR Plus will send the data back to the host. If NO is selected and the MICR Plus finds an error, it will discard the data and nothing will be sent. The error conditions are listed in Table 4-4.

Send Status After Data

The Send Status After Data option makes the MICR Plus append a two-digit error/status code to the end of the MICR data. For most formats (See Appendix A) the error/status code will always be preceded by a forward slash (/). The error/status codes are listed in Table 4-4.

For example, if a Canadian check (code 08) is read and had no errors, and the MICR data is "1234567890", then the message from the MICR Plus will look like this:

```
MICR Data:  1234567890/08
```

The status code is always at the end of the data, not the end of the message. For example, using the above conditions, with the message format set to send <STX> and <ETX>, the message from the MICR Plus will look like this:

```
MICR Data:  <STX>1234567890/08<ETX>
```

Table 4-4. Error and Status Codes

PRIORITY	CODE	TYPE	DESCRIPTION
9	01	Error	No MICR data: no transit and no account found
8	09	Status	Mexican check
7	08	Status	Canadian check
6	05	Error	Transit error: No transit, bad character, bad length, bad check digit
5	07	Error	Account error No account, bad character
4	04	Error	Check # error: Bad character in check number
6	05	Error	No transit, bad character, bad length, bad check digit
5	07	Error	No account, bad character
4	04	Error	Bad character in check number
4	04	Status	No check number
3	03	Status	Low MICR signal, good read
2	10	Status	Business check
1	11	Status	Amount field present
0	00	Status	No error, check OK

Notes:

- The LED indicator will turn red on all error conditions.
- The absence of a check number is not considered an error.
- If a multiple error occurs, the error or status code with the highest priority is reported.
- All unreadable MICR characters are transmitted as an "?" ASCII character (hex 3F), except for Format 00xx (See Appendix A).

SWC - SWITCH C COMMAND

The SWC command controls miscellaneous functions, shown in Table 4-5. The data for this command consists of 8 ASCII bits ("0" = hex 30 and "1" = hex 31).

To execute, send the SWC command as follows:

SWC 01010101<CR> (with data)
or
SWC <CR> (without data)

When sending data, all 8 bits must be provided. The MICR Plus will execute the command but it will not reply. The new settings become effective immediately. To make this command permanent, use the SA command described below.

If no data is sent, the MICR Plus responds with the current settings for SWC.

Table 4-5. SWC Command

BITS								FUNCTION
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
							0	CMC-7 Character Set: No
							1	CMC-7 Character Set: Yes
					0	0		Invalid Commands: ?<CR>
					0	1		Invalid Commands : No Reply (Header Required)*
					1	0		Invalid Commands: No Reply (no header required)
					1	1		Ignore all Commands
				0				Active RTS: No
				1				Active RTS: Yes
			0					Data Header: No
			1					Data Header: Yes
		0						Card Data Message: Multiple
		1						Card Data Message: Single
0	0							These bits are always set to 0 but must be included.

*Header Required means all commands must be preceded by a GS character (Hex 1D).

SWC PARAMETERS

The SWC functions are listed in Table 4-5 and described below.

CMC-7 Character Set

If NO is selected, the MICR Reader will only read E13-B characters. When YES is selected, the MICR Reader will read both CMC-7 and E13-B characters (see Appendix B). However, the MICR Reader will only output raw data ("as is" on the check) for checks with CMC-7 characters.

Invalid Command Response

Invalid command response is the action the MICR Plus takes upon receipt of a command it does not recognize. It can also be used to stop the MICR Plus from receiving any more commands.

The first option "?<CR>" is the default. If the MICR Plus receives an unrecognized command, it will return a question mark and carriage return to the host. The MICR Plus will then return to an idle state and wait for further commands or check/credit card reads.

For the second option, "no reply - header required," the MICR Plus will only execute commands preceded by a GS ASCII character (hex 1D), and it will not reply to invalid commands. When this option is selected, all messages received without a GS header will be transmitted "as received" through the RS-232 auxiliary port.

For the third option, "no reply," the MICR Plus will execute all valid commands, but it will not reply to invalid commands.

The fourth option, "ignore all commands," causes the MICR Plus to stop receiving any Host data and to ignore any further commands. Even the SA (Save) command is ignored and therefore this fourth option is only temporary. To make this option permanent or to reset it, you must use an Insta-Change check.

Active RTS

When this function is set to YES, the MICR Plus will raise RTS and wait 5 seconds for CTS to become active before sending any data. If the 5 seconds expire and CTS is not active, the data message will be discarded and nothing will be sent.

Data Header

If YES is selected, a single character header precedes the data. For MICR data, the message is transmitted as follows:

MICR data: 'C'[data]

For card data, the header position on the message is controlled by the Card Data Message parameter (see below). Therefore, the message may be transmitted as follows:

If Multiple Message: 'M'[TK1]'M'[TK2]'M'[TK3]

If Single Message: 'M'[TK1] [TK2] [TK3]

It is important to note that the Data Header precedes the data and not the message. For example, if <STX>, <ETX> and Data Header are set to YES, a MICR data message will be transmitted as follows:

MICR data: <STX>'C'[data]<ETX>

Card Data Message

This option determines the structure of the output message for the individual tracks when a credit card is read. If Multiple is selected, the Control Characters (see SWB, below) and Data Header (see Data Header, above) are added to each track individually. On the other hand, if Single is selected, all available tracks are lumped together into a single message. For example, if <STX>, <ETX> and Data Header are set to YES, the output message may be transmitted as follows:

If Multiple Message: <STX>'M'[TK1]<ETX><STX>'M'[TK2]<ETX><STX>'M'[TK3]<ETX>

If Single Message: <STX>'M'[TK1] [TK2] [TK3]<ETX>

HW - HARDWARE COMMAND

This command controls miscellaneous hardware options, shown in Table 4-6. The data for this command consists of 8 ASCII bits ("0" = hex 30 and "1" = hex 31).

To execute, send the HW command as follows:

HW 01010101<CR> (with data)
or
HW <CR> (without data)

When sending data, all 8 bits must be provided. The MICR Plus will execute the command but it will not reply. The new settings become effective immediately. To make this command permanent, use the SA command described below.

If no data is sent, the MICR Plus responds with the current settings for HW.

Table 4-6. HW Command

7	6	5	4	3	2	1	0	PARAMETERS
							0	Y Option: Disable
							1	Y Option: Enable
					0			Track 3: Disable
					1			Track 3: Enable
				0				Track 2: Disable
				1`				Track 2: Enable
			0					Track 1: Disable
			1					Track 1: Enable
		0						ID Card decoding: Disable
		1						ID Card decoding: Enable
0	0					0		These bits are always set to 0

HW PARAMETERS

Disable/Enable Y Option

Enable this option when using a Y-cable to connect an additional device on the RS-232 Host Port. This option allows the MICR Plus and the additional device to receive/transmit data from the Host.

One important consideration is to determine how the MICR Plus should respond to all data received from the Host. This response is controlled by the Invalid Command Response. (See SWC Command, above.)

Disable/Enable Tracks

Each Track can be enabled or disabled individually. The tracks are always transmitted in ascending order: TK1, TK2, TK3. For example, if TK1 and TK3 are enabled and TK2 is disabled, the reader will transmit TK1, TK3.

ID Card Decoding

The MSR has two modes of operation. In the first mode, ID Card decoding disabled, the MSR will only read ISO encoded cards. In the second mode, ID Card decoding enabled, the MSR will read and autodiscriminate ISO, AAMVA, and CDL encoded cards. When a card is swiped, the LED indicator will turn red and indicated an error if any of the enabled tracks read is incompatible with the selected mode of operation. Tk2 is a standard track for all types of cards.

LE - LED COMMAND

To control the LED, the LE command is sent with a hexadecimal digit (use ASCII characters for the hex digit):

LE *X*<Enter>

Where ***X*** = Hex digit **0-F**.

An example of the "Blink Red Fast" command is:

LE 9<Enter>

LE PARAMETERS

The codes and descriptions are shown in Table 4-7. The color cycle is four equal time periods of 0.1 second. This cycle is repeated for three seconds and then goes off. The description column is a common expression of the state of the LED.

Table 4-7. LED Control

Color Cycle	Hex Digit	Description
Off/Off/Off/Off	0	LED Off
Green/Green/Green/Green	1	Steady Green
Red/Red/Red/Red/	2	Steady Red
Amber/Amber/Amber/Amber	3	Steady Amber
Green/Green/Off/Off	4	Blink Green Slow
Red/Red/Off/Off	5	Blink Red Slow
Amber/Amber/Off/Off	6	Blink Amber Slow
Red/Red/Green/Green	7	Blink Red/Green Slow
Green/Off/Green/Off	8	Blink Green Fast
Red/Off/Red/Off	9	Blink Red Fast
Amber/Off/Amber/Off	A	Blink Amber Fast
Red/Green/Red/Green	B	Blink Red/Green Fast
Red/Green/Off/Off	C	Fast Red/Green Off
Green/Green/Green/Red	D	Green + Fast Red
Red/Red/Red/Green	E	Red + Fast Green
Off/Off/Off/Off	F	Off

FC - FORMAT CHANGE COMMAND

Formats are used by the MICR Plus to process and transmit the MICR fields. The Format command allows the selection of a format from the Format List, Appendix A. The data for this command consists of 4 digits (ASCII characters 0-9). To execute, send the command as follows:

FC 6600<CR> (with data)
or
FC <CR> (without data)

When sending data, all 4 digits must be provided. The MICR Plus will execute the command but it will not reply. The new settings become effective immediately. To make this command permanent, use the SA command described below.

If no data is provided, the MICR Plus will respond with the current setting.

VR - VERSION COMMAND

The Version command gives the current firmware revision in the MICR Plus. To execute, send the VR command followed by a carriage return as follows:

VR<CR>

The MICR Plus responds as follows:

MICR data: [firmware revision]<CR>

SA - SAVE COMMAND

All changes are considered temporary until the Save command is executed. The Save command saves all changes to the MICR Plus memory and makes them permanent. The MICR Plus will execute the command but it will not reply. To execute, send the SA command followed by a carriage return as follows:

SA<CR>

RS - RESET COMMAND

The Reset command resets the MICR program, and it resets the serial port to the most recent settings provided by the SWA command. To execute, send the RS command followed by a carriage return as follows:

RS<CR>

APPENDIX A. FORMAT LIST

For check reading, the MICR Plus provides the flexibility to format the MICR fields and build a specific output string that will be transmitted to the Host. These output strings are referred to as Formats. The MICR Plus has a built-in list of Formats (described below) from which the user may select one to become the active Format every time a check is read. The Formats may be selected using the FC command (Section 4, Commands) or Insta-Change Checks provided by MagTek.

Each Format is assigned a 4-digit number. The first two digits indicate the Format number, and the last two digits are specific parameters used for various functions by each Format. For example, in Format "0415", the "04" refers to Format number 4 and the 15 refers the maximum number of characters allowed for the account field.

Note

The formats listed in this section apply only to U.S. and Canadian checks. The MICR line on checks from other countries will not be broken or parsed as described in these formats.

A complete description for each Format follows.

Fmt 00xx: Raw Data Format - sends the entire MICR line - where:

```
xx          - specify what symbol set to use.  Choose from the table
Add xx + 16 - change multiple spaces to one space
Add xx + 32 - Remove all spaces

Examples:

   MICR LINE:  T122000218T  1234 5678 9U   1321
      FC0001 - t122000218t  1234 5678 9o   1321
(+16) FC0017 - t122000218t 1234 5678 9o 1321
(+32) FC0033 - t122000218t123456789o1321.
```

xx	Transit	On-Us	Amount	Dash	Error
00	T	U	$	-	?
01	t	o	a	d	?
02	T	O	A	D	?
03	T	U	$	-	*
04	T	U	$	0	?
05	T	U	$	0	*
06	t	o	a	0	?
07	T	U	$	none	?

Fmt 01xx: Parsed Text Format

FC0100 - Parsed text with dashes
FC0101 - Parsed text, replace dashes with “d”
Field Labels - TR-transit, AC-account #, CK-check #, AM-amount, TP-tpc, EP-epc
Example: - PTTR444455556;AC 999-222-3;CK11045

Fmt 02xx: Parsed Text Format with Error Labels

FC0200 - Parsed text with dashes
FC0201 - Parsed text, replace dashes with “d”
Error Labels - PE-parsed error, NE-no error, TR-transit error, CK-chk # error, TC-transit check digit error, AM-amount error, OU-on us/account# error, TP-tpc error
Examples: - PTTR444455556;AC999-222-3;CK11045/PENE
- PTTR111?11111;AC123456/PETR (“?” = unreadable character)

Fmt 03xx: **[acct #]**

• **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
- keep spaces and dashes

Fmt 04xx: **[acct #]**

• **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
- remove spaces and dashes

Fmt 05xx: **[acct #]**

• **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
- replace spaces and dashes with zeros

Fmt 06xx: **[acct #]**

• **[acct #]**: - always xx characters, zero filled; when xx=00 all characters are sent
- replace spaces and dashes with zeros

Fmt 07xx: **[acct #]**

• **[acct #]**: - always xx characters, zero filled; when xx=00 all characters are sent
- remove spaces and dashes

Fmt 08xx: **[transit] [acct #]**

• **[transit]**: - all characters in the field
- keep dashes

• **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
- remove spaces and dashes

Fmt 09xx: [transit] [acct #]

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - replace spaces and dashes with zeros

Fmt 10xx: [transit] [acct #]

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - always xx characters, zero filled; when xx=00 all characters are sent
  - replace spaces and dashes with zeros

Fmt 11xx: [transit] 'T' [acct #] 'A' [check #]

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field

Fmt 12xx: [transit] 'T' [acct #] 'A' [check #]

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 6 characters, zero filled

Fmt 13xx: [transit] 'T' [acct #] 'A' [check #] '000'

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 6 characters, zero filled

Fmt 14xx: **[transit] [acct #] [check #]**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 6 characters, zero filled

Fmt 15xx: **[bank #] [acct #]**

- **[bank #]**: - all characters in the field
  - keep spaces and dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 16xx: **[bank #] [chk dgt] [acct #]**

- **[bank #]**: - all characters in the field
  - keep spaces and dashes
- **[chk dgt]**: - all characters (one character long)
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 17xx: **[transit] [acct #]**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - keep spaces and dashes

Fmt 18xx: **[acct #] "/" [check #]**

- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - keep spaces and dashes
- **[check #]**: - all characters in the field

Fmt 19xx: **[transit] [acct #] [check #]**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - replace spaces and dashes with zeros
- **[check #]**: - all characters in the field

Fmt 20xx: **[transit] [acct #] <CR> [check #]**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - replace spaces and dashes with zeros
- **[check #]**: - all characters in the field

Fmt 21xx: **[transit] [acct #] [check #]**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - always xx characters, zero filled;
  when xx=00 all characters are sent
  - replace spaces and dashes with zeros
- **[check #]**: - all characters in the field

Fmt 22xx: **[bank #] [acct #] [check #]**

- **[bank #]**: - all characters in the field
  - keep dashes
- **[acct #]**: - always xx characters, zero filled;
  when xx=00 all characters are sent
  - replace spaces and dashes with zeros
- **[check #]**: - all characters in the field

Fmt 23xx: **[error #] [transit] [acct #] [check #] 'S'**

- **[error #]**: - one digit, always present
  - '0' read OK
  - '1' read error: bad char, empty field, invalid length, validation
- **[transit]**: - always 9 characters, zero filled
  - keep dashes
- **[acct #]**: - always xx characters, trailing spaces;
  when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 6 characters, zero filled
  - remove spaces and dashes

Fmt 24xx: **[transit] 'T' [acct #] 'A' [check #] 'C' [amount] '$'**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 6 characters, zero filled
- **[amount]**: - all characters in the field

Fmt 25xx: **'M' 'C' [transit] 'D' [acct #] 'E' [check #]**

- **[transit]**: - all characters in the field
  - remove dashes and keep spaces (contig spcs = 1 spc)
  - if the field is empty, remove 'C'
- **[acct #]**: - include leading characters
  - maximum of xx characters; when xx=00 all characters are sent
  - remove dashes and keep all spaces
  - if the field is empty, remove 'D'
- **[check #]**: - all characters in the field
  - if the field is empty, remove 'E'

Fmt 26xx: **[acct #]**

- **[acct #]**: - work with characters in acct and transit fields
  - a window of xx characters; xx must be greater than 00
  - remove spaces and dashes

Fmt 27xx: **[acct #]**

- **[acct #]**: - work with characters in the acct field only
  - a window of xx characters; xx must be greater than 00
  - remove spaces and dashes

Fmt 28xx: **[acct #]**

- **[acct #]**: - work with characters in the acct field only
  - a window of xx characters; xx must be greater than 00
  - minimum of 6 digits, fill with zeros if necessary
  - remove spaces and dashes

Fmt 29xx: **'C' '/' [transit] '/' [acct #] '/' [check #] '/' [status]**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - maximum of 6 digits
- **[status]**: - this is a programmable option that must be enabled (See Table 4-4).

Fmt 30xx: **[zero fill] [transit] [acct #]**

- **[zero fill]**: - if length of (transit+account) is less than xx; xx must be greater than 00
- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - all characters in the field
  - remove spaces and dashes

Fmt 31xx: **[transit] '/' [acct #] '/' [check #]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - maximum of 10 digits
  - remove spaces and dashes
  - if no check number, remove preceding slash ('/')

Fmt 3200: **'^' [transit] '^' [acct #] '^' [check #] '^' [status]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - all characters in the field
  - remove spaces and dashes
- **[check #]**: - all characters in the field
  - remove spaces and dashes
- **[status]** : - this is a programmable option that must be enabled (See Table 4-4).

Fmt 3300: **'=' [transit] '=' [acct #] '=' [check #] '=' [status]**

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]** : - maximum of 14 digits
  - remove spaces and dashes

- **[check #]**: - maximum of 8 digits
  - remove spaces and dashes

- **[status]**: - this is a programmable option that must be enabled (See Table 4-4).

Fmt 34xx: **[transit] [acct #] [zero fill]**

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]**: - all characters in the field
  - remove spaces and dashes

- **[zero fill]**: - zero filled up to xx; xx must be greater than 00

Fmt 3500: **MA [aux] B [epc] C [tran] D [acct] E [chk] F [tpc] G [amt]**

This format is defined specifically for Target Test Checks. A description of the Target Test Check must be loaded in the exception table.

- **[aux], [epc], [tran], [chk], [tpc], [amt]**:
  - all characters in the field
  - keep spaces and dashes

- **[acct]**: - all characters in the field
  - keep spaces and remove dashes

Fmt 36xx: Read OK : **[transit] [acct #] [check #] '/'**
Read error: **'0' '/'**

- **[transit]**: - all characters in the field
  - remove spaces and dashes

- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

- **[check #]**: - always 6 characters, zero filled
  - remove spaces and dashes

Fmt 37xx: **[ABA] [chk dgt] [acct #]**

- **[ABA], [chk dgt]**:
  - all characters in the field
  - keep spaces and dashes

- **[acct #]**:
  - work with characters in the acct field only
  - window of xx characters; xx must be greater than 00
  - remove spaces and dashes

Fmt 38xx: **'T' [transit] 'A' [acct #] 'C' [check #]**

- **[transit]**:
  - all characters in the field
  - keep dashes

- **[acct #]**:
  - maximum of xx characters; when xx=00 all characters are sent
  - include leading characters
  - keep spaces and dashes

- **[check #]**:
  -all characters in the field

Fmt 39xx: **[transit] <CR> [acct #]**

- **[transit]**:
  all characters in the field
  - remove dashes

- **[acct #]**:
  - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and keep dashes

Fmt 40xx: **[country code] [transit] [acct #]**

- **[country code]**:
  - '1' for US checks
  - '2' for Canadian checks

- **[transit]**:
  - all characters in the field
  - remove dashes

- **[acct #]**:
  - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 4100: **'S' 'T' [transit] 'A' [acct #] 'C' [check #]**

- **[transit]**:
  - all characters in the field
  - remove dashes

- **[acct #]**:
  - all characters in the field
  - place a slash ('/') after 10th character
  - if 10 characters or less, precede with a slash ('/')
  - remove spaces and dashes
- **[check #]**:
  - always 6 characters, zero filled
    remove spaces and dashes

Fmt 42xx: US check : **[transit] [acct #]**

Can check: **'9' [transit] [acct #]**

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]**: - always xx characters; zero filled;
  when xx=00 all characters are sent.
  - remove spaces and dashes

Fmt 43xx: **[check #] <CR> <CR> [transit] <CR> [acct #]**

- **[check #]**: - maximum of 6 digits
  - remove spaces and dashes

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 44xx: **[transit] [acct #]**

- **[transit]**: - all characters in the field
  - if Canadian check, replace dash with a space

- **[acct #]**: - always xx characters, trailing spaces,
  when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 45xx: **[transit] <CR> [acct #] <CR> [check #]**

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces, dashes and leading zeros

- **[check #]**: - all characters in the field

Fmt 46xx: **[transit] [acct #] [check #]**

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]**: - always xx characters, zero filled;
  when xx=00 all characters are sent
  - remove spaces and dashes

- **[check #]**: - always 6 characters, zero filled
  - remove spaces and dashes

Fmt 47xx: **[transit] 'T' [acct #] 'A' [check #]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field

Fmt 48xx: **[transit] 'T' [acct #] 'A'**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 49xx: **[transit] '/' [acct #] '/' [check #] '/' [check type]**

- **[transit]**: - always 9 characters, zero filled
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - maximum of 9 digits
- **[check type]**:- personal checks ('1'); commercial checks ('2')

Fmt 50xx: **'T' [transit] 'T' 'O' [acct #] 'O' [check #]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field

Fmt 51xx: **'=' [transit] '=' [acct #] '='**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 52xx: **'T' [transit] 'T' [acct #] 'A' [check #]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field
  - remove dashes and spaces

Fmt 53xx: **'/' [transit] '/' [acct #] '/' [check #] '/' [tpc] '/' [status] '/'**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field
- **[tpc]**: - all characters in the field
- **[status]**: - this is a programmable option that must be enabled (See Table 4-4)

Fmt 54xx: **[transit] [acct #] [check #] [status]**

- **[transit]**: - always 12 characters, zero filled
  - remove dashes
- **[acct #]**: - always xx characters, zero filled;
  when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 12 characters, zero filled
  - remove dashes and spaces
- **[status]**: - this is a programmable option that must be enabled (See Table 4-4)

Fmt 55xx: **'C' '/' [acct #] '/' [transit] '/' [check #] '/' 0000000000**

- **[acct #]**: - always xx characters, zero filled;
  when xx=00 all characters are sent
  - remove spaces and dashes
- **[transit]**: - all characters in the field
  - remove dashes
- **[check #]**: - always 6 characters, zero filled
  - remove dashes and spaces

Fmt 56xx: **[transit] <CR> [acct #] <CR> [check #] <CR> [amount]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field
  - remove dashes and spaces
- **[amount]**: - all characters in the field
  - remove dashes and spaces

Fmt 57xx: **[acct #] <CR> [amount]**

- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[amount]**: - all characters in the field
  - remove dashes and spaces

Fmt 58xx: **[short transit] [acct #] ':'**

- **[transit]**: - 3 rightmost characters
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 59xx: **[transit] [acct #] <TAB> [check #] [amount]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 9 characters, zero filled
  - remove dashes and spaces
- **[amount]**: - all characters in the field
  - remove dashes and spaces
  - insert decimal point ('.') before 2nd rightmost digit

Fmt 60xx: **[transit] '/' [acct #] '/' [check #] '/' [check type]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - maximum of 10 characters
  - remove spaces and dashes
  - if no check #, remove preceding slash ('/')
- **[check type]**:- personal checks ('1'); commercial checks ('2')

Fmt 61xx: **[transit] <TAB> [acct #] <TAB> [check #] <TAB>**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces, dashes and leading zeros
- **[check #]**: - all characters in the field

Fmt 62xx: **'T' [transit] 'T' [acct #] 'A' [check #] 'S' [status]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field
  - remove dashes and spaces
- **[status]**: - this is a programmable option that must be enabled (See Table 4-4).

Fmt 63xx: **[transit] [acct #] [check #]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - always 4 characters, zero filled
  - remove spaces and dashes

Fmt 64xx: **[transit] [acct #] [check #] [amount]**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - always xx characters, trailing spaces;
  when xx=00 all characters are sent
  - keep spaces and dashes
- **[check #]**: - always 6 characters (N is on quick-init check), trailing spaces
  - remove spaces and dashes
- **[amount]**: - all characters in the field
  - remove spaces and dashes
  - insert decimal point ('.') before 2nd rightmost digit

Fmt 65xx: **'!' [transit] '/' [acct #] '/' [check #] '/' [amount]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes
- **[check #]**: - all characters in the field
  - remove dashes and spaces
- **[amount]**: - all characters in the field
  - remove dashes and spaces

Fmt 66xx: **[transit] [acct #] <CR> '7' '1' <CR>**

- **[transit]**: - all characters in the field
  - keep dashes
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

Fmt 67xx: **<CR> <CR> [check #]**

- **[check #]** : - maximum of xx characters; when x=00 all characters are sent
  - remove spaces and dashes

Fmt 68xx: **[transit] <TAB> [acct #] <TAB> [check #] <TAB> [amount] <TAB>**

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove spaces and dashes

- **[check #]**: - all characters in the field
  - remove dashes and spaces

- **[amount]**: - all characters in the field
  - remove dashes, spaces and leading zeros
  - insert decimal point ('.') before 2nd rightmost digit

Fmt 69xx: Read OK : **[transit] [acct #] [check #]**

Read error: **'0' '/'**

- **[transit]**: - all characters in the field
  - remove dashes

- **[acct #]**: - always xx characters, trailing spaces;
  when xx=00 all characters are sent
  - remove spaces and dashes

- **[check #]**: - always 6 characters, zero filled
  - remove dashes and spaces

Fmt 70: **[transit] ',' [acct #] ',' [check #] ',' [amount]**

- **[transit]**: - all characters in the field
  - keep dashes

- **[acct #]**: - always N characters (N is on quick-init check), space filled
  - remove spaces and dashes from the account

- **[check #]**: - always 8 characters, zero filled
  - remove dashes and spaces

- **[amount]**: - all characters in the field
  - remove dashes and spaces
  - if amount is not present, remove last ','

Fmt 71: **[acct #] '?' [check #]**

- **[acct #]**: - work with a window of N characters in the acct field
  - always N characters (N is on quick-init check), zero filled
  - remove spaces and dashes

- **[check #]**: - maximum of 4 characters
  - remove spaces and dashes

Fmt 72: **[transit] <TAB> [acct #]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of N characters (N is on quick-init check)
  - remove spaces and dashes

Fmt 73: **[transit] <CR> [acct #] <CR> [check #]**

- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - maximum of N characters (N is on quick-init check)
  - remove spaces and dashes
- **[check #]**: - all characters in the field
  - remove dashes and spaces

Fmt 74: **[transit] [acct #] [check #]**

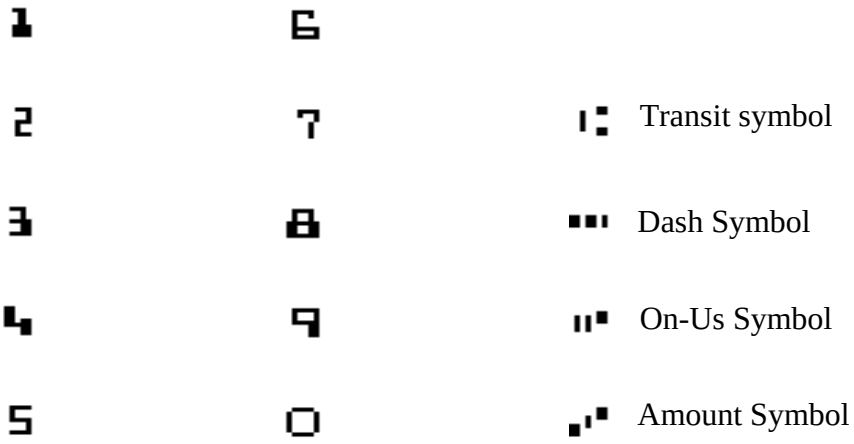
- **[transit]**: - all characters in the field
  - remove dashes
- **[acct #]**: - always N characters (N is on quick-init check), zero filled
  - remove spaces and dashes
- **[check #]**: - always 8 characters, zero filled
  - remove spaces and dashes

APPENDIX B. CHECK READING

The characters printed on the bottom line of commercial and personal checks are special. They are printed with magnetic ink to meet specific standards . These characters can be read by a MICR Reader at higher speeds and with more accuracy than manual data entry. Two MICR character sets are used world-wide; they are: E13-B and CMC-7. The E13-B set is used in the US, Canada, Australia, United Kingdom, Japan, India, Mexico, Venezuela, Colombia, and the Far East. The CMC-7 set is used in France, Spain, other Mediterranean countries, and most South American countries.

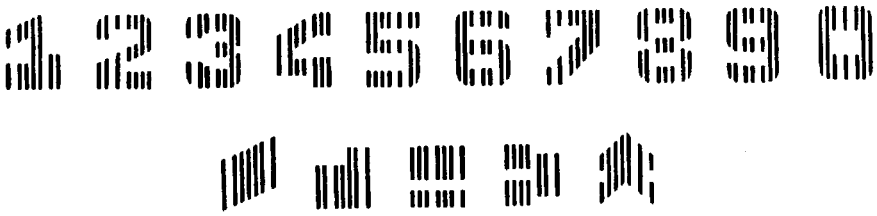
E13-B CHARACTER SET

The MICR font character set E13-B includes digits 0 through 9 and four symbols. The numbers found on U.S. checks are of the E13-B character set. The numbers and symbols of E13-B are as follows:



CMC-7 CHARACTER SET

The numbers and symbols of the CMC-7 character set are as follows:



SI SII SIII SIV SV

The nonnumeric CMC-7 characters are translated by the MICR Plus as shown in Table B-1.

Table B-1. CMC-7 Nonnumeric Characters

CMC-7 Character	MICR Plus Output
SI	A
SII	B
SIII	C
SIV	D
SV	E

CHECK LAYOUTS

Personal checks with MICR fields are shown in Figure B-1. Business checks are shown in Figure B-2. The digits 1 through 4 in the illustrations are described below under MICR Fields.

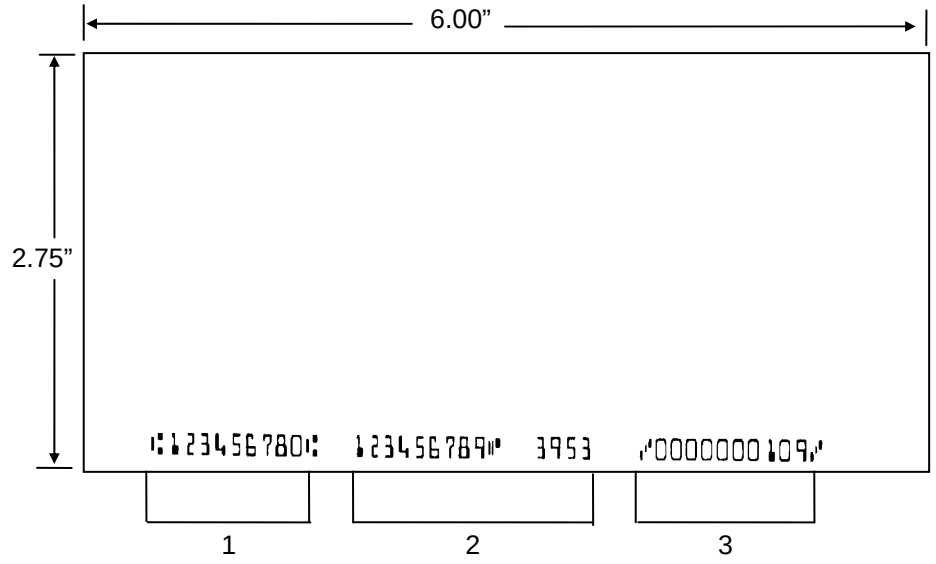


Figure B-1. Personal Checks

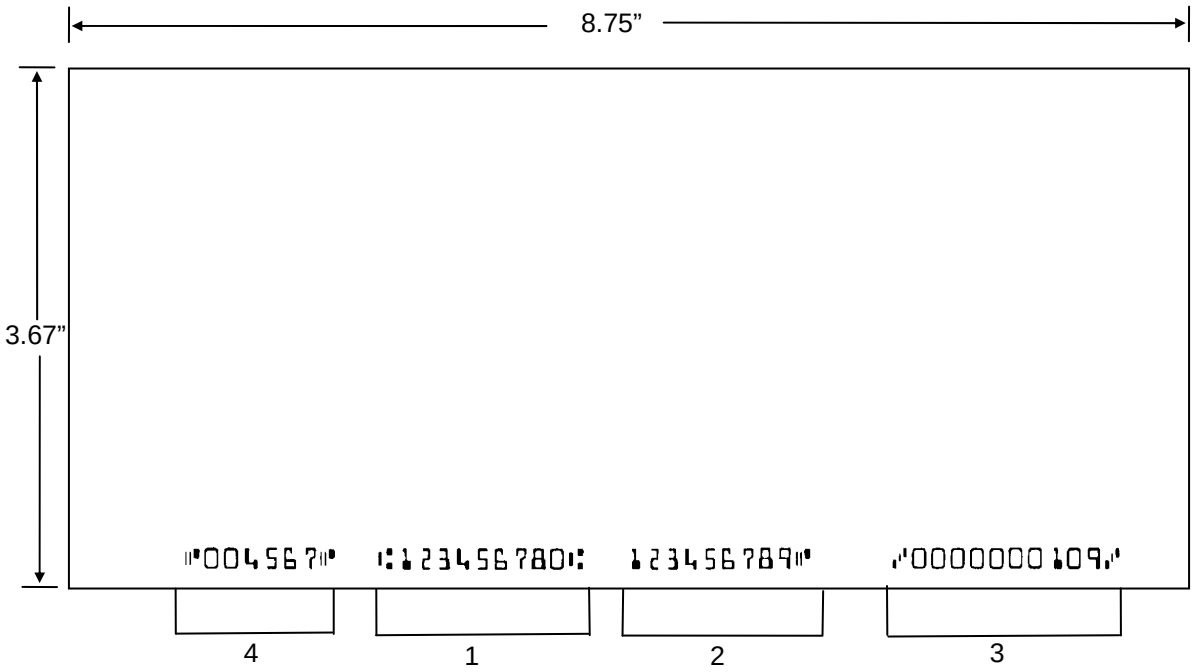


Figure B-2. Business Checks

MICR FIELDS

The numbers 1 through 4 refer to the numbers below the checks on the illustration and represent the 4 MICR fields.

1-Transit Field

The Transit field is a 9-digit field bracketed by two Transit symbols. The field is subdivided as follows:

- Digits 1-4 Federal Reserve Routing Number
- Digits 5-8 Bank ID Number (American Banking Association)
- Digit 9 Check Digit

2-On-Us Field

The On-Us field is variable, up to 19 characters (including symbols). Valid characters are digits, spaces, dashes, and On-Us symbols. The On-Us field contains the account number and may also contain a serial number (Check number) and/or a transaction code. Note that an On-Us symbol must always appear to the right of the account number.

3-Amount Field

The Amount field is a 10-digit field bracketed by Amount symbols. The field is always zero-filled to the left.

4-Auxiliary On-Us Field

The Auxiliary On-Us field is variable, 4-10 digits, bracketed by two On-Us symbols. This field is not present on personal checks. On business checks, this field contains the check serial number.

APPENDIX C. TROUBLESHOOTING GUIDE

REQUIREMENTS

- Personal Computer.
- RS-232 cable, P/N 22617504 or 22617506.
- AC adapter, P/N 64300050.
- SET-MICR program, P/N 22000020 (Rev P or higher).
- Sample checks, P/N 96530005.
- A small bottle of compressed air.
- A cleaning card, P/N 96700006.

SET-UP

1. Plug the 9-pin din connector of the RS-232 cable into the MICR Plus.
2. Plug the DB25 or DB9 connector of the RS-232 cable into the PC.
3. Power on the MICR Plus.
4. Run the SET-MICR program on the PC.
5. Press <F9> to establish communication between the PC and the MICR Plus.

PROCEDURE

Start trouble-shooting procedure at Step 00.

00	**Check LED**

Check the status of the LED indicator:

◊ off, continue to step 01.
◊ green, continue to step 02.
◊ blinking red, continue to step 11.
◊ blinking green, continue to step 17.
◊ blinking red/green, continue to 13.
◊ red or orange, continue to step 18.

01	Check the Power to the MICR Plus

Possible causes for this problem are:

- AC adapter connection to outlet - make sure the AC adapter is securely connected to outlet on the wall or power strip.
- AC adapter connection to MICR Plus - make sure the AC adapter is securely connected to the power jack on the MICR Plus.
- Power strip - if using a power strip, make sure the strip is connected to outlet on the wall and the switch on the strip is turned on.
- AC adapter is defective - replace the AC adapter.

Determine if any of the conditions described above are true:

◊ If yes, rectify and continue to step 00.
◊ If no, continue to step 18.

02	Read a check

Read a check through the MICR Plus:

◊ If the check is transported all the way around the check path, continue to step 03.
◊ If the check gets "stuck" in the check path, continue to step 10.
◊ If the motor does not turn on, continue to step 18.

03	Did PC receive data?

After the check is read, did the PC receive any data?

◊ If yes, continue to step 04.
◊ If no, continue to step 05

04	Analyze data

Analyze the data received by the PC:

◊ If the data is good, continue to step 16.
◊ If the data contains one or more '?', continue to step 06.
◊ If the data is missing characters, continue to step 07.
◊ If the data is garbled, continue to step 08.
◊ If the data is good but not what is expected, continue to step 09.

05	Verify parameters

Use SET-MICR to verify the following parameters:

- "Send Data After Error" - if this option is set to NO, the MICR Plus will not send any data after a read error. Use SET-MICR to change this option to YES.
- "Use CTS/DSR" - if this option is set to USE, the MICR Plus will not send any data unless the CTS and DSR signals are enabled. Use SET-MICR to change this option to IGNORE.

Determine if any of the conditions described above are true:

◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 14.

06	Read error

Possible causes for this problem are:

- Printing problem - the check being read may not meet the requirements of the ANSI Standards. Use one the sample checks provided by MagTek .
- Feeding the check - do not hold on to the check as it goes around the path. Release the check immediately after the MICR Plus "grabs" it. Also, make sure that the front end is not tilted up while the check is being read.
- Foreign debris – power off the MICR Plus and try to push out any loose debris on the check path. Grab the cleaning card and force it through the check path (this is a manual process, the motor will not turn on). Try this procedure several times until the debris comes out. Power on the MICR Plus again.

Determine if any of the conditions described above are true:
◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 15.

07	Missing characters

Possible causes for this problem are:
- Character rate - the character rate at which the MICR Plus is transmitting data may be too fast for the PC. Use SET-MICR to set the "Inter-character Delay" option to YES.
- Feeding the check - When feeding the check, make sure that the MICR line is at the bottom and the printed side of the check is facing the MagTek logo on the MICR Plus.

Determine if any of the conditions described above are true:
◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 08.

08	Communication parameters do not match

Verify that the communication parameters of the MICR Plus match the parameters of the PC. Use SET-MICR to verify/change the communication parameters.

Determine if the condition described above is true:
◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 15.

09	Incorrect Format

Possible causes for this problem are:
- Incorrect Format Number - the current Check data format in the MICR Plus is not the desired format. Use SET-MICR to verify/change the format.
- Incorrect Message Format - the current Message format in the MICR Plus is not the desired format. Use SET-MICR to verify/change the Message format.

Determine if any of the conditions described above are true:
◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 18.

10	**Path is obstructed**

Foreign debris is obstructing the check path:

- Loose debris - power off the MICR Plus and try to push out any loose debris on the check path. Grab the cleaning card and force it through the check path (this is a manual process, the motor will not turn on). Try this procedure several times until the debris comes out. Power on the MICR Plus.
- Wedged debris - the debris is wedged in and cannot be removed with the procedure described above.

Is the foreign debris removable?

◊ If yes, remove and continue to step 02.
◊ If no, continue to step 18.

11	**Motor sensor is blocked (motor does not run)**

The Motor sensor may be blocked by dust build-up or foreign debris (see Figure C-1). Use forced air to clean the sensor.

Power off the MICR Plus and then power on again, observe the LED indicator:

◊ If the LED indicator blinks red, continue to step 18.
◊ Any other LED indicator status, continue to step 00.

12	**EMF noise/interference**

When idle, The MICR Plus monitors the signal coming from the MICR head. If any signal (noise/interference) with amplitude large enough to affect check reading is detected, the LED indicator blinks red/green. Possible sources of EMF are monitors, AC adapters, or magnetic devices. Move the MICR Plus at least 6 inches away from the source of noise/interference.

Determine if the condition described above is true:

◊ If yes, rectify and continue to step 00.
◊ If no, continue to step 13.

13	Data sensor is blocked (motor does not run)

The data sensor may be blocked (see Figure C-1). Try one or both of the following procedures:

- Forced air - use forced air to clean the sensor.
- Cleaning card - power off the MICR Plus and try to push out any loose debris on the check path. Grab the cleaning card and force it through the check path (this is a manual process, the motor will not turn on). Try this procedure several times until the debris comes out.

Power off the MICR Plus and then power on again, observe the LED indicator:

◊ If the LED indicator blinks red/green, continue to step 18.
◊ Any other LED indicator status, continue to step 00.

14	No MICR data detected

Possible causes for this problem are:

- No MICR characters - the ink used to print the MICR characters does not have magnetic properties. Try one of the sample checks provided by MagTek.
- Feeding the check - When feeding the check, make sure that the MICR line is at the bottom and the printed side of the check is facing the MagTek logo on the MICR Plus.

Determine if any of the conditions described above are true:

◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 15.

15	Cable problem

Possible causes for this problem are:

- Loose connection - the cable connector on the PC or the MICR Plus may be loose. Make sure that both connectors are tightly connected.
- Damaged cable - the connectors, pins or wires in the cable may be damaged. Replace cable.

Determine if any of the conditions described above are true:

◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 18.

16	**No problem found**

The MICR Plus is operating properly. If you have additional concerns or requirements please contact your MagTek representative.

17	**Read Insta-Change check**

Read Insta-Change check with the appropriate settings. Return to step 00. If condition persists, continue to step 18.

18	**Return MICR Plus to MagTek**

The MICR Plus has a problem that needs further analysis, testing, and possibly repair. Please contact the MagTek Help Desk at (651) 415-6800, and make arrangements to send the unit back to MagTek. Include a detailed description of the problem.

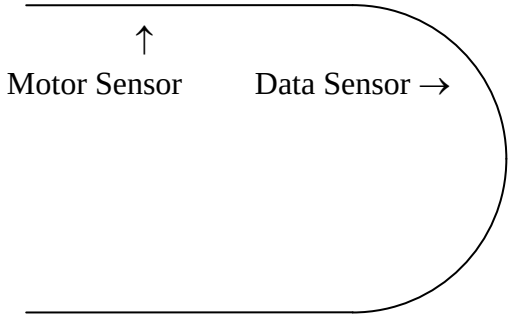


Figure C-1. Sensor Location

APPENDIX D. RS-232 AUXILIARY PORT

In addition to the RS-232 Host port, the MICR Plus offers the added benefit of an RS-232 auxiliary port. This port allows the user to attach an additional RS-232 device to the Host through the MICR Plus. For example, Figure D-1 shows Mag-Tek's IntelliPIN device attached the RS-232 auxiliary port.

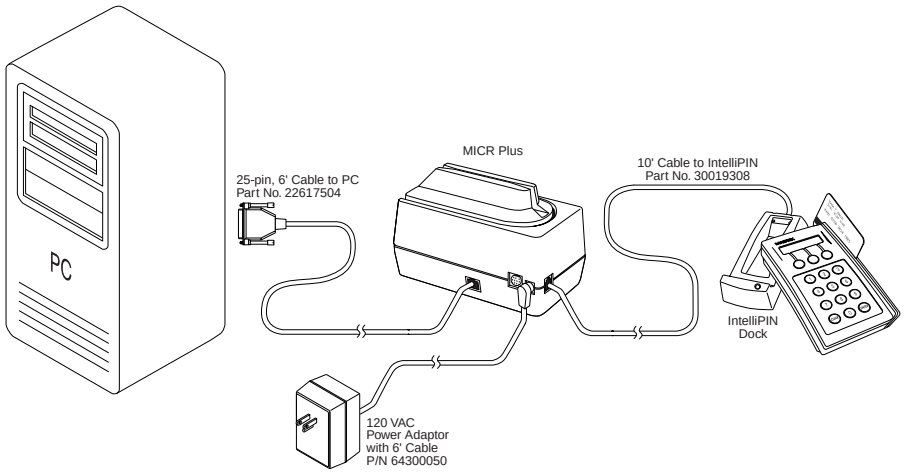


Figure D-1. IntelliPIN on Auxiliary RS-232 Port

In this setup, the MICR Plus serves as a communication bridge between the Host and the RS-232 device, directing the data flow between the Host and the RS-232 device. It should be noted that at all times, the Host is responsible for the control and operation of the RS-232 device.

COMMUNICATION PARAMETERS

The RS-232 auxiliary port is set to the same communication parameters used by the RS-232 Host port.

HOST DATA TO AUXILIARY PORT

The MICR Plus will not automatically direct Host data to the auxiliary port. There are two methods available to direct data to the auxiliary port.

The first method involves bracketing the data with <STX> and <ETX> as follows:

```
<STX>[Host data]<ETX>
```

The MICR Plus will automatically remove <STX> and <ETX> from the message, and will transmit the Host data through the auxiliary port.

The second method requires selecting the “No Reply, Header Required” option for the Invalid Command Response parameter (See SWC Command, Section 4). When this is selected, any Host message intended for the MICR Plus must be preceded with a GS (hex 1D) and terminated with a CR (hex 0D). Any Host message received by the MICR Plus without the GS header will be transmitted, as received, through the auxiliary port. For example:

Message for MICR Plus: `<GS>[Host data]<CR>`

Message for Auxiliary port: `[Host data]`

APPENDIX E. DOWNLOADING

The MICR Plus uses flash memory to store all of its firmware. This technical feature allows firmware changes to be downloaded to the MICR Plus through the RS-232 Host port. The firmware changes are implemented by MagTek and they are distributed in the form of a special file called the BIN file (a file with the .BIN extension). This file can then be downloaded to the MICR Plus with some tools provided by MagTek. The downloading procedure is described below.

REQUIREMENTS

- Laptop or PC
- MICR Plus, RS-232 interface
- Interface cable, DB25, female, Part Number 22617504 or
  Interface cable, DB9, female, part Number 22617506
- AC power adapter with cable, 120V AC to 12 VAC, 1 amp, Part Number 6430050
- Program Mode Insta-Change check for MICR Plus, Part Number 96530022
- Software Utility for Downloading , PROGMICR.EXE, Part Number 22596805
- The BIN file to be downloaded, provided by MagTek

PROCEDURE

1. On the interface cable, connect the DB25/DB9 connector to serial communications port on the PC. Make a note of the port you are using.
2. On the interface cable, connect the 8-pin RJ connector to the RS-232 Host port.
3. On the AC power adapter, connect the jack to the plug on the MICR Plus.
4. On the AC power adapter, connect the plug to the wall outlet.
5. On the PC, create a directory or folder for the MICR Plus. Copy the PROGMICR.EXE program and the BIN file to this directory or folder.
6. Read the Program Mode Insta-Change check through the MICR Plus. The LED indicator will blink red/green fast.
7. On the PC, download the BIN file to the MICR Plus by typing:

`PROGMICR xxxxxxxx.BIN` (for COMM1)

`PROGMICR xxxxxxxx.BIN /2` (for COMM2)

While the BIN file is being downloaded to the MICR Plus, there will be a series of dots on the screen and the LED indicator will blink red/green slow. When the download is complete, the LED indicator will be a steady green and there will be a 4-digit CRC number on the screen. Please make a note of this number for verification.

APPENDIX F. PIN LISTS

The pin list for the DB25 Interface Cable is shown in Table F-1, and the pin list for the DB9 Interface Cable is shown in Table F-2.

Table F-1. DB25 Interface Cable Pin List

PIN	SIGNAL (Host as Reference)	DESCRIPTION
2	TXD	Transmitted Data. Transmits data from the Host to the MICR Plus.
3	RXD	Received Data. Receives data from the MICR Plus to the Host.
4	RTS	Request to Send. Sends a signal to the MICR Plus to indicate that the Host is ready to receive data.
5	CTS	Clear to Send. Receives a signal from the MICR Plus to indicate that the MICR Plus is ready to send data.
6 8	DSR DCD	Data Set Ready. Receives a signal from the MICR Plus to indicate that the MICR Plus is active, i.e., power is on.
7	GND	Ground
20	DTR	Data Terminal Ready. Transmits a signal to the MICR Plus to indicate that the Host is active, i.e., power is on.

Table F-2. DB9 Interface Cable Pin List

PIN	SIGNAL (Host as Reference)	DESCRIPTION
2	RXD	Received Data. Receives data from the MICR Plus to the Host.
3	TXD	Transmitted Data. Transmits data from the Host to the MICR Plus.
4	DTR	Data Terminal Ready. Transmits a signal to the MICR Plus to indicate that the Host is active, i.e., power is on.
5	GND	Ground
6	DSR	Data Set Ready. Receives a signal from the Host to indicate that the MICR Plus is active, i.e., power is on.
7	RTS	Request to Send. Sends a signal to the MICR Plus to indicate that the Host is ready to receive data.
8	CTS	Clear to Send. Receives a signal from the MICR Plus to indicate that the MICR Plus is ready to send data.

The 8-pin RJ socket for the RS-232 Host Port is shown in Figure F-1. The Pin List is shown in Table F-3.

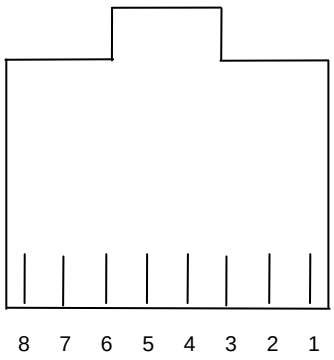


Figure F-1. RS-232 Host Port, 8-Pin, RJ Socket

Table F-3. RS-232 Host Port Pin List

PIN NUMBER	SIGNAL (MICR Plus as Reference)
1	+16V DC, unregulated
2	RXD
3	TXD
4	CTS
5	RTS
6	DSR
7	DTR
8	GND

The 6-pin RJ socket for the RS-232 Auxiliary Port is shown in Figure F-2. The Pin List is shown in Table F-4.

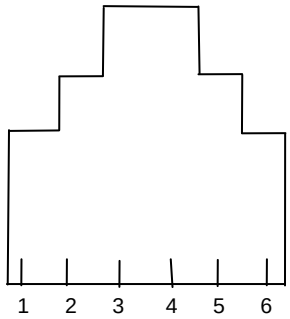


Figure F-2. RS-232 Auxiliary Port, 6-Pin, RJ Socket

Table F-4. RS-232 Auxiliary Port Pin List

PIN NUMBER	SIGNAL (MICR Plus as Reference)
1	+16V DC, unregulated
2	RXD
3	TXD
4	CTS
5	RTS
6	GND

APPENDIX G. ASCII CODES

The following is a listing of the ASCII (American Standard Code for Information Interchange) codes. ASCII is a 7-bit code, which is represented here with a pair of hexadecimal digits. The decimal equivalent follows the hexadecimal value.

ASCII	Hex	Dec	ASCII	Hex	Dec	ASCII	Hex	Dec	ASCII	Hex	Dec
NUL	00	0	SP	20	32	@	40	64	`	60	96
SOH	01	1	!	21	33	A	41	65	a	61	97
STX	02	2	"	22	34	B	42	66	b	62	98
ETX	03	3	#	23	35	C	43	67	c	63	99
EOT	04	4	$	24	36	D	44	68	d	64	100
ENQ	05	5	%	25	37	E	45	69	e	65	101
ACK	06	6	&	26	38	F	46	70	f	66	102
BEL	07	7	'	27	39	G	47	71	g	67	103
BS	08	8	(	28	40	H	48	72	h	68	104
HT	09	9	)	29	41	I	49	73	i	69	105
LF	0A	10	*	2A	42	J	4A	74	j	6A	106
VT	0B	11	+	2B	43	K	4B	75	k	6B	107
FF	0C	12	,	2C	44	L	4C	76	l	6C	108
CR	0D	13	-	2D	45	M	4D	77	m	6D	109
SO	0E	14	.	2E	46	N	4E	78	n	6E	110
SI	0F	15	/	2F	47	O	4F	79	o	6F	111
DLE	10	16	0	30	48	P	50	80	p	70	112
DC1	11	17	1	31	49	Q	51	81	q	71	113
DC2	12	18	2	32	50	R	52	82	r	72	114
DC3	13	19	3	33	51	S	53	83	s	73	115
DC4	14	20	4	34	52	T	54	84	t	74	116
NAK	15	21	5	35	53	U	55	85	u	75	117
SYN	16	22	6	36	54	V	56	86	v	76	118
ETB	17	23	7	37	55	W	57	87	w	77	119
CAN	18	24	8	38	56	X	58	88	x	78	120
EM	19	25	9	39	57	Y	59	89	y	79	121
SUB	1A	26	:	3A	58	Z	5A	90	z	7A	122
ESC	1B	27	;	3B	59	[	5B	91	{	7B	123
FS	1C	28	<	3C	60	\	5C	92	|	7C	124
GS	1D	29	=	3D	61	]	5D	93	}	7D	125
RS	1E	30	>	3E	62	^	5E	94	~	7E	126
US	1F	31	?	3F	63	_	5F	95	DEL	7F	127