MINI MICR WEDGE
WITH OPTIONAL 3-TRACK MSR
TECHNICAL REFERENCE MANUAL

Manual Part Number: 99875074 Rev 3

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	15 May 97	Initial Release
2	20 Sep 99	Sec 1: Added EMF and Device Driver references; Sec 4: Added MICRbase, Device Drivers, Card Data Message, and EMF detect; Appendix A: Added formats 7500, 7600, and 7700; Appendix C: Added Step 12 to Troubleshooting Guide; Appendix G: Added ASCII Codes.
3	18 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 a Class A 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 commercial 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. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

CANADIAN DOC STATEMENT

This digital apparatus does not exceed the Class A 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 A prescrites dans le Réglement sur le brouillage radioélectrique édicté par les ministère des Communications du Canada.

SAFETY

This product has been evaluated, tested, and certified by the Canadian Standards Association (CSA 22.2, No. 950, Underwriters Laboratories (UL1950), and TUV Rheinland (TUV/EN60950). In order to insure that it maintains the safety integrity that was designed into the product, and for which it has been evaluated by the Safety Certification Agencies, compliance with all Installation Instructions and Safety Requirements is essential.

TABLE OF CONTENTS

SECTION 1. OVERVIEW ---------- 1
FEATURES 1
ACCESSORIES 1
SPECIFICATIONS 2

SECTION 2. INSTALLATION ---------- 3
REQUIREMENTS 3
PROCEDURE 3

SECTION 3. OPERATION ---------- 5
CHECK READING PROCEDURE 5
CARD SWIPE PROCEDURE 5
LED INDICATOR 6

SECTION 4. COMMANDS ---------- 7
INSTA-CHANGE CHECKS 7
MICRBASE SETUP PROGRAM FOR WINDOWS 7
MAGTEK DEVICE DRIVERS FOR WINDOWS 8
USER APPLICATIONS WITHOUT MAGTEK DEVICE DRIVERS 8
PC Commands To Enable and Disable Keyboard Wedge 8
PC Timing 8
Keyboard Wedge Port IO Addresses 8
COMMAND FORMAT 9
DK – DISABLE KEYBOARD 9
EK – ENABLE KEYBOARD 9
SWA - SWITCH A COMMAND 9
SWA PARAMETERS 10
Wedge Type 10
Send Data After Error 10
Keypad Type 11
SWB - SWITCH B COMMAND 12
SWB PARAMETERS 12
Control Characters and MICR Data 13
<ENTER> Key Options 13
Send Status After Data 15
SWC - SWITCH C COMMAND 15
SWC PARAMETERS 16
CMC-7 Character Set 16
Card Data Message 16
HW - HARDWARE COMMAND 17
HW PARAMETERS 17
Disable/Enable Tracks 17
ID Card Decoding 18
EMF Detect 18
FC - FORMAT CHANGE COMMAND 18
VR - VERSION COMMAND 18
SA - SAVE COMMAND 19
RS - RESET COMMAND 19

APPENDIX A. FORMAT LIST ---------- 21

APPENDIX B. CHECK READING ---------- 39
E13-B CHARACTER SET 39
CMC-7 CHARACTER SET 39
CHECK LAYOUTS 40
MICR FIELDS 41
1-Transit Field 41
2-On-Us Field 41
3-Amount Field 42

4-Auxiliary On-Us Field 42

APPENDIX C. TROUBLESHOOTING GUIDE ----------43
REQUIREMENTS 43
SET-UP 43
PROCEDURE 43

APPENDIX D. PINPAD INTERFACE TO MICR READER----------51

APPENDIX E. INTERFACE CABLE PIN LISTS----------53

APPENDIX F. ASCII CODES----------55

INDEX ----------57

FIGURES

Figure 1-1. MINI MICR WEDGE with 3-Track MSR..... viii
Figure 2-1. MICR Interface Cabling 3
Figure 3-1. Check Orientation 5
Figure 4-1. Keypad Types 11
Figure 4-2. <Enter> Key Positions 14
Figure B-1. Personal Checks 40
Figure B-2. Business Checks 41
Figure C-1. Sensor Location 49
Figure D-1. PINPad RJ Socket 51
Figure E-1. Large DIN 5-Pin Connectors 53
Figure E-2. Mini DIN 6-Pin Connectors..... 54

TABLES

Table 1-1. Specifications 2
Table 3-1. LED indicators..... 6
Table 4-1. SWA Command 10
Table 4-2. Error and Status Codes 11
Table 4-3. SWB Command 12
Table 4-4. Control Characters..... 13
Table 4-5. SWC Command 16
Table 4-6. HW Command 17
Table B-1. CMC-7 Nonnumeric Characters 40
Table D-1. PINPad RJ Socket..... 52
Table E-1. Large DIN 5-Pin Interface Cable 53
Table E-2. Mini DIN 6-Pin Interface Cable 54

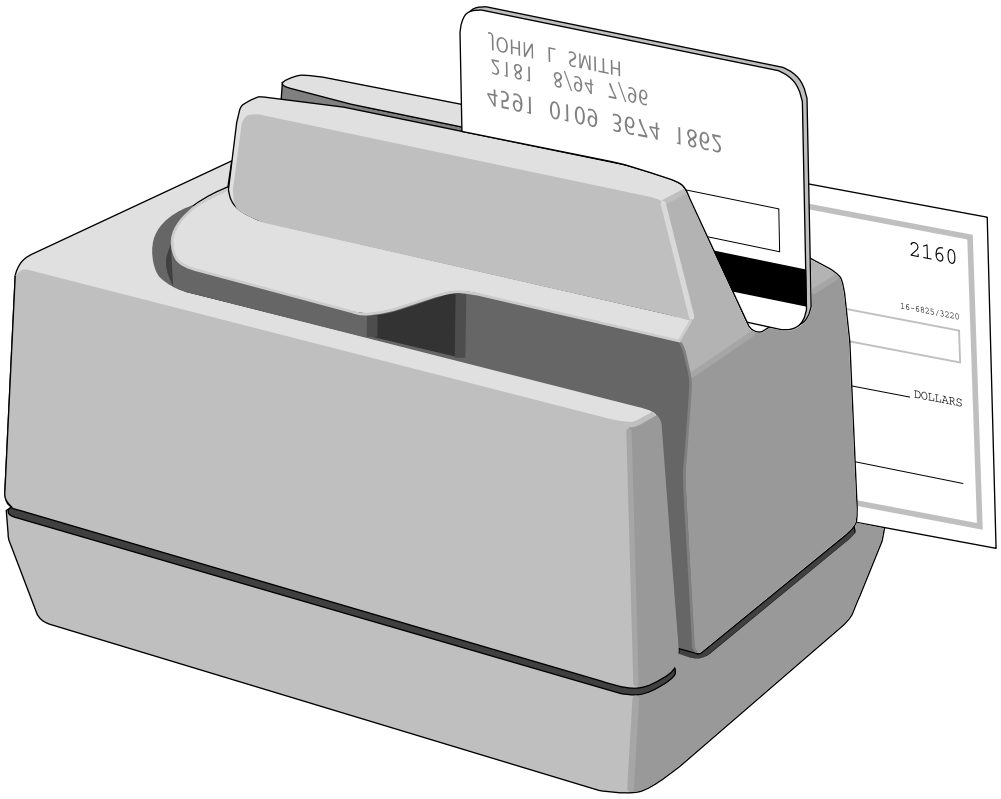


Figure 1-1. MINI MICR WEDGE with 3-Track MSR

SECTION 1. OVERVIEW

The MINI MICR WEDGE With Optional 3-Track MSR is both a MICR (Magnetic Ink Character Recognition) Check Reader and an MSR (Magnetic Stripe Reader).

The MICR Reader, 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 Reader improves accuracy and speed because there is no manual data entry; therefore there are no keying errors or unwanted delays.

The MICR Reader will connect between the PC and the keyboard using the split wedge interface cable. The original connection between the PC and the keyboard is maintained. The MICR Reader will transmit all data as scan codes and the PC will receive it as if it was entered using the keyboard.

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.
- Ability to create custom formats to transmit MICR data, plus an extensive list of built-in formats.
- Optional error/status reporting for check reading.
- Reads E13-B and CMC-7 MICR fonts.
- EMF noise detection
- In addition to the Wedge interface, the MICR Reader is also available with other interfaces.

ACCESSORIES

Accessories available for the MICR Reader are as follows:

- MagTek Device Drivers for Windows, Part Number 30037385
- MagTek Device Drivers for Windows, Programming Reference Manual, Part Number 99875125

- MICRbase Program, P/N 22000021
- MICRbase, Setup Program for MICR Readers, Programming Reference Manual, P/N 99875102
- Interface Cable, Mini DIN 6 pin, Part Number 22517501
- Interface Cable, Large DIN 5 pin, Part Number 22517503
- AC Power Adapter with Cable, 120VAC to 12 VAC, 1 Amp, Part Number 64300050
- MICR Reader Cleaning Card, Part Number 96700006
- Sample Checks, Part Number 96530005
- Encrypting PINPad and Cable, Part Number 30015061 (See Appendix D)
- Encrypting PINPad Specification, Part Number 99815042
- Nonencrypting PINPad and Cable, Part Number 30015026 (See Appendix D)
- Nonencrypting PINPad Technical Description, Part Number 99833004

SPECIFICATIONS

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

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 4.25”
Weight:	3.0 lbs. MSR and Adapter included
Cable length	6’
Interface Connectors	Large DIN 5-pin Mini DIN 6-pin
ENVIRONMENTAL	
Temperature	
Operating	0°C to 50°C (32°F to 122°F)
Storage	-30°C to 70°C (-22°F to 158°F)
Humidity	
Operating	10% to 90% noncondensing
Storage	Up to 100% noncondensing

SECTION 2. INSTALLATION

The installation for the MICR Reader is as follows:

REQUIREMENTS

The following is required for the Installation:

- MINI MICR WEDGE With Optional 3-Track or 2-Track MSR
- Interface Cable, Mini DIN 6 pin, Part Number 22517501 or
  Interface Cable, Large DIN 5 pin, Part Number 22517503
- AC Power Adapter with Cable, 120VAC to 12 VAC, 1 Amp, Part Number 64300050

PROCEDURE

To install the MICR Reader, refer to Figure 2-1 and perform the following steps:

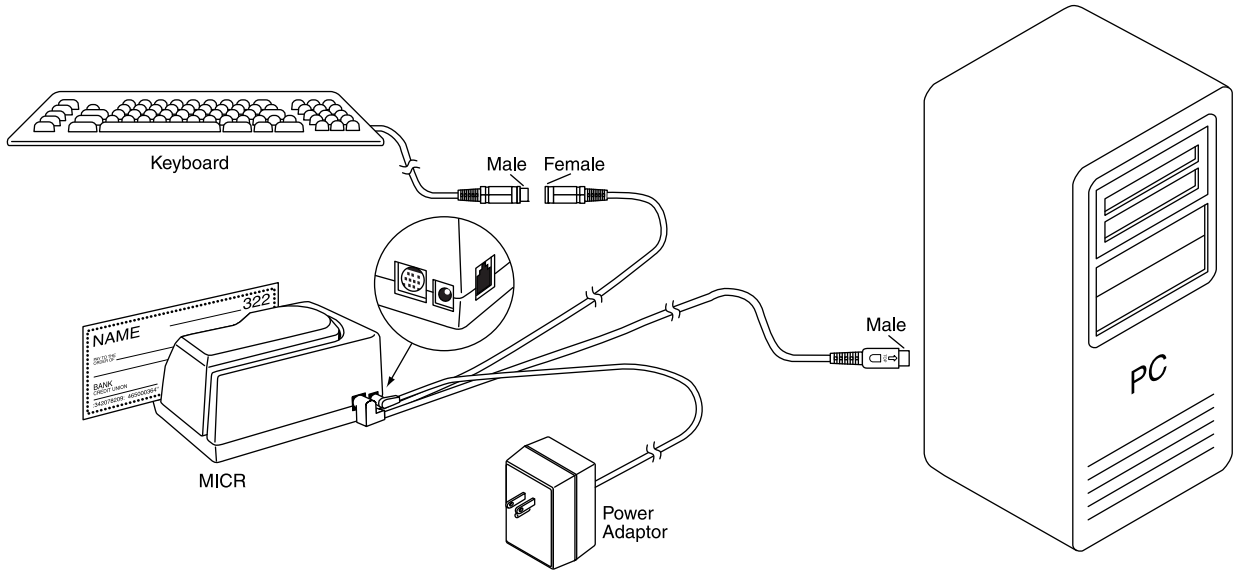


Figure 2-1. MICR Interface Cabling

1. Power off the PC.

2. Disconnect the keyboard from the PC.

3. On the interface cable, connect the 9-pin male DIN connector to the MICR Reader.

4. On the interface cable, connect female DIN connector to the keyboard cable. Pin lists and connector illustrations are shown in Appendix E.

5. On the interface cable, connect the male DIN connector to the PC.

6. On the AC power adapter, connect the jack to the plug on the MICR Reader.

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

8. Power the PC on. The LED indicator on the MICR Reader 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 Reader 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 Reader as shown in Figure 3-1.

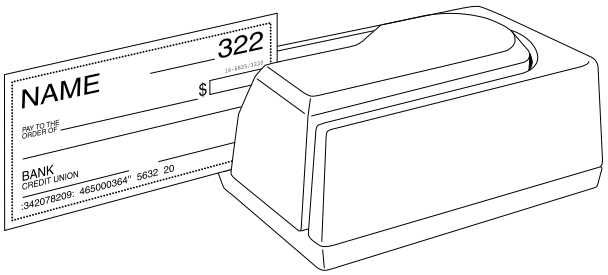


Figure 3-1. Check Orientation

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

3. When the MICR Reader 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 MICR 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 MSR in either direction, but the magnetic stripe must be oriented in only one direction as shown in Figure 1-1. The MICR 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 (Hardware) command (Section 4, Commands).

LED INDICATOR

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.

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
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 Reader.

Note

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

The following methods are available to execute MICR commands:

- Insta-change checks
- MICRbase setup program for Windows
- MagTek Device Drivers for Windows
- User application without drivers

INSTA-CHANGE CHECKS

The Insta-Change check is a MICR encoded document that contains commands and options used to reset the parameters of the MICR Reader. Multiple commands and options may be contained on one Insta-Change check. When used, the Insta-Change checks are run through the MICR Reader 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 Reader as described in Section 2, and watch the LED indicator. When the Insta-Change check is run through the MICR Reader 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.

MICRBASE SETUP PROGRAM FOR WINDOWS

The MICRbase setup program (P/N 22000021) allows the user to control all the programmable options available in the MICR Reader.

The program provides a graphical, user-friendly interface that hides the complexities involved in manually entering MICR commands. The user is no longer required to know the specific commands or the detailed data associated with each command. However, the program still allows manual entry of commands for advanced users. For more detailed information refer to the MICRbase Setup Program Reference Manual (P/N 99875102).

The MICRbase setup program may also be downloaded from the internet at www.magtek.com under Software/Demo Programs.

MAGTEK DEVICE DRIVERS FOR WINDOWS

For Windows applications, the MagTek Device Drivers for Windows (P/N 30037385) are available to simplify the programming of the MICR Reader. The drivers are easy to install, and they facilitate the execution of MICR Reader commands. For more detailed information, refer to the MagTek Device Driver for Windows, Programming Reference Manual (P/N 99875125).

The drivers may also be downloaded from the Internet at www.magtek.com Software/Device Drivers.

USER APPLICATIONS WITHOUT MAGTEK DEVICE DRIVERS

For applications that communicate directly with the MICR Reader, the following programming procedures must be considered:

PC Commands To Enable and Disable Keyboard Wedge

Because the system is connected from the PC to the MICR Reader to the PC keyboard, the user must cut the communication between the MICR Reader and the keyboard. This is accomplished by sending a Disable Keyboard (DK) command. This command permits the PC to send commands to the MICR Reader. After the commands have been sent to the MICR Reader, the PC must send an Enable Keyboard (EK) command to reestablish communication between the PC and the keyboard.

PC Timing

The timing between each character sent to the MICR Reader must be in the range of 5 milliseconds to 150 milliseconds.

The timing between the last key pressed (from the keyboard) and the first character of the MICR Reader command should be 500 milliseconds.

Keyboard Wedge Port IO Addresses

For the PC program, the keyboard port IO address is 60 HEX for AT and PS/2 model 50. The address for all other PS/2 models is 68 HEX.

COMMAND FORMAT

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

[COMMAND][DATA] <ENTER>

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

DK – DISABLE KEYBOARD

When a PC program is used, the keyboard must be disabled to allow programming of the MICR Reader. Commands used for programming the MICR Reader are described below. The command to disable the keyboard is:

DK <ENTER>

The MICR Reader will execute the command and will reply with lower case 'k'.

EK – ENABLE KEYBOARD

After all commands have been sent to the MICR Reader, communication between the PC and the keyboard must be established. The command to enable the keyboard is:

EK <ENTER>

The MICR Reader will execute the command and will reply with lower case 'k'.

SWA - SWITCH A COMMAND

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

To execute, send the SWA command as follows:

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

When sending data, all 8 bits must be provided. The MICR Reader will execute the command but it will not reply. The SWA options are saved immediately.

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

Table 4-1. SWA Command

BITS								PARAMETERS
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
					1	1	0	Wedge AT
					0	1	0	Wedge NCR
			0	0				Send data after error: Yes
			0	1				Send data after error: No
		0						Keypad: calculator (Wedge NCR only)
		1						Keypad: telephone (Wedge NCR only)
0	0							These bits are always set to 0

SWA PARAMETERS

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

Wedge Type

The Wedge type can be for the IBM AT or NCR 7052/7053 cash registers.

Send Data After Error

The request Send Data After Error specifies whether the MICR Reader will return data to the Host after a read error. If YES is selected and the MICR Reader detects a read error, the MICR Reader will still send the data back to the Host. If NO is selected and the MICR Reader finds an error, it will discard the data and nothing will be sent. The error conditions are listed in Table 4-2.

Table 4-2. 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
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 will turn red on all error conditions.
- The absence of a check number is not considered an error.
- If a multiple error condition occurs, the error or status code with the highest priority is reported.
- All unreadable MICR characters are transmitted as a "?" ASCII character (hex 3F), except for Format 00xx (See Appendix A).

Keypad Type

The Keypad selection is for the Wedge NCR only. The selection is the calculator type of layout or telephone layout. Both are shown in Figure 4-1.

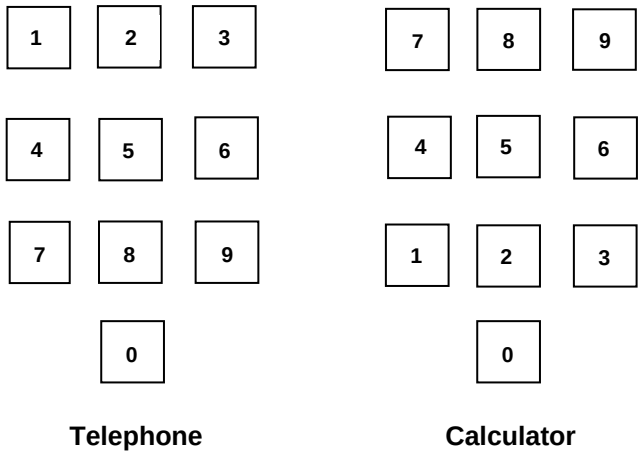


Figure 4-1. Keypad Types

SWB - SWITCH B COMMAND

The SWB command controls the message format, shown in Table 4-3. 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<ENTER> (with data)
or
SWB <ENTER> (without data)

When sending data, all 8 bits must be provided. The MICR Reader will execute the command but it will not reply. The new settings become effective immediately. To make this command permanent, use the SA (Save) command described at the end of this section.

If no data is sent, the MICR Reader responds with the current settings for SWB.

Table 4-3. SWB Command

BIT								PARAMETERS
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
							0	Trailing <ESC>: No
							1	Trailing <ESC>: Yes
0		0				0		<ENTER> key: No - Do Not Send
0		0				1		<ENTER> key: Yes - Default single
1		0				1		<ENTER> key: Yes - Default double
0		1				1		<ENTER> key: Yes - Custom single
1		1				1		<ENTER> key: Yes - Custom double
					0			<ETX>: No
					1			<ETX>: Yes
				0				Leading <ESC>: No
				1				Leading <ESC>: Yes
			0					<STX>: No
			1					<STX>: Yes
	0							Sent Status: No
	1							Sent Status: Yes

SWB PARAMETERS

The SWB functions are listed in Table 4-3 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> <ENTER> <ESC>
```

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

Table 4-4. Control Characters

CONTROL CHARACTER	DESCRIPTION
<STX>	Start of Text
<ESC>	Escape (Leading)
<ETX>	End of Text
<ENTER>	Enter
<ESC>	Escape (Trailing)

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

```
MICR Data: <STX>[data]<ENTER>
```

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>`

<ENTER> Key Options

The <ENTER> key options may be required for cash registers but not for PCs. They provide flexibility for cash register applications where the <ENTER> key may be relocated from its original position to a new position on the keyboard. The original position is called the *default* position and the new position is called is called the *custom* position. The MICR Reader must know whether a *default* or *custom* position is being used to properly emulate the pressing of the <ENTER> key after reading a check or a card.

In addition to relocating the <ENTER> key, the user may require a two-position <ENTER> key. On the keyboard, this is accomplished by placing a plastic cap over two adjacent keys. Hence, when the key is pressed, two signals are sent to the PC or cash register. The one-key option is referred to as *single*, and the two-key option is referred to as *double*.

The concepts of *default*, *custom*, *single*, and *double* are illustrated in Figure 4-2.

Every key-position on a keyboard is associated with a fixed number. This number is called *scan code*. When a key is pressed, the keyboard transmits the corresponding scan code. Similarly, the MICR Reader transmits data using scan codes, effectively emulating keys being pressed on the keyboard.

When a *custom* option is required, the scan code information must be determined by the customer or with the help of MagTek. The scan code must then be programmed into the MICR Reader, and this can be accomplished at the factory or with Insta-Change checks provided by MagTek.

Note

The default scan codes are 5A (hex) for PC/AT and 1C (hex) for PS/2.

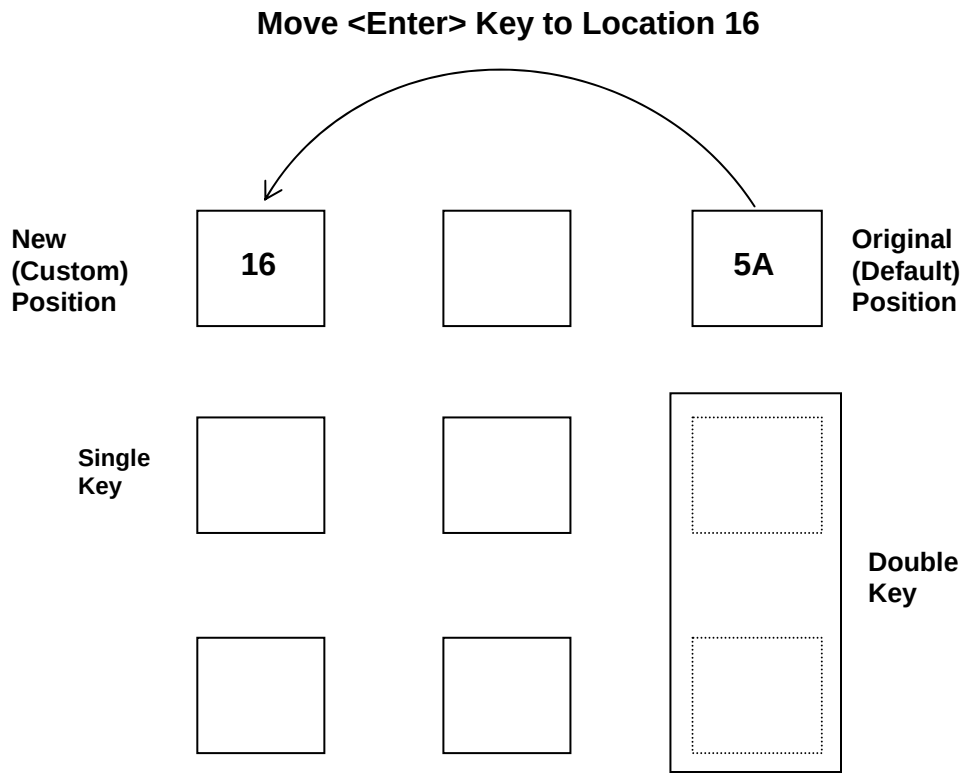


Figure 4-2. <Enter> Key Positions

Combinations of these options as shown in Table 4-3 are as follows:

Do not Send – The MICR Reader will not send the <ENTER> key to the PC.

Default Single – The MICR Reader will use the default scan code for a one-key <ENTER>. Use this option for all PC applications.

Default Double – The MICR Reader will used two default scan codes for a two-key <ENTER>.

Custom Single – The keyboard has a one-key <ENTER>, but it has been moved from its default position. The scan code must be programmed into the MICR Reader.

Custom Double – The keyboard has a two-key <ENTER>, but it has been moved from its default position. The scan codes must be programmed into the MICR Reader

Send Status After Data

The Send Status After Data option makes the MICR Reader 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 4-2.

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 Reader 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 Reader will look like this:

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

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<ENTER> (with data)
or
SWC <ENTER> (without data)

When sending data, all 8 bits must be provided. The MICR Reader will execute the command but will not reply. The new settings become effective immediately. To make this command permanent, use the SA (Save) command described at the end of this section.

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

Table 4-5. SWC Command

BITS								PARAMETERS
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
							0	CMC-7 Character Set: No
							1	CMC-7 Character Set: Yes
		0						Card Data Message: Single
		1						Card Data Message: Multiple
0	0		0	0	0	0		These bits are always set to 0

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.

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>, and <ETX> are set to YES, the output 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>`

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<ENTER> (with data)
or
HW <ENTER> (without data)

When sending data, all 8 bits must be provided. The MICR Reader will execute the command but it will not reply. The new settings become effective immediately. To make this command permanent, use the SA (Save) command described at the end of this section.

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

Table 4-6. HW Command

7	6	5	4	3	2	1	0	PARAMETERS
					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		EMF detect: Yes
						1		EMF detect: No
0	0						0	These bits are always set to 0

HW PARAMETERS

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 MICR 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 indicate 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.

EMF Detect

The EMF Detect option allows the MICR Reader, when idle, to monitor EMF interference in its immediate environment. If YES is selected, the LED indicator will blink red/green when the MICR Reader detects a signal with amplitude large enough to affect check reading. If NO is selected, the MICR Reader will not monitor nor indicate the presence of EMF interference.

FC - FORMAT CHANGE COMMAND

Formats are used by the MICR Reader 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<ENTER> (with data)
or
FC <ENTER> (without data)

When sending data, all 4 digits must be provided. The MICR Reader 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 at the end of this section.

If no data is provided, the MICR Reader will respond with the current format number.

VR - VERSION COMMAND

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

VR<ENTER>

The MICR Reader responds as follows:

MICR data: [software revision]<ENTER>

SA - SAVE COMMAND

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

SA<ENTER>

RS - RESET COMMAND

The Reset command resets the MICR Reader firmware to the normal operating state of waiting for a check or card to be read. To execute, send the RS command followed by a carriage return as follows:

RS<ENTER>

APPENDIX A. FORMAT LIST

For check reading, the MICR Reader 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 Reader 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.

If none of the built-in formats meets the necessary requirements, a custom format can be created using the Flexible Format feature. These custom formats can be easily created and downloaded using the MICRbase Setup Program for Windows provided by MagTek (P/N 22000021). For more detailed information refer to Section 7 in the MICRbase Reference Manual (P/N 99875102).

Each of the built-in formats 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.

A complete description for each format follows.

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.

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

Fmt 75xx: **[transit] <CR> [acct #] <CR> [check #] <CR> [status]**

- **[transit]**: - always 9 characters, zero filled
  - keep dashes; remove spaces

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

- **[check #]**: - maximum of 12 characters
  - remove dashes and spaces

Fmt 76xx: 'T' [transit] 'A' [acct #] 'C' [check #] 'M' [raw data]

- **[transit]**: - all characters in the field
  - remove dashes and spaces
- **[acct #]**: - maximum of xx characters; when xx=00 all characters are sent
  - remove dashes and spaces
- **[check #]**: - all characters in the field- remove dashes and spaces
- **[raw data]**: - translate MICR symbols to t,o,a,d

Fmt 7700: The Flexible Format

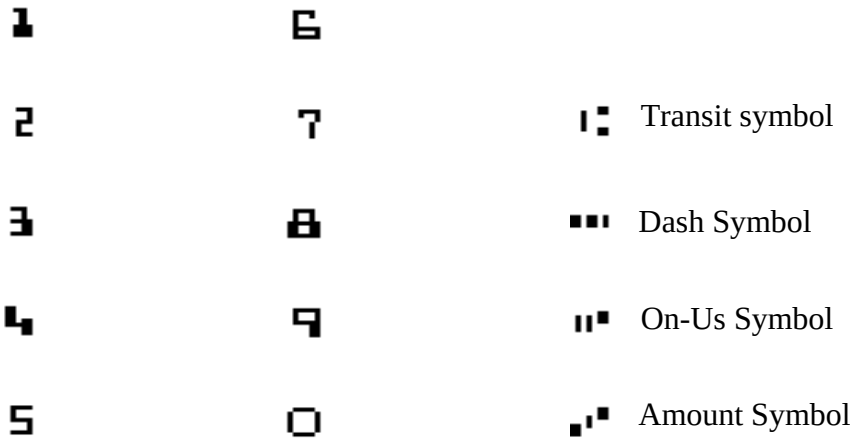
Select this format to activate a preloaded Flexible Format. The Flexible Format is a feature that allows the user to create custom MICR formats. The Flexible formats can be easily created and downloaded using the Windows based MICRbase program provided by MagTek (P/N 22000021). For more detailed information refer to Section 7 in the MICRbase reference manual (P/N 99875102).

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.

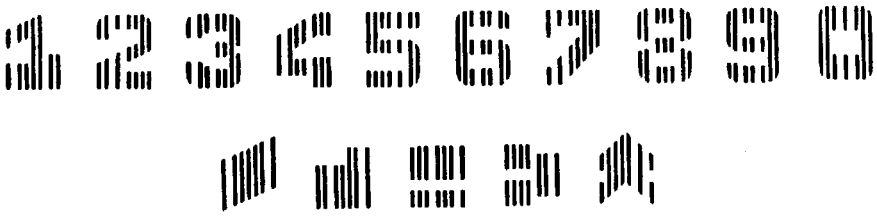
E13-B CHARACTER SET

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:



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

Table B-1. CMC-7 Nonnumeric Characters

CMC-7 Character	MICR Reader 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.

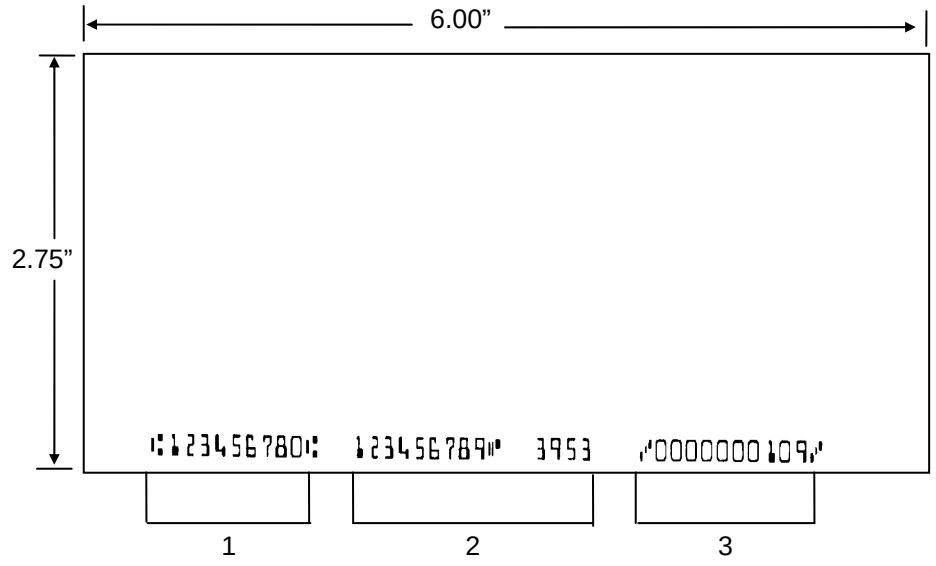


Figure B-1. Personal Checks

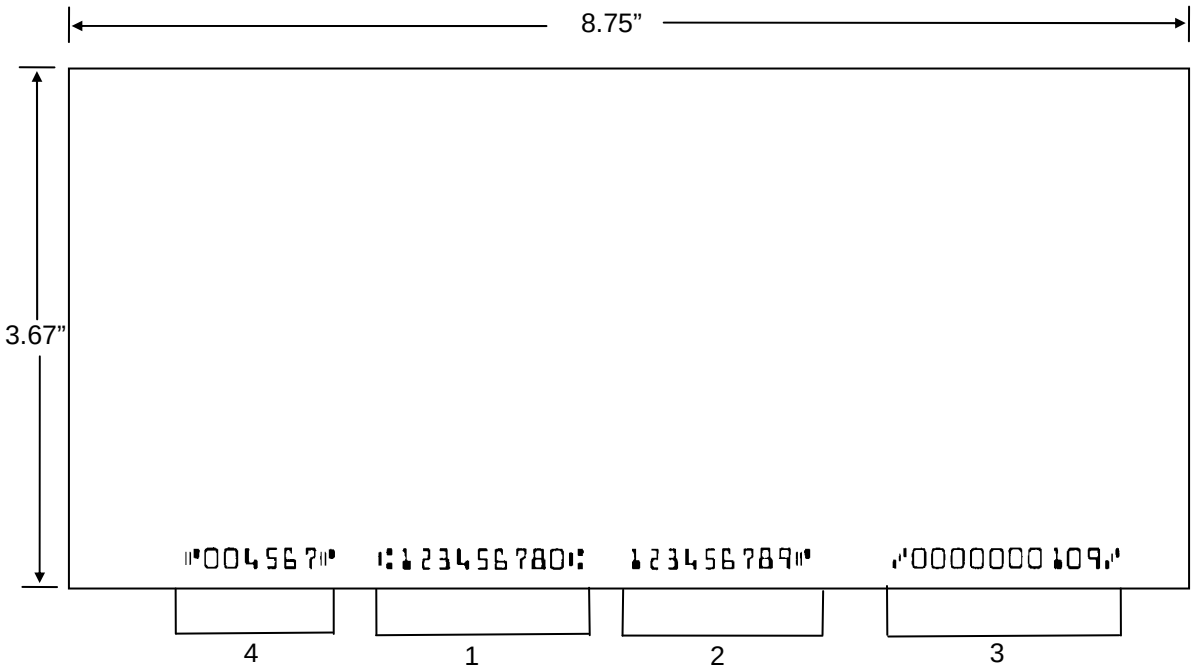


Figure B-2. Business Checks

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.
- Interface Cable, Mini DIN 6 pin, Part Number 22517501 or
  Interface Cable, Large DIN 5 pin, Part Number 22517503
- AC adapter, P/N 64300050.
- Sample checks, P/N 96530005.
- A small bottle of compressed air.
- A cleaning card, P/N 96700006.

SET-UP

1. Install the MICR Reader as described in Section 2, Installation.
2. On the PC, go to the DOS prompt or any other PC application that accepts and displays keyboard input (e.g., the MICRbase Setup Program). Observe data from the MICR Reader throughout the procedure described below.
3. Start the Troubleshooting procedure at Step 00.

PROCEDURE

00	**Check LED Indicator**

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 12.
◊ red or orange, continue to step 18.

01	Check the Power to the MICR Reader

Possible causes for this problem are:

- PC power - power PC on.
- 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 Reader - make sure the AC adapter is securely connected to the power jack on the MICR Reader.
- 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 Reader:

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

05	Verify send data after error

If Send Data After Error is set to NO and the MICR Reader detects a read error (the LED indicator turns red), no data will be sent. Use an Insta-Change check or the PC program (Section 4, Commands) to set Send Data After Error to YES

Determine if the above condition is true:

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

06	Read error

Possible causes for this problem are:

- Interference - the MICR Reader may be too close to a monitor, AC adapter or magnetic device. Move the MICR Reader away from the source of interference.
- 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 Reader "grabs" it. Also, make sure that the front end is not tilted up while the check is being read.
- Foreign debris – power the MICR Reader off 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 the MICR Reader on 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 cause for this problem is:
- 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 Reader.

Determine if the above condition is true:
◊ If yes, rectify and continue to step 02.
◊ If no, continue to step 08.

08	Character Rate

Possible cause for this problem is:
- The character rate at which the MICR Reader is sending may be too fast. Check to see if the last few characters in the data message are consistently missing.

Determine if the above condition is true.
◊ If yes, continue to step 18.
◊ 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 Reader is not the desired format. Verify/change the format. Use the FC command (refer to Section 4).
- Incorrect message format - the current message format in the MICR Reader is not the desired format. Verify/change the message format. Use the SWB command (refer to Section 4).

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 the MICR Reader off 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 the MICR Reader on again.
- 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**

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 Reader 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, if EMF detect is set to YES (see HW Command, Section 4), the MICR Reader 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. Set EMF to NO, or move the MICR Reader 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

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 the MICR Reader off 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 Reader 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 Reader (See Figure 3-1).

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 Reader 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 Reader 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 Reader to MagTek**

The MICR Reader has a problem that needs further analysis, testing, and possibly repair. Please contact the MagTek Help Desk at (888) 624-8350, and make arrangements to send the unit back to MagTek. Include a detailed description of the problem.

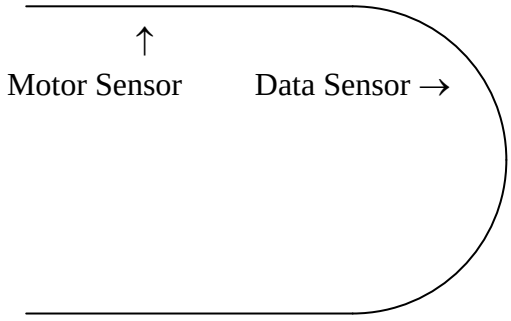


Figure C-1. Sensor Location

APPENDIX D. PINPAD INTERFACE TO MICR READER

The PINPad device connects to the 6-pin RJ socket on the MICR Reader. This socket is referred to as the PINPad port. This port uses the RS232 protocol at TTL voltage levels with settings fixed at 300 baud rate, 7 data bits and odd parity.

In a typical connection, the PINpad connects to the MICR Reader, and the MICR Reader connects to the Host. The MICR Reader serves as a communication bridge between Host and PINPad. It should be noted that the MICR Reader simply directs the data flow from the Host to the PINPad (and vice versa) and is never responsible for the PINPad operation. At all times, the Host is responsible for the control of the PINPad operation.

Connect the PINPad to the RJ socket on the MICR Reader. The RJ socket is shown in Figure D-1, and the pin descriptions are listed in Table D-1.

To communicate with the PINPad through the MICR Reader, all PINPad commands from the Host must be in the following format:

<STX>[PINPad Command]<ETX>

When the MICR Reader receives commands for the PINPad, the MICR Reader just passes the commands along to the PINPad.

In the same manner all PINPad data to the Host will be in the following format:

<STX>[PINPad Data]<ETX>

For further information on PINPad operation refer to the following documents:

- Encrypting PINPad Specification, Part Number 99815042
- Nonencrypting PINPad Technical Description, Part Number 99833004

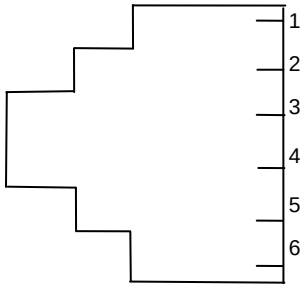


Figure D-1. PINPad RJ Socket

Table D-1. PINPad RJ Socket

PIN NUMBER	SIGNAL (MICR Reader as Reference)	DESCRIPTION
1	GND	Ground
4	TXD	Transmitted Data, RS-232 Signal. Transmits data from the MICR Reader to the PINPad.
5	RXD	Received Data, RS-232 Signal. Receives data from the PINPad to the MICR Reader.
6	+5V	+5 volt DC Power

APPENDIX E. INTERFACE CABLE PIN LISTS

Table E-1 shows the listings for the Large DIN 5-pin for P/N 22517502 and P/N 22517503. Figure E-1 shows the pin number locations for the Large DIN 5-pin connectors.

Table E-1. Large DIN 5-Pin Interface Cable

Signal Name	Mini DIN 9-pin	Large DIN 5-pin female	Large DIN 5-pin male
PC Data	1		2
PC Strobe	2		1
PC +5V	4		5
PC Ground	9		4
Keyboard Data	6	2	
Keyboard Strobe	7	1	
Keyboard +5V	5	5	
Keyboard Ground	8	4	
	3	Shield	Shield

Figure E-1. Large DIN 5-Pin Connectors

Table E-2 shows the listings for the Mini DIN 6-pin for P/N 22517501.
Figure E-2 shows the pin number locations for the Mini DIN 6-pin connectors.

Table E-2. Mini DIN 6-Pin Interface Cable

Signal Name	Mini DIN 9-pin	Mini DIN 6-pin female	Mini DIN 6-pin male
PC Data	1		1
PC Strobe	2		5
PC +5V	4		4
PC Ground	9		3
Keyboard Data	6	1	
Keyboard Strobe	7	5	
Keyboard +5V	5	4	
Keyboard Ground	8	3	
	3	Shield	Shield

Figure E-2. Mini DIN 6-Pin Connectors

APPENDIX F. 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

INDEX

A

Accessories 1
ASCII Codes 55

C

Card Data Message 16
Card Swipe Procedure 5
Check Layouts 40
Check Reading Procedure 5
CMC-7 Character Set 16, 39
Command Format 9
Commands 7–19
  Disable Keyboard, DK 9
  Enable Keyboard, EK 9
  Format Change, FC 18
  Hardware, HW 17
  Reset, RS 19
  Save, SA 19
  Switch A, SWA 9
  Switch B, SWB 12
  Switch C, SWC 15
  Version, VR 18
Control Characters and MICR Data 13

D

Disable Keyboard Command, DK 9
Disable/Enable Tracks 17

E

E13-B Character Set 39
EMF Detect 18
Enable Keyboard Command, EK 9
Enter Key Options 13

F

Flexible Format 38
Format Change Command, FC 18
Format List 21–38

H

Hardware Command, HW 17

I

ID Card Decoding 18
Insta-Change Checks 7
Installation 3–4
Interface Cable Pin Lists 53

K

Keypad Type 11

L

LED Indicator 6

M

MagTek Device Drivers for Windows 8
MICR Fields 41
MICRbase Setup Program for Windows 7

O

Operation 5–6

P

Parameters, HW 17
Parameters, SWA 10
Parameters, SWB 12
Parameters, SWC 16
Pinpad Interface to MICR Reader 51–52

R

Reset Command, RS 19

S

Save Command, SA 19
Send Data After Error 10
Send Status After Data 15
Specifications 2
SWA Parameters 10
SWB parameters 12
SWC Parameters 16
Switch A Command, SWA 9
Switch B Command, SWB 12
Switch C Command, SWC 15

T

Troubleshooting 43–49

U

User Applications without Drivers 8

V

Version Command, VR 18

W

Wedge Type 10