Intermec
Technologies Corporation

M90 Series Portable Data Entry Unit

REFERENCE MANUAL

P/N: M90902
Revision A
September 1999

▶ NOTICE

The information contained herein is proprietary and is provided solely for the purpose of allowing customers to operate and service Intermec manufactured equipment and is not to be released, reproduced, or used for any other purpose without written permission of Intermec.

Disclaimer of Warranties. The sample source code included in this document is presented for reference only. The code does not necessarily represent complete, tested programs. The code is provided **"AS IS WITH ALL FAULTS." ALL WARRANTIES ARE EXPRESSLY DISCLAIMED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.**

We welcome your comments concerning this publication. Although every effort has been made to keep it free of errors, some may occur. When reporting a specific problem, please describe it briefly and include the book title and part number, as well as the paragraph or figure number and the page number.

Produced by:
Intermec International Inc.
Sovereign House
Vastern Road
Reading RG1 8BT
UK

This publication printed on recycled paper.

Acknowledgments

CONTENTS

SECTION 3

Operation Modes

APPENDIX C

Bar Code Symbologies

APPENDIX D

FIGURES

TABLES

Section 1

Introduction

This document is a guide for the M90 System Software Project. The M90 is a programmable and flexible data collection device that can serve in a variety of data acquisition markets. The target market segments range from VAR, OEM, and end-user applications. The M90 can act as a portable data entry device and as a programmable dedicated computer receiving application programs from a host system.

The M90 Computer contains 256 KB of ROM-based software or firmware. This provides a rich set of DOS functions and device drivers for application development, including bar code decoding, display, keypad, communications, real-time clock, calendar, and alarm.

System Block Diagram

An NEC V25 Microprocessor controls the M90. Figure 1-1 shows the system block diagram. The system has a 256 KB EEPROM program. An M90 Kernel program fits in 128 KB of the EEPROM space. The remaining 128 KB is for special application software, allowing the M90 to run stand-alone applications.

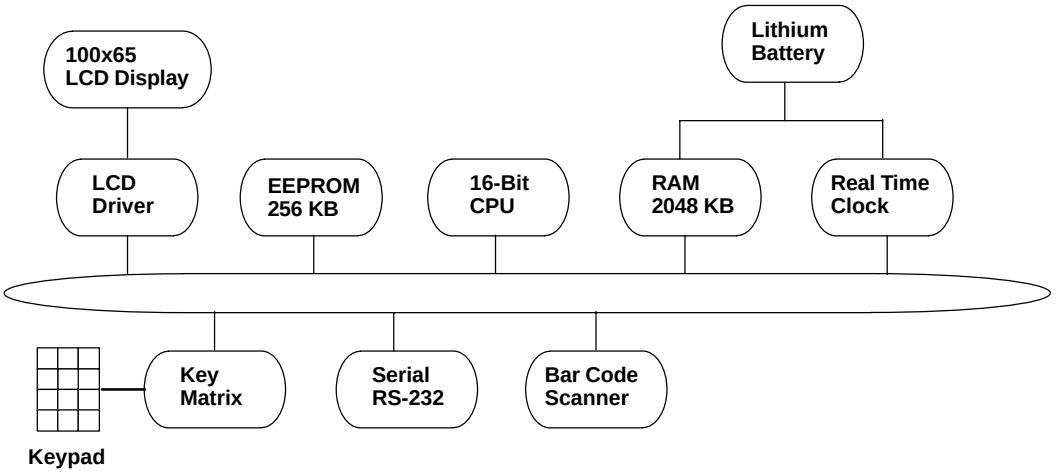


Figure 1-1
System Block Diagram

The maximum memory available in the M90 is 2048 KB RAM. The minimum RAM requirement is 128 KB. The main battery powers all of the RAM memory present. Otherwise, there is a backup lithium battery in case the main battery is absent or depleted. The RAM stores data and programs.

The M90 has a 12-character and 4-line, or 16-character and 8-line LCD backlight display (depending on setting). An RS-232 Serial Port communicates with a host system.

Programming

Use any of the following to program M90:

- Microsoft C 4.0 or later
- Borland C 3.0 or later
- TURBO PASCAL 6.0 or later
- IBM PC macro assembler version 1.0 or later

The downloaded application program determines the intelligence level of the M90. Once started, the M90 operates as a stand-alone unit or can be combined with a host or personal computer.

Sufficient energy stays in the main or backup battery to retain the program and data even when the power is down. Transaction data transfers to the computer or stays in the M90 RAM area.

Application Program Interface

The M90 Kernel includes three basic modules: device driver, file manager, and DOS manager.

The programmer can design the application programs by calling those functions just like in a PC DOS environment. The programs are in the .EXE format.

For end-users who use the C/PASCAL compiler and assembler, the ROM-based hardware of the M90 provides emulated MS-DOS function calls. The calling and parameter passing conventions are identical with MS-DOS. Table 1-1 describes several subsystems and related I/O interface functions, and the DOS and file manager functions.

▶ **NOTE:** *For the detailed calling process, see Section 4.*

Keypad Subsystem

The keypad subsystem scans the key matrix, converts the scan code to its associated key value, and stores the value in the input buffer of the keyboard. Table 1-1 indicates the scan code and its default key ASCII value and display of each key on the keypad.

▶ **NOTE:** *Table 1-1 lists the keys from left to right, from the upper left corner of the M90 keypad. Default ASCII values are from the English keypad. Table 1-2, on page 1-10, contains ASCII values that differ on Swedish, Finnish, Danish, Spanish, French, German, and Italian keypad subsystems.*

Table 1-1
English Keypad Subsystem

Shift	Key	Hex Value	ASCII Character
	[ENT SCAN]	0D	<CR>
[S1]	**[ENT SCAN]**	0D	<CR>
[S2]	**[ENT SCAN]**	0D	<CR>
[S3]	**[ENT SCAN]**	0D	<CR>
[FN]	**[ENT SCAN]**	0D	<CR>
	[S1]	00	
[S1]	**[S1]**	00	
[S2]	**[S1]**	00	
[S3]	**[S1]**	00	
[FN]	**[S1]**	00	
	[S2]	00	
[S1]	**[S2]**	00	
[S2]	**[S2]**	00	
[S3]	**[S2]**	00	
[FN]	**[S2]**	00	

Table 1-1 (Continued)
English Keypad Subsystem

Shift	Key	Hex Value	ASCII Character
	[S3]	00	
[S1]	**[S3]**	00	
[S2]	**[S3]**	00	
[S3]	**[S3]**	00	
[FN]	**[S3]**	00	
	[FN]	00	
[S1]	**[FN]**	00	
[S2]	**[FN]**	00	
[S3]	**[FN]**	00	
[FN]	**[FN]**	00	
	[7]	37	“7”
[S1]	**[7]**	41	“A”
[S2]	**[7]**	42	“B”
[S3]	**[7]**	43	“C”
[FN]	**[7]**	00	
	[8]	38	“8”
[S1]	**[8]**	44	“D”
[S2]	**[8]**	45	“E”
[S3]	**[8]**	46	“F”
[FN]	**[8]**	00	
	[9]	39	“9”
[S1]	**[9]**	47	“G”
[S2]	**[9]**	48	“H”
[S3]	**[9]**	49	“I”
[FN]	**[9]**	00	

Table 1-1 (Continued)
English Keypad Subsystem

Shift	Key	Hex Value	ASCII Character
	[4]	34	"4"
[S1]	**[4]**	4A	"J"
[S2]	**[4]**	4B	"K"
[S3]	**[4]**	4C	"L"
[FN]	**[4]**	00	
	[5]	35	"5"
[S1]	**[5]**	4D	"M"
[S2]	**[5]**	4E	"N"
[S3]	**[5]**	4F	"O"
[FN]	**[5]**	00	
	[6]	36	"6"
[S1]	**[6]**	50	"P"
[S2]	**[6]**	51	"Q"
[S3]	**[6]**	52	"R"
[FN]	**[6]**	00	
	[1]	31	"1"
[S1]	**[1]**	53	"S"
[S2]	**[1]**	54	"T"
[S3]	**[1]**	55	"U"
[FN]	**[1]**	00	
	[2]	32	"2"
[S1]	**[2]**	56	"V"
[S2]	**[2]**	57	"W"
[S3]	**[2]**	58	"X"
[FN]	**[2]**	00	

Table 1-1 (Continued)
English Keypad Subsystem

Shift	Key	Hex Value	ASCII Character
	[3]	33	"3"
[S1]	**[3]**	59	"Y"
[S2]	**[3]**	5A	"Z"
[S3]	**[3]**	20	Space
[FN]	**[3]**	00	
	[DEL]	08	BS
[S1]	**[DEL]**	08	BS ***** (See footer page 1-9)
[S2]	**[DEL]**	08	BS ***** (See footer page 1-9)
[S3]	**[DEL]**	08	BS ***** (See footer page 1-9)
[FN]	**[DEL]**	08	BS
	[0]	30	"0"
[S1]	**[0]**	2A	"*"
[S2]	**[0]**	2B	"+"
[S3]	**[0]**	2C	"–"
[FN]	**[0]**	00	
	[.]	2E	"."
[S1]	**[.]**	2F	"/"
[S2]	**[.]**	3A	":"
[S3]	**[.]**	3D	"="
[FN]	**[.]**	00	
	◀	11	
[S1]	◀	11	
[S2]	◀	11	
[S3]	◀	11	
[FN]	◀		* (See footer page 1-9)

Table 1-1 (Continued)
English Keypad Subsystem

Shift	Key	Hex Value	ASCII Character
	▼	12	
[S1]	▼	12	
[S2]	▼	12	
[S3]	▼	12	
[FN]	▼		** (See footer page 1-9)
	▲	13	
[S1]	▲	13	
[S2]	▲	13	
[S3]	▲	12	
[FN]	▲		*** (See footer page 1-9)
	▶	10	
[S1]	▶	10	
[S2]	▶	10	
[S3]	▶	10	
[FN]	▶	84	“ä” **** (See footer page 1-9)
	[F1]	86	“å”
[S1]	**[F1]**	86	“å”
[S2]	**[F1]**	86	“å”
[S3]	**[F1]**	86	“å”
[FN]	**[F1]**	8A	“è”
	[F2]	87	“è”
[S1]	**[F2]**	87	“è”
[S2]	**[F2]**	87	“è”
[S3]	**[F2]**	87	“è”
[FN]	**[F2]**	8B	“ç”

Table 1-1 (Continued)
English Keypad Subsystem

Shift	Key	Hex Value	ASCII Character
	[F3]	88	"ê"
[S1]	**[F3]**	88	"ê"
[S2]	**[F3]**	88	"ê"
[S3]	**[F3]**	88	"ê"
[FN]	**[F3]**	8C	"î"
	[F4]	89	"ë"
[S1]	**[F4]**	89	"ë"
[S2]	**[F4]**	89	"ë"
[S3]	**[F4]**	89	"ë"
[FN]	**[F4]**	8D	"ì"

* *The **[FN]**, ◀ key combination turns the backlight on and off.*
** *The **[FN]**, ▼ key combination adjusts the contrast.*
*** *The **[FN]**, ▲ key combination adjusts the speaker volume.*
**** *The **[FN]**, ▶ key combination activates the "User Menu" when the M90 is in "Ready Mode." When the M90 is in the "User Menu," the key combination takes you back to "Ready Mode." When **[FN]** and ▶ are pressed simultaneously pressed, M90 performs a WARM START.*
***** *Table 1-2, on the next page, has other values for non-English subsystems.*

Table 1-2
Non-English Keypad Subsystems

Shift	Key	Language	Hex Value	ASCII Character
[S1]	[DEL]	Swedish	8F	"Å"
		Finnish	8F	"Å"
		Danish	92	"Æ"
		Spanish	AD	"Ì"
		French	F8	"◦"
		German	8E	"Ä"
		Italian	F9	
[S2]	[DEL]	Swedish	8E	"Ä"
		Finnish	8E	"Ä"
		Danish	9D	"Ø"
		Spanish	A5	"Ñ"
		French	87	"Ç"
		German	99	"Ö"
		Italian	5C	"\"
[S3]	[DEL]	Swedish	99	"Ö"
		Finnish	99	"Ö"
		Danish	8F	"Å"
		Spanish	A8	"¿"
		French	26	"&"
		German	9A	"Ü"
		Italian	82	"é"

Shift Keys

S1 0000: 1B7E
S2 0000: 1B7F
S3 0000: 1B80

Value: 0 Not in shift mode.
255 In shift mode.

Key in Keyboard Buffer

0000:1BAF

Value: 0 Buffer is not empty.
1 Buffer is empty.

File System

Directory address: 0000:3A64

Table 1-3
File System

Filename Address	Contents
0	Name
16	Start address
18	Low word of size
20	High word of size
26	Used or not used, zero is not used
32	Next name

Display Subsystem

The M90 Display Subsystem supports a character-oriented 8-line by 16-character or 4-line by 12-character display with backlight control. The origin (0,0) is always at the upper left-hand corner.

Real-Time Clock Subsystem

The real-time clock subsystem keeps system time and date values for the M90. The subsystem also provides the alarm or wake-up functions.

Serial Port Subsystem

The M90 has an RS-232 Serial Port for data communication. Its communication system consists of point-to-point connection type for general processing. The operator may press the **[FN]** key, then the ▶ key to enter "User Mode," then select COM to invoke the built-in Kermit server for point-to-point communication. The RI signal of the port can turn on the M90.

Bar Code Input Port

There is one bar code input port plus an integrated bar code scanner on the M90. The bar code port is for bar code scanning devices such as bar code wand, wand-emulation CCD, or laser-diode scanners. Depending on its connection, the scanner may turn the unit on by scanning without pressing the ON or OFF keys.

M90 Port Usage

The M90 defines four ports. Table 1-4 shows the various port usages. See the Glossary for usage abbreviations.

Table 1-4
M90 Port Usage

Port	Bit	Usage	I/O
0	0	RTS	OUT
	1	CTS	OUT
	2	DTR	OUT
	3	SHUT213	OUT
	4	KEY_OUT0	OUT
	5	KEY_OUT1	OUT
	6	KEY_OUT2	OUT
	7	KEY_OUT3	OUT
1	0	NMI/POWER FAIL	IN
	1	WAND	IN
	2	KEYBOARD	IN
	3	RTC	IN
	4	BACKLIGHT	OUT
	5	SPEAKER	OUT
	6	SOS	IN
	7	OUT0	OUT
2	0	GOOD READ	OUT
	1	CS	OUT
	2	No connection	N/A
	3	SCAN_EN	OUT
	4	LCD_VO	OUT
	5	CN6–2 (Laser control)	OUT
	6	No connection	N/A
	7	AUTOOFF	OUT

Table 1-4 (Continued)
M90 Port Usage

Port	Bit	Usage	I/O
20h	0	BAT2_LOW	IN
	1	PF0	IN
	2	No connection	N/A
	3	No connection	N/A
	4	BAT1_LOW	IN
	5	Switch detect (Laser)	IN
	6	COLDSTART	IN
	7	DSR/RI	IN

Port T (keyboard) Serial channel 1 (COM port)

M90 Memory Map

Table 1-5
M90 Memory Map

Address	Chip
0 – 7FFFFh	Working RAM memory
80000h – BFFFFh	Bank switched RAM memory (RAM DISK)
C0000h – FFFFFh	EEPROM (disk and OS)

M90 Input and Output

All V25 Processor ports and registers in the M90 are mapped to memory. The read-and-write-to ports act as memory that can be moved to different memory pages. Read the Internal Data in Base register (IDB) to find the actual port address. The V25 Processor does not have the same interrupt Vectors as an 8086 Processor.

Table 1-6
Vector Summary

Vector Number	Assigned Use
0	Divide error
1	Break flag
2	NMI
3	BRK3 instruction
4	BRKV instruction
5	CHKIND instruction
6	General purpose
7	FPO instructions
8	General purpose
9	General purpose
A	General purpose
B	General purpose
C	INTSER0 (serial channel 0)
D	INTSR0
E	INTST0
F	General purpose
10	INTSER1 (serial channel 1)
11	INTSR1
12	INTST1
13	I/O trap

Table 1-6 (Continued)
Vector Summary

Vector Number	Assigned Use
14	INTD0 (DMA channel 0)
15	INTD1
16	General purpose
17	General purpose
18	INTP0 (peripheral 0)
19	INTP1
1A	INTP2
1B	General purpose
1C	INTTU0 (timer 0)
1D	INTTU1
1E	INTTU2
1F	INTB (time base counter)
20–FF	General purpose

Section 2

System Organization

The software organization consists of the kernel and application modules.

The kernel modules provide basic system services for the applications. The application program of an M90 may be either resident or transient. A resident application is an application stored in the M90 EEPROM. A transient application is an application that resides on the RAM. The host uses host file downloading commands to download the transient application.

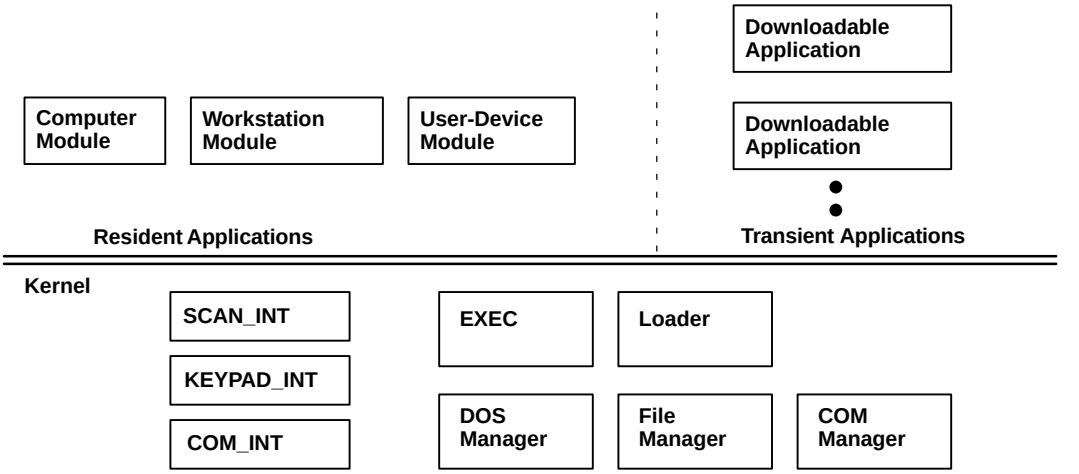


Figure 2-1
Software Modules

An EXECutable program fundamentally controls the M90 Program. When the system powers up, EXEC performs the Power-On Test (POT). During a *cold start* process, the system restores all default configurations. A *warm start* process resets the system without erasing any RAM program or data files. Device configurations preserve these program or data values as they were before the warm boot.

▶ **NOTE:** ***Warm start*** *is referred to as* ***restart*** *in this document.*

EXEC initializes all M90 Peripheral Devices respectively, according to their configuration parameters. M90 Peripherals include a bar code scanner and a serial communication port (RS-232).

EXEC also creates and initializes all dynamic data structures during the start-up process, such as a keypad queue and a COM buffer. After the initialization procedure, EXEC checks for a request for a "Supervisor Menu" (▶ key pressed while the device turns on).

If there is an AUTOEXEC.EXE application in the M90, EXEC executes that application. Otherwise, the M90 enters the "User Menu" state, where the operator can execute M90 functions. A description of the EXEC function appears later.

Kernel

The M90 Kernel provides basic system services. The kernel modules include various interrupt service routines and DOS, File, and COM managers.

- **DOS Manager:** Emulates most MS-DOS function calls to control M90 Peripherals and Files.
- **File Manager:** Implements a DOS-like file subsystem to support file operations (read, write, open, close).
- **COM Manager:** Controls the communication link between the M90 and the host.

- **EXEC:** A job scheduler that manages top-level operation flow of the M90 System. The EXEC program is analogous to the COMMAND.COM program in MS-DOS operating systems:

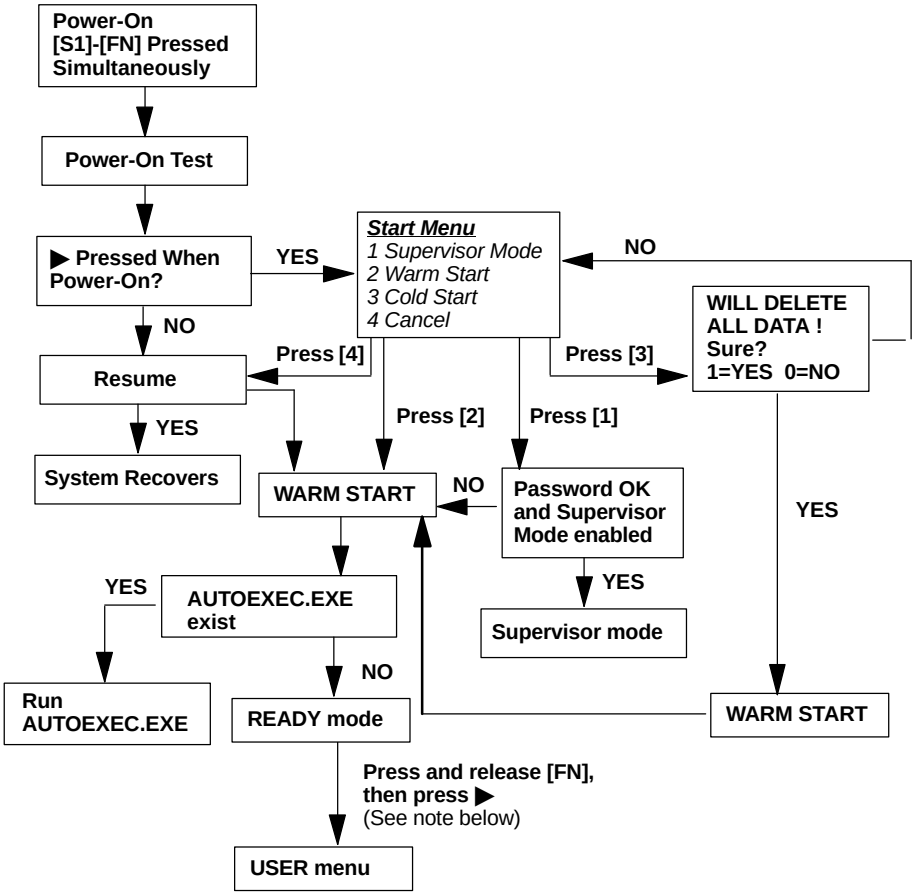


Figure 2-2
EXEC Flow Chart

▶ **NOTE:** *Press and release **[FN]**, then press ▶ to go from one mode to the other.*

When the M90 is on, the system performs a comprehensive POT. If the system passes the POT, EXEC determines whether the M90 is in "Supervisor Mode" or "Ready Mode." The "Supervisor Mode" appears if the ▶ key is pressed while the M90 is turned on. M90 EXEC displays the supervisor menu while in "Supervisor Mode." A supervisor password protects the M90 from unauthorized entry into supervisor state.

In normal operation M90 EXEC initializes all the input and output devices, according to the configuration. If there is an AUTOEXEC.EXE application in the unit, EXEC starts executing the application. Otherwise, EXEC displays a prompt and waits for the operator to enter further commands.

- **Loader:** The function is identical to the MS-DOS loader. The loader is an MS-DOS clone, except for the simplified memory allocation and deallocation scheme. The loader's operations are:
  a. Read *.EXE header.
  b. Find enough memory for .EXE.
  c. Read binary into RAM.
  d. Reallocate based on RAM allocation.
  e. Set up all registers.
  f. Jump to the entry point.
- **LCD Manager:** All displayable characters can appear on the LCD screen. In addition, Table 2-1 shows the special control sequences supported by the M90 Display Handler:

Table 2-1
LCD Manager Sequences

Key	Hex	Operation
BEL	0x07	Activate beeper for 500 ms (0.5 seconds)
LF	0x0A	New line
CR	0x0D	Cursor return

▶ **NOTE:** *Refer to Section 4 for detailed LCD control commands.*

Workstation Mode Application

Although applications are classified into resident and transient categories, the M90 EXEC always loads the executable file from the RAM/ROM into the RAM executive area. It then begins execution.

Workstation mode is the default resident M90 Operation. In workstation mode, an M90 configures as a portable programmable device. The system receives transient application programs from the host using a download channel. The system also receives the resident application, from ROM.59 programs or data files, that can reside on an M90. Up to eight resident applications can reside on an M90. The operator can then select one of the applications using the keypad host command sequence.

The transient application operates as a file and purges when the M90 performs a *cold start*. While File Manager does not support the resident application, it is always in ROM.

Data files or recorded information collected from workstation applications are uploaded to the host when the host program decides to retrieve the collected data. Some applications may also require prepared data files, such as database information. These data files are loaded into the M90 in the same way that the program is downloaded.

When the M90 is in "User Mode," the user menu is displayed. The operator may select one of the following functions:

- RUN Runs executable program
- TER Terminal mode operation
- COM Communicates in Kermit protocol
- DIR Displays M90 RAM disk directory
- ERA Erases file
- TYP Types file
- CPY Copies file
- SET Sets M90 Parameters

Use host communication commands to perform all keypad controls.

▶ **NOTE:** *If the M90 Communication Parameters are not set correctly, the host system cannot send any control commands.*

Section 3

Operation Modes

Ready Mode

Turn on the M90 if it is not already on. Press and release **[FN]**, then press ▶ to go from "User Menu" to "Ready Mode." The following screen appears:

```
M90 Ver X.XX
MEM XXXX KB
>
```

The first line shows the model code and version number (such as 1.60). The second line shows the total installed RAM size (such as 2048 KB). The third line is the prompt ">," meaning that the M90 is in *Ready Mode*.

There are eight system commands:

- **1. RUN** Run program
- **2. TER** Terminal mode
- **3. COM** Communications
- **4. DIR** Directory
- **5. ERA** Erase file
- **6. TYP** Type or list file
- **7. CPY** Shows system parameter status
- **8. SET** M90 set up

Input the command name at the "Ready Mode" prompt, or access the "User Mode," starting on the next page.

User Mode

Press and release the **[FN]** key, then press the ▶ key to invoke the "User Mode." The "User Mode" menu appears:

```
1. RUN      2. TER
3. COM      4. DIR
5. ERA      6. TYP
7. CPY      8. SET
```

Select the corresponding number, 1–8, or use ▲ or ▼ to move the cursor. Press **[ENT SCAN]** to select a function. Press and release **[FN]**, then press ▶ to return to "Ready Mode."

Run Program

Select option **"1. RUN"** from the "User Mode" menu for the "Run Program" screen:

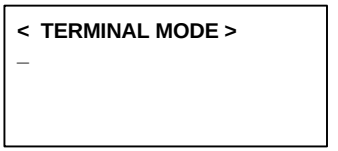
```
< RUN PROGRAM >

PROGRAM.EXE
```

The "Run Program" function gives you the possibility to run any program on the M90. Use the ◀ or ▶ keys to scroll to the program you want run, then press **[ENT SCAN]** to start the program shown on the screen. The ◀ key will only scroll until it reaches the first file on the disk.

Press and release **[FN]**, then press ▶ to invoke the "User Mode."

Terminal Mode

Select option **"2. TER"** from the "User Mode" menu for the "Terminal Mode" screen:



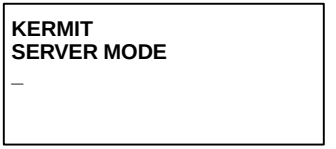
In this function, the M90 serves as a dumb computer that transmits data to or receives data from a host. Bar code label data, either scanned or keyed in, go out through the RS-232 port. Data received from the serial port appear on screen. Communication parameters must be compatible between Host and M90 to send data properly.

▶ **NOTE:** *The M90 has a combined "SCANNER" and "ENTER" key, thus the "ENTER" function is not active in "Terminal Mode". Pressing the* ***[ENT SCAN]*** *key will activate the scanner.*

Press and release **[FN]**, then press ▶ to return to "Ready Mode." Press and release **[FN]**, then press ▶ again to go to the "User Mode."

Communications

Select option **"3. COM"** from the "User Mode" menu for the "Kermit Server Mode" screen to enter the Kermit server mode:



Press and release **[FN]**, then press ▶ to go to the "User Mode."

Table 3-1 indicates the available Kermit commands in the computer:

Table 3-1
Kermit Commands

Command	Description
SEND filename	Sends host or computer file to M90.
GET filename	Sends M90 File to host or computer disk.
REMOTE DIR	Displays directory of files stored in M90 RAM disk.
REMOTE DEL filename	Deletes program or data file in M90 RAM disk.

Directory

Select option **"4. DIR"** from the "User Mode" menu to list the file directory of the M90, for example:

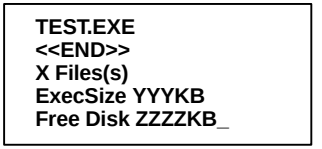
```
ASET.EXE
AUTOEXEC.EXE
BATCHK.EXE
BCRTST2.EXE
```

The file directory in the RAM disk appears with the following information:

- ROM disk filenames
- RAM disk filenames
- Execution area size
- Free RAM disk space

If there are more files than can be shown on the display, press **[ENT SCAN]** to see more files. If you are at the end of the list, pressing **[ENT SCAN]** takes you back to the "User Mode."

You can see that you are at the end of the list when you see the following at the bottom of the screen:

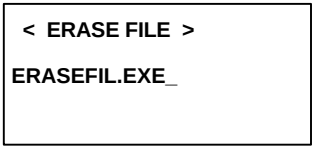


- "X" is the number of files on the disk.
- "YYY" is the size of the execution area in KB.
- "ZZZZ" is the amount of free disk space in KB.

Press and release **[FN]**, then ▶ to return to the "Ready Mode."

Erase File

Select option **"5. ERA"** from the "User Mode" menu for the "Erase File" screen:



This function erases files present on the M90. Use the ◀ and ▶ keys to scroll to the file you want erased, then press **[ENT SCAN]** to erase it. The ◀ key only scrolls until it reaches the first file on the disk.

When you press **[ENT SCAN]**, you get the following message on the bottom of the screen to confirm whether you do want to erase the selected file:

Are you sure ?
1=YES/0=NO

Press **[1]** if you want to continue with the erase or press **[0]** to abort.

Press and release **[FN]**, then press ▶ to go to the "User Mode."

Type File

Select option **"6. TYP"** from the "User Mode" menu to display M90 file contents. Unintelligible characters may appear when attempting to view a program or binary file:

```
< TYPE FILE >
FILE.TXT_
```

Use the ◀ and ▶ keys to scroll to the file you want displayed, then press **[ENT SCAN]** to display it. The ◀ key only scrolls until it reaches the first file on the disk

The file displays 128 (8 lines x 16) characters per screen. Press any key to show the next page.

Press and release **[FN]**, then press ▶ to go to the "User Mode."

Copy Setup

Select option **"7. CPY"** from the "User Mode" menu for the "Copy Setup" screen. Use this screen to copy files:

```
< COPY SETUP >
SOURCE FILE :
_
```

Enter the source filename, then press **[ENT SCAN]** for the destination prompt:

```
< COPY SETUP >
DESTINATION :
_
```

Enter the destination filename, then press **[ENT SCAN]** for the M90 to execute the copy, then return to the "User Mode" screen.

The **CPY** command allows you to copy data from a source and transfer the information to a different location or destination. The source and destination can be a file or device, such as "COM," "serial port," or "CON." "CON" specifies the LCD for the destination and the keyboard for the source.

Table 3-2
CPY Source and Destination Files

Source	Destination	Function
File1	File2	Copy File1 to File2
File1	COM	Output content of File1 to serial port
File1	CON	Output content of File1 to LCD
COM	File2	Input data from serial port and store in File2
CON	File2	Input data from keyboard and store in File2, press and release the **[FN]** key, then press the ▶ key from the keyboard to end the data input

Setup Menu

Select option **"8. SET"** from the "User Mode" menu to display the "System Setup" menu. Press and release the **[FN]** key, then press ▶ to return to the "User Mode."

```
  <SYSTEM SETUP>
1.DATE & TIME
2.SCANNER
3.DISPLAY
4.EXIT
```

The "System Setup" menu contains three functions to set system parameters. Press **[1]** or **[3]** to select a category, press either **[4]** or **[FN]**, then release, then press ▶ to return to the "User Mode."

Press **[4]** or press and release **[FN]**, then ▶ to go to the "User Mode" without any changes.

Date & Time

Select option **"1. DATE & TIME"** for the "Set Date&Time" screen:

```
<SET DATE&TIME>
mm–dd–yyyy
08–09–1999
```

Press **[ENT SCAN]** if the date is correct or change the date in the month/day/year format. Be sure to enter a zero before the single digit entries. The time field appears:

```
<SET DATE&TIME>
mm–dd–yyyy
08–09–1999
hh–mm–ss
18:31:36
```

Press **[ENT SCAN]** if the time is correct to return to the "System Setup" menu, or change the time in the hour/minute/second format. Be sure to enter a zero before the single digit entries. The M90 uses a 24-hour clock. After you enter the last digit for the second, the M90 returns to the "System Setup" menu.

Use the ◀ and ▶ keys to move between fields.

Press and release **[FN]**, then press ▶ to return to the "User Mode" without changes to the date or time.

Scanner

Select option **"2.SCANNER"** from "System Setup" to enable or disable the internal bar code laser scanner:

```
<SYSTEM SETUP>

SCANNER
ENABLE
```

The M90 supports a bar code pen connected to the 10-pin modular connector at the bottom of the M90. Some models also come with a built-in laser scanner.

Press **[ENT SCAN]** to enable the internal scanner. The M90 returns a verification screen:

```
<SYSTEM SETUP>

VERIFICATION
DISABLE
```

This screen lets you set the M90 laser scanner to decode the bar code twice before it accepts the data. Use the ▶ key to enable or disable the two scans. When ready, press **[ENT SCAN]** to return to the "System Setup" menu.

Press and release **[FN]**, then press ▶ to return to the "System Setup" menu without changes.

Display

Select option **"3.DISPLAY"** from the "System Setup" menu to dictate how the cursor is to appear on the display:

```
<SYSTEM SETUP>

CURSOR SHAPE
UNDERLINE
```

With this screen, use the ▶ key to select either BLOCK, such as **"▮"** or UNDERLINE, such as **"_"**. When ready, press **[ENT SCAN]** for the "Display Format" screen:

```
<SYSTEM SETUP>

DISPLAY FORMAT
4*12 (LARGE)
```

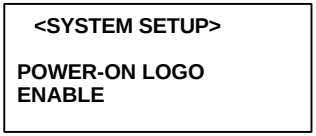
Use this screen to set a display format for the M90:

- 4*12 (LARGE) 4 lines by 12 characters wide
- 8*16 (SMALL) 8 lines by 16 characters wide
- 4*20 (M80) 4 lines by 20 characters wide virtual screen, last four characters truncated
- 6*20 (M80) 6 lines by 20 characters wide virtual screen, last four characters truncated

The “4*20” and “6*20” formats are M80-compatible, such that they will only display four or six lines with the first 16 characters visible and the last four characters (positions 17–20) truncated.

A program written for the M80 is able to run without problems as the system ignores the character found outside the physical screen.

Use the ▶ key to scroll between formats. When ready, press **[ENT SCAN]** for the “Power-On Logo” screen:



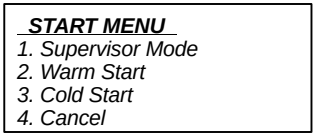
Use this screen to dictate whether to have the initial power-on logo appear. Press the ▶ to toggle between “ENABLE” and “DISABLE.” When ready, press **[ENT SCAN]** to return to the “System Setup” screen.

Press and release **[FN]**, then press ▶ to return to the “System Setup” menu without changes.

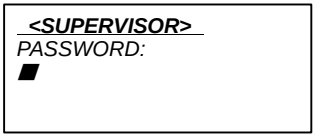
Supervisor Mode

The M90 has a "Supervisor Mode" to set up system configurations and verify computer hardware. Do the following to enter the "Supervisor Mode:"

1. Turn off the M90.
2. Press and hold the ▶ key, then simultaneously press **[S1]** and **[FN]**. The M90 powers on with the following screen:



3. Press **[1]** to access the "Supervisor Mode:"



▶ **NOTE:** *A supervisor password prevents unauthorized users from changing configuration parameters. The M90 system forces the user to enter "Ready Mode" after five unsuccessful attempts to enter a correct password.* ***"M90" is the default password.***

The “Supervisor Mode” menu appears:

```
1.DEV      2.TERM
3.MEM      4.ALRM
5.PWR      6.PSWD
7.SYS      8.DIAG
```

These are the parameter descriptions:

- **1.DEV** Device Configuration
- **2.TERM** Terminal Configuration
- **3.MEM** Memory Configuration
- **4.ALRM** Alarm Date and Time
- **5.PWR** Auto Off and Resume Configuration
- **6.PSWD** Supervisor Mode Password
- **7.SYS** System Initialization and Update
- **8.DIAG** System Diagnostics

Device Configuration

▶ **NOTE:** *Defaults are given with each screen.*

Select option **“1.DEV”** from the “Supervisor Mode” menu to configure the M90 device parameters:

```
 <DEVICE CONFIG>
1.KEYPAD
2.SERIAL
3.BARCODE _
```

Keypad

Select option **"1.KEYPAD"** from the "Device Config" menu to select one of seven languages for the keypad:

```
< KEYPAD SETUP >

LANGUAGE
English _
```

Press ◀ or ▶ to toggle between the following languages, then press the **[ENT SCAN]** key:

- Italian
- English *(default)*
- Sweden/Finland
- Danish
- Spanish
- French
- German

Serial

▶ **NOTE:** *Defaults are given in each screen, and are listed in* ***bold.***

Select option **"2.SERIAL"** from the "Device Config" menu to send or receive data or programs using RS-232, and to set these and other M90 communication parameters. Press and release **[FN]**, then press ▶ to return to the "Device Config" menu without changes.

Press **[ENT SCAN]** to scroll through the following parameter menus:

- **Baud Rate:**

```
< COM SETUP>

BAUD
 19200   _
```

  Press ▶ to toggle between these available baud rates, then press **[ENT SCAN]**:

  - 150, 300, 600, 1200, 2400, 4800, 9600**, 19200**, 38400, 57600

- **Length**

```
< COM SETUP>

LENGTH
8 BITS   _
```

  Press ▶ to toggle between these bit lengths, then press **[ENT SCAN]**:

  - 7, **8**

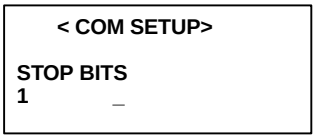
- **Parity**

```
< COM SETUP>

PARITY
NONE     _
```

  Press ▶ to toggle between these three parities, then press **[ENT SCAN]**:

  - **None**, Odd, Even

- **Stop Bits**

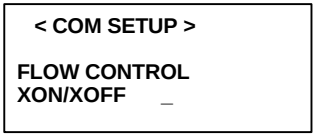


Press ▶ to toggle between these stop bits, then press **[ENT SCAN]**:

- **1,** 2

▶ **NOTE:** ***Flow Control*** *controls the RS-232 port sending or receiving data in a character-by-character mode.* ***Protocol*** *controls file transfers.*

- **Flow Control**



Press ▶ to toggle between these flow control options, then press **[ENT SCAN]**:

- XON/XOFF, CTS/RTS, **NONE**

Bar Code

▶ **NOTE:** *Defaults are given in each screen, and are listed in **bold.***

Select option **"3.BARCODE"** from the "Device Config" menu to enter the appropriate bar code symbology supported by the M90. All bar code symbology can be separately enabled or disabled. See Appendix C for bar code symbologies.

- **Code 39:**

```
<BARCODE SETUP>

CODE 39
ON        _
```

Press ◀ or ▶ to toggle this bar code symbology "ON" or "OFF," then press **[ENT SCAN]**.

- **Code 39 Full ASCII:**

```
<BARCODE SETUP>
CODE 39
FULL ASCII
OFF       _
```

Press ◀ or ▶ to toggle this bar code symbology "ON" or "OFF," then press **[ENT SCAN]**.

- **Code 39 Start/Stop:**

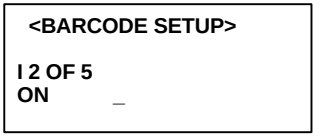
```
<BARCODE SETUP>
CODE 39
START/STOP
NO SEND     _
```

Press ◀ or ▶ to toggle either "NO SEND" or "SEND" to dictate whether to send the Code 39 start and stop character as part of the decoded data, then press **[ENT SCAN]**.

- **Code 39 Check Digit:**

Press ◀ or ▶ to select one of three verification types: "OFF," "ON & NO SEND," or "ON & SEND," then press **[ENT SCAN]**.

- **I 2 of 5:**

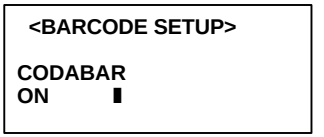


Press ◀ or ▶ to toggle this bar code symbology "ON" or "OFF," then press **[ENT SCAN]**.

- **I 2 of 5 Check Digit:**

Press ◀ or ▶ to select one of three verification types: "OFF," "ON & NO SEND," or "ON & SEND," then press **[ENT SCAN]**.

- **Codabar:**



Press ◀ or ▶ to toggle this bar code symbology "ON" or "OFF," then press **[ENT SCAN]**.

- **Codabar Start/Stop:**

```
 <BARCODE SETUP>
CODABAR
START/STOP
SEND          _
```

Press ◀ or ▶ to toggle either "NO SEND" or "SEND" to dictate whether to send the Codabar start and stop character as part of the decoded data, then press **[ENT SCAN]**.

- **Codabar Check Digit:**

```
 <BARCODE SETUP>
CODABAR
CHECK DIGIT
OFF       _
```

Press ◀ or ▶ to select one of three verification types: "OFF," "ON & NO SEND," or "ON & SEND," then press **[ENT SCAN]**.

- **UPC-A:**

```
 <BARCODE SETUP>

UPC–A
ON        _
```

Press ◀ or ▶ to toggle this bar code symbology "ON" or "OFF," then press **[ENT SCAN]**.

- **UPC-A Leading Digit:**

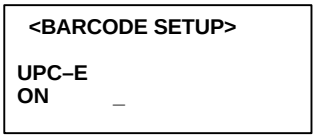
```
 <BARCODE SETUP>
UPC–A
LEADING DIGIT
SEND          _
```

Press ◀ or ▶ to toggle either "NO SEND" or "SEND," then press **[ENT SCAN]**.

- **UPC-A Check Digit:**

Press ◀ or ▶ to toggle either "NO SEND" or "SEND," then press **[ENT SCAN]**.

- **UPC-E:**

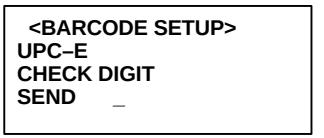


Press ◀ or ▶ to toggle this bar code symbology "ON" or "OFF," then press **[ENT SCAN]**.

- **UPC-E Leading Digit:**

Press ◀ or ▶ to toggle either "NO SEND" or "SEND," then press **[ENT SCAN]**.

- **UPC-E Check Digit:**

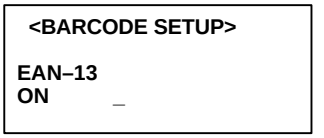


Press ◀ or ▶ to toggle either "NO SEND" or "SEND," then press **[ENT SCAN]**.

- **UPC-E Zero Expansion:**

Press ◀ or ▶ to toggle either "ON" or "OFF," then press **[ENT SCAN]**.

- **EAN-13:**

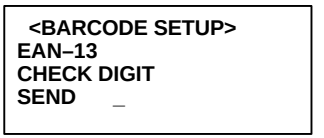


Press ◀ or ▶ to toggle either "ON" or "OFF," then press **[ENT SCAN]**.

- **EAN-13 Leading Digit:**

Press ◀ or ▶ to toggle either "NO SEND" or "SEND," then press **[ENT SCAN]**.

- **EAN-13 Check Digit:**



Press ◀ or ▶ to toggle either "NO SEND" or "SEND," then press **[ENT SCAN]**.

- **EAN-8:**

Press ◀ or ▶ to toggle either "ON" or "OFF," then press **[ENT SCAN]**.

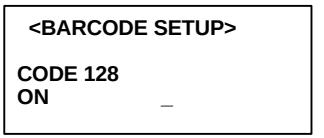
- **EAN-8 Check Digit:**

Press ◀ or ▶ to toggle either "NO SEND" or "SEND," then press **[ENT SCAN]**.

- **EAN/UPC Add-On:**

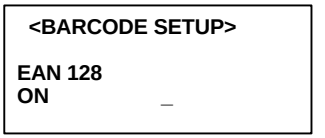
Press ◀ or ▶ to toggle among "DISABLE," "OPTIONAL," or "REQUIRED," then press **[ENT SCAN]**.

- **Code 128:**



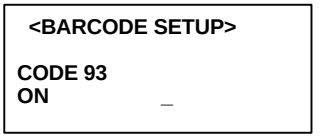
Press ◀ or ▶ to toggle either "ON" or "OFF," then press **[ENT SCAN]**.

- **EAN 128:**



Press ◀ or ▶ to toggle either "ON" or "OFF," then press **[ENT SCAN]**.

- **Code 93:**



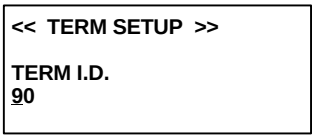
Press ◀ or ▶ to toggle either "ON" or "OFF," then press **[ENT SCAN]**.

Terminal Configuration

Select option **"2.TERM"** from the "Supervisor Mode" menu for the "Terminal Setup" menu. Parameters are set up in this menu like they are set up in the "Device Config" menu. The M90 supports a terminal emulator that acts as a dumb ASCII computer.

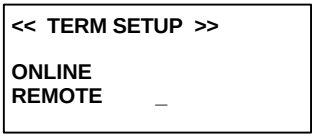
Terminal ID

ASCII string IDs identify each M90 Computer. IDs take up to eight characters. The initial ID is "M90." Alphanumeric characters ("A"–"Z", "a"–"z", "0"–"9") are valid. Press **[ENT SCAN**] to continue.



Online

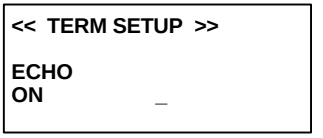
Press ▶ to toggle between "REMOTE" or "LOCAL," then press **[ENT SCAN]**:



- **REMOTE:** M90 immediately transmits scanned bar code data or key data to the host or PC using the RS-232 port.
- **LOCAL:** M90 does not transmit any data.

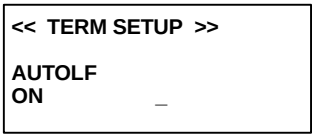
Echo

Press ▶ to toggle between "ON" or "OFF," then press **[ENT SCAN]**. Set to "ON" to view any collected data.



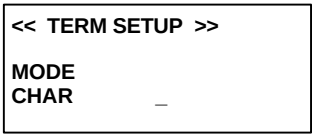
AutoLF

Press ▶ to toggle between "ON" or "OFF," then press **[ENT SCAN]**. Set to "ON" to have the M90 append a line-feed (10 hex) character to the input data block.



Mode

Press ▶ to toggle between "BLOCK" or "CHAR," then press **[ENT SCAN]**.

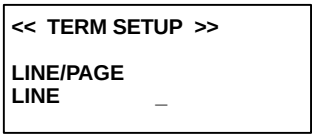


Line/Page

If "BLOCK" is selected under **Mode**, then **LINE/PAGE** is enabled. The **LINE/PAGE** entry designates the termination character set as:

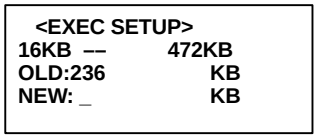
- "LINE" Line termination: CR (0D hex)
- "PAGE" Page termination: CTRL-Z (1A hex)
- "BOTH" Terminates both the line and page: CR and CTRL-Z (0D and 1A hex)

Press ▶ to toggle among "LINE," "PAGE," or "BOTH," then press **[ENT SCAN]**.



Memory Configuration

Select option **"3.MEM"** from the "Supervisor Mode" menu to access the "Execution Setup" menu:



Enter the new Execution Program Area size, then press **[ENT SCAN]**. The M90 system RAM memory is in three sections:

- **System Variable Area:** Has about 29 KB reserved for system parameters.
- **RAM Disk:** Stores programs and data files, like a physical PC disk.
- **Execution Program Area:** Loads applications and data, like a PC's main memory.

Allocate the available RAM between the Execution Program Area and RAM disk. The RAM disk size decrements when the Execution Program area size is incremented, and vice versa.

Alarm

Select option **"4.ALRM"** from the "Supervisor Mode" menu to access the "Alarm Set" menu:

```
< ALARM SET >

ALARM FUNCTION
Disable                    _
```

Press ◀ or ▶ to toggle among "Disable," "Everyday/Time," or "Date/Time," then press **[ENT SCAN]**.

If "Everyday/Time" is selected, this screen appears:

```
< SET ALARM >
hh:mm:ss
Date: Everyday
Time: 08:00:00
```

Enter the time (24-hour format) that the alarm is to be set for every day. Press **[ENT SCAN]** to save or press and release **[FN]**, then ▶ to exit without saving the entry.

▶ **NOTE:** *If an incorrect entry is made, the system puts the cursor back to the beginning of the field.*

If "Date/Time" is selected, this screen appears:

```
< SET ALARM >
mm–dd–yyyy
Date:      08–10–1999
Time:       08:00:00
```

Enter the date and time (24-hour format) when the alarm is to be set. Press **[ENT SCAN]** to save or press and release **[FN]**, then ▶ to exit without saving the entry.

▶ **NOTE:** *If an incorrect entry is made, the system puts the cursor back to the beginning of the field.*

Power

Select option **"5.PWR"** from the "Supervisor Mode" menu to set the power management system. The power management system is in the M90 Hardware and Software to increase battery service time:

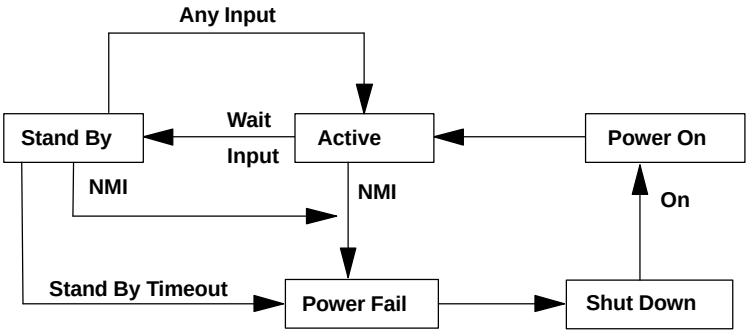


Figure 3-1
M90 Power Transition Flow

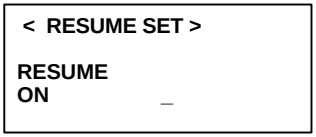
- **Active:** All system hardware and software continue to work if the M90 has no activity. The system may go to Stand-by mode after a timeout period.
- **Stand By:** If the keyboard, scanner, or RS-232 input are activated, the system returns to Active mode. Otherwise, the M90 waits for the specified setting timeout period, then shuts down the system.
- **Shut Down:** The M90 turns the main power off and disables all hardware devices. RAM is left on in retention mode.

Press **[1]** to enter the "RESUME" set up menu or **[2]** to go to the "AUTO-OFF" set up menu. Press and release the **[FN]** key, then the ▶ key to return to the "Supervisor Mode" menu.

```
<SYSTEM ON/OFF>
1. RESUME
2. AUTO-OFF_
```

Resume

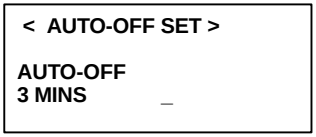
The "Resume" function lets the M90 resume processing from where it was when it was shut down. If off, the M90 will perform a warm start each time it is powered on again.



Press the ◀ or ▶ keys to toggle between "ON" or "OFF," then press **[ENT SCAN]** to return to the "System On/Off" screen. Press and release the **[FN]** key, then the ▶ key to return to the "System On/Off" screen without changes.

Auto-Off

Set the timeout period from 1 to 9 MINS (minutes), or DISABLE. The unit remains on until **[S1]** and **[FN]** are pressed (off).



Press the ◀ or ▶ keys to toggle among the values, then press **[ENT SCAN]** to return to the "System On/Off" screen. Press and release the **[FN]** key, then the ▶ key to return to the "System On/Off" screen without changes.

Password Change

Select option **"6.PSWD"** from the "Supervisor Mode" menu to change the password:

```
OLD PASSWORD:
M90
NEW PASSWORD:
_
```

A supervisor password prevents unauthorized users from entering the Supervisor Mode and changing configuration parameters. This function assigns passwords. Using up to ten alphanumeric characters ("A"–"Z", "a"–"z", "0"–"9"), key the password. Press **[ENT SCAN]** to save or press and release **[FN]**, then ▶ to exit without saving the entry.

System Initialization

▶ **NOTE:** *Back up data or programs to the host or PC before doing this function.*

▶ **NOTE:** *The M90 **must** be connected to a charger before entering this option.*

Select option **"7.SYS"** from the "Supervisor Mode" menu for either the "Cold Start" screen or the "Programming" screen:

```
   < SYSTEM INITIAL  >
1.COLD START
2.PROGRAMMING _
```

Cold Start

Select option **1. COLD START** to activate the *cold start*. A warning screen appears:

```
   <    SYS INIT    >

Will Flush RAM &
RAM Disk  !

Continue  ?_
 1=YES/0=NO
```

Press **[0]** to return to the "Supervisor Mode" menu, or press **[1]** to continue with the *cold start*.

Programming

Select option **2. PROGRAMMING** to access the following screen:

```
<  PROGRAMMING   >
1. ADD PROGRAM
2. DELETE ALL
3. UPDATE BIOS _
```

Add Program

Press **[1]** to add a program from the RAM disk to the FLASH disk.

```
<  PROGRAMMING   >

127 KB FLASH Free
FILE1.EXE_
```

Press the ◀ or ▶ keys to toggle among the available programs on the RAM disk, then press **[ENT SCAN]** to add the program to the FLASH disk. While the file is added to the FLASH disk, "Programming. . ." appears on the display.

The following prompt appears when the programming is ready:

```
< PROGRAMMING >
Delete program
in RAM Disk ?_
1=YES/0=NO
```

Press **[1]** to erase the program from the RAM disk, press **[0]** otherwise. The last screen appears with the change in flash size:

```
< PROGRAMMING >
Done !
116KB FLASH Free
Press any key..._
```

"116 KB" is an example if the file added was 11 KB. Press any key to return to the "Programming" menu.

Delete All

Press **[2]** to remove all programs on the FLASH disk. The following verification screen appears:

```
Delete all PROGs
in FLASH memory!
Continue ?_
1=YES/0=NO
```

Press **[1]** to delete all programs stored on the FLASH disk, press **[0]** to return to the "Programming" menu without deleting the programs.

If you pressed **[1]**, the following verification screen appears. Press **[1]** to confirm.

```
< PROGRAMMING >

Are you sure?
1=YES/0=NO
```

Two additional screens appear, the first will show "Erasing," the second displays the flash size after the programs have been erased:

```
< PROGRAMMING >
Done !
127KB FLASH Free
Press any key..
```

Press any key to return to the "Programming" menu.

Update BIOS

▶ **NOTE:** *Before you can update the BIOS, you must download a BIOS file onto the RAM disk. If there is no BIOS file, the update will fail and the M90 will perform a cold start.*

Press **[3]** to update the BIOS. The following verification screen appears:

```
Erase all data
in RAM & FLASH !
Continue ?
1=YES/0=NO
```

Press **[1]** to continue, or press **[0]** to return to the "System Initialization" menu.

If you pressed **[1]** to erase all data, the following verification screen appears:

```
< PROGRAMMING >

Are you sure?
1=YES/0=NO
```

Press **[1]** to verify and the M90 updates the BIOS. "Wait" will appear briefly.

If a BIOS file is not found on the RAM disk, the following message appears:

```
< PROGRAMMING >
Please download
BIOS file . . . .
```

If you see this message, turn off the M90, then cold start it or wait for the "Start Menu" to cold start the unit. Set up the communications parameters and download the BIOS to the RAM disk. When done, return to the "Programming" menu.

If BIOS is present on the RAM disk, you will see the following. Press **[1]** to change the BIOS. The M90 displays a "Wait" message, then an "Erase" message.

```
< PROGRAMMING >
Change BIOS to
VX.XX ?
1=YES/0=NO
```

After the "Erase" message, a status screen appears with the address count. "XXXXXX" counts from "0" to "1C0000:"

```
< PROGRAMMING >
Updating BIOS
ADDR=XXXXXX
```

After the count is complete, the final screen appears. Press any key to restart the M90.

```
< PROGRAMMING >
BIOS Update OK!
Press Any Key to
Restart. . .
```

System Diagnostic

Select option **"8.DIAG"** from the "Supervisor Mode" menu to access the M90 diagnostic program to verify the M90. The diagnostic routines run these tests: RAM, keypad, RS-232, LCD, real-time clock, scanner, and RAM backup.

▶ **NOTE:** *Execute this diagnostic program when there is a service process, such as maintenance, repair, or upgrade to verify that system.*

```
 0.All      1.RAM
2.KEY      3.232
4.LCD      5.RTC
6.SCANNER
7.RAM BACKUP
8.EXIT

SELECT(0—8)?_
```

▶ **NOTE:** *The diagnostic program **may** destroy data. Back up the M90 data.*

Run All Seven Tests

Select option **"0. All"** from the "Diagnostic" menu to run all of the following seven tests. *(Requires RS-232 loopback.)*

RAM Test

Select option **"1.RAM"** from the "Diagnostic" menu to flush the M90 RAM. The first screen appears with a warning:

```
    ! ! WARNING ! !
Will Flush RAM &
RAM Disk  !

Continue  ?_
 1=YES/0=NO
```

Press **[0]** to return to the "Diagnostic" menu, or press **[1]** to continue with the memory flush.

- *Fixed pattern stuck at fault check (00, FF, 55, AA)*
  Writes a fixed pattern of data to the RAM chip, then verifies that the data had stored properly.
- *Address Test:*
  Writes odd, even address data into corresponding memory location, then verifies it.

Keypad Test

Select option **"2.KEY"** from the "Diagnostic" menu for the "Keypad Test" screen:

```
KEYPAD TEST

Wait key . . .■
```

Press any key and the ASCII symbol should appear. Press ▶ to exit, and the following prompt appears:

```
KEYPAD TEST

Exit_? ■
1=YES/0=NO
```

Press **[1]** to return to the "Diagnostic" menu, or press **[0]** to continue with the keypad test.

RS-232 Loopback Test

Select option **"3.232"** from the "Diagnostic" menu for the "RS-232 Test" testing status screen:

```
RS232 TEST
Baud= 9600   PASS
Baud=19200   PASS
Baud=38400   PASS
Baud=57600   PASS

Press any key to
continue . . . _
```

A signal is sent from the TXD pin through a loopback connector to the RXD pin. The received signal is verified to match the transmitted signal. A full ASCII table code (0–255) test pattern should be in each test cycle. Communication parameters should equal default values, except the baud rate.

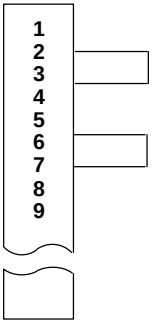


Figure 3-2
Loopback Connector Wiring

The loopback plug is a 9-pin DSUB that can be made to connect either to the M90-to-PC communication cable or at the back of the M90 office dock.

To connect it to the M90 PC cable, it should be made of a male DSUB. For connection to the M90 office dock, it should be made of a female DSUB.

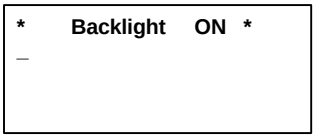
If the loopback test fails, or the test connector is not plugged in, the following screen appears briefly after the testing status screen:

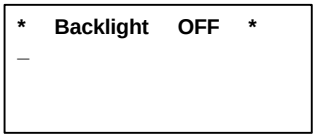
```
RS232 TEST
Baud=  9600   FAIL
Baud=19200   FAIL
Baud=38400   FAIL
Baud=57600   FAIL

Press any key to
continue . . . _
```

Press any key to return to the “Diagnostic” menu.

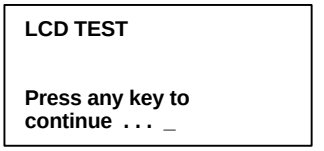
LCD Screen Test

Select option **"4.LCD"** from the "Diagnostic" menu for the "LCD Test" screen to see graphic patterns on the M90. The patterns start two sets of dark and light screens, followed by two sets of the following:



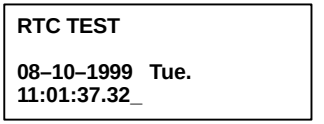


The backlight is more noticeable if the M90 is tested in a dimly lit area. After two sets of "Backlight ON" and "Backlight OFF," the final screen appears. Press any key to return to the "Diagnostic" menu.



Real-Time Clock Test

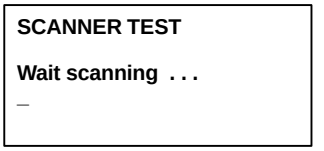
Select the **"5.RTC"** option from the "Diagnostic menu" for the "RTC Test" screen:



The date and time given should be current. If not, exit the "Supervisor Mode" and access the "User Mode." Go into the "Set" option and select option 1 to reset the date or time. Press any key to return to the "Diagnostic" menu.

Scanner Test

Select option **"6.SCANNER"** from the "Diagnostic" menu for the "Scanner Test" screen to test the M90 scanner on bar code labels. A successful scan will give the data, type, and scan information:

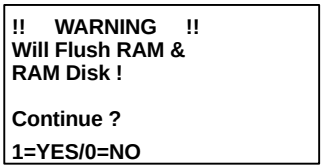


When doing a test, aim the M90 at a bar code, then press **[ENT SCAN]** to start the scan. A red laser should appear on the bar code to read, then if successful, the M90 emits a beep and lists the bar code information on the display.

Press the **[ENT SCAN]** key to do another bar code, or press any key to exit.

RAM Backup Test

Select option **"7.RAM BACKUP"** from the "Diagnostic" menu for the RAM back up test screen. Press **[0]** to return to the "Diagnostic" menu, or press **[1]** to continue with the memory flush.



If you pressed **[1]** to continue, the following instructions appear on the display:

```
RAM Backup TEST
Wait 30 seconds
After power off,
Then turn it on
Again

Press any key to
Turn it off . . . _
```

To do the test, remove the battery pack from the M90 to turn the unit off, wait 30 seconds, reinsert the battery pack, then turn on the M90. A "Please Wait" message appears on the display, followed by this pass/fail screen:

```
RAM Backup TEST

PASS !


Press any key to
Continue . . . _
```

Return to Supervisor Mode

Press option **"8.EXIT"** from the "Diagnostic" menu to return to the "Ready Mode." The M90 will then perform a warm start. If the M90 memory was tested, the M90 will perform a cold start.

Section 4

DOS System Call

DOS Call (INT 21H)

This section describes the supported DOS calls of the M90 System Kernel. All DOS calls are compatible with MS-DOS version 2.0 invoked by DOS INT 21H with functions defined in register AH.

00 Terminate Program
Entry Parameter: AH=0
Return Value: None

01 Read Stdin (wait if no key) and Write to Stdout Excluding Shift Keys
Entry Parameter: AH=1
Return Value: AL = 8-bit data ASCII code

02 Write Stdout
Entry Parameter: AH=2
DL = 8-bit data ASCII
Return Value: None

03 Read Stdaux (wait if no input) Excluding ESC Command
Entry Parameter: AH=3
Return Value: 8-bit data ASCII code

04 Write Stdaux
Entry Parameter: AH=4
DL= 8-bit data ASCII code
Return Value: None

06 Read/Write Stdin

Entry Parameter: AH=6
DL = 0x0FF Read
Other values Write
Return Value: If DL = 0x0FF
AL = 8-bit data ASCII code
Clear ZERO flag If char ready
Set ZERO flag If not ready
If DL = other values
None

07 Read Stdin (wait if no key) Excluding Shift Keys

Entry Parameter: AH=07
Return Value: AL=8-bit data ASCII code

08 Read Stdin (wait if no key) Excluding Shift Keys

Entry Parameter: AH=08
Return Value: AL=8-bit data ASCII code

09 Write Character String to Stdout

Entry Parameter: AH=09
DS:DX=Points to string buffer w/ "$" end
Return Value: None

0A Key Buffer Input

Entry Parameter: DS:DX = Points to an input buffer
Return Value: Buffer filled with CR as last character

0B Keyhit Check

Entry Parameter: AH=0B
Return Value: AL=0 If character is not ready
AL=FF If character is ready

1A LCD Backlight ON/OFF Control

Entry Parameter: AH=0x1A
BH=0
AL=0 Set LCD backlight OFF
AL=1 Set LCD backlight ON
Return Value: None

1A Buzzer ON/OFF Control

Entry Parameter: AH=0x1A
BH=1
AL=0 Set Buzzer OFF
AL=1 Set Buzzer ON
Return Value: None

1A Key Enable/Disable Setting

Entry Parameter: AH=0x1A
BH=2
AL=0 All keys except ON, OFF, four function keys
AL=1 Supervisor mode key
AL=2 Cold start key
AL=3 Warm start key
AL=4 User menu key
BL=0 Disable
BL=1 Enable
Return Value: None

1A Beeper Volume

Entry Parameter: AH=0x1A
BH=3
AL=0 Set low beeper volume
AL=1 Set middle beeper volume
AL=2 Set high beeper volume
Return Value: None

1A Battery Check

Entry Parameter: AH=0x1A
BH=8
Return Value: AL=0 Main battery normal
AL=1 Main battery low
AH=0 Backup battery normal
AH=1 Backup battery low

1A Keypad Language Setting

Entry Parameter: AH=0x1A
BH=9
AL=0 English
AL=1 Swedish or Finnish
AL=2 Danish
AL=3 Spanish
AL=4 French
AL=5 German
AL=6 Italian

Return Value: None

1A Set Good-read LED (green light)

Entry Parameter: AH=1AH
BH=0AH
AL=0 Set Good-read LED OFF
AL=1 Set Good-read LED ON
AL=2 Set Good-read LED controlled by system

▶ **NOTE:** *If the **Set Good-read LED** function is called by AL=0 or AL=1, the system will not control **Good-read LED** ON/OFF when a bar code label is decoded successfully.*

1A Buzzer Sound

Entry Parameter: AH=0x1A
BH=0x0B

Return Value: None

1A Enable/Disable Double Verification When Reading Bar Code Label

Entry Parameter: AH=1AH
BH=0CH
AL=0 Disable double verification
AL=1 Enable double verification

Return Value: None

1A Check Laser Scanner

Entry Parameter: AH=1AH
BH=0DH

Return Value: AL=0 Has no built-in laser scanner
AL=1 Has built-in laser scanner

1B Get Scanner Port Status

Entry Parameter: AH=1BH
BH=5

Return Value: AL=0 Scanner port is disabled
AL=1 Scanner port is enabled

1C Communication Parameter Setting

Entry Parameter: AH=0x1C
BH=1
AL= Bits 4–7: 0001xxxx Baud 150
0010xxxx Baud 300
0011xxxx Baud 600
0100xxxx Baud 1200
0101xxxx Baud 2400
0110xxxx Baud 4800
0111xxxx Baud 9600
1000xxxx Baud 19200
1001xxxx Baud 38400
1010xxxx Baud 57600
Bits 2–3: 00 None parity
01 Odd parity
11 Even parity
Bit 1: 0 1 stop bit
1 2 stop bits
Bit 0: 0 7 data bits
1 8 data bits

Return Value: None

1C Communication Control Flow Setting

Entry Parameter: AH=0x1C
BH=2
AL=0 Null
AL=1 XON/XOFF control flow
AL=2 CTS/RTS control flow
Return Value: None

1D Set Terminal ID

Entry Parameter: AH=0x1D
BH=0
DS:DX=String of ASCIIZ
Return Value: None

1D Set Online/Local

Entry Parameter: AH=0x1D
BH=1
AL=0 Online
AL=1 Local
Return Value: None

1D Set Echo ON/OFF

Entry Parameter: AH=0x1D
BH=2
AL=0 Set echo ON
AL=1 Set echo OFF
Return Value: None

1D AutoLF ON/OFF

Entry Parameter: AH=0x1D
BH=3
AL=0 Set auto-line-feed ON
AL=1 Set auto-line-feed OFF
Return Value: None

1D Mode Setup

Entry Parameter: AH=0x1D
BH=4
AL=0 Character
AL=1 Block
DX=0 Line
DX=1 Page
DX=2 Both

Return Value: None

1D Line Terminal Character

Entry Parameter: AH=0x1D
BH=5
AL = ASCII code

Return Value: None

1D Page Terminal Character

Entry Parameter: AH=0x1D
BH=6
AL = ASCII code

Return Value: None

1E Key Map Definition

Entry Parameter: AH=0x1E
BH=1
DS:DX = Keyboard map with 128 bytes in ASCII code corresponding to unshifted, shift 1, shift 2, shift 3 and function scan codes *(see Keypad Subsystem).*

Return Value: None

1E Get Key Map Definition

Entry Parameter: AH=0x1E
BH=0
DS:DX = Pointer to 160 bytes buffer

Return Value: Buffer with keyboard map of 160 bytes in ASCII code corresponding to unshifted, shift 1, shift 2, shift 3, and function scan codes *(see Keypad Subsystem).*

1E Set Key Map Definition

Entry Parameter: AH=0x1E
BH=1
DS:DX = Keyboard map with 160 bytes in ASCII code corresponding to unshifted, shift 1, shift 2, shift 3, and function scan codes *(see Keypad Subsystem)*.

Return Value: None

1F Enable/Disable Decoding of All Bar Code Symbologies

Entry Parameter: AH=0x1F
AL=0 Disable all codes
AL=1 Enable all codes
BH=1
BL=0

Return Value: None

1F Enable/Disable the Decoding of a Bar Code Symbology

Entry Parameter: AH=0x1F
BH=1
AL=0 Disable
AL=1 Enable
BL=1 Code 39
BL=2 I 2 of 5
BL=3 Codabar
BL=4 EAN/UPC
BL=5 Code 128
BL=6 EAN 128
BL=7 Code 93
CL *(See next note)*

Return Value: None

▶ **NOTE:** *If AL=1 and BL=1 (enable Code 39) when the **Enable/Disable the Decoding . . .** function is called, do CL=0 to disable Full ASCII decoding or CL=1 to enable Full ASCII decoding.*

*If AL=1 and BL=4 (enable EAN/UPC) when the **Enable/Disable the Decoding . . .** is called, do CL=0 to disable 2- or 5- digit add-on decoding; CL=1 to make 2- or 5-digit decoding optional; or CL=2 to require 2- or 5- digit decoding.*

1F Get the Decoding Status of Bar Code Symbology

Entry Parameter: AH=0x1F
BH=2
BL=1 Code 39
BL=2 I 2 of 5
BL=3 Codabar
BL=4 EAN/UPC
BL=5 Code 128
BL=6 EAN 128
BL=7 Code 93

Return Value: AL=0 Disable
AL=1 Enable
CL *(See next note)*

▶ **NOTE:** *If BL=1 (get Code 39 status) when the **Get the Decoding . . .** function is called, CL=1 is returned if Full ASCII decoding is enabled and CL=0 is returned if it is disabled.*

*If BL=4 (get EAN/UPC status) when the **Get the Decoding . . .** function is called, CL=0 (2- or 5-digit add-on) is off, CL=1 (2- or 5-digit add-on) is optional, and CL=2 (2 or 5 digits add-on) is required.*

1F Code 39 Settings

Entry Parameter: AH=1FH
BH=7
BL=1
AL= Bit 0: 0 Disable Code 39 decoding
1 Enable Code 39 decoding
Bit 1: 0 Disable Check Digit verification
1 Enable Check Digit verification
Bit 2: 0 No-send Check Digit
1 Send Check Digit
Bit 3: 0 No-send Start/Stop characters
1 Send Start/Stop characters
Bit 4: 0 Full ASCII OFF
1 Full ASCII ON

Return Value: None

1F Interleaved 2 of 5 Settings

Entry Parameter: AH=1FH
BH=7
BL=2
AL= Bit 0: 0 Disable I 2 of 5 decoding
1 Enable I 2 of 5 decoding
Bit 1: 0 Disable Check Digit verification
1 Enable Check Digit verification
Bit 2: 0 No-send Check Digit
1 Send Check Digit

Return Value: None

1F Codabar Settings

Entry Parameter: AH=1FH
BH=7
BL=3
AL= Bit 0: 0 Disable Codabar decoding
1 Enable Codabar decoding
Bit 1: 0 Disable Check Digit verification
1 Enable Check Digit verification
Bit 2: 0 No-send Check Digit
1 Send Check Digit
Bit 3: 0 No-send Start/Stop characters
1 Send Start/Stop characters

Return Value: None

1F Code 128 Setting

Entry Parameter: AH=1FH
BH=7
BL=5
AL=Bit 0: 0 Disable Code 128 decoding
1 Enable Code 128 decoding

Return Value: None

1F EAN 128 Setting

Entry Parameter: AH=1FH
BH=7
BL=6
AL=Bit 0: 0 Disable EAN 128 decoding
1 Enable EAN 128 decoding

Return Value: None

1F Code 93 Setting

Entry Parameter: AH=1FH
BH=7
BL=7
AL=Bit 0: 0 Disable Code 93 decoding
1 Enable Code 93 decoding

Return Value: None

1F UPC-A Settings

Entry Parameter: AH=1FH
BH=7
BL=11H
AL= Bit 0: 0 Disable UPC-A decoding
1 Enable UPC-A decoding
Bit 2: 0 No-send Check Digit
1 Send Check Digit
Bit 3: 0 No-send Leading Digit
1 Send Leading Digit

Return Value: None

1F UPC-E Settings

Entry parameter: AH=1FH
BH=7
BL=12H
AL= Bit 0: 0 Disable UPC-E decoding
1 Enable UPC-E decoding
Bit 2: 0 No-send Check Digit
1 Send Check Digit
Bit 3: 0 No-send Leading Digit
1 Send Leading Digit
Bit 4: 0 Disable Zero Expansion
1 Enable Zero Expansion

Return Value: None

1F EAN-13 Settings

Entry Parameter: AH=1FH
BH=7
BL=13H

AL=	Bit 0:	0	Disable EAN-13 decoding
		1	Enable EAN-13 decoding
	Bit 2:	0	No-send Check Digit
		1	Send Check Digit
	Bit 3:	0	No-send Leading Digit
		1	Send Leading Digit

Return Value: None

1F EAN-8 Settings

Entry Parameter: AH=1FH
BH=7
BL=14H

AL=	Bit 0:	0	Disable EAN-8 decoding
		1	Enable EAN-8 decoding
	Bit 2:	0	No-send Check Digit
		1	Send Check Digit

Return Value: None

1F Code 39 Settings

Entry parameter: AH=1FH
BH=8
BL=1

Return Value: AL=Bit 0: 0 Disable Code 39 decoding
1 Enable Code 39 decoding
Bit 1: 0 Disable Check Digit verification
1 Enable Check Digit verification
Bit 2: 0 No-send Check Digit
1 Send Check Digit
Bit 3: 0 No-send Start/Stop characters
1 Send Start/Stop characters
Bit 4: 0 Full ASCII OFF
1 Full ASCII ON

1F Interleaved 2 of 5 Settings

Entry Parameter: AH=1FH
BH=8
BL=2

Return Value: AL= Bit 0: 0 Disable I 2 of 5 decoding
1 Enable I 2 of 5 decoding
Bit 1: 0 Disable Check Digit verification
1 Enable Check Digit verification
Bit 2: 0 No-send Check Digit
1 Send Check Digit

1F Codabar Settings

Entry Parameter: AH=1FH
BH=8
BL=3

Return Value: AL= Bit 0: 0 Disable Codabar decoding
1 Enable Codabar decoding
Bit 1: 0 Disable Check Digit verification
1 Enable Check Digit verification
Bit 2: 0 No-send Check Digit
1 Send Check Digit
Bit 3: 0 No-send Start/Stop characters
1 Send Start/Stop characters

1F Code 128 Setting

Entry Parameter: AH=1FH
BH=8
BL=5

Return Value: AL=Bit 0: 0 Disable Code 128 decoding
1 Enable Code 128 decoding

1F EAN 128 Setting

Entry Parameter: AH=1FH
BH=8
BL=6

Return Value: AL=Bit 0: 0 Disable EAN 128 decoding
1 Enable EAN 128 decoding

1F Code 93 Setting

Entry Parameter: AH=1FH
BH=8
BL=7

Return Value: AL=Bit 0: 0 Disable Code 93 decoding
1 Enable Code 93 decoding

1F UPC-A Settings

Entry Parameter: AH=1FH
BH=8
BL=11H

Return Value:	AL=	Bit 0:	0	Disable UPC-A decoding
			1	Enable UPC-A decoding
		Bit 2:	0	No-send Check Digit
			1	Send Check Digit
		Bit 3:	0	No-send Leading Digit
			1	Send Leading Digit

1F UPC-E Settings

Entry parameter: AH=1FH
BH=8
BL=12H

Return Value:	AL=	Bit 0:	0	Disable UPC-E decoding
			1	Enable UPC-E decoding
		Bit 2:	0	No-send Check Digit
			1	Send Check Digit
		Bit 3:	0	No-send Leading Digit
			1	Send Leading Digit
		Bit 4:	0	Disable Zero Expansion
			1	Enable Zero Expansion

1F EAN-13 Settings

Entry Parameter: AH=1FH
BH=8
BL=13H

Return Value:	AL=	Bit 0:	0	Disable EAN-13 decoding
			1	Enable EAN-13 decoding
		Bit 2:	0	No-send Check Digit
			1	Send Check Digit
		Bit 3:	0	No-send Leading Digit
			1	Send Leading Digit

1F EAN-8 Settings

Entry Parameter: AH=1FH
BH=8
BL=14H

Return Value: AL= Bit 0: 0 Disable EAN-8 decoding
1 Enable EAN-8 decoding
Bit 2: 0 No-send Check Digit
1 Send Check Digit

25 Set Interrupt Vector

Entry Parameter: AH=0x25
AL=Interrupt number
DS:DX=Address of interrupt routine

Return Value: None

2A Get System Date

Entry Parameter: AH=0x2A

Return Value: CX=Year (1980 through 2079)
DH=Month (1–12)
DL=Day (1–31)
AL=Day of week (0–6)

2B Set System Date

Entry Parameter: AH=0x2B
CX=Year (1980 through 2079)
DH=Month (1–12)
DL=Day (1–31)

Return Value: AL=0 OK
AL=FFH Input parameter error

2C Get System Time

Entry Parameter: AH=0x2C

Return Value: CH=Hour (0–23)
CL=Minutes (0–59)
DH=Seconds (0–59)
DL=1/100th of a second (0–99)

2D Set System Time

Entry Parameter: AH=0x2D
CH=Hour (0–23)
CL=Minutes (0–59)
DH =Seconds (0–59)
Return Value: AL=0 OK
AL=FFH Input parameter error

2E Set Alarm Date

Entry Parameter: AH=0x2E
AL=0 Disable alarm
AL=1 Enable Every Day/Time type alarm
AL=2 Enable Specific Date/Time type alarm
If AL=2
CX=Year (1980 through 2079)
DH=Month (1–12)
Dl=Day (1–31)
Return Value: AL=0 OK
AL=FFH Setting error

2F Set Alarm Time

Entry Parameter: AH=0x2F
CH=Hour (0–23)
CL=Minutes (0–59)
DH=Seconds (0–59)
Return Value: AL=0 OK
AL=FFH Input parameter error

30 Get M90 DOS Version Number

Entry Parameter: AH=0x30
Return Value: AL=2
AH=0
CL=Major firmware version number
CH=Minor firmware version number

▶ **NOTE:** *"Major" indicates the prefix in the version number and "Minor" indicates the suffix in the version number (i.e. ".YY"). For example, "X.YY" has "X" for the major version number and "YY" for the minor version number.*

35 Get Interrupt Vector

Entry Parameter: AH=0x35
AL=Interrupt number
Return Value: ES:BX=Address of interrupt routine

36 Get Free Disk Cluster

Entry Parameter: AH=0x36
Return Value: AX=1 (Number of sectors per cluster)
BX=Number of available clusters
CX=1024 (Number of bytes per sector)
DX=Number of total clusters in RAM disk

37 Set EXEC Memory Size

Entry Parameter: AH=0x37
AL=0
DX=size in KB 1–488
Return Value: AL=0 OK
AL=1 Error

37 Get EXEC Memory Size

Entry Parameter: AH=0x37
AL=1
Return Value: AX=Current EXEC memory size in KB
CX=Maximum KB EXEC memory size

3C Create or Truncate a File

When a file is opened, the file manager searches the file table for a match. If the manager finds a match, the corresponding file handle is returned and the current pointer resets to the beginning of the file. The actual file is reset to zero. If the file does not exist in the file table, a file entry is allocated and memory is assigned.

Entry Parameter: AH=0x3C
DS:DX=Segment:offset of filename
Return Value: 1) AX=Handle
Clear carry flag if successful
2) AX=3
Set carry flag if failed

3D Open a File

Entry Parameter: AH=0x3D
AL=0 Open a file for read only
AL=1 Open a file for write only
AL=2 Open a file for read and write
DS:DX=Segment:offset of filename
Return Value: 1) AX=Handle
Clear carry flag if successful
2) AX=2
Set carry flag if failed

3E Close a File

Entry Parameter: AH=0x3E
BX=File handle
Return Value: If successful:
Carry flag is cleared
If not successful:
Carry flag is set

3F Read a File
Copy (CX) bytes from current address to DS:DX.
Advance the current address (CX) number of bytes.

Entry Parameter: AH=0x3F
BX=File handle
CX=Number of bytes to read
DS:DX=Segment: offset of buffer area
Return Value: 1) AX=Number of bytes read, 0 if EOF,
Clear carry flag if successful
2) AX=6
Set carry flag if failed

40 Write a File
Copy (CX) bytes from DS:DX to file (BX).
Update BX current address and ending address.

Entry Parameter: AH=0x40
BX=File handle
CX=Number of bytes to write
DS:DX=Segment: offset of buffer area
Return Value: 1) AX=Number of bytes written, 0 if full,
Clear carry flag if successful
2) AX=6
Set carry flag if failed

41 Delete a File
Entry Parameter: AH=0x41
DS:DX=Segment: offset of filename
Return Value: 1) Clear carry flag if successful
2) AX=2
Set carry flag if failed

42 Move File Pointer

Entry Parameter: AH=0x42
AL=0 Offset from beginning position
AL=1 Offset from current position
AL=2 Offset from end position
BX=File handle
CX=Most significant half of offset
DX=Least significant half of offset

Return Value: 1) AX= Least significant half of new current position
DX= Most significant half of new current position
Clear carry flag if successful
2) AX=6 Set carry flag if failed

42 Search Character Beginning at the Current File Position

Entry Parameter: AH=42H
AL= 3 Search forward (to end of file)
4 Search backward (to top of file)
BX=File handle
CX=n Search nth matched character
DL=Character

Return Value: 1) If character is found:
Carry flag=Clear
DX:AX=Pointer to current file position (at position of matched character)
2) If character is not found:
Carry flag=Set
CX=Total matched times
DX:AX=Pointer to current file position (not changed)

42 Search String in Formatted Data File Beginning at Current Position

Entry Parameter: AH=42H
AL= 5 Search forward (to end of file)
6 Search backward (to beginning of file)
BX=File handle
CH=n Total field number in data record
CL=m Search mth field
DS:DX=Pointer to parameter block

Structure of parameter block for variable-length record:

String length:	1 byte
String without "\0" terminator:	N bytes
0x00	1 byte
Field separator character:	1 byte

Structure of parameter block for fixed-length record:

String length:	1 byte
String without "\0" terminator:	N bytes
Field #1 length:	1 byte
Field #2 length:	1 byte
. .	
. .	
. .	
Field #n length:	1 byte

Return Value: 1) If string is found:
Carry flag=Clear
DX:AX=Pointer to current file position (at beginning of the matched string)
2) If string is not found:
Carry flag=Set
DX:AX=Pointer to current file position (not changed)

42 Insert/Delete Data Block to/from File at Current Position

Entry Parameter: AH=42H
AL= 7 Insert
8 Delete
BX=File handle
CX=Block length in bytes

Return Value: 1) If the function is successful:
Carry flag=Clear
DX:AX=Pointer to current file position (not changed)
2) If the function fails:
Carry flag=Set
DX:AX=Pointer to current file position (not changed)

▶ **NOTE:** *For insertion, the content of the inserted data block is undefined.*

43 Get File Attribute

Entry Parameter: AH=0x43
AL=0
DS:DX=Segment: offset of filename

Return Value: 1) CX=0
Clear carry flag if file found
2) AX=2
Set carry flag if file not found

44 Device-Driver Control IOCTL

Entry Parameter: AH=0x44
AL=0 Getting device information
AL=1 Setting device information
BX=Handle
DX=Device information

Return Value: 1) DX=Device information
Clear carry flag if successful
2) AX=6
Set carry flag if failed

44 Device-Driver Control IOCTL

Entry Parameter: AH=0x44
AL=2 Read to buffer
AL=3 Write buffer to device
BX=Handle
CX=Number of bytes to read or write
DS:DX=Segment: offset of buffer area

Return Value: 1) AX=Number of bytes transferred
Clear carry flag if successful
2) AX=6
Set carry flag if failed

44 Device-Driver Control IOCTL

Entry Parameter: AH=0x44
AL=6 Get input status
AL=7 Get output status
BX=Handle

Return Value: 1) AL=0 if not ready
AL=0xFF if ready
Clear carry flag if successful
2) AX=6
Set carry flag if failed

48 Allocate Specified Number of Paragraphs in Memory

Entry Parameter: AH=0x48
BX=Number of segments

Return Value: AX=Segment address of allocate blocks,
Error code, if carry flag
BX=Largest available block (on failure)

49 Free Allocated Memory

Entry Parameter: AH=0x49
ES=Segment of block to free

Return Value: AX=Error code if carry flag set

4A Modify Allocated Block

Entry Parameter: AH=0x4A
ES=Segment of the block modified
BX=New number of segments wanted
Return Value: AX=Error code, if carry flag is set
BX=Largest available block (on failure), if carry flag is set

4B Call Application Program

Entry Parameter: AH=0x4B
AL=0
DS:DX=String of ASCIIZ
Return Value: Carry flag = 0 (Success)
1 (Fail)

4B Run Application Program

Entry Parameter: AH=0x4B
AL=3
DS:DX=String of ASCIIZ
Return Value: Carry flag = 0 (Success)
1 (Fail)

50 Get Bar Code Data from Scanner Port

Entry Parameter: AH=0x50
DS:DX=Buffer pointer

Return Value: 1) AL= 0 Data in buffer
CH= 0 Scan from start to stop
1 Scan from stop to start
CL= 1 Code 39
2 I 2 of 5
3 Codabar
4 EAN/UPC
5 Code 128
6 EAN 128
7 Code 93
BL= 1 Code 39
2 Interleaved 2 of 5
3 Codabar
5 Code 128
6 EAN 128
7 Code 93
11H UPC-A
12H UPC-E
13H EAN-13
14H EAN-8
2) AL=1 No data input
CL=0 No failed decoding since last time function was called.
CL=1 Failed decoding since last time function was called.

51 Enable or Disable Scanner Port

Entry Parameter: AH=0x51
AL=0 Disable scanner port
AL=1 Enable scanner port

Return Value: None

56 Rename a File

Entry Parameter: AH=0x56
DS:DX=Pointer to a filename
ES:DI=Pointer to new filename
Return Value: Clear carry flag if successful
AH=2 if carry flag is set

5B Create New File

Entry Parameter: AH=5BH
DS:DX Pointer to file name string
Return Value: If successful:
Carry flag is cleared
AX=File Handle
If not successful:
Carry flag is set
AX= 04H Too many open files
50H File exists

▶ **NOTE:** *If the specified file already exists, the **Create New File** function fails.*

Table 4-1
Implemented DOS Functions (INT 21h)

Function	Description	Comment
00h	Terminate Program	
01h	Read Stdin	
02h	Write Stdout	
03h	Read Stdaux	
04h	Write Stdaux	
06h	Write Stdout	
07h	Read Stdin	
08h	Read Stdin	
09h	Write Character String	
0Bh	Keyboard Hit	
1Ah	Device Configuration	M90 special

Table 4-1 (Continued)
Implemented DOS Functions (INT 21h)

Function	Description	Comment
1Ch	Communication Configuration	M90 special
1Dh	Terminal Configuration	M90 special
1Eh	Keyboard Configuration	M90 special
1Fh	Bar Code Control	M90 special
25h	Set Interrupt Vector	
2Ah	Get System Date	
2Bh	Set System Date	
2Ch	Get System Time	
2Dh	Set System Time	
2Eh	Set Alarm Date	M90 special
2Fh	Set Alarm Time	M90 special
30h	Get M90 Version Number	
35h	Get Interrupt Vector	
36h	Get Free Disk Clusters	
37h	Get/Set EXEC size	M90 special
3Ch	Create File	
3Dh	Open File	
3Eh	Close File	
3Fh	Read File	
40h	Write File	
41h	Delete File	
42h	Move File Pointer	
43h	Get File Attribute	
44h	Device IOCTL	Only Console I/O supported
48h	Allocate Memory	

Table 4-1 (Continued)
Implemented DOS Functions (INT 21h)

Function	Description	Comment
49h	Free Allocated Memory	
4Ah	Modify Allocated Memory	
4Bh	Execute or Call Program	
4Ch	Terminate Process	
50h	Get Bar Code Data	M90 special
51h	Bar Code Port ON or OFF	M90 special
56h	Rename File	
5Bh	Create New File	

▶ **NOTE:** *All of these INT 21h functions are compared to INT 21h functions in DOS 3.*

Table 4-2
Unsupported DOS Functions (INT 21h)

Function	Description
0Ch	Flush Buffer, Read Keyboard
0Dh	Disk Reset
0Eh	Select Disk
0Fh	Open File (FCB)
10h	Close File (FCB)
11h	Find First File (FCB)
12h	Find Next File (FCB)
13h	Delete File (FCB)
14h	Sequential Read (FCB)
15h	Sequential Write (FCB)
16h	Create File (FCB)

Table 4-2 (Continued)
Unsupported DOS Functions (INT 21h)

Function	Description
17h	Rename File (FCB)
19h	Get Current Disk
1Ah	Set DTA Address
1Bh	Get Default Drive Data
1Ch	Get Drive Data
21h	Random Read (FCB)
22h	Random Write (FCB)
23h	Get File Size (FCB)
24h	Set Relative Record (FCB)
26h	Create New PSP
27h	Random Block Read (FCB)
28h	Random Block Write (FCB)
29h	Parse Filename
2Fh	Get DTA Address
31h	TSR
33h	Get or Set Ctrl-C Check
34h	Get Addr of InDOS Flag
38h	Get or Set Country
39h	Create Directory
3Ah	Remote Directory
3Bh	Change Current Directory
45h	Duplicate File Handle
46h	Force Duplicate File Handle
47h	Get Current Directory
4Dh	Get Return Code

Table 4-2 (Continued)
Unsupported DOS Functions (INT 21h)

Function	Description
4Eh	Find First File
4Fh	Find Next File
54h	Get Verify File
57h	Get or Set Date and Time of File

BIOS Call

Display Font Functions: INT 09H

0 Select Large Font
Entry Parameter: AH=0 Select 8x16-dot character font (4-line x 12-column display)
Return Value: None

1 Select Small Font
Entry Parameter: AH=1 Select 6x8-dot character font (8-line x 16-column display)
Return Value: None

2 Set Font Type
Entry Parameter: AH=2
AL=0 Set to large font
AL=1 Set to small font
Return Value: None

3 Get Font Type
Entry Parameter: AH=3
Return Value: AL=0 Large font
AL=1 Small font

4 Set User-Defined Font for All Characters

Entry Parameter: AH=4
AL=0 Large font
AL=1 Small font
DS:DX=Pointer to buffer with font data
(for large font: buffer size=16x256
=4096 bytes
(for small font: buffer size= 6x256
=1536 bytes)

Return Value: None

5 Get Font Data for All Characters

Entry Parameter: AH=5
AL=0 Large font
AL=1 Small font
DS:DX=Pointer to the buffer
(for large font: buffer size=16x256
=4096 bytes
(for small font: buffer size= 6x256
=1536 bytes)

Return Value: Font data in the buffer

6 Set User-Defined Font for One Character

Entry Parameter: AH=6
AL=0 Large font
AL=1 Small font
CL=0–255 character
DS:DX=Pointer to buffer with font data
(for large font: buffer size=16 bytes
for small font: buffer size=6 bytes)

Return Value: None

7 Get Font Data for One Character

Entry Parameter: AH=7
AL=0 Large font
AL=1 Small font
CL =0–255 characters
DS:DX=Pointer to the buffer
(for large font: buffer size=16 bytes
for small font: buffer size=6 bytes)

Return Value: Font data in the buffer

Table 4-3
Implemented BIOS Functions (INT 09h)

Function	Description	Comment
00h	Select Large Font	M90 special
01h	Select Small Font	M90 special
02h	Set Font Type	M90 special
03h	Get Font Type	M90 special
04h	Set User-Defined Font for All Characters	M90 special
05h	Get Font Data for All Characters	M90 special
06h	Set User-Defined Font for One Character	M90 special
07h	Get Font Data for One Character	M90 special

Kermit Function: INT 0x0F

0 Kermit Application Invoke
Entry Parameter: AH=0
Return Value: None

▶ **NOTE:** *To exit Kermit, send command "BYE" to M90 or press the **[4]** key.*

Table 4-4
Implemented Kermit Functions (INT 0x0f)

Function	Description	Comment
00h	Kermit Application Invoke	M90 special

LCD Function: INT 10H

0 Clear LCD Screen
Entry Parameter: AH=0
Return Value: None

1 Enable/Disable Scroll
Entry Parameter: AH=1
AL=0 Disable
AL=1 Enable
Return Value: None

2 Set Cursor Position
Entry Parameter: AH=2
DH=0–3 (Row)
DL=0–19 (Column)
Return Value: None

3 Get Cursor Position
Entry Parameter: AH=3
Return Value: DH=0–3 (Row)
DL=0–19 (Column)

4 Display 5x8 Bit Map Pattern Font

Entry Parameter: AH=4
DH=0–3 (Row)
DL=0–19 (Column)
DS:BX=Pattern data (5 bytes)
Return Value: None

5 Enable/Disable Cursor

Entry Parameter: AH=5
AL=0 Disable
AL=1 Enable
Return Value: None

6 Set Cursor Shape

Entry Parameter: AH=6
AL=0 Block
AL=1 Underline
Return Value: None

7 Set Display Size

Entry Parameter: AH=7
AL=0 4x20 character display
AL=1 6x20 character display
Return Value: None

8 Get Display Size

Entry Parameter: AH=8
Return Value: AL=0 4x20 character display
AL=1 6x20 character display

9 Enable/Disable Power-on Logo Display

Entry Parameter: AH=6
AL=0 Disable
AL=1 Enable
Return Value: None

A Display Character

Entry Parameter: AH=0AH
AL=0–255 character to display
Return Value: None

4F Display 16x16 Bitmap at Current Cursor Position

Entry Parameter: AH=4FH
DS:BX=Pointer to bitmap (32-byte pattern data)
Return Value: None

▶ **NOTE:** *Function 4F is available only in large font, the bit map is twice as wide as two normal characters. You need to consider this when you mix bit maps and characters and when you set cursor positions.*

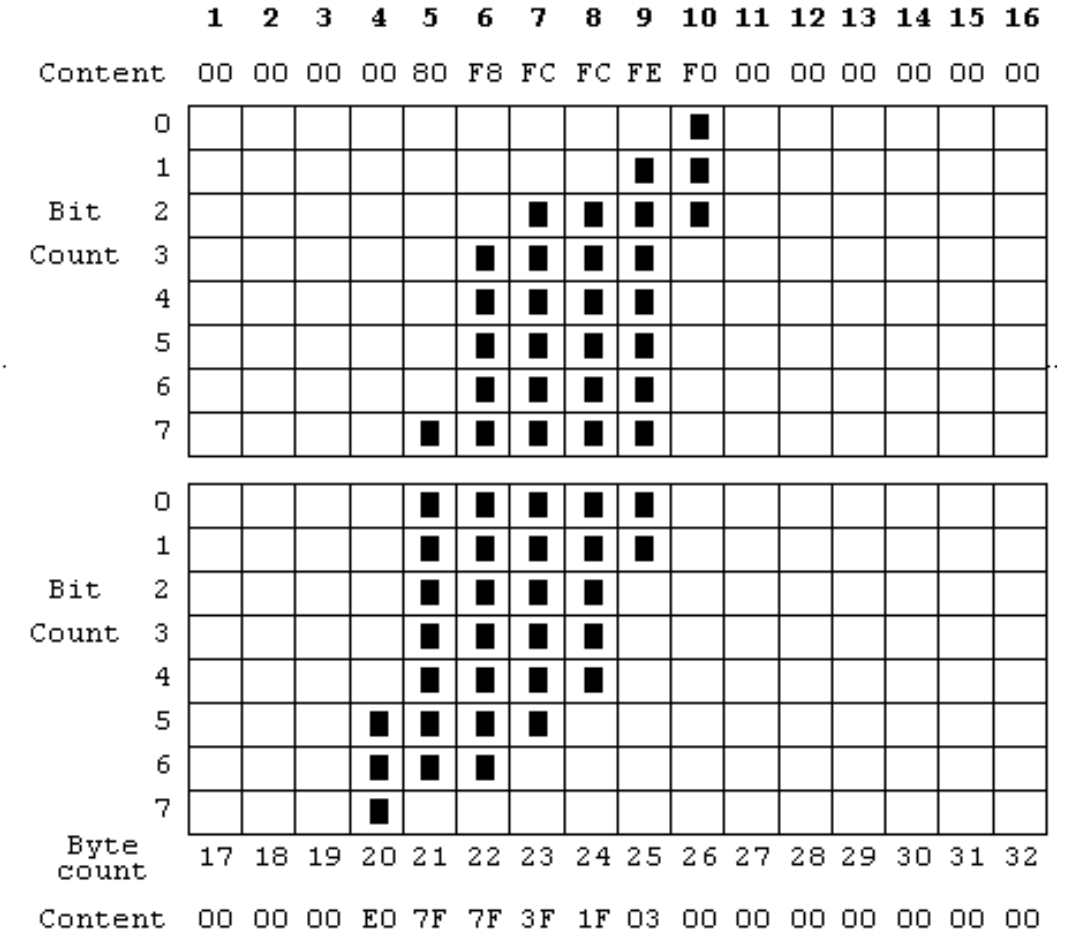


Figure 4-1
Sample Bit Map

EXAMPLE: C string example of above bit map:

```
Unsigned char
logo[32]=  {0x00,0x00,0x00,0x00,0x80,0xf8,0xfc,0xfc,0xfe,0x0f,0x00,0x00,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xe0,0x7f,0x7f,0x3f,0x1f,0x03,0x00,
            0x00,0x00,0x00,0x00,0x00,0x00};
```

▶ **NOTE:** *User Menu/Set/Cursor also sets the cursor shape.*

Table 4-5
Implemented BIOS Functions (INT 10h)

Function	Description	Comment
00h	Clear LCD Screen	M90 special
01h	Enable or Disable Scroll	M90 special
02h	Set Cursor Position	
03h	Get Cursor Position	
04h	Display 8x5 bit font	M90 special
05h	Enable or Disable Cursor	M90 special
06h	Set Cursor Shape	M90 special
07h	Set Display Size	M90 special
08h	Get Display Size	M90 special
09h	Enable/Disable Power-on Logo	M90 special
0ah	Display Character	
4fh	Display 16x16 bit bitmap at Current Cursor	M90 special

Power Management Function: INT 22H

0 Power Off
Entry Parameter: AH=0
Return Value: None

1 Wait Interrupt — Input Data
Entry Parameter: AH=1
Return Value:
AL=0 Keypad
AL=1 Scanner
AL=2 RS-232 receiver
AL=3 Other or user-defined interrupts
AL=4 Power-on

▶ **NOTE:** *The following function does not check system input-data buffers. It waits for interrupts and new input data. M90 shuts off according to system settings if there is no data input during the auto-off time period.*

2 Wait Interrupt - Input Data with Buffer Check

Entry Parameter: AH=2

Return Value: AL=(bit sets)

- Bit 0: Keypad
- Bit 1: Scanner
- Bit 2: RS-232 receiver
- Bit 3: Other or user-defined interrupts
- Bit 4: Power-on

▶ **NOTE:** *If there is input data in the system buffers, the function returns directly. Otherwise, it waits for interrupts and new input data. M90 shuts off according to system settings if there is no data input during the auto-off time period.*

3 Wait Interrupt - Input Data with Timeout

Entry Parameter: AH=3

CX=Timeout 10–30,000 ms

Return Value:

- AL=0 Keypad
- AL=1 Scanner
- AL=2 RS-232 receiver
- AL=3 Other or user-defined interrupts
- AL=4 Power-on
- AL=5 Timeout

▶ **NOTE:** *The **Wait Interrupt – Input Data with Buffer Check and Timeout** function does not check system input data buffers. If there is input data in the system buffers, the function returns directly. Otherwise, it waits for interrupts and new input data. M90 will not turn off according to system settings.*

4 Wait Interrupt — Input Data with Buffer Check and Timeout

Entry Parameter: AH=4
CX=Timeout 10–30,000 ms

Return Value: AL=(bit sets)
- Bit 0: Keypad
- Bit 1: Scanner
- Bit 2: RS-232 receiver
- Bit 3: Other or user-defined interrupts
- Bit 4: Power-on

Beeper Frequency and Time Control: INT 31H

Entry Parameter: AX=Frequency
BX=Time duration

Return Value: None

Note:

AX	Frequency(Hz)	BX	Time Duration
0	200	0	10 ms
1	400	1	50 ms
2	600	2	100 ms
3	800	3	200 ms
4	1K	4	500 ms
5	2K	5	800 ms
6	2.5K	6	1 second
7	3K	7	1.5 seconds
8	5K	8	2 seconds

Table 4-6
Implemented BIOS Functions (INT 31h)

Function	Description	Comment
(Not applicable)	Beeper frequency and Time Control	Added function

RS-232 Function: INT 33H

0 Set Communication Parameters

Entry Parameter: AH=0

AL	Bits 7–4:	0001xxxx	Baud 150
		0010xxxx	Baud 300
		0011xxxx	Baud 600
		0100xxxx	Baud 1200
		0101xxxx	Baud 2400
		0110xxxx	Baud 4800
		0111xxxx	Baud 9600
		1000xxxx	Baud 19200
		1001xxxx	Baud 38400
		1010xxxx	Baud 57600
	Bits 3–2:	xxxx00xx	No parity
		xxxx01xx	Odd parity
		xxxx11xx	Even parity
	Bit 1:	xxxxxx0x	One stop bit
		xxxxxx1x	Two stop bits
	Bit 0:	xxxxxxx0	7 data bits
		xxxxxxx1	8 data bits

Return Value: None

1 Input Character

Entry Parameter: AH=1

Return Value: If no character received:
AH=1
AL=Unpredictable
If a character received:
AH=0
AL=Character input

2 Output Character

Entry Parameter: AH=2
AL=Character output
Return Value: None

3 Enable RS-232 Port

Entry Parameter: AH=3
Return Value: None

4 Disable RS-232 Port

Entry Parameter: AH=4
Return Value: None

5 Set RTS/DTR

Entry Parameter: AH=5

AL=	Bit 0:	0	Set RTS low level
		1	Set RTS high level
	Bit 1:	0	Set DTR low level
		1	Set DTR high level

Return Value: None

6 Get RS-232 Hardware Status

Entry Parameter: AH=6

Return Value:

AL=	Bit 0:	0	Get RTS low level
		1	Get RTS high level
	Bit 1:	0	Get DTR low level
		1	Get DTR high level
	Bit 2:	0	Get OUT0 low level
		1	Get OUT0 high level
	Bit 3:	0	Get CTS low level
		1	Get CTS high level
	Bit 4:	0	Get DSR low level
		1	Get DSR high level
	Bit 5:	0	Get RI low level
		1	Get RI high level
	Bit 6:	0	Get DCD low level
		1	Get DCD high level

▶ **NOTE:** *Bits 4, 5, and 6 (DSR, RI, and DCD) are returning the same value as RI. If **Function 6** is called when the M90 is in an office dock, RI is the only signal present in the connector. If the M90 is connected via the direct connect cable, all of those bits are undefined as neither signal is present in the Modular connector at the bottom of the M90.*

7 Set OUT0 Signal

Entry Parameter: AH=7

AL=0 Set OUT0 signal in Modular 10-pin connector to low

AL=1 Set OUT0 signal in Modular 10-pin connector to high

Return Value: None

Table 4-7
Implemented BIOS Functions (INT 33h)

Function	Description	Comment
00h	Set Communication Parameters	Added function
01h	Input RS-232 character	Added function
02h	Output RS-232 character	Added function
03h	Enable RS-232 port	Added function
04h	Disable RS-232 port	Added function
05h	Set RTS/DTR	Added function
06h	Get RS-232 hardware status	Added function
07h	Set OUT0 signal in Modular 10-pin connector	Added function

Appendix A

Connector Pin Assignments

Table A-1
10-Pin Modular Connector

Pin #	Signal	Direction	Description
1	DC 9 V	Power	9-volt charge power
2	TXD	Output	Transmitted Data
3	RXD	Input	Receive Data
4	I/O 5 V	Power	Regulated 5-volt max 100 mA
5	GND	Power	Ground
6	DTR	Output	Data Terminal Ready
7	CTS	Input	Clear to Send
8	RTS	Output	Request to Send
9	WANDSIG1	Input	Wand input
10	OUT0	Output	Digital OUT0 signal

Appendix B
Programming Applications

The M90 provides a utility for putting application programs into the system Read-Only Memory (ROM) area. Once in the ROM, the programs become user-resident applications in the M90 system. The advantage of the user-resident application is that they do not have to be downloaded from a host computer through M90 RS-232 port and cannot be accidentally deleted. It can also save the Random Access Memory (RAM) space for the system. Up to eight user-resident applications can be in the M90, with the total file size less than or equal to 128 KB.

The M90 system EXEC job scheduler supports the user-resident application. The ROM resident user applications must be of *.EXE type. The system does not allow data files. EXEC searches user-resident applications first when a RUN command executes. If there is an AUTOEXEC.EXE user-resident application, it runs after power-on when the M90 is in "Ready Mode."

To add a program to the ROM disk, download the program to the RAM disk as described later in this appendix, then start the M90 in the "Supervisor Mode." Select **System** → **Programming** → **Add Program**. Adding a program is fully described under "Operation Modes" in Section 3.

▶ **NOTE:** *As this requires reprogramming of the FLASH in the M90, make sure that either the M90 battery is fully charged or that the M90 is on charge (office dock). A power failure during flash programming could corrupt the FLASH. If this occurs, you must have the M90 FLASH EEPROM replaced at a service center.*

Reprogramming the M90 Flash

The following steps explain how to update the existing flash program in the M90. The M90 Flash can be updated to a new version.

Download the new flash to the M90 RAM disk, then start the M90 in "Supervisor Mode." Select **System → Programming → Update Flash** (see "Operation Modes" in Section 3).

▶ **NOTE:** *As this requires reprogramming of the FLASH in the M90, make sure that either the M90 battery is fully charged or that the M90 is on charge (office dock). A power failure during flash programming could corrupt the FLASH. If this occurs, the M90 has to have the FLASH EEPROM replaced at a service center.*

Download Application or BIOS to M90

Set Up M90

Downloading from Host Computer

Connect the host personal computer (PC) to the M90 unit with a serial null modem cable. Select "COM" in the "User Mode" and the M90 enters the "Kermit server" mode with default communication parameters 19200 bps, 8 bit, 1 stop, no parity, and no flow control.

Kermit Communications Program

To set up the PC to talk to the M90 unit using a Kermit communications program, consult your manual for the Kermit communications program.

Windows Hyper Terminal Method

If you are using "Hyper Terminal" on a Windows 95 or 98 PC, do as follows to download a file to the M90. "Hyper Terminal" is not optimized for large files or for daily use in an application, but is an easier method of downloading files to the M90 as compared to the Kermit method.

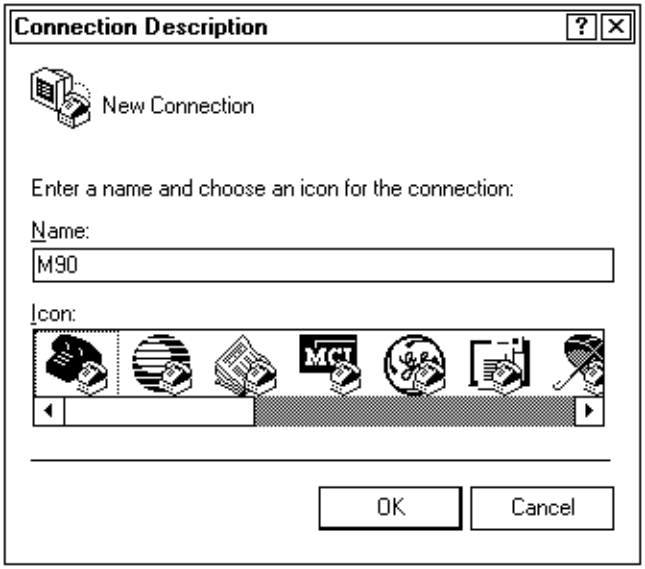
Set Up Host Computer

Do the following to set up the host computer:

1. From the Windows desktop, select **Start** → **Programs** → **Accessories** → **Hyper Terminal** to access the contents of the "Hyper Terminal" directory.


Hypertrm.exe

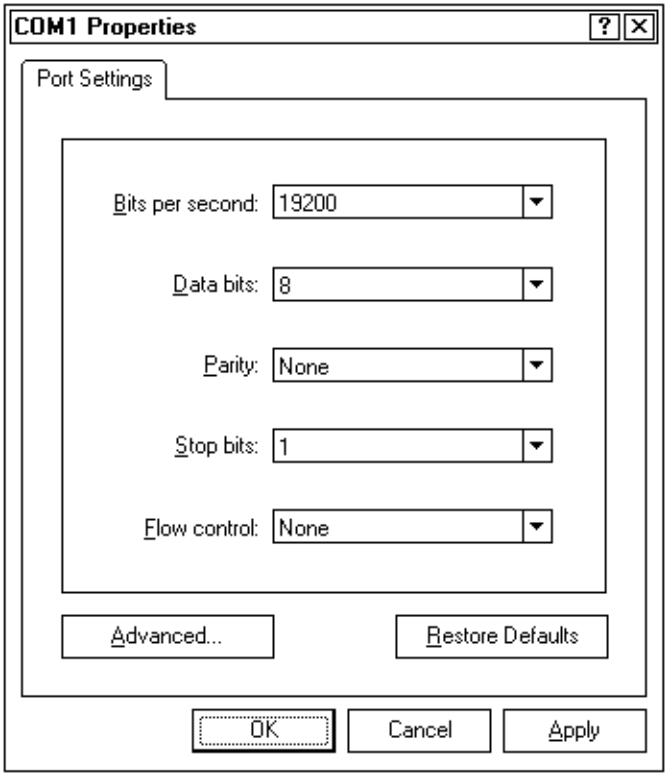
2. Double-click the **Hypertrm.exe** icon (shown left) to access the "Connection Description" window to create a new connection icon.



3. Enter a descriptive title for this connection in the **Name** field, such as "M90," and select an icon from the **Icon** field to illustrate this connection.

4. Click the **OK** button to continue to the "Phone Number" window. Select the applicable "Direct to Com" option from the **Connect using** drop-down list, then click the **OK** button to continue.

5. Use the applicable "COM Properties" window to set up the proper port settings.



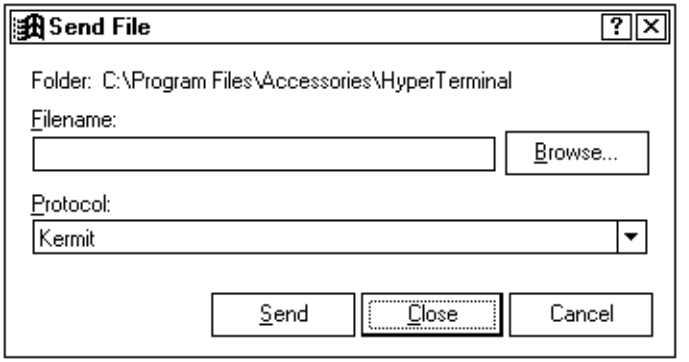
- Set the **Bits per second** field to "19200."
- Set the **Flow control** field to "None."

Click the **Apply** button, then the **OK** button. A communications icon (like the one shown left) appears for this connection within the "Hyper Terminal" directory.

Set Up Protocol Via Hyper Terminal

Double-click the communications icon to access the "Hyper Terminal" window.

1. From the "Hyper Terminal" title bar, select **Transfer** → **Send File** to access the "Send File" window.



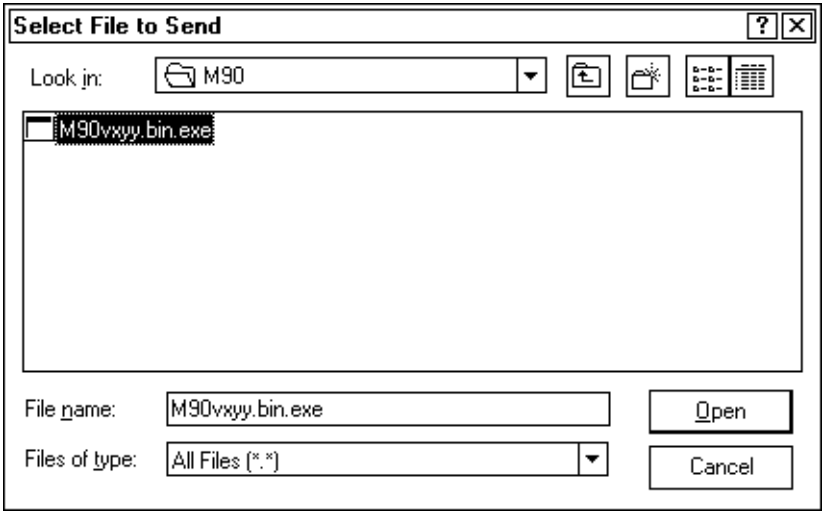
2. Select "Kermit" from the **Protocol** drop-down list, then click the **Close** button to return to the "Hyper Terminal" window.
3. From the "Hyper Terminal" window, select **File** → **Save** to exit.

The host computer is now ready to send and receive files. Your connection is saved, for later use, under the name and icon created.

Download M90VXYY.BIN to the M90

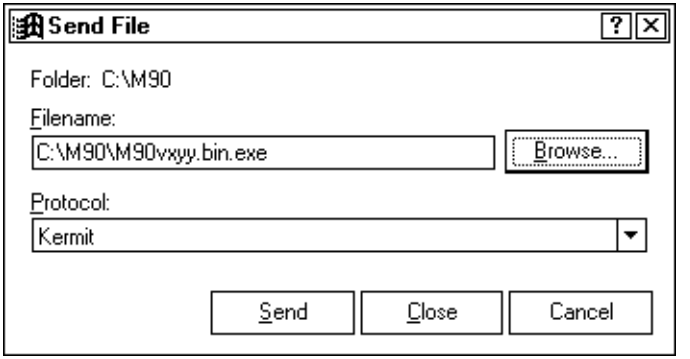
Do the following to download files from the host computer to the M90. Repeat these steps for each file to be downloaded:

1. From the "Hyper Terminal" window, select **Transfer** → **Send File** to access the "Send File" window.
2. Use the **Browse** button to access the "Select File to Send" window and locate the directory that contains files for the M90.

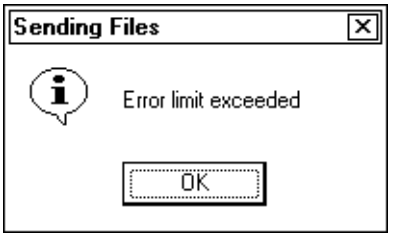


3. Select the ***M90VXYY.BIN*** file, then click the **Open** button to access the "Send File" window.

4. ***M90VXYY.BIN*** should appear in the **Filename** field in the "Send File" window. If not, go back to the "Select File to Send" window and select M90VXYY.BIN again.



5. Click the **Send** button to download M90VXYY.BIN to the M90. The "Kermit file send to M90" window appears with the progress of the file transmission.
   - If the file is sent successfully, the status screen disappears.
   - If the transmission fails, the following "Sending Files" message appears. Click the **OK** button to exit, then try to send M90VXYY.BIN again. Likely problems could include faulty baud rate settings or cable connections.



6. From the M90, press and release **[FN]** and press ▶ to exit "Server Mode" and return to "Ready Mode."

7. Type “DIR” at the “Ready Mode” prompt, then press **[ENT SCAN]** for a list of files. Ensure that M90VXYY.BIN is listed.

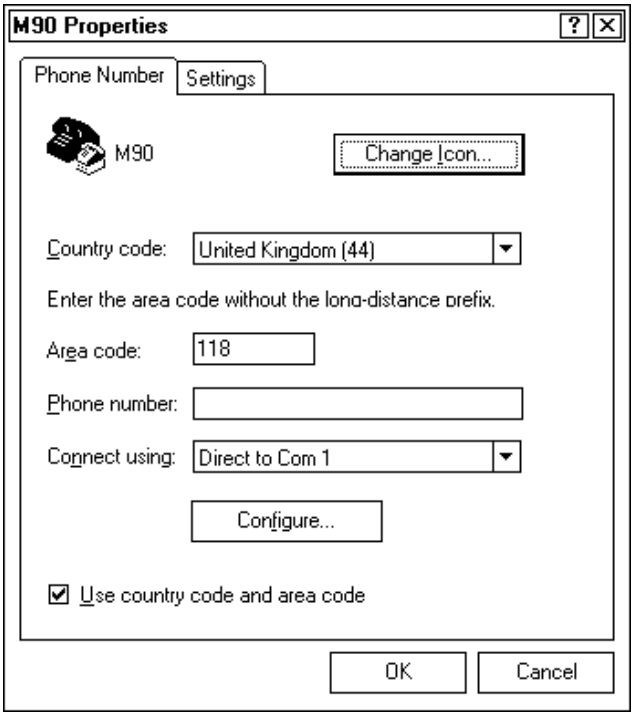
   If not listed, go to page B-7 to download the program again.

   If listed, you may want to review the Hyper Terminal settings, instructions start on the next page.

Review Hyper Terminal Settings

Do the following if you want to change or view the Hyper Terminal settings:

1. Double-click the communications icon (like the one shown left) to access the "Hyper Terminal" window.
2. From the "Hyper Terminal" window, select **File** → **Properties** to access the "Properties" window.



3. Change the COM port in the **Connect using** field and the icon with the **Change Icon** button. Click the **Configure** button to access the "COMx Properties" window and change the communication port settings.

Click the **OK** button to save your changes or click the **Cancel** button to exit this screen without saving your changes.

Appendix C

Bar Code Symbologies

This appendix contains a brief explanation of each bar code symbology that the M90 portable data entry unit decodes. It explains some of the general characteristics and uses of these bar code types.

Specific bar code algorithms can be enabled using the setup menus or the host computer. Once the computer correctly decodes a bar code, the computer encodes data with descriptive information about the symbol.

You can improve response time by limiting the computer to the bar codes being used. Table C-1 shows the bar code string formats.

▶ NOTE: *These bar code data definitions apply to the Data Format column in Table C-1 below:*

- *a Add-on code digits*
- *c Check digits*
- *d Bar code digits*
- *f EAN flag 1 characters*
- *n Number system digits*
- *s Start and stop digits*

If MOD 10 or MOD 11 check digits are enabled, the digit falls at the end of a bar code data string. Each check digit enabled extends the length of the bar code data string by one character.

Table C-1
Bar Code Data String Formats

Data Bar Code Type	Data Format	Data Length
UPC short (UPC-E)	ndddddc	8
EAN short (EAN-8)	fnddddc	8
UPC long (UPC-A)	nddddddddddc	12
EAN long (EAN-13)	fnddddddddddc	13
UPC short add-on 2	ndddddcaa	10
EAN short add-on 2	fnddddcaa	10
UPC long add-on 2	nddddddddddcaa	14
EAN long add-on 2	fnddddddddddcaa	15
UPC short add-on 5	ndddddcaaaaa	13
EAN short add-on 5	fnddddcaaaaa	13
UPC long add-on 5	nddddddddddcaaaaa	17
EAN long add-on 5	fnddddddddddcaaaaa	18
Interleaved 2 of 5	d.....d	1 to 31
Codabar	sd....ds	3 to 31
Code 39	d.....d	1 to 31
Code 128	d.....d	1 to 31
EAN 128	d.....d	1 to 31

The M90 unit recognizes 15 of the most widely used bar code symbologies. With bar code symbologies, like languages, there are many different types. A bar code symbology provides the required flexibility for a particular inventory tracking system.

A symbology may be for particular industries, such as food and beverage, automotive, railroad, or aircraft. Some of these industries have established their own bar code symbology because other symbologies may not meet their needs.

Without going into great detail on the bar code structure, note that no two products use the same bar code. Each product gets a unique bar code.

Industries that use a particular type of bar code symbology have formed regulating committees or are members of national institutes that issue and keep track of bar codes. This ensures that each organization that contributes to a particular industry conforms to its standard. Without some form of governing body, bar coding would not work.

These are the bar codes described in this appendix:

- UPC (Universal Product Code) with or without add-ons
- EAN (European Article Numbering Code) with or without add-ons
- Codabar
- C39 (Code 39)
- C128 (Code 128)
- I 2 of 5 (Interleaved 2 of 5 Code)

UPC

The UPC (Universal Product Code) is the symbology used throughout the grocery and retail industries. This bar code symbology contains two pieces of numerical information encoded on the bar code, producer identification, and product identification information.

The UPC symbol is 12 characters long. The first character of the UPC symbol is a number system character, such as "0" for grocery items and "3" for drug- and health-related items.

The UPC symbology is for retail environments such as:

- Grocery stores
- Convenience stores
- General merchandise stores

Some retail items are so small that a standard UPC bar code cannot fit on the packaging. When this occurs there is a permitted shorter version of the UPC symbology, referred to as UPC-E. UPC-E is six characters long (eight including number system and check digit), approximately half the size of a standard UPC bar code.

EAN

The EAN (European Article Numbering Code) symbology is similar to the UPC symbology, except that it contains 13 characters and uses the first two to identify countries.

The EAN symbology is used throughout most of Europe in the retail environment. Although similar to UPC symbology, the two are not interchangeable.

Codabar

Codabar was for retail price-labeling systems. Today it is widely accepted by:

- Libraries
- Medical industries
- Photo finishing services

Codabar is a discrete, self-checking code with each character represented by a stand-alone group of four bars and three intervening spaces.

Four different start or stop characters get defined and designated “a”, “b”, “c”, and “d”. These start and stop characters are constructed using one wide bar and two wide spaces. A complete Codabar symbol begins with one of the start or stop characters followed by some number of data characters and ending in one of the start or stop characters.

Any of the start or stop characters may be used on either end of the symbol. It is possible to use the 16 unique start or stop combinations to identify label type or other information.

Since Codabar is variable-length, discrete, and self-checking, it is a versatile symbology. The width of space between characters is not critical and may vary significantly within the same symbol. The character set consists of “0” through “9”, “–”, “$”, “:”, “/”, “.”, and “+”.

The specific dimensions for bars and spaces in Codabar optimize performance of certain early printing and reading equipment. Codabar has 18 different dimensions for bar and space widths. So many different dimensions often result in labels printed out of specification and cause Codabar printing equipment to be more expensive.

Code 39

Code 39 (C39) is the most widely used symbology among the industrial bar codes. Most major companies, trade associations, and the federal government find this code to fit their needs. The main feature of this symbology is the ability to encode messages using the full alphanumeric character set, seven special characters, and ASCII characters.

Programming for this symbology can be for any length that the application requires. The application program for the M90 handles symbology that is at least one character but no more than 32 characters in length.

When programming the computer for Code 39, it is important to set the symbology limit as close as possible (minimum and maximum bar code lengths being scanned). Doing so keeps the computer bar code processing time to a minimum and conserves battery power.

Bar code readers can respond to Uniform Symbology Specification symbols in non-standard ways for particular applications. These methods are not for general applications, because of the extra programming required. Code 39 Full ASCII is one example of non-standard code.

Encoded Code 39 (Full ASCII)

If the bar code reader has been programmed for the task, it is possible to encode the entire ASCII character set (128 characters). This is done using two character sequences made up of one of the symbols (“$”,“.”,“%”,“/”) followed by one of the 26 letters.

Code 128

Code 128 (C128) is one of the newest symbologies used by the retail and manufacturing industries. It responds to the need for a compact alphanumeric bar code symbol that could encode complex product identification.

The fundamental requirement called for a symbology capable of being printed by existing data processing printers (primarily dot-matrix printers) that produce daily, work-in-progress, job, and product traceability documents. The ability to print identification messages between 10 and 32 characters long, on existing forms and labels deemed an important requirement.

Code 128 uniquely addresses this need as the most compact, complete, alphanumeric symbology available.

Additionally, the Code 128 design with geometric features, improves scanner read performance, does self-checking, and provides data message management function codes.

Code 128 encodes the complete set of 128 ASCII characters without adding extra symbol elements. Code 128 contains a variable-length symbology and the ability to link one message to another for composite message transmission. Code 128, being a double-density field, provides two numeric values in a single character.

Code 128 follows the general bar code format of start zone, data, check digit, stop code, and quiet zone. An absolute minimum bar or space dimension of nine mils (0.010 inch minimum nominal ± 0.001 inch tolerance) must be maintained.

Characters in Code 128 consist of three bars and three spaces so that the total character set includes three different start characters and a stop character.

UCC/EAN-128 Shipping Container Labeling is a versatile tool that can ease movement of products and information. The Shipping Container Labeling bar code can take any

form and usually has meaning only within the company or facility where applied.

Because this *random* data can get mistaken later for an industry standard code format, the UCC and EAN chose a symbology uniquely identified from these other bar codes. This standard is for maximum flexibility, to handle the diversity of distribution in global markets by cost efficiency.

The UCC/EAN-128 Container Labeling specification calls for a FUNC1 to immediately follow the bar code's start character. FUNC1 also follows any variable-length application field. The specification also calls for the computer to send "]C1" for the first FUNC1. The specification requires that the computer send a "<GS>" (hex 1D) for subsequent FUNC1 codes in the bar code.

Because "<GS>" is not compatible with computer emulation data streams, the Uniform Code Council has been asked to change the specification. This change is made to send the same three character sequence "]C1" to identify the embedded FUNC1 codes.

This implementation should provide for clean application coding by identifying the same sequences for the same scanned codes. If the communication of Norand bar code types is enabled, the Shipping Container Label codes precede with a "J". These strings will appear on the computer display. The application may have to allow for strings longer than 48 characters (maximum length indicated in the specification). Actual length variance depends on the number of variable-length data fields. Allowing for 60 characters should be sufficient. Within the Code 128 specification, the computer can link bar codes together. If this is to happen, allow for more characters (computer limit is 100 characters).

The Application Identifier Standard, that is part of the UCC/EAN Shipping Label concept, complements, rather than replaces, other UCC/EAN standards. Most UCC/EAN standards primarily identify products.

Several industries expressed the need to standardize more than product identification. The UCC/EAN Code 128 Application Identifier Standard supplies this tool. The standard adds versatility for inter-enterprise exchanges of perishability dating, lot and batch identification, units of use measure, location codes, and several other information attributes.

For more detailed information on Code 128 UCC/EAN Shipping Label bar code and Application Identifier Standard, refer to the UCC/EAN-128 Application Identifier Standard specification.

I 2 of 5

I 2 of 5 (Interleaved 2 of 5 Code) is an all-numeric symbology, widely used for warehouse and heavy industrial applications. Its use has been particularly prevalent in the automobile industry. The I 2 of 5 symbology can be placed on smaller labels than what the standard UPC symbology requires.

I 2 of 5 also provides a little more flexibility on the type of material it can print on. Interleaved 2 of 5 Code has its name because of the way the bar code is configured.

I 2 of 5 bars and spaces both carry information. The bars represent the odd number position digits, while spaces represent the even number position digits. The two characters are interleaved as one. Messages encoded with this symbology have to use an even number of characters since two numeric characters always get interleaved together.

Appendix D

Cables

Cables that work with the M90 Computer appear in this appendix. They attach to modems, cradles, any serial printer, or a PC. None of these cables are standard.

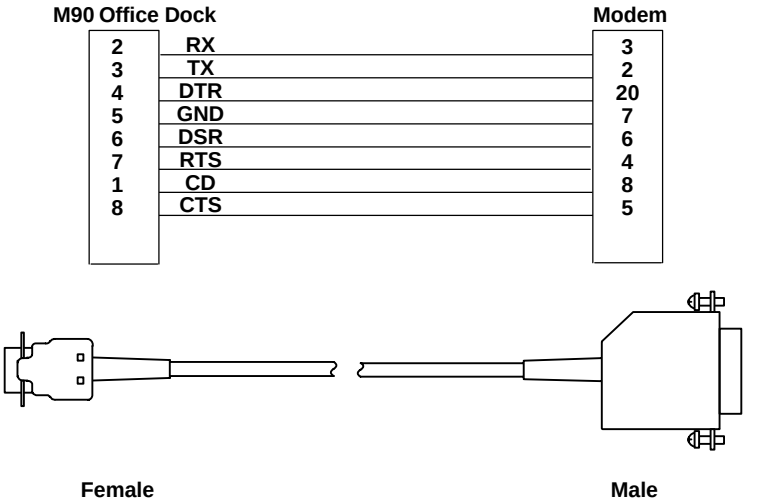


Figure D-1
M90 Office Dock to Modem Cable (M90503)

▶ **NOTE:** *This cable is **not** designed for NORAND® printers.*

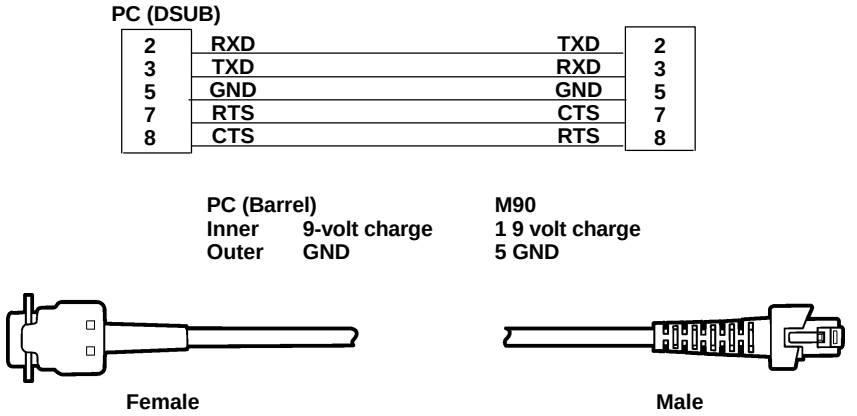


Figure D-2
M90 to PC Cable (M90403)

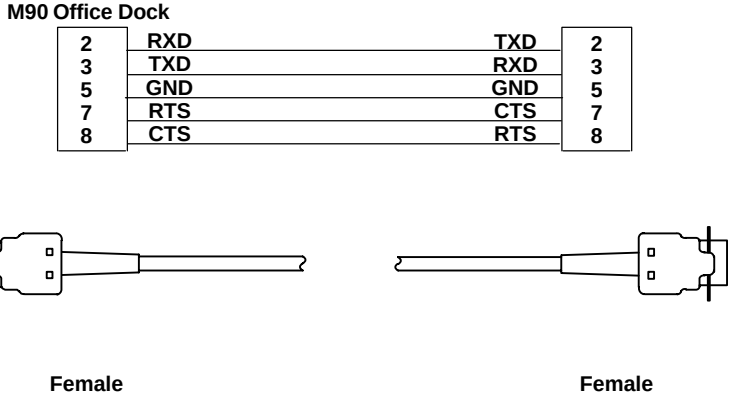


Figure D-3
M90 Office Dock to PC Cable (M90504)

Glossary

ACK (ACKnowledgement)

A type of message sent to show that a previous message arrived at its destination without error.

ANSI (American National Standards Institute)

A private organization that coordinates some United States (US) standards setting. It also approves some US standards that are often called ASNI standards. ANSI also represents the US to the International Organization for Standards (ISO).

ASCII (American Standard Code for Information Interchange)

A standard character set that typically assigns a 7-bit binary code to each letter, number, and selected control character. Erroneously used now to refer to 8-bit Extended ASCII. The other major encoding standard is EBCDIC.

BIOS (Basic Input and Output System)

A set of programs, usually in ROM, that lets each computer's central processing unit communicate with printers, disks, keyboards, consoles, and other attached input and output devices.

***Bus** (network)*

The main (multiple access) network cable or line that connects network stations. Also refers to a network topology of multiple stations communicating directly with the same cable with terminators at both ends, like an Ethernet or token bus.

CCITT (Comite Consultatif International de Telegraphique et Telephonique)

This organization is part of the United National International Telecommunications Union (ITU) and is responsible for making technical recommendations about telephone and data communications systems. It is best known for its "V" and "X" recommendations. X.25 is one such recommendation.

CD

Carrier Detect

Codabar *(Bar Code Symbology)*

A discrete self-checking bar code symbology with each character represented by a stand alone group of four bars and three intervening spaces. It was developed for retail price-labeling systems and is currently accepted in libraries, medical industries, and photo finishing services. (See also: Self-checking and Discrete Code)

Code 39 (Code 3 of 9) *(Bar Code Symbology)*

An alphanumeric bar code symbology with a set of 43 characters, including uppercase and seven special characters. The name comes from the idea that 3 of the 9 elements representing a character are wide while the remaining 6 are narrow.

It is the most widely used industrial bar code. This code fits the needs of most major companies, trade associations, and the federal government.

Code 93 (Code 3 of 9) *(Bar Code Symbology)*

A bar code symbology compatible with Code 39 that offers a full ASCII character set and a higher coding density than Code 39.

Code 128 *(Bar Code Symbology)*

A high density bar code symbology, allowing encoding of all 128 ASCII characters without adding extra symbol elements. It is used by retail and manufacturing industries.

CPU

Central Processing Unit.

CR

Carriage Return.

CTS (Clear To Send)

A time delay inserted after a data terminal RTS to allow a modem to turn carrier on and establish equalization and synchronization; also known as RTS-CTS delay and turnaround delay.

DCD (Data Carrier Detect)

A signal in EIA RS-232-C specification.

DCE (Distributed Computing Environment)

An architecture of standard programming interfaces, conventions, and server functionality for distributing applications across networks of different computers.

DOS (Disk Operating System)

A program or set of programs that tells a disk-based computer system to schedule and supervise work, manage computer resources, and operate and control its peripheral devices.

DSR (Data Set Ready)

An RS-232 modem interface control signal (sent from the modem to the DTE on pin 6) that indicates the modem is connected to the telephone circuit.

DTE (Data Terminal Equipment)

The devices in a category that includes terminals and computers. Also refers to the interface to users' equipment as opposed to the DCE interface to the network.

DTR (Data Terminal Ready)

An RS-232 modem interface control signal (sent from the DTE to the modem on pin 20) that indicates the DTE is ready for data transmission and requests the modem be connected to the circuit.

EAN (European Article Numbering) *(Bar Code Symbology)*

European Article Numbering Code. A bar code symbology similar to the UPC symbology except that EAN contains 13 characters and uses the first two to identify a country.

EBCDIC (Extended Binary Coded Decimal Interchange Code)

An 8-bit character code scheme used in IBM environments.

EEPROM

Electrically Erasable Programmable Read Only Memory.

EIA (Electronics Industries Association)

A United States trade organization that issues its own standards and contributes to ASNI. Best known for its development RS-232 and the building wiring standard, 568. Membership includes US manufacturers.

ENQ

Enquiry or request for header block. A request for a response from another terminal. It obtains identification and an indication of the other station's status.

EOF (End Of File)

A constant following the last data in a file that signals its end.

EOT

End Of Transmission or End Of Tape.

ETX (End Of Text)

A control character used to indicate the conclusion of a message.

FAT

File Allocation Table.

FHT

File Handle Table.

Flash

A technology for nonvolatile memory storage. A special type of EEPROM that can be erased and reprogrammed.

HHC (Hand-Held Computer)

A generic acronym for a NORAND Hand-Held Computer, including the 4000 Series (43XX, 44XX, 4500) and the PEN*KEY or 6000 Series (61XX, 62XX, 63XX, 6400, 65XX, 66XX).

Also the trademark of another company.

Host Computer

A large computer that serves many users, such as a PC, minicomputer, or mainframe.

IBM (International Business Machines)

Developers of mainframe technologies, minicomputer technologies, cabling systems, and the IBM PC family of products.

Interleaved 2 of 5 (I 2of5 Code) *(Bar Code Symbology)*

An all numeric bar code symbology, widely used for warehouse and heavy industrial applications, such as the automobile industry.

Interleaved Bar Code

A bar code that pairs characters together, where the bars represent the first character and the interleaving characters to represent the second character, providing greater density of information with no intercharacter spaces.

IOCTL (I/O Control)

UNIX function call used to control a device.

IP (Internet Protocol)

The network layer for the TCP/IP Protocol Suite. It is a connectionless, best-effort packet switching protocol that offers a common layer over dissimilar networks.

IRQ (Interrupt Request)

A method used in PCs and other computer architectures to let a subdevice, like a serial port or network adapter, request service from a central processor.

Kermit

A popular file transfer protocol developed by Columbia University. By running in most operating environments, it provides an easy method of file transfer. Kermit is *not* the same as FTP.

LAN (Local Area Network)

A group of network devices in which each device can communicate through a wired or wireless link. The wired link may have several segments joined by repeaters and bridges. The LAN is characterized by the relatively short distance it is designed to cover, a high speed of operation, and relatively low error rates. The geographic scope of LANs is limited to thousands of feet or closely-spaced building complexes.

LCD

Liquid Crystal Display.

LF

Line Feed. Advancing the cursor or print head one line.

Loopback

A method of performing transmission tests on a circuit not requiring the assistance of personnel at the distant end. Usually involves physically connecting send lines to receive lines.

LRC (Longitudinal Redundancy Check)

A system of error detection and correction based on transmission of a block check character based on preset rules. The check character formation rule is applied in the same manner to each character on a bit by bit basis.

MS-DOS (Microsoft Disk Operating System)

A master control program for 16-bit, Intel-based system. One of the more common operating systems on PC systems.

NAK (Negative Acknowledgement)

Response to receipt of a corrupted packet of information.

PAD (Packet Assembler or Disassembler)

A protocol conversion device or program that lets devices access a packet switched network such as X.25.

PC (Personal Computer)

1. A desktop computer developed by IBM or a clone based on the same architecture developed by a third party vendor. 2. Sometimes used more generically to refer to other desktop systems, such as the Apple Macintosh. 3. The original IBM computer using an Intel 8088 CPU and an 8-bit internal bus.

RAM (Random Access Memory)

Dynamic memory, sometimes known as main memory or core.

RI

Ring Indicator.

ROM (Read Only Memory)

Contains read–only information that is protected from being overwritten, such as BIOS.

Root Subnet

The Ethernet segment to which the access point super root connects, which is the distribution LAN. For Enterprise OWL, the root subnet is the Ethernet link of the access point that originates an IP tunnel, which is the super root.

Router (SNMP)

A device that forwards traffic between networks or subnetworks. It operates at the OSI Network Layer (layer 3) or the IP layer in TCP/IP. See also: bridge, gateway, EGP, IGP.

RS-232 C (Recommended Standard 232)

An Electronic Industries Association standard interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) with serial binary data interchange.

RTS (Request To Send or Ready To Send)

A modem control signal on a standard RS-232-C connector that puts the modem in originate mode to start sending data.

RXD

Received Data.

SG

Signal ground.

SOH (Start Of Header)

A control character that identifies the beginning of the header field of a message block.

STX (Start Of Text)

A communication control character which precedes the text in the message block.

TCP/IP (Transmission Control Protocol over Internet Protocol) Suite

This is a common shorthand which refers to the suite of transport and application protocols which run over IP.

UNIX

A multiuser operating system developed by Bell Laboratories.

UPC (Universal Product Code)

A bar code symbology used throughout the grocery and retail industries.

X.25

A CCITT data communications interface specification to describe how data passes into and out of public data networks. The protocol suite defines layers 1 through 3.

INDEX

NOTE:

This index covers all topics. Page numbers in italics are figures, those in bold are tables.

NUMBERS

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W