PMD 400 Series

user's manual

PMD 400 SERIES

USER'S MANUAL

WARNING

In the application of UTICOR Technology, LP programmable control devices, you should consider them components. Therefore, provisions other than the programmable control device must be taken to protect personnel in the event of a programmable control device malfunction. Programmable control devices should not be used as stand-alone protection in any application. Unless proper safeguards are used, unwanted start-ups could result in equipment damage or personal injury. If programmable controllers are used with operator interface and like devices, this hazard should be of primary importance. The operator should be made aware of this hazard and appropriate precautions should be taken.

In addition, consideration should be given to the use of an emergency stop function that is independent of the programmable controller.

The diagrams and examples in this user's manual are included for illustrative purposes only. UTICOR Technology, LP cannot assume responsibility or liability for actual use based on the diagrams and examples.

CAUTION

Static messages that remain on the display for extended periods of time may cause uneven illumination of the dot matrix grid. This is caused by inherent properties of all vacuum fluorescent display technology. To avoid unevenness of display characters, power down unit when not in use. Do not leave the same message on the display for indefinite periods.

Brightness damage may be partially restored to normal by performing the "TEST FUNCTION — DISPLAY" option on the PMD.

UTICOR Technology, LP
AVG Automation
P.O. Box 1327
4140 Utica Ridge Road
Bettendorf, IA 52722-1327

5th printing – 08/00 Manual Number 79728-3

WARNING

In the application of UTICOR Technology, LP programmable control devices, you should consider them components. Therefore, provisions other than the programmable control device, must be taken to protect personnel in the event of a programmable control device malfunction malfunction. Programmable control devices should not be used as stand-alone protection on any application. Unless proper safeguards are used, unwanted start-ups could result in equipment damage or personal injury. If programmable controllers are used with operator interface and like devices, this hazard should be of primary importance. The operator should be made aware of this hazard and appropriate precautions should be taken.

In addition, consideration should be given to the use of an emergency stop function that is independent of the programmable controller.

The diagrams and examples in this user manual are included for illustrative purposes only. UTICOR Technology, LP cannot assume responsibility or liability for actual use based on the diagrams and examples.

CAUTION

Static messages that remain on the display for extended periods of time may cause uneven illumination of the display's dot matrix grid. This is caused by inherent properties of all vacuum fluorescent display technology. To avoid uneven character display, use the Display Saver Timeout feature described in the programming sections. Do not leave the same message on the display for indefinite periods of time.

The brightness damaged screen may be partially restored by performing the Test Functions - Display option on the PMD.

UTICOR Technology, LP
AVG Automation
P.O. Box 1327
4140 Utica Ridge Road
Bettendorf, IA 52722

Sixth printing X 08/03 Form Number 79728-3

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

SECTION 1 - INTRODUCTION

Congratulations on your decision to enhance your intra-plant communication by incorporating a UTICOR Programmable Message Display into your system. With the help of this manual, you will soon be able to implement the diverse capabilities of the PMD. What follows is a brief description of each unit in the 400 Series. The "Getting Started" and programming sections in this manual explain how to communicate with your unit and utilize the software by catering to your programming preference: online or offline. The additional sections provide reference information and include a glossary and index for quick help. Of course, if you still encounter difficulty, UTICOR backs up every one of its products with superior customer service: 319-359-7501.

Introduction

PMDs are designed to give you fast, accurate information when you need it most. They give you maximum flexibility with minimum complexity. The primary requirement for application of the PMD is to understand the system it will be incorporated into and the needs of your personnel. Because the PMD is so versatile, it is used in a wide variety of applications. And since machinery, processes, and electronic equipment vary from application to application, there are numerous ways to enter information into and extract information from your 400 Series unit. With appropriate planning, you can always be aware of the current status of your process and react immediately to unwanted situations.

UTICOR PMD products provide information from your controlled process or automated operation. Simple ON/OFF signals from your programmable controller enable a PMD to translate current conditions into plain English by selecting a pre-programmed message. This message information may be displayed on the PMD's alpha-numeric display, sent to an online printer, stored in the PMD's data log, and sent to and displayed on PMD slaves.

PMDs are programmed with any ASCII terminal or personal computer with an RS-232C interface. Programming Software (Part Number 10F50) available from UTICOR lets you program online, offline, or while residing in a network. All programming is menu- and prompt-driven so that creating messages and setting unit parameters is relatively simple.

PMDs are easy to use and have many standard features that let you set up your intra-plant communication system in a way that best accommodates your applications. They can be as simple or as complex as your operation.

This manual covers several Programmable Message Displays: PMD 400, PMD 450, PMD 460, PMD 470, PMD 475 and PMD 480.

SECTION 1 - INTRODUCTION

400 Series Features:

Messages can...

*display on masters
*display on virtually unlimited number of slaves
*display time, date and variable data
*display punctuation as well as letters
*be triggered by time of day
*scroll left or up
*blink words or characters
*log with time/date stamp
*print
*chain messages (up to 115)

PMD 400

The PMD 400 is a compact, 80-character vacuum fluorescent display with 16 configurational function keys and LEDs, and a numeric keypad for complete operator interface. The .35"-high (9.1mm) characters are arranged in four rows of 20 character locations and can be seen up to eighteen feet away. The back panel contains the connectors for interfacing to the unit.

PMD 450

The PMD 450 is essentially a PMD 400 with an optional direct interface board which allows the unit to directly interface to an Allen-Bradley PLC2, PCL3 or PLC5 through Remote I/O, Block Transfer or Data Highway/Plus. Each of these modes operates independently from the other and the 450 can be configured to communicate using any one of them. It has all of the PMD 400 capabilities, but the PMD 450 receives communication through twinaxial cable ("blue hose").

PMD 460

The PMD 460 is also very similar to the PMD 400 and contains an interface to Siemens/Texas Instruments Series 545 CPU (and the 560 and 565 CPUs used in conjunction with the TI RCC module) which have the RS-485 remote I/O module. The PMD 460 will appear as an RBC (Remote Base Controller) to the TI PLC. The PMD 460 can also listen to an existing RBC and use the information from it.

PMD 470

The PMD 470 is essentially a PMD 400 with an optional Modicon interface board which can directly interface to a Modicon S908 RIO processor. The interfacing is done through a coaxial connector on the bottom of the unit and is accomplished by the PMD 470 emulating a D908 module in the remote I/O communication system.

PMD 475

The PMD 475 is a PMD 400 that interfaces to Modicon MODBUS PLUS local area network. The 475 is a node on the network with its own network address and monitors PLC holding registers for message control and data set information.

SECTION 1 - INTRODUCTION

PMD 480

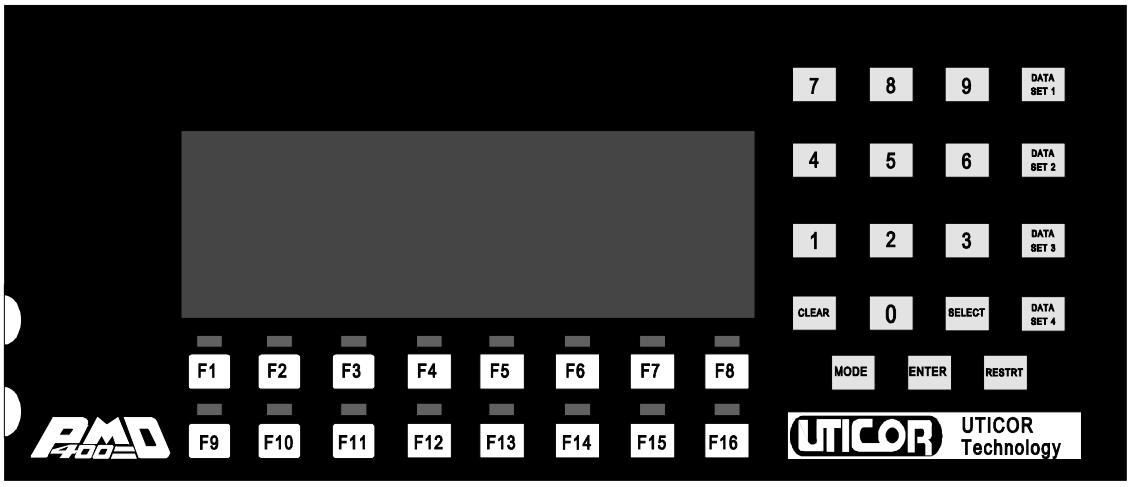
The PMD 480 has all of the PMD 400 capabilities, but the 480 contains support for a Genius Network Adapter ("GENA") board which allows the PMD to be configured as a node on the Genius I/O system. The 480 can be configured as an I/O block on a Genius I/O system and will receive data from a bus interface module. A bus interface module is typically a PLC with a Genius bus controller module or a PCIM card installed in a personal computer. The PMD 480 will exist on the Genius I/O network as an I/O block broadcasting its inputs to the bus and reading the outputs sent to it by the bus controller.

Conclusion

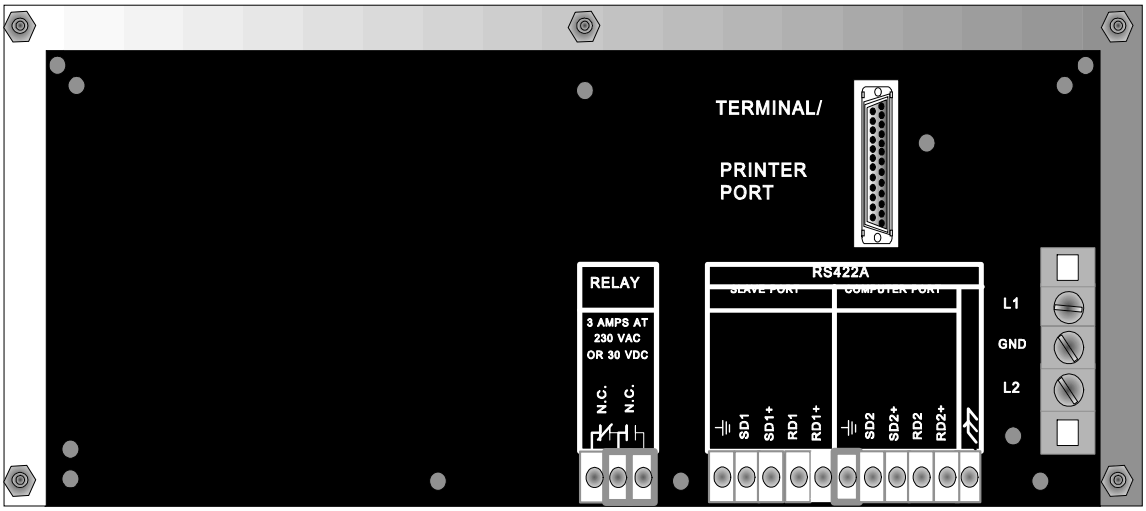
Get to know your 400 unit by browsing through this manual. Then program your unit and experiment with your messages to obtain the most effective method for displaying them. View your messages individually or in various sequences when you are programming so that unforeseen effects can be easily detected and corrected. Combine this manual information with your knowledge of your needs, your other equipment, and your imagination to make your control network successful.

SECTION 1 - INTRODUCTION

400 Series front view

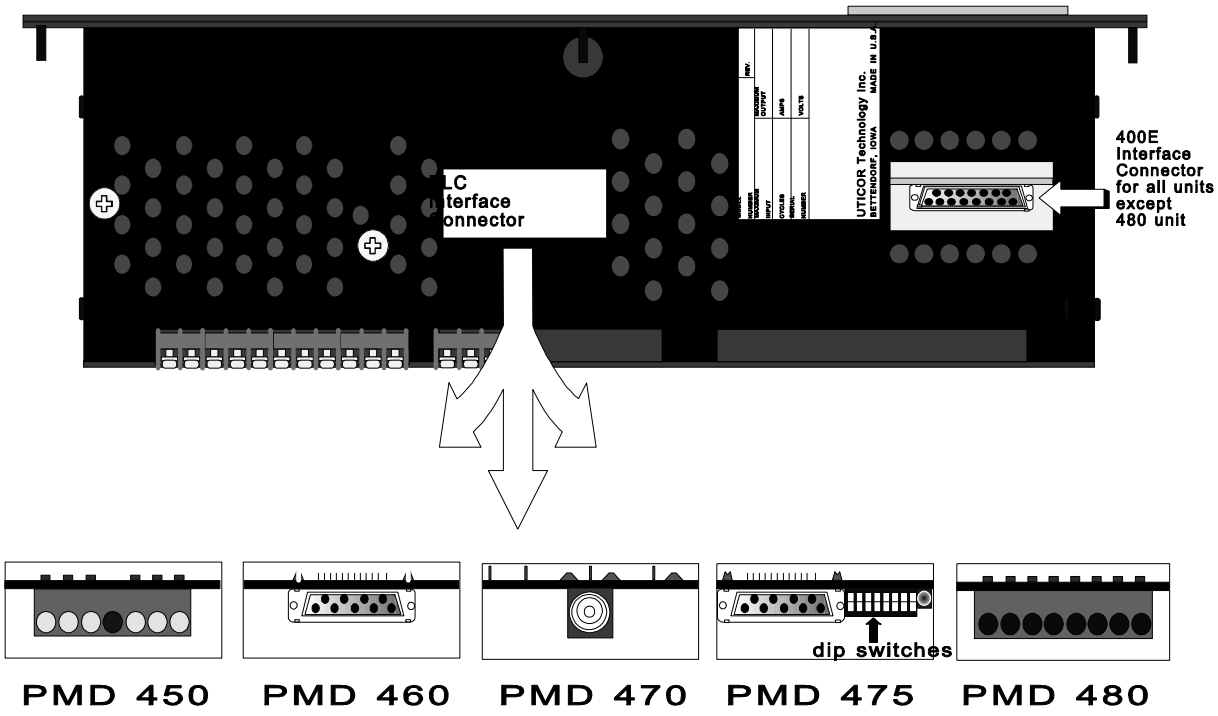


400 Series back view



SECTION 1 - INTRODUCTION

PMD 400 Series bottom view



SECTION 1 - INTRODUCTION

SECTION 2 - GETTING STARTED

Manual Conventions

!A **bold** word is a "command" or the action required to operate your 400 unit or 10F50 software.

!A gray section indicates that the information or reference enclosed within it is unique to the various 400 Series interface units.

These conventions, in addition to the strict division of offline and online programming, are intended to aid you in implementing the directions in this manual more quickly for your particular use.

Hardware Requirements

!IBM or compatible computer
!450K RAM (minimum)
!Hard disk space: 10F50 will use 1 Megabyte
!1 Megabyte of EMS or XMS memory is recommended but not necessary.
!One RS-232 serial port (COM1, COM2, COM3 or COM4)
!DOS 3.3 or later

!The CONFIG.SYS file in the computer's root directory should have the following two statements:
FILES=20 (or more)
BUFFERS=20 (or more)

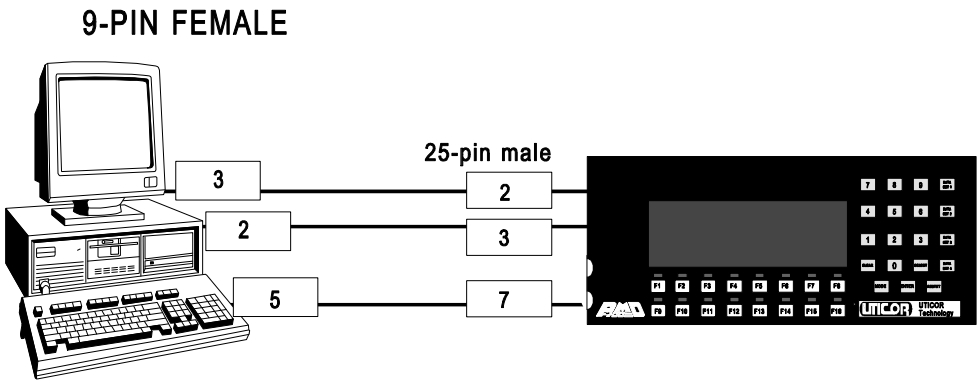
Software Requirements

!UTICOR programming software (Part Number 10F50)

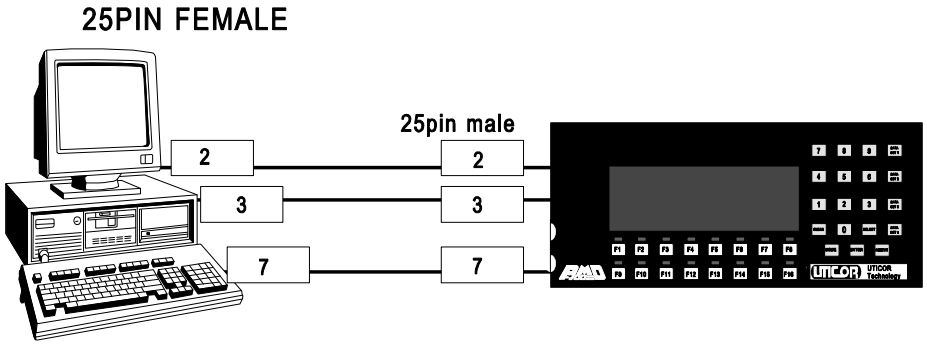
Note: Any terminal emulation software can also be used for online programming.

Interconnect Cable

!An RS-232 interconnect cable is required for proper communication. This cable will connect to the COM1, COM2, COM3 or COM4 port of the computer and the programming port of the 400 unit. UTICOR makes a universal programming cable (Part Number 43962) which accommodates both 25- and 9-pin connectors.

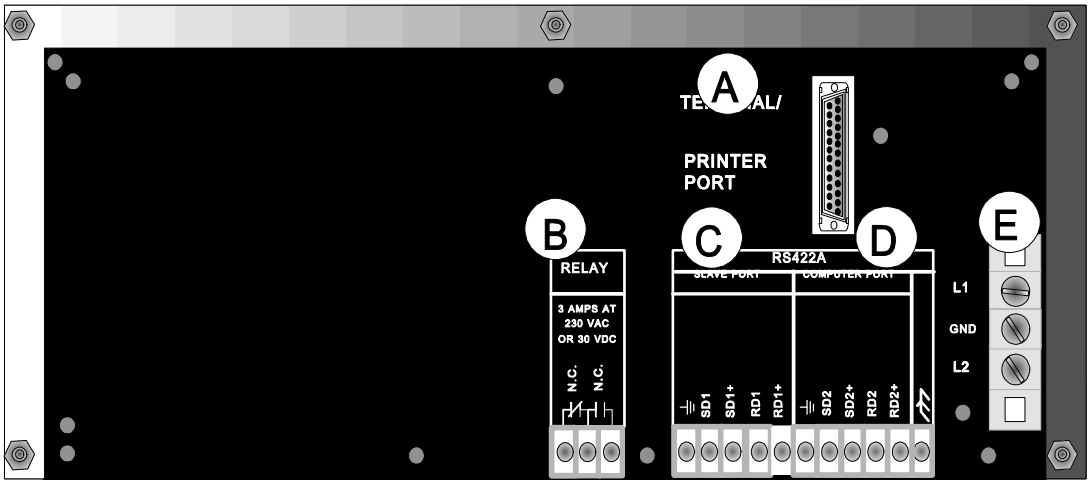

9-PIN FEMALE
25-pin male
3
2
5
2
3
7

SECTION 2 - GETTING STARTED



Note: Additional wires or jumpers in the cable may cause communication problems between the computer and the 400 Series unit.

User Connections What follows is a brief description for each connector, labeled by letters on the back panel of the 400 shown.



A. *Terminal/Printer Port*- is a program development, documentation, and storage port that connects to a program terminal, printer, or personal computer.

B. *Relay Output Terminals*- are connected to an internal Form C relay which is controlled by the message currently triggered. These terminals can be interfaced to an external device for a message-selected or timer-activated alarm.

C. *Slave Port*- is an RS-422 port used for interfacing to slave units. Up to 15 groups of 4095 slave units can be addressed from one PMD master unit.

Note that master/slave communications are uni-directional: from master to slave. For more slave information, see **APPENDIX D -ACCESSORIES**.

D. *Computer Port*- is an RS-422 port used for interfacing the PMD to a controlling computer which accepts UTICOR or ASCII protocol (main frame, mini, personal, or PLC ASCII interface unit). Computers with RS-232C ports can be adapted to RS-422 operation by using the Isolated Bidirectional Adapter, Part Number 76535.

E. *Power Input Terminal Block*- This three-position terminal block is for connecting to an external power source to power the unit. Always connect the ground terminal to the safety ground.

WARNING! GND terminals on communication ports are signal grounds and should never be attached to this safety ground.

Front Panel Hardware:

Function Keys

The 16 (or more if you also have a 400E expander module) function keys on the front of the unit are used to indicate a response to the information you observe on the display. You can map the function keys to the PMD or to the PLC if you have that option. They are always mapped to the Computer Interface. Mapping the function keys to the PMD will cause the associated LED to light up when you press the function key. Mapping them to the PLC Interface will allow the key to appear as an input that is activated momentarily (one scan). This input can be latched via the PLC program.

LEDs

The LED indicators aligned directly above each function key can be used in three different ways. The first is to indicate which function key has been pressed. This is accomplished by mapping the LEDs to the PMD. The second way is to map them so that they will be turned on and off by computer commands which come through the Computer Interface. The third way is only possible with the direct PLC interface units. With this mode of operation, the PLC determines when the LEDs are on and off.

SECTION 2 - GETTING STARTED

Numeric Keypad

The numeric keypad on the front of the unit consists of digits 0-9, DATA SETS 1, 2, 3 and 4, CLEAR and SELECT push buttons which give you the ability to enter new variable data into the unit or to a PLC if the option exists. You press the SELECT button after you enter your data into a data set so that it is entered into the unit and/or PLC. The CLEAR button is used like a clear button on a calculator; it clears the data you have entered if you make a mistake.

Notes: Only data sets that are currently in the displayed message can be manipulated by the DATA SET buttons on the front panel e.g. If data sets 1 and 2 are in your current message, you can change data sets 1 and 2, but not data sets 3 and 4. ! Only the number of digits you program for a data set value will appear in your data set i.e. If you program data set 1 with only two (of the five) digit place holders, only a two-digit number can be inserted or changed in data set 1. ! When a data set is manipulated by the DATA SET buttons on the front panel, its data will not be updated from any other source until you **hit** the SELECT button. This prevents another source (PLC, Computer Interface) from overwriting data that has been entered from the front panel until operator data entry is complete.

Push Buttons

The 400 unit also has three push buttons on the front panel which operate the unit and provide some parameter setting functions: MODE toggles the unit to the various modes of operation. When toggling the MODE push button, select ENTER when your desired mode or setting appears on line 1 or line 2 of the display.

Use ENTER to select your desired mode of operation, a baud rate setting, real-time clock settings etc. RESTART discontinues your current operation and starts the push button mode sequence again, letting you change modes of operation. After you press RESTART, the unit self-tests and then enters the DISPLAY MODE. Press MODE to change to a different mode of operation.

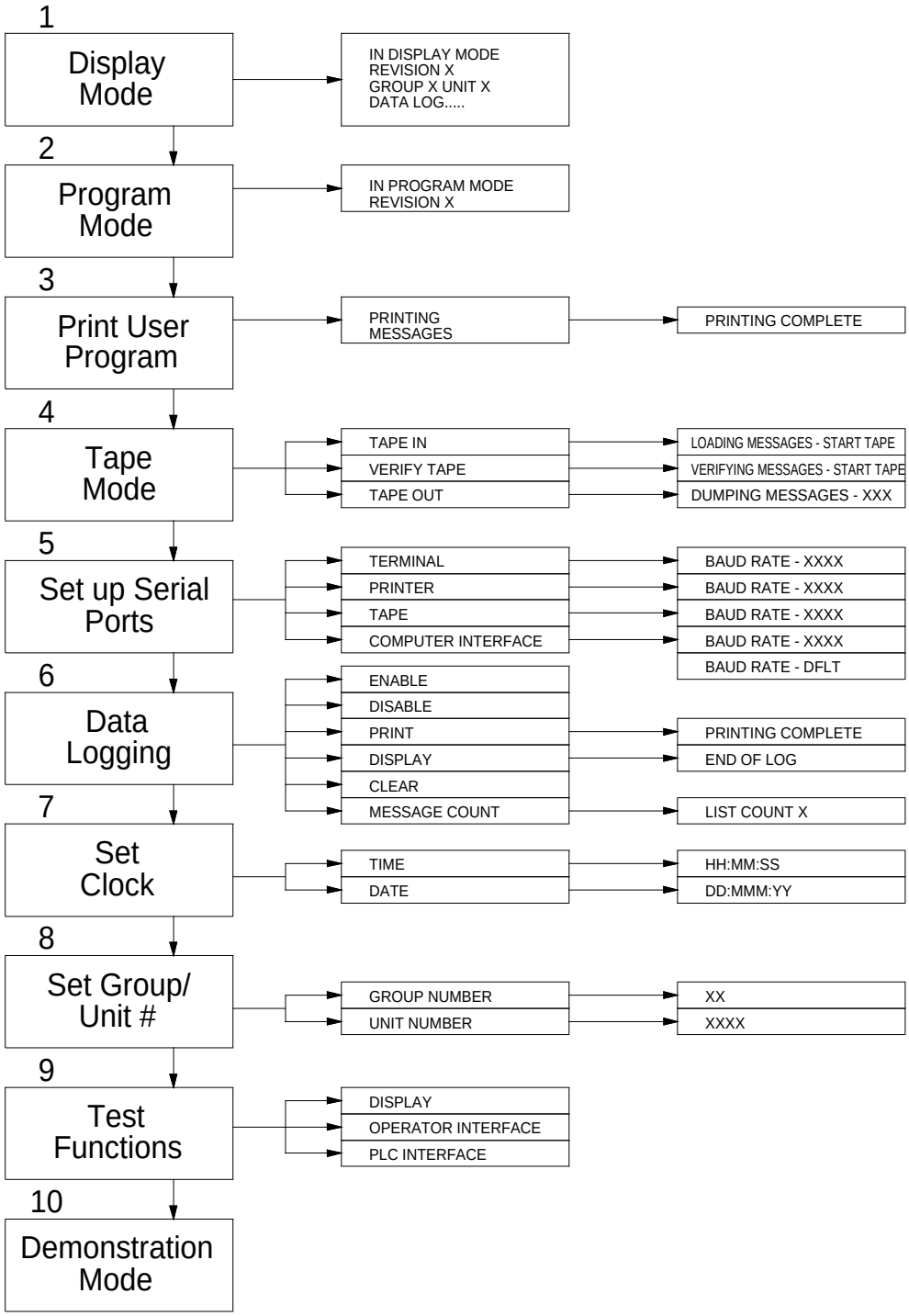
The "Front Panel Push Button Flow Chart" on the following page illustrates all of the modes. Brief descriptions of each follow.

Note: An internal jumper and your wiring configuration allow you to enable or disable operation of the front panel push buttons and/or the inputs for safety purposes. See **APPENDIX E - INTERNAL ADJUSTMENTS** to do this.

SECTION 2 - GETTING STARTED

FRONT PANEL PUSHBUTTON FLOWCHART

MODE, ENTER, RESTART



SECTION 2 - GETTING STARTED

Push Button Mode Definitions: The following numbers explain their corresponding number on the previous page: "Front Panel Push Button Flow Chart." You might find it easier to perform these operations through the online programming section.

1. Display Mode This is the normal operating mode which makes the unit ready to accept communication through the RS-422A *Computer Port* or the *PLC Interface Port*.

2. Program Mode This is used to program the unit with a dumb ASCII terminal or computer with 10F50 in **`Terminal Emulator`** mode.

3. Print User Program This prints out the programmed memory of the unit.

4. Tape Mode This allows message files to be uploaded from your computer to the unit or downloaded from the unit to your computer and verified.

5. Set up Serial Ports This sets the baud rate for the RS-232 *Terminal/Printer Port* (Terminal, Printer or Tape) and the *RS-422A Computer Port* (Computer Interface). Baud rate settings for these functions are 300, 600, 1200, 2400, 4800 and 9600. When the baud rate for the PROGRAM MODE is set externally, the unit is set for eight data bits, no parity and one stop bit. The "DFLT" option will set the selected port to the settings that are the defaults when memory is cleared. This lets you return any port to a known setting using only the push button menu.

6. Data Logging This turns the internal data logging function on and off, prints, displays, or clears data log memory. It also displays message count.

7. Set Clock This sets the clock's time and date. Time (shown in 24-hour format) and date both have three fields. The current field to set will blink. **Press** MODE to change the setting; **press** ENTER to select the setting shown.

8. Set Group/ Unit # This sets group number (0-15) and unit number (0-4095) for the *Computer Interface Port*. Enter your selection digit-by-digit by pressing MODE and ENTER. The PMD does not accept numbers larger than the maximum allowed.

WARNING! Do NOT use the same group/unit address for PMDs that are connected serially (using bidirectional communication through the Computer Port). Computer interface communication can cause internal damage to displays that are set to the same group/unit address.

SECTION 2 - GETTING STARTED

9. Test Functions

This provides two test functions:
1. *Display* tests the dots in the matrix of each character location of the display. It will also put slave displays into the display test mode.

For PLC Interface units, the test functions check the *PLC Port* and the dot matrix on the vacuum fluorescent display.

Note: After you power up the unit and set it to test the display, it will do so every time you press RESTART and automatically re-enter the DISPLAY MODE. To disable this process, **hit** ENTER twice when the unit displays DISPLAY MODE... but before it starts the test function.

2. *Operator Interface* scrolls through the LEDs, illuminating them one at a time. If you press a function key, the display will report which key you have pressed e.g. **`Function Key xx`** .

10. Demonstration Mode

This is essentially an illustration tool. When a function key is pressed, the associated LED lights up and the associated message displays e.g. F2 = LED 2 = Message 2.

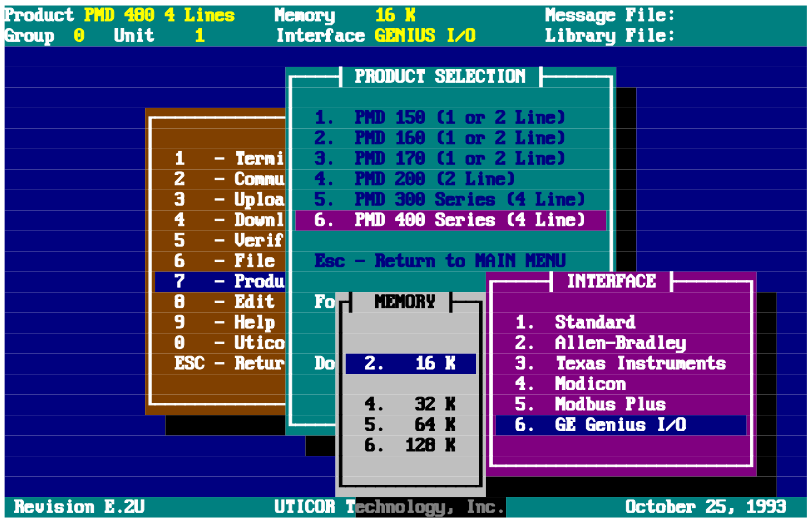
UTICOR Software 10F50

The following two sections are brief descriptions on how to get started with UTICOR software for both online and offline users. Please refer to the programming sections, **"ONLINE PROGRAMMING"** or **"OFFLINE PROGRAMMING"** to proceed with your method of programming.

Getting Started with Online Programming

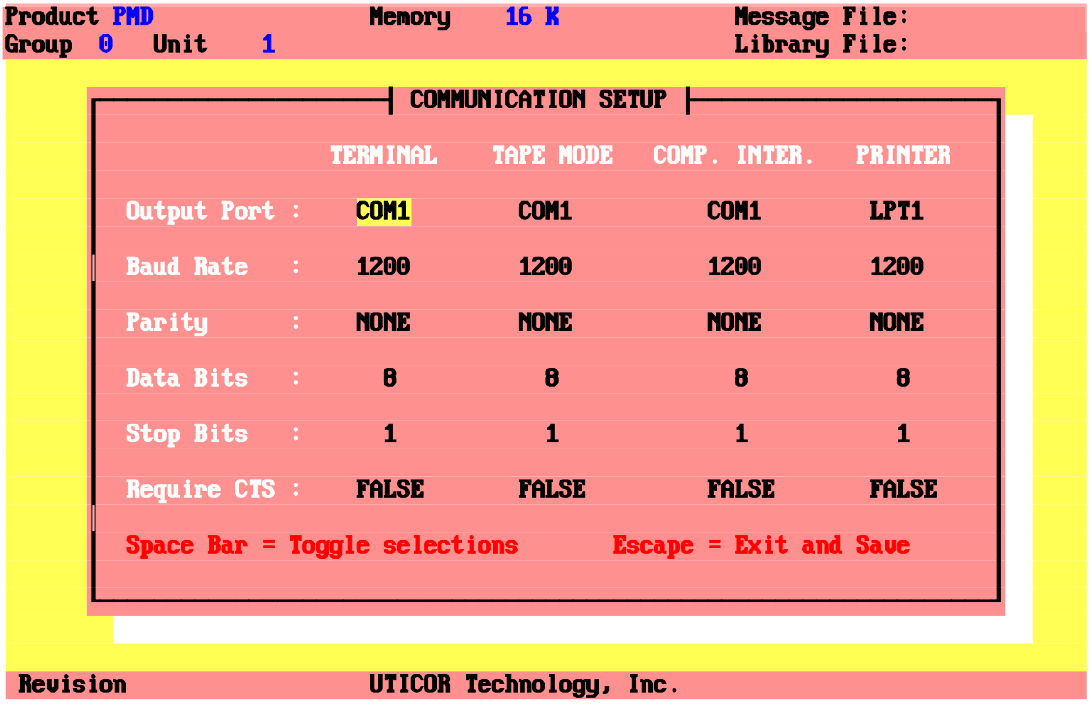
1. **Connect** your RS-232 cable to the *serial port* (COM1, COM2, COM3, or COM4) and to the *Terminal/Printer Port* on the PMD.
2. **Install** 10F50 to your hard disk.
3. **Select** the drive and path where the 10F50 has been installed.
4. For color monitor, **type**: PMD
   For monochrome monitor, **type**: PMD -MON
   For LCD monitor, **type**: PMD -LCD
   Type PMDP if using a PMD 450 with pass-through programming.
5. **Select** your UTICOR display by selecting **`Product Selection`** from the **`MAIN MENU: PMD 400`**.
6. **Select** a memory option: **`16K, 32K, 64K, 128K`**.

SECTION 2 - GETTING STARTED



7. **Select** your interface: **`Standard, Allen-Bradley, Texas Instruments, Modicon, Modbus Plus, GE Genius I/O`**.

8. **Select `Communication Setup`** from the **`MAIN MENU`**.



9. **Select** the output port and baud rate for the terminal/computer. This baud rate MUST match the baud rate setting of the PMD. All PMDs shipped from the factory are set to 1200 baud. You can change the terminal mode baud rate of the 400 unit by **selecting `SETUP SERIAL PORTS`** on it and using the MODE and ENTER push buttons on the front panel.

10. **Select**: **`Parity = none; Data bits = 8; Stop bits = 1`**.

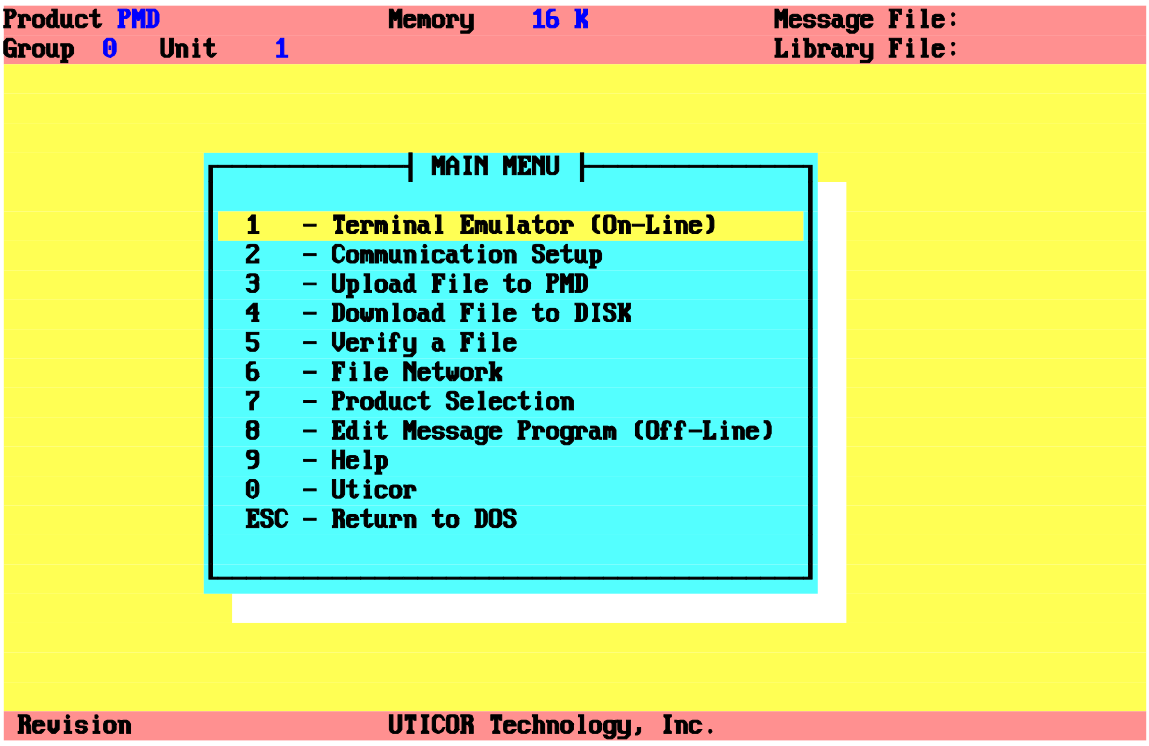
11. The **`CTS`** (Clear to Send) option is not required by UTICOR but if you want to use handshaking, toggle the option to **`TRUE`**.

12. **Select** Esc to exit **`COMMUNICATION SETUP`** and save changes.

13. **Put** the 400 Series unit into the PROGRAM MODE by using the MODE and ENTER buttons.

14. **Select `Terminal Emulator`** from the **`MAIN MENU`** on the computer.

Note: All changes are stored in the PMD **only**; nothing is stored in the computer. Online programming puts your computer in a dumb ASCII terminal mode.



15. **Press** Enter on the computer, then H to list the PMD's **COMMANDS**.

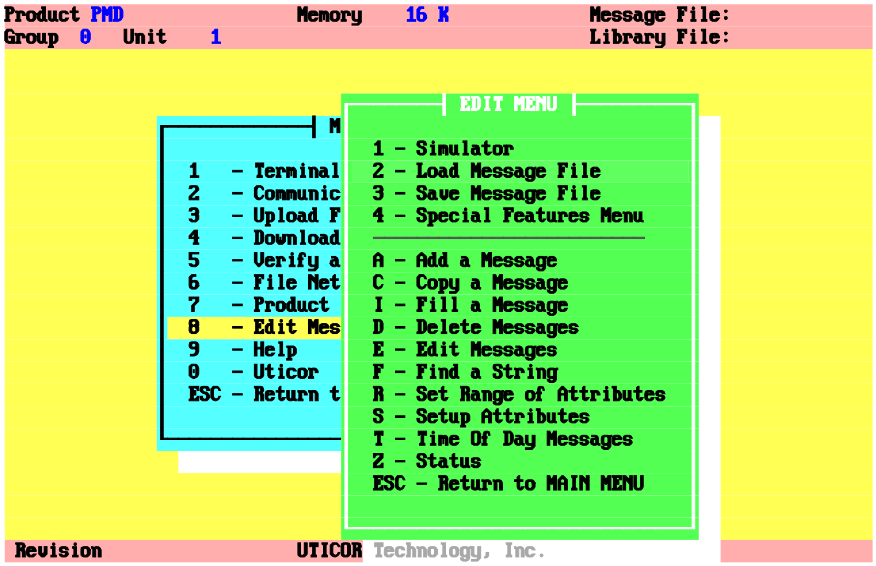
```
COMMANDS ARE:
A          ADD A MESSAGE
C          COPY MESSAGES
D          DELETE MESSAGES
E          EDIT A MESSAGE
F          FIND A STRING
H          LIST VALID COMMANDS
I          INITIALIZE PARAMETERS TO DEFAULTS
           AND CLEAR ALL MESSAGES
L          LIST MESSAGES
N          DISPLAY NUMBER OF FREE BYTES
S          SET UP PARAMETERS
V          VIEW MESSAGES
<CR>       LIST NEXT MESSAGE
ESC        ABORT COMMAND
```

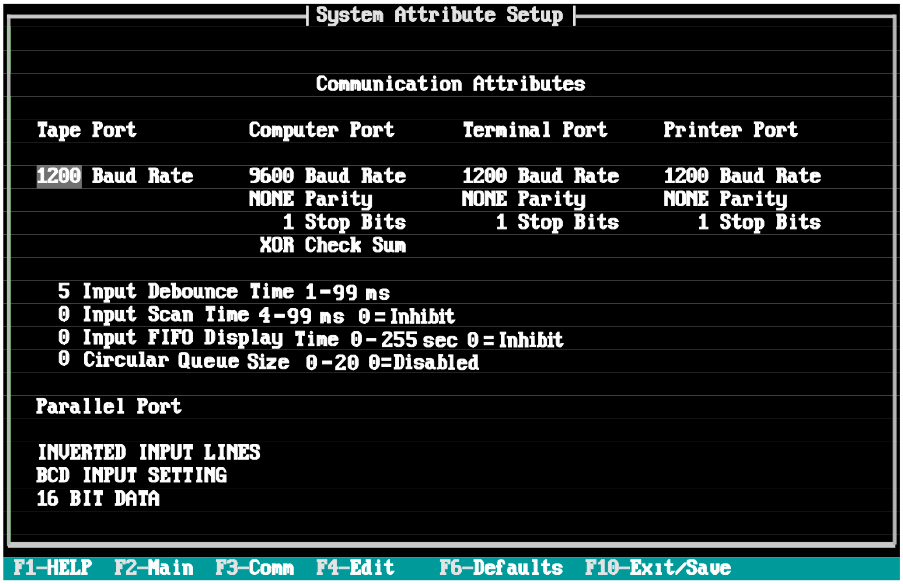
You should now be communicating with your unit. If communication has not been established, check your cable connections, proper COM port on the computer, and verify that the baud rate setting on the computer (10F50) matches that of the PMD. Now go to **SECTION 3 - ONLINE PROGRAMMING** to proceed.

SECTION 2 - GETTING STARTED

Getting Started with Offline Programming

1. Follow steps 1-12 in the **Online Programming** section above.
2. After you **select** Esc, you will be returned to the **`MAIN MENU`** screen.
3. **Select `Edit Message Program`** and then **`Setup Attributes`**.





4. **Press** F3 - Comm and match the tape and terminal port baud rate settings to those that you set in step 9.

WARNING! Baud rates are stored with the message file.

5. **Press** F10 to save the changes and exit. The **`Edit Menu`** will be returned to your screen and you are ready to start programming offline. Now go to **SECTION 4 - OFFLINE PROGRAMMING** to proceed.

SECTION 2 - GETTING STARTED

SECTION 3 - ONLINE PROGRAMMING

In this section you will learn how to use each option on the **MAIN> COMMAND** menu, the key to programming online. For your convenience, the **ONLINE PROGRAMMING** section is divided into three parts: **A. DEVELOPING A MESSAGE FILE B. MESSAGE ATTRIBUTES C. WORKING WITH MESSAGE FILES.** Part **A** takes you through the basics of online programming and gives you the opportunity to get hands-on familiarity with the product right away as you start to develop a message file. Part **B** is an extension of Part **A**, as it helps you assign message attributes and add versatility to your message files. Part **C** leads you through ways to utilize the message files you have developed.

As you have already been introduced to some of the options on the **MAIN MENU** (**Communication Setup, Product Selection**), Parts **A** and **B** guide you through **Terminal Emulator** and part **C** explains **Upload file to PMD, Download file to DISK, Verify a File, File Network**. (**Edit Message Program** is primarily for offline programming).

To continue from the **GETTING STARTED** section, you have selected **Terminal Emulator** from the **MAIN MENU**, your 400 unit is in **PROGRAM MODE** and you have pressed H to see the menu below. This is your "Main Menu" for online programming. Instructions for each option follow. Of course, remember that you always have help function keys at the bottom of your terminal screen. When you complete **SECTION 3**, you will know how to program online and put your unit to work for you!

```
COMMANDS ARE:
A       ADD A MESSAGE
C       COPY MESSAGES
D       DELETE MESSAGES
E       EDIT A MESSAGE
F       FIND A STRING
H       LIST VALID COMMANDS
I       INITIALIZE PARAMETERS TO DEFAULTS
        AND CLEAR ALL MESSAGES
L       LIST MESSAGES
N       DISPLAY NUMBER OF FREE BYTES
S       SET UP PARAMETERS
V       VIEW MESSAGES
<CR>    LIST NEXT MESSAGE
ESC     ABORT COMMAND
```

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

Add a Message

Use this command to program new messages. **Press** A to add a message. The terminal responds:

ADD MESSAGE #

!**Enter** a number between 0 and 65534 (you designate each of your messages with a number) and **hit** Enter. The terminal responds with the current message option defaults, followed with:

REPLACE MESSAGE CONTROL OPTIONS (Y/N) ?

!For now, **enter** N to leave the control options as they are. [Entering Y will open the **OPTION MENU**, which is where you can change message options. This is explained in Part **B**.]

!When the terminal displays @, you may enter message text. When the twentieth character is entered on one message line, the program terminal will sound a bell or beep to indicate that a 400 Series standard line has been programmed, and you should move to the next line to complete your message. (You may enter up to 80 characters if you are sending messages to the larger slaves or your message scrolls left). **Press** Enter to advance to the next line and continue entering text, or **press** Ctrl Z to end the message.

!The terminal now asks:

ADD MESSAGE #xx (Y/N) <Y> ?

xx is one number larger than the message you just programmed. Continue the process or **select** N to stop adding messages.

The **ADD MESSAGE #** loop continues until:

1. The N response to **ADD MESSAGE #** returns you to **MAIN>** and you **press** H to return to the **MAIN> COMMAND** menu.

2. The next message number is already programmed and you see **MESSAGE #xx ALREADY EXISTS**. You are returned to the **MAIN> COMMAND** menu.

Copy Messages

Use this command to copy one existing message to other message number location(s). **Press** C to copy a message. The terminal responds:

COPY MESSAGE NUMBER >

!**Enter** a message number and **hit** Enter. The terminal responds:

TO
STARTING AT MESSAGE NUMBER >

!**Enter** the first message number location at which you wish to have this message copied and **press** Enter. The terminal responds:

ENDING AT MESSAGE NUMBER >

!**Enter** the ending message number in the range of message locations to copy to and **press** Enter. If you only want to copy the message into this one location, make the **STARTING AT MESSAGE #** and **ENDING AT MESSAGE #** the same number.

!The terminal responds:

COPY DONE

x MESSAGE(S) COPIED
TO MESSAGE NUMBER xx
THROUGH MESSAGE NUMBER xx

You are returned to **MAIN>**.

Note: When programmed messages exist within the "copy to" location specified, no messages are copied. The terminal will say:

x MESSAGES IN COPY RANGE

Delete Messages

Use this command to delete one or more messages. **Press** D to delete messages. The terminal responds:

DELETE
STARTING RANGE (<CR> = LAST):

!**Enter** a number and **press** Enter. The terminal responds:

ENDING RANGE (<CR> = LAST):

!Here you have three options: 1. Enter the same number to delete only one message. 2. Enter a higher number to delete a range of messages (deletes the first and last messages and all messages in between). 3. **Press** Enter to delete all messages from the message number entered at the first prompt to the end of the entire message program. The terminal will return a third prompt depending on which option you choose:

DELETE MSG xx (Y/N)?(for option 1)
or **DELETE FROM MSG xx TO MSG xx (Y/N)?** (for options 2 or 3)

!**Enter** Y to confirm deletion, or N to cancel. You will be returned to **MAIN>**.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

Note: The programming terminal does not accept non-digits or numbers greater than the largest programmed message.

Edit a Message

Use this command to edit an existing message. **Press** E to edit a message. The terminal responds:

`EDIT MSG #`

!**Enter** the message number and **press** Enter. The terminal responds with the programmed message options, followed by:

`REPLACE MESSAGE CONTROL OPTIONS (Y/N)?`

!For now, **enter** N to leave the options as they are. [Entering Y will open the **`OPTION MENU`**, which is where you can change message options. This is explained in Part **B**.] The terminal displays the first line of the message with the prompt:

`REPLACE LINE (Y/N) ?`

!**Enter** N to leave the line of text as it is, or **enter** Y to replace it. If you choose Y, re-enter the entire line and **press** Enter. This process continues for each line of message text. Additional lines of text can be added to the end of messages. Terminate the editing of messages with Ctrl Z. You are returned to **`MAIN>`**.

Find a String

Use this to find messages containing a particular character string. **Enter** F; **enter** the word to be searched for; then **press** Enter. The 400 unit searches your program from the beginning. When the string is found, the terminal displays the message number and the message. It also displays:

`FIND>`

!**Press** Enter only to find the next message containing the string. Repeat until the terminal responds:

`WAS NOT FOUND`

!**Press** Esc to end **`FIND A STRING`** before receiving the **`WAS NOT FOUND`** message. You are returned to **`MAIN>`**.

Notes: **Find a String** is upper- and lower-case sensitive. This procedure may take a few minutes when the message program is large.

List Valid Commands

Use this to list the **`MAIN> COMMANDS`** menu. **Press** H to see the menu.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

Initialize Parameters to Defaults and Clear all Messages

Use this to clear your message program in the 400 unit and initialize parameters to defaults (See **APPENDIX B - DEFAULT SETTINGS**). **Press** I. The terminal asks you to confirm this:

```
CLEARING ALL MESSAGES
ARE YOU SURE?
```

!**Press** Y to confirm. **Enter** N to abort this command. You are returned to **MAIN>**.

Note: This command resets the Unit Address to Group #0 Unit #0. It also initializes the data log to the default size (1000), clears the data log, time-and date-triggered message memory.

List Messages

Use this to list one or more messages on the terminal screen. **Press** L. The terminal responds:

```
STARTING RANGE (<CR> = LAST):
```

!Here you have two options: 1. **Hit** Enter to list the highest programmed message. 2. **Enter** a number and **hit** Enter. You see the prompt:

```
ENDING RANGE (<CR> = LAST):
```

!Here you have three options: 1. **Enter** the same number and **hit** Enter to list only one message. 2. **Enter** a higher number and **hit** Enter to list all programmed messages within that range. 3. **Hit** Enter only to list all messages from the message number indicated to the end of the program. The terminal will display the message number, options, and the text for all message numbers specified.

!**Press** Ctrl S to stop scrolling; **press** Ctrl Q to resume scrolling; **press** Esc to abort listing. You are returned to **MAIN>**.

Note: The programming terminal does not accept non-digits or numbers greater than the largest programmed message.

Display Number of Free Bytes

Use this to display message program information on the terminal screen. **Press** N. The terminal might respond with this example:

```
LARGEST MESSAGE NUMBER:  103
TOTAL NUMBER OF MESSAGES:  84
NUMBER OF FREE BYTES:  12031
NUMBER DELETED MESSAGES:  4
```

You are returned to **MAIN>**.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

Set up Parameters As this command has a long sub-menu of its own, its examples and instructions are found at the end of Part **A**.

View Messages on Display Use this command to view a programmed message on the PMD. **Press** V. The terminal responds:

```
VIEW>
```

!**Enter** a message number and **press** Enter to view that message on the display.

Notes: Messages programmed to be sent to slaves will do so in the VIEW MODE. ! No messages are stored in the data log while in the VIEW MODE. ! Messages programmed to be sent to the *RS-232 Port* will be displayed on the terminal screen provided your printer baud rate is the same as the terminal baud rate. ! Variable data digits are represented as numbers (1-5) in the VIEW MODE.

Esc Use this to abort a command and return to **`MAIN>`**. Esc does not abort **`Delete, Initialize Parameters to Defaults and Clear All Messages,`** or **`Memory Tests`**, once they have been initiated.

Set up Parameters (resumed) Use this to see the **`SETUP COMMANDS`** sub-menu and implement its commands. **Press** S to put the 400 unit into this mode. It will display:

```
SETUP  PARAMETERS
REVISION x
```

while it's in this mode. The terminal displays the menu on the following page. Instructions for each option follow.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

```
SET UP COMMANDS:
1) BLINK ON AND OFF DELIMITERS
2) NUMBER OF NULLS PRINTED AFTER A <CR>
3) SERIAL PORTS
4) CLOCK
5) PARALLEL PORT*
6) RATE FOR SCROLL AND CHAIN MESSAGE
7) BLINK ON AND OFF RATES
8) DEBOUNCE TIME
9) PRINT TIME AND DATE FOR MESSAGES
10)DATA LOGGING
11)RELAY CONTROL
12)GROUP AND UNIT NUMBER
13)MEMORY TESTS
14)DEFINE CLEAR FIFO MESSAGE
15)PROGRAM TIME OF DAY MESSAGES
16)PROGRAM DISPLAY SAVER TIMEOUT
17)FUNCTION KEYS/LED CONTROL
<CR>PRINT THIS LIST

ENTER VALUE FOLLOWED BY <CR>
```

* Will reflect PLC protocol

Blink On and Off Delimiters

Use this command to change character definition for the **`Blink On`** and **`Blink Off`** delimiters. The PMD defaults to [for the **`Blink On`** delimiter and] for the **`Blink Off`** delimiter. If these characters are needed for display in a message, however, either or both can be exchanged for other characters. When they are changed, all usages of the blink characters in the program will automatically be changed to the new characters. **Press** 1 and Enter to select new **`BLINK ON AND OFF DELIMITERS`**. The terminal responds:

```
BLINK ON AND BLINK OFF DELIMITERS
CANNOT BE THE SAME CHARACTERS.
ENTER BLINK ON CHARACTER <[>?
```

!**Enter** your new **`Blink On`** delimiter and **press** Enter, or Enter only to leave it in its current state, shown within the brackets. The terminal responds:

```
ENTER BLINK OFF CHARACTER <]>?
```

!Respond to this prompt in the same manner. You are returned to **MAIN>**.

!To program a message in which some, but not all, characters blink, use blink delimiters to toggle blink mode on and off. **Enter** the **Blink On** delimiter before the first character to blink; **enter** the **Blink Off** delimiter after the last character of blinking text. Blink delimiters have no effect on time, date, or variable data sets of messages. The **Blink Entire Message** option must be enabled to blink these items (see Part **B - MESSAGE ATTRIBUTES**).

Note: Delimiters can be any printable characters except control characters and the ^ character. Characters used as delimiters cannot be used for any other purpose.

Number of Nulls Printed After a <CR>

Use this command to program a specific number of null characters to be sent to the *RS-232 port* after each <CR>. **Press** 2 and Enter to program your number. The terminal responds:

```
ENTER NUMBER OF NULLS <0>?
```

!**Enter** a number between 0 and 255 and **press** Enter, or Enter only to leave it in its current state, shown within the brackets. The terminal does not accept non-digits or a number higher than 255. You are returned to **MAIN>**.

Note: Some printers require a delay time after receiving a carriage return to prevent loss of characters at the beginning of lines. This command allows the creation of a delay (after <CR>) which may be adjusted to the required value.

Serial Ports

Use this command to set up the *Computer Port* and/or the *Terminal/Printer Port* and *Tape Port*. (This process can also be completed using the push buttons on the front of the unit and selecting **SET UP SERIAL PORTS**). **Press** 3 and Enter. The terminal responds with the sub-menu:

```
SELECT WHICH MODE TO SETUP
1) COMPUTER INTERFACE
2) PROGRAM TERMINAL
3) PRINTER
4) TAPE IN AND OUT
?
```

!**Enter** an item (1-4) to set up the port for that mode. The following prompts must be answered:

```
COMPUTER INTERFACE MODE (0=PMD, 1=ASCII) <0>
BAUD RATE (1=300, 2=600, 3=1200, 4=2400
          5=4800, 6=9600) <9600>?
STOP BITS (SELECT 1 OR 2) <1>?
PARITY ENABLED (SELECT Y OR N) <N>?
PARITY MODE (0=ODD, 1=EVEN) <1>?
CHECKSUM TYPE (0=CRC, 1=EOR, 2=NONE) <1>?
```

!**Enter** the proper letter or digit to select each item, or Enter to retain the current setting. Note than in some cases, parameters are fixed and won't be included in prompts:

Computer Interface:	Uses all prompts.
Program Terminal:	No Computer Interface or Checksum Type.
Printer:	No Computer Interface or Checksum Type.
Tape In and Out:	Select Baud Rate Only.

Notes: Use Esc to exit the **`Serial Ports`** sub-menu. ! The Parity Mode prompt is not displayed when the parity is not enabled. Parity Enabled = 7 Bit Word Length -- (RS-232 port only!!)/ Parity Disabled = 8 Bit Word Length -- (RS-232 port only!!) ! The setting of the PROGRAM TERMINAL port does not take effect until the unit is restarted.

Clock

Use this command to set time and date or check the current settings. **Press** 4 and Enter. The terminal responds with the sub-menu:

```
SELECT CLOCK
     DATE:  23-NOV-91
     TIME:  2:23:35 PM
  1) SETUP DATE
  2) SETUP TIME
  ?
```

!Select an item (1 or 2) for desired option. Option **`1) SETUP DATE`** displays the following prompts:

```
MONTH (SELECT 1 TO 12) <11>?
DAY OF MONTH (SELECT 1 TO 31) <23>?
YEAR (SELECT 0 TO 99) <91>?
```

!Option **`2) SETUP TIME`** displays the following prompts:

```
HOUR FORMAT (SELECT 1=12 HR, 2=24 HR) <12>?
HOURS (SELECT 1 TO 12) <2>? *
```

```
HOURS (SELECT 1=AM OR 2=PM) <PM>? *
HOURS (SELECT 0 TO 23) <14>? *
MINUTES (SELECT 0 TO 59) <23>?
SECONDS (SELECT 0 TO 59) <35>?
```

*Prompts returned depend on response to **Hour Format** prompt.

Notes: Time is not updated on the terminal screen. Time is updated only when the **Clock** sub-menu is re-displayed. **Press** Esc to exit the **Clock** sub-menu and return to **MAIN>**.

Parallel Port
AB
TI
MODICON
MODBUS PLUS
GE GENIUS I/O

Although **Parallel Port** appears as a menu option, it is not applicable to operation of your 400. If you have a 450, 460, 470, 475 or 480 unit, however, **press** 5 and Enter. Refer to the following parts of **SECTION 7 - PLC INTERFACING** for information about your particular unit:

PART A: 450 A-B INTERFACE
PART B: 460 TI INTERFACE
PART C: 470 MODICON INTERFACE
PART D: 475 MODICON MODBUS PLUS INTERFACE
PART E: 480 GE GENIUS I/O

Rate for Scroll and Chain Message

Use this command to set up default time rates for chained and scrolling messages. **Press** 6 and Enter. The terminal responds with the sub-menu:

```
SETUP RATE FOR SCROLL AND CHAIN MESSAGE
1) DEFAULT TIME RATE FOR CHAIN MESSAGES <100>
2) DEFAULT TIME RATE FOR SCROLL UP <5>
3) DEFAULT TIME RATE FOR SCROLL LEFT <5>
?
```

!**Select** an option (1-3). The terminal displays:

```
ENTER TIME IN TENTHS OF SECONDS <5>?
```

!For option **1) DEFAULT TIME RATE FOR CHAIN MESSAGES**, **enter** a number between 1 (0.1 second) and 255 (25.5 seconds) and **hit** Enter. This is the amount of time each message within a chained message will remain on the display. If a scrolling message is in the chain, however, it will display until it is done scrolling.

!For options **2) DEFAULT TIME RATE FOR SCROLL UP** and **3) DEFAULT TIME RATE FOR SCROLL LEFT**, enter a number between 1

(0.1 second) and 99 (9.9 seconds) and **press** Enter.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

This is the amount of time each message line will remain on each line of the display (option 2) or the amount of time each character will remain at each character location (option 3). The terminal does not accept non-digits or numbers above the range specified. You are returned to **MAIN>**.

Blink On and Off Rate

Use this command to select the rate at which blinking characters blink on and off. **Press** 7 and Enter. The terminal responds:

```
CHANGE BLINK ON AND OFF RATES
BLINK ON TIME IN TENTHS OF SECONDS <5>?
BLINK OFF TIME IN TENTHS OF SECONDS <5>?
```

!**Enter** a number between 1 (0.1 second) and 99 (9.9 seconds) and **press** Enter for each prompt, or just Enter to leave them at the current setting shown within brackets. The terminal does not accept non-digits or numbers above the specified range.

Debounce Time

Although **Debounce Time** appears as a menu option, it is not applicable to operation of your 400 Series unit; it applies only to products with *Parallel Ports*.

Print Time and Date for Messages

Use this command to enable/disable printing time and date before messages. **Press** 9 and Enter. The terminal responds:

```
PRINT TIME AND DATE BEFORE EACH
MESSAGE SENT TO RS-232 PORT
(1=ENABLE, 2=DISABLE) <2>?
```

!**Enter** 1 to have the time and date sent to the printer before each message is sent to the *Terminal/Printer Port*. **Enter** 2 to disable this option. You are returned to **MAIN>**.

Data Logging

Use this command to enable/disable the data log or clear the data log buffer. **Press** 10 and Enter. The terminal responds with the sub-menu:

```
SETUP>10
DATA LOGGING <DISABLED>
1) ENABLE
2) DISABLE
3) CLEAR
4) SIZE OF LOG<1000>
?
```

!**Select** an option (1-4) for the desired function. The current state of options 1 and 2 is shown within brackets (**DATA LOGGING <ENABLED/ DISABLED>**). If you select option **1) ENABLE**, logged messages will be stored in the data log battery-backed RAM memory when selected. If you select option **2) DISABLE**, no messages will be stored in the data log memory when selected. If you select option **3) CLEAR**, messages currently stored in the data log memory will be cleared from it. Option **4) SIZE OF LOG** determines the number of entries (message numbers and variable data values) that the data log will hold. When enabled, the data log will store this number of readings and discard the oldest records. When you select this option, the terminal responds:

!!! WARNING - WHEN SIZE OF LOG CHANGED THE LOG IS CLEARED !!! SIZE OF LOG IN NUMBER OF ENTRIES <0>

!**Press** Esc to save data log contents or enter a number from 0 to 3071 to set log size. Entries larger than 3071 prompts the message:

!!! ERROR - INSUFFICIENT MEMORY !!!

!When this message is displayed, log size is not altered, log memory is not cleared, and you are returned to **MAIN>**.

Notes: Clearing data log memory does not clear time- and date-triggered message memory. This procedure may also be done externally by manipulating the MODE and ENTER push buttons on the PMD.

Printing Logged Messages

This procedure is not a part of the menu but is used to print the messages stored in the 400 unit's data logger as well as their variable data and the time and date that they were selected. Make sure the baud rate, stop bits and parity mode on the 400 unit match those of the printer. **Connect** the printer to the 400 Series unit at the *Terminal/Printer Port*. **Set** the 400 to DATA LOGGING using the MODE and ENTER push buttons on its front panel, and then **select** Print.

!The 400 will display **PRINTING DATA LOG** and the printer will print the current time and date before printing the logged messages. When the data logging memory is finished printing, the unit displays **PRINTING COMPLETE - PRESS RESTART**.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

Relay (Output Terminals) Control

Use this command to determine the length of time that the *Relay Output Terminals* will be energized when activated by a selected message, or to set the internal timer for the relay output terminals. **Press** 11 and Enter. The terminal responds:

```
ALARM RELAY DURATION (0-255 sec) <0>?
```

!**Enter** 0 to have the *Relay Output Terminals* energized until a new message is selected. **Enter** a number between 1 and 255 for the number of seconds that you wish to have the *Relay Output Terminals* energized each time that a message programmed to energize the relay output is selected. Now the terminal lets you set or change the clock setting for a timed *Relay Output Terminal* energization. The terminal responds:

```
TIME OF DAY RELAY
  RELAY ON:  00:00:00
  RELAY OFF:  00:00:00
  (1=ENABLE, 2=DISABLE) <1>?
```

!The numbers shown on lines 1 and 2 are the current clock setting for the timed relay output. These settings are in the form of HH:MM:SS. If you **enter** 2 to disable the timed alarm, you are returned to **`MAIN>`**. If you **enter** 1 to enable the timed alarm, you will receive the prompt:

```
SETUP ALARM RELAY ON TIME (Y/N)?
```

!This prompt lets you set the time of day to energize the *Relay Output Terminals*. **Enter** N to leave the **`RELAY ON`** time as it is set, or **enter** Y to set the **`RELAY ON`** time. A Y response returns the prompts:

```
HOURS (SELECT 0 TO 23) <0>?
MINUTES (SELECT 0 TO 59) <0>?
SECONDS (SELECT 0 TO 59) <0>?
```

!Respond to these prompts and then the terminal displays:

```
SETUP ALARM RELAY OFF TIME (Y/N)?
```

!Use this prompt to set the time of day to de-energize the *Relay Output Terminals*. Respond to the prompts that follow in the same manner as with the **`SETUP ALARM RELAY ON TIME`**. Once the relay is turned on, it stays on until it is turned off. You will be returned to **`MAIN>`**.

Group and Unit Number

Use this command to set the unit address of the PMD if you are communicating through the *RS-422A Computer Interface Port*. This unit address is the group and unit numbers to which the 400 unit(s) will respond. **Press** 12 and Enter.

The terminal responds:

```
GROUP = 0
UNIT = 0
SETUP GROUP NUMBER (Y/N)?
```

!**Enter** N to keep the group number at its current setting or Y to change the group number shown. The terminal responds:

```
GROUP<0>?
```

!**Enter** a number between 0 and 15 and **hit** Enter. The terminal responds:

```
SETUP UNIT NUMBER (Y/N)?
```

!**Enter** N to keep the unit number at its current setting or Y to change the unit number shown. The terminal responds:

```
UNIT<0>?
```

!**Enter** a number between 0 and 4095 and **press** Enter. You are returned to **MAIN>**.

Note: The terminal will not accept invalid numbers for the unit address.

Memory Tests

Use this command to perform self-testing of the program memory and/or data log memory chips. **Press** 13 and Enter. The terminal responds with the sub-menu:

```
MEMORY TEST FUNCTIONS
1) E2 TEST 1
2) E2 TEST 2
3) DATA LOG RAM TEST
?
```

!**Select** an option (1-3) and **press** Enter to test the integrity of the user-programmable chips. Option **1) E2 TEST 1** tests the program memory chips to see that they retain memory properly. The unit displays the number and size of the memory chips and indicates the condition of the chips upon completion of the test. When you select this item, you see:

```
CAUTION - THIS WILL INITIALIZE MESSAGE MEMORY!!
CONTINUE (Y/N)?
```

!**Press** N to abort or Y to continue testing. Example:

```
EEPROM TEST #1  *
  TESTING 2 8k EEPROM CHIPS
  EEPROM #1 IS OK
  EEPROM #2 IS OK
CLEARING MESSAGES
```

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

WARNING! WHEN THIS TEST IS COMPLETE, THE 400 UNIT EXECUTES A CLEAR MEMORY COMMAND. MESSAGE MEMORY IS CLEARED WHEN THIS TEST IS COMPLETED.

!Option **2) E2 TEST** 2 tests the program memory chips and halts the test when an error occurs. The display shows the error address and bit pattern at which the memory failed to program and does not continue testing. If no error occurs, the display shows the message **TEST DONE**. When you select option 2, you see:

```
CAUTION - THIS WILL INITIALIZE MESSAGE MEMORY!!
CONTINUE (Y/N)?
```

!**Press** N to abort or Y to continue testing. Examples:

```
EEPROM TEST 2      * EEPROM TEST 2 *
  2 8k CHIPS         2 8k CHIPS
  0603AE   55        TEST DONE
```

WARNING! WHEN THIS TEST IS COMPLETE, THE 400 UNIT EXECUTES A CLEAR MEMORY COMMAND. MESSAGE MEMORY IS CLEARED WHEN THIS TEST IS COMPLETED.

!Option **3) DATA LOG RAM TEST** tests the RAM chip used for storing data log entries. When you **select** 3, you see:

```
DATA LOG WILL BE CLEARED
CONTINUE (Y/N)?
```

!**Press** N to abort or Y to continue testing. (Time- and date-triggered message memory is also cleared). When complete, the display shows **RAM TEST COMPLETE** or **RAM TEST FAILURE**. If these tests indicate bad PMD memory chips, contact UTICOR Technology customer service: 563-359-7501.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

Define Clear FIFO

Use this command to select a message number to be used on the *PLC Port* to clear the incoming message FIFO (First-In-First-Out). **Press** 14 and Enter. The terminal responds:

```
A CLEAR FIFO MESSAGE IS NOT DEFINED.
```

or

```
CLEAR FIFO IS DEFINED AS MESSAGE <0>
DO YOU WANT TO CHANGE IT?
```

!**Enter** N to leave this option undefined or **press** Enter to enter a message number value for clearing the FIFO. The terminal responds:

```
ENTER CLEAR FIFO MESSAGE NUMBER
WHERE 65535 INDICATES NOT DEFINED <65535>:
```

!**Enter** a number between 0 and 65535 and **press** Enter, or just Enter to leave it at its current setting shown within brackets. When selected in the DISPLAY MODE, this message number will cause the incoming message FIFO to empty (clear memory). The FIFO must contain less than 32 entries for it to accept the clear FIFO number. You are returned to **MAIN>**.

Program Time of Day Messages

Use this command to program up to 16 time- and date-triggered messages. **Press** 15 and Enter. The terminal responds with the following sub-menu:

```
                    Date Triggered Messages

Default Time Of Day Message:  XXXXX

MSG#   On             Off           MSG#  ON             OFF
 1.   MMM DD HH:MM   MMM DD HH:MM  9.  MMM DD HH:MM  MMM DD HH:MM
 2.                                 10.
 3.                                 11.
 4.                                 12.
 5.                                 13.
 6.                                 14.
 7.                                 15.
 8.                                 16.

 1. Modify An Entry
 2. Initialize List
 3. Modify Default Time Of Day Message
 4. Exit
```

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

!**Select** an option. Option **1) MODIFY AN ENTRY** lets you change existing time/date triggered message information. When you select option 1, you see the prompt:

ENTRY NUMBER

!**Enter** a number between 1 and 16 and **press** Enter. The terminal responds:

DELETE (Y/N) <N>:

!**Enter** Y to delete previously programmed information for this message location, or **press** N if this location has not been previously programmed. The terminal responds:

MESSAGE NUMBER <#####>:

!**Enter** a message number between 0 and 65534 and **press** Enter. Or, **press** Enter to select the number shown in brackets (the last number selected). This will be the message that will be displayed at the time/date for this time-and date-triggered message. Now the terminal responds:

Every Day <Y>:

!**Press** Y to have this message be displayed at the programmed time every day. (Month and day setting locations will display *** *** when updated and you will skip the next two prompts). Or, **enter** N to program this location for a specific date. Depending on your selection for this option, the terminal responds with these prompts to set the "on time":

ON MONTH (1-12) <MMM>:
ON DAY (1-31) <DD>:
ON HOUR (0-23) <HH>:
ON MINUTE (0-59) <MM>:

This sequence repeats (month, day, hour, and minutes) for the "off time" of the message.

!Option **2) INITIALIZE LIST** clears the date-triggered message list memory. When you select option 2, you see the prompt:

INITIALIZE ALL DATE TRIGGERED MESSAGES (Y/N) <N>:

!**Enter** Y to clear the date-triggered message list (stored in RAM). **Enter** N or Enter to abort this clear-memory command.

!Option **3) MODIFY DEFAULT TIME OF DAY MESSAGE** lets you set or change the **Default Time Of Day Message**.

This message is displayed after a date-triggered message's display time has lapsed, and the message is cleared from the display. The **`Default Time of Day Message`** is then displayed until new information is received and displayed (either from the *Computer Port* or from the time- and date-triggered message information). Selecting this option displays the prompt:

```
Default Time Of Day Message Number (0-65534
65535 = NONE) <65535>:
```

!Enter a number between 0 and 65535 and **press** Enter, or just Enter to leave it at its current setting shown in brackets. The setting of 65535 will cause the display to go blank.

!Option **`4) EXIT`** returns you to **`MAIN>`**.

Program Display Saver Timeout

Use this command to clear the display of the same message after a set number of hour(s). **Press** 16 and Enter. The terminal responds:

```
DISPLAY SAVER TIMEOUT IN HOURS (0-24) <0>?
```

!**Enter** a number between 1 and 24 to select the number of hours you wish to leave the message unchanged on the display if no other messages are selected, or **press** Enter to leave this option undefined or at a prior setting.

Note: If you **select** 0, you will **disable** this option.

Function Keys/ LED Control

Use this command to map the function keys. **Press** 17 and Enter. The response you get will vary depending on which 400 Series product you have. With the **PMD 400**, the terminal responds with:

```
LED CONTROL (SELECT 1=PMD, 2=CI) <1>
```

Since the 400 does not have a PLC interface board, this will be the only prompt. If your choice is **`1=PMD`**, the LEDs will turn on when their corresponding function keys are pressed. If you choose **`2=CI`** (Computer Interface), the LEDs will be illuminated by commands received through the computer interface. You can turn off the LEDs only through the Computer Interface.

The 450, 460, 470 and 475 products also come with a 15-pin "D"-type connector on the PLC interface board which is used to connect either a 16- or 32-point function keypad with associated LEDs.

SECTION 3 - ONLINE PROGRAMMING
PART A: DEVELOPING A MESSAGE FILE

Refer to **APPENDIX D - ACCESSORIES** and **SECTION 7 - PLC INTERFACING** for more information about the 400E expander modules, and refer to the following parts of **SECTION 7** for information about your particular unit:

PART A: 450 A-B INTERFACE
PART B: 460 TI INTERFACE
PART C: 470 S908 MODICON INTERFACE
PART D: 475 MODICON MODBUS PLUS INTERFACE
PART E: 480 GE GENIUS I/O

<CR> Print this List

This command simply provides another way to view the **`Setup Parameters`** sub-menu when the **`SETUP>`** prompt appears on your screen.

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

This part is an extension of part **A** as it explains how to set up the attributes of the messages you program. As shown in the **`Edit a Message`** portion of part **A**, you can "replace message control options" when editing a messages.

Message options let you format messages to blink, scroll up or left, chain, and overlay each other to create multiple messages. The message **`OPTION MENU`** also lets you define various "outputs" for the message such as energizing the *Relay Output Terminals* to cause an alarm to sound, sending it to an online printer, slaves or the data log.

After hitting H to enter the **`MAIN> COMMANDS`** online programming menu, then E and Enter to enter the **`Edit a Message`** option, you must respond to the prompt: **`EDIT MSG #`**. **Enter** the message number to be edited and **press** Enter. The terminal responds with the programmed message options, followed with: **`REPLACE MESSAGE CONTROL OPTIONS (Y/N)?`** **Enter** N to leave the options as they are or Y to change the message options. The terminal shows the **`OPTION MENU`** below.

```
                              OPTION MENU

1. CHAIN MESSAGE           <N>   15. CLEARS DISPLAY          <Y>
2. REPEAT CHAIN MESSAGE    <N>   16. CENTER MESSAGE          <N>
                                 17. STARTING DATA SET       <1>
3. SCROLLS UPWARD          <N>
4. BOTTOM LINE             <1>   18. SEND TO SLAVE           <N>
5. TOP LINE                <1>   19. TO ALL SLAVES           <Y>
6. REPEAT                  <N>   20. GROUP                   <0>
7. SCROLL TIME INTERVAL    <0>   21. UNIT                    <0>
8. SCROLLS LEFT            <N>   22. ALARM RELAY             <N>
9. SCROLL LINE             <1>   23. INVISIBLE MESSAGE       <N>
10.REPEATS                       <N>  24. LOG MESSAGE              <N>
11.SCROLL TIME INTERVAL    <0>   25. BLINK ENTIRE MSG        <N>
12.SEND MSG TO RS232 PORT  <N>
13.ON ONE LINE             <N>   26. DS PROTECT N N N N
14.<CR><LF> AT END OF MSG  <N>   98. Help
                                 99. Exit

Enter Selection:
```

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

Enter the number that corresponds to the item you wish to change, then **press** Enter. If the option is **<Y>/<N>**, you toggle the status with the space bar. If the option requires number entry, you are prompted for a number.

Enter the desired number and Enter, or just Enter to keep the setting shown in brackets. Non-digits and illegal numbers are not accepted. The menu is updated on the screen each time you change it. When options are set as desired, **select 99. Exit** to enter or edit message text.

Chain Message

A chained message contains a list of message numbers (up to 115) and causes the unit to sequence through this list of programmed messages whenever the chained message number is triggered.

!From the **Add a Message** option of the **MAIN> COMMAND** menu, **select** a number you haven't programmed yet (Let's say message 33 for this example) and **press** Enter. The **REPLACE MESSAGE CONTROL OPTIONS (Y/N)** will appear on the terminal screen. **Select** Y and Enter. **Press** 1 to enable **CHAIN MESSAGE** from the **OPTION MENU** and Enter. Now **press** Exit and Enter. Begin to make your chain by **entering** the first message number to be in the chain and then **pressing** Enter (up to 115 entries). Example:

```
1>11
2>23
3>92
4>
```

With this example, the chain would appear on your display with messages in this order: 11, 23, 92 whenever you **select** message 33. To end a chain, **press** Enter twice.

Notes: You can repeat message numbers. The unit accepts any valid message number for the chain list as long as it is currently programmed in the unit and it is not another chained message. ! Unlike other kinds of messages, a chained message contains no text of its own and therefore ignores all message options except for **REPEAT CHAIN MESSAGE** (explained next); selecting **CHAIN MESSAGE** toggles all other message options (except **REPEAT CHAIN MESSAGE**) to their defaults. ! Individual messages in a chain message maintain their own options while part of a chain (blinking, logging etc).

Repeat Chain Message - **Enter** Y and the chain message repeats until new message information is processed by the *Computer Port*. **Enter** N and the chain message sequences through only once.

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

Scrolls Upward

Enter Y to have your message scroll up from a lower line of the display to an upper line of the display. The message can scroll up starting from and ending on any of the unit's lines (described next). **Enter** N if you do not want your message to scroll up. (If **SCROLLS LEFT**, explained next, is selected Y, **SCROLLS UPWARD** is automatically set to N).

Note: The maximum number of characters in a scroll up message is 235.

Bottom Line - This number indicates the lowest line on which the **SCROLLS UPWARD** message begins. **Enter** a number between 1 and 4 and **press** Enter, or just Enter to select the number shown within brackets.

Top Line - This number indicates the top line on which the **SCROLLS UPWARD** message ends. **Enter** a number between 1 and 4 and **press** Enter, or just Enter to select the number shown within brackets. This number must be less than or equal to the number indicated in **BOTTOM LINE**.

Repeat - This option determines if the **SCROLLS UPWARD** message will repeat. **Enter** Y if you want it to repeat or N if you do not want it to repeat.

Scroll Time Interval - This option lets you set an individually assigned **SCROLL UPWARD** rate/speed for this message. **Enter** a number between 1 and 99 (0.1 to 9.9 seconds) and **press** Enter, or just Enter to select the setting shown within brackets. A setting of 00 indicates no scroll rate is stored with the message and the system scroll rate is used.

Scrolls Left

Enter Y to have your message scroll left. The message can scroll right to left on any one line of the display. **Enter** N if you do not want your message to scroll left. (If **SCROLLS UPWARD**, explained previously, is selected Y, **SCROLLS LEFT** is automatically set to N).

Scroll Line - This option indicates on which display line a **SCROLL LEFT** message is displayed. **Enter** a number between 1 and 4 and **press** Enter, or just Enter to select the setting shown within brackets.

Repeats - This option determines if the **SCROLL LEFT** will repeat. **Enter** Y if you want it to repeat or N if you do not want it to repeat.

Scroll Time Interval - This option lets you set an individually assigned **SCROLL LEFT** rate/speed for this message.

A setting of 00 indicates no scroll rate is stored with the message and the system scroll rate is used. **Enter** a number between 1 and 99 (0.1 to 9.9 seconds) and **press** Enter, or just Enter to select the setting shown within brackets.

Send Msg to RS232 Port

This option lets you send your message to the RS-232 *Terminal/Printer Port* to be printed each time it is selected. **Press** Y to implement this option or N to deny the output via the RS-232 *Terminal/Printer Port*. (UTICOR offers a Serial Impact Printer, Part Number 58372).

Notes: Characters that remain on the display from previous (non-scrolling) messages are also printed. ! If the message to be printed has imbedded data sets, then these data sets must be updated with new values before the message text will be printed.

On One Line - This option lets you select how you would like your message text to appear on the printed out sheet if you opt to print. **Enter** Y to have your message printed on one line (If it is not a **SCROLL LEFT** message, one space will appear between each line of your message). Line wrap behavior is printer dependent. **Enter** N to have the printed message appear as the text appears on the display. Each Enter pressed during programming inserts an end-of-line marker for the printer. This marker advances the printing element to the beginning of the next line.

<CR><LF> at End of Message - This option lets you designate your message as a marker to tell the printer when to advance to the next line. **Enter** Y to tell the printer to advance to the beginning of the next line after this message has printed. If you **enter** N, the next message will begin printing on the same line as the previous one.

Clears Display

This option lets you designate your message as one that will clear the rest of the display screen when it displays. **Enter** Y to clear text and data set values (standard 400 only) before this message is displayed. **Enter** N to allow previously displayed characters to remain in locations on the display screen where this message does not overlay them.

Note: This option will set data set values to 0.

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

Center Message This option will center your message on the display. **Enter** Y to center message text on the display lines. **Enter** N and the message will not be centered; no "blanks" are placed before or after the text.

Note: This option will set data set values to 0 and clear the display.

Note: For the 450, 460, 470, 475 and 480 units, the variable data sets will not be cleared when the display is cleared.

Starting Data Set If this message does not contain variable data, ignore this option. It allows you to designate which of the four data sets the 400 product will read first, as the first data set it reads will be considered the "Starting Data Set."

These four possible states have been assigned numbers (1 through 4) so that you know the order they are received and inserted into the message. They are used in a 1, 2, 3, 4, 1 order, and you set which one is used first by setting this option. **Enter** a number between 1 and 4 and **press** Enter, or just Enter to select the number as shown.

Note: Implementing variable data sets into your message(s) is described later in this section.

Send to Slave This option lets you determine whether or not your message is displayed on slave units (Group and unit numbering slaves is described below). **Enter** Y to send the message to the *Slave Port* and see it displayed on specified slave(s). If you **enter** N, message display options and message text are not sent to the *Slave Port* when the message is displayed.

To All Slaves - This option allows you to broadcast the message to all slave displays connected to the master's *Slave Port*. **Enter** Y to send the message to all slaves (Setting **GROUP** and **UNIT** options to zero toggles this option to Y). **Enter** N so that only messages addressed in the **GROUP** and **UNIT** options will receive and display the message (Setting **GROUP** and **UNIT** options to non-zero toggles this option to N).

Group - This option indicates the group number for the slave(s) that will respond. **Enter** a number between 0 and 15 and **press** Enter to select the group number, or just Enter to select the current setting.

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

Unit - This option indicates the unit number, within the above group number, of the slave(s) that will respond. **Enter** a number between 0 and 4095 and **press** Enter to select the unit number, or just Enter to select the current setting.

Notes: A zero group and zero unit number addresses all slaves. A set group number with a zero unit number displays the messages on all slaves within that one group.

A specified group and unit number displays the message on that one slave only. -- Slaves are described in more detail in **APPENDIX D - ACCESSORIES**.

Alarm Relay This option lets you designate your message as one that will energize the *Relay Output Terminals* on the back of the unit every time the message is selected. **Enter** Y to select this option or N so that the *Relay Output Terminals* will not activate with this message.

Notes: Energizing these terminals stops when a new message that does not energize the relay is selected. ! The timed relay works independently and takes precedence over this prompt.

Invisible Message This option lets you print, log or display your message on slave units without displaying the message on the master. **Enter** Y to implement this option or N so that the message will be displayed on the master unit.

Log Message This option tells the master to store the message (every time it is selected) as well as the time/date it was triggered. Variable data values displayed within the message are also stored. Note that only the original data set value and the first change to it will be logged. All changes in the data that may occur while the message is being displayed will not be logged. **Enter** Y to select this option. If you **enter** N, the message is not stored in the data log.

Note: The **`Data Log`** must be enabled (See Data Logging in **PART A**).

Blink Entire Message This option blinks the entire message on and off when displayed (including time, date and variable data). **Enter** Y to select this option or N so that the message blinking behaves only according to use of the blink delimiters.

(See **PART A** to program blink delimiters).

Note: Characters that remain on the display from previous messages will blink also.

DS Protect This option protects the data sets from being changed by the operator. Toggle the N to Y to implement this option and protect the selected data set(s). When a data set is protected, it can be displayed but not changed by the user.

Help This option simply tells how to set message options.

Exit Select this option to exit from the **OPTION MENU** to begin entering or editing message text.

PMD 3000: Frame Definition The PMD 3000 slave displays messages sent to it from a master or computer and abides by the message attribute commands. In addition to these options, "escape codes" can be embedded in a message to control the PMD 3000's LED field. Frame definition lets you specify where to display the message on the LED field by embedding a frame definition code immediately before the first character of message text. The frame definition is defined by "sticks." A stick is 20 characters, two inches high and is denoted in the PMD 3000 part number with 1W2H number designators. The number preceding the W in the part number tells you how many sticks wide the PMD 3000 is. The number preceding the H in the part number tells you how many sticks high this PMD 3000 is. For example:

! 1 stick wide and 2 sticks high (3100-N1W2H) = 2 lines of 20, two-inch characters

! 2 sticks wide and 4 sticks high (3100-N2W4H) = 4 lines of 40, two-inch characters

! 2 sticks wide and 8 sticks high (3100-N2W8H) = 8 lines of 40, two-inch characters

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

You will need at least a 1W2H to display four-inch characters; and at least a 1W4H to display six- and eight-inch characters. The full width of the display is enabled. If frame specifiers are not used, the unit will use its default value. Frame definition is a 6 byte code that is defined as follows: <ESC><F> (n1) (n2)

<ESC><F>	Frame definition escape code
(n1)	Top LED stick for message display
(n2)	Bottom LED stick for message display

Before changing the character height, you must enter a frame definition. A frame definition enables the stick numbers on which the characters will be displayed. For an eight-inch character, you would need to select four sticks since each stick is two inches. The field is defined by choosing the top and bottom stick.

To display four-inch characters, you need to have at least two rows open.

Example:	Upper Row = 1	or:	Upper Row = 2
	Lower Row = 2		Lower Row = 3

To display six-inch characters, you need to have at least three rows open.

Example:	Upper Row = 1	or:	Upper Row = 2
	Lower Row = 3		Lower Row = 4

To display eight-inch characters, you need to have at least four rows open.

Example:	Upper Row = 1	or:	Upper Row = 3
	Lower Row = 4		Lower Row = 6

Character Height

Character height can be changed at any time in a message. Character height is changed with "escape codes" embedded within the message text exactly as follows:

Type in **^[0** to designate 2-inch characters.
Type in **^[1** to designate 4-inch characters.
Type in **^[2** to designate 6-inch characters.
Type in **^[3** to designate 8-inch characters.
Type in **^[4** to designate thin 8-inch characters.

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

Examples:

!To display an eight-inch character on sticks 1 through 4, the command would be ^[F0104^[3message .

^[= the escape code
F = frame definition
01 = the top stick of the field
04 = the bottom stick of the field
^[= the escape code again
3 = the 8 inch character
message= enter the message text

!To display a six-inch character on sticks 2 through 4, the command would be ^[F0204^[2message.

^[= the escape code
F = frame definition
02 = the top stick of the field
04 = the bottom stick of the field
^[= the escape code again
2 = the 6 inch character
message= enter the message text.

Be sure there are no spaces between the commands and the stick numbers are padded with leading zeroes.

Frame definition can be larger than the character size. However, if the frame definition is smaller than the character size selected, the largest character size that will fit in the frame definition will be displayed.

Embedded Codes

Some features are handled by codes embedded within the message itself. This segment covers these options.

!**Time** - To have the time displayed in a message, **enter** Ctrl T. The terminal displays: **HH:MM:SS** to indicate that the time code is programmed. Time occupies eight character locations of the message.

!**Date** - To have the date displayed in a message, **enter** Ctrl D. The terminal displays: **DD-MMM-YY** to indicate that the date code is programmed. Date occupies nine character locations of the message.

SECTION 3 - ONLINE PROGRAMMING
PART B: MESSAGE ATTRIBUTES

!**Variable Data** - Each message can have four separate data sets of up to five significant digits each. To program variable data digits in a message, use the control characters shown below.

CTRL U	**CTRL V**	**CTRL W**	**CTRL X**	**CTRL Y**
MSD (most significant digit)				**LSD** (least significant digit)

For data sets with less than five digits, simply omit unwanted control codes from the MSD side of the chart. The PMD will display leading zeros e.g. 0004. Data sets can be combined to form up to a 20-digit number. The first set of control characters represents the first data set; the next set of control characters is the second set etc.

BINARY users can use all five control characters (0-65,535)-- from the least significant digit to the most significant digit. For example:

9	**CTRL Y**
99	**CTRL X CTRL Y**
999	**CTRL W CTRL X CTRL Y**
9999	**CTRL V CTRL W CTRL X CTRL Y**
65535	**CTRL U CTRL V CTRL W CTRL X CTRL Y**

BCD users use four control characters (0-9999). For example:

9	**CTRL Y**
99	**CTRL X CTRL Y**
999	**CTRL W CTRL X CTRL Y**
9999	**CTRL V CTRL W CTRL X CTRL Y**

You may place the following ASCII characters within data sets:

! " # $ % & ' ( ) * + , - . /

For example, let's say you want to show the variable cost of gold. You would enter:

$ CTRL V CTRL W CTRL X CTRL Y #

This code might then represent **$ 4000#** to report that it is four thousand dollars per pound.

Note: In online programming, you can't put any characters in between the control codes without overlaying messages.

As another example, let's say you want to show the variable percentage of metal scrapped for one day. You would enter:

CTRL X CTRL Y %

This code might then represent **12%**.

!**Null Characters** - Null characters are useful when overlaying messages on the display. Overlaid messages do not clear the display and allow previous messages to remain in non-programmed or null areas. To program a null character in the message, **enter ^**, followed by **@** (for each null location).

Notes: If this message does not clear the display, characters from previous messages can remain in null locations. **`Edit`** and **`List`** commands show nulls as periods.

!**Control Characters** - Standard ASCII control characters can be embedded in messages. To program ASCII control characters, **enter** ^, followed by the conventional control code character.

!**Functional Commands** - Specific embedded commands in message text cause the unit to perform particular duties. These programmed codes are displayed along with the message text when triggered. The unit is instructed with the text string **#$** followed by an additional letter. These commands are listed below and must be entered in the first three character positions of the first line.

#$A Enable Data Log - This turns data logging function on. (This message can be logged). This option does not change "programmed" data log setting. If you restart the unit, Data Log Enable will return to the software-controlled setting.

#$B Disable Data Log - This turns data logging function off. (This message cannot be logged). This option does not change "programmed" data log setting. If you restart the unit, Data Log Enable will return to the software-controlled setting.

#$C Clear Data Log - This clears data log memory. (This message can be logged. It will be the first logged message in the new data log).

#$D Print Data Log - This causes the data log contents to be sent to the *Terminal/Printer Port*. (Normal PMD operation is suspended while data log is being printed).

Data log is printed from first logged message to most recently logged message. The unit outputs one logged message per scan. Scan time becomes much slower while printing messages online. Due to decrease in DISPLAY MODE performance, we suggest that you print the data log offline.

#$E Stop Data Log Print - This terminates data log printing and allows message logging to resume. (This message can be logged).

#$F Increment Clock 1 hour (for daylight-savings time) - This does not affect the date setting. The hours will roll over.

#$G Decrement Clock 1 Hour (for daylight-savings time) - This does not affect the date setting. The hours will roll over.

#$H Enable Displaying of Time of Day Messages - When this is embedded at the beginning of a message (the first characters on line one of the message text), this code turns on the time of day message display function. The unit powers up with this option enabled. This function is used to turn the time of day option back on after it has been disabled, and has no effect on the display if the option is already enabled.

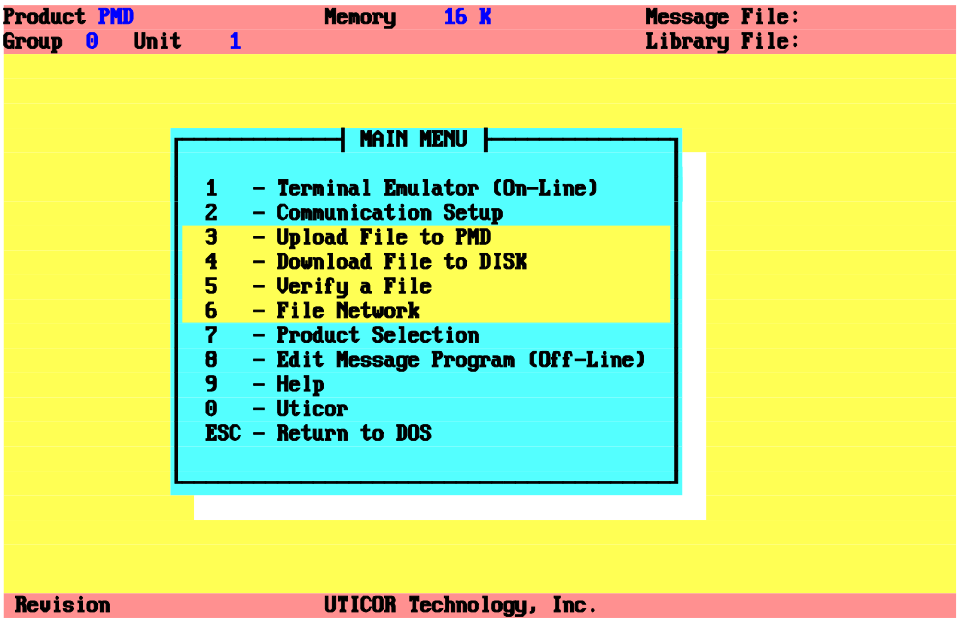
#$I Disable Displaying of Time of Day Messages - When embedded at the beginning of a message (the first characters on line one of the message text), this code turns off the time of day message display function.

No date-triggered messages will be displayed when this option is disabled. This code has no effect on the display if the option is already disabled.

Note: These **Functional Commands** cannot be seen using the View command of the online programming **`MAIN MENU`**. They will simply appear in their code form e.g. #$I, #$G etc. You must be in the DISPLAY MODE, described in **SECTION 6 - DISPLAY MODE** to see their commands applied.

SECTION 3 - ONLINE PROGRAMMING
PART C: WORKING WITH MESSAGE FILES

In this final part, you will learn about the remaining options on the **MAIN MENU** which help you utilize the message files you have created: **Upload file to PMD, Download file to DISK, Verify a File** and **File Network**. You'll also be given instructions for printing message files, as it is not part of the **MAIN MENU**, but maintains a unique process of its own. The network connects to the unit's *RS-422 ComputerInterface Port*.



Before you begin to use the options, make sure that the unit baud rate matches the TAPE MODE baud rate (for **Upload/Download**), and the **COMP. INTER.** baud rate (for **File Network**). Refer to the **GETTING STARTED** section or simply **select Communication Setup** from the **MAIN MENU**. To set the 400 unit's baud rate, use the MODE and ENTER push buttons on the front panel to manipulate **SET UP SERIAL PORTS**.

Upload File to PMD

Use this command to transfer message file(s) from the computer to the 400 unit. **Select** Upload File to PMD from the **MAIN MENU**. Make sure that **Directory:** is your message source--where the message(s) are going to be uploaded from. **Select** one of your created message files and the terminal will tell you to put the 400 in TAPE IN mode. Use the push buttons on the front panel to do this.

Select [OK]. The display screen will respond:

```
LOADING MESSAGES
START TAPE
LOAD COMPLETE
    x MESSAGES LOADED
PRESS RESTART
```

!The terminal screen will ask you:

```
Do you want to verify the Upload/Download?
```

!If you **select [OK]**, the terminal will tell you to "**put PMD in TAPE OUT mode**." If you follow this procedure, the 400 unit will respond:

```
DUMP COMPLETE
   x MESSAGES DUMPED
PRESS RESTART
```

and the terminal screen will tell you the status of the verification.

Notes: When the 400 is set to TAPE IN, the unit is set for one stop bit, no parity, and eight-bit word length. ! Load memory ends when an error occurs. Memory loaded successfully prior to the error is usable. ! Loading memory into the 400 from digital tape does NOT change its unit address setting. ! The TAPE IN function clears time- and date-triggered message memory from the display. Also, time- and date-triggered message information is not loaded by the TAPE IN function (programmed time- and date-triggered message information is stored in RAM).

Download File to DISK

Use this command to transfer a message file from the 400 unit to disk. **Select Download File to DISK** from the **MAIN MENU**. Make sure that **Directory:** is your message destination--where your message(s) are going to be downloaded to. **Select** one of your created message files or **type** in a new file name, and the terminal will tell you to put the 400 in TAPE OUT mode. Use the push buttons on the front panel to do this. The display screen will respond:

```
DUMP COMPLETE
   x MESSAGES DUMPED
PRESS RESTART
```

!The terminal will ask you:

```
Do you want to verify the Upload/Download?
```

!If you **select [OK]**, the terminal will tell you to put to "**put PMD in TAPE OUT mode**."

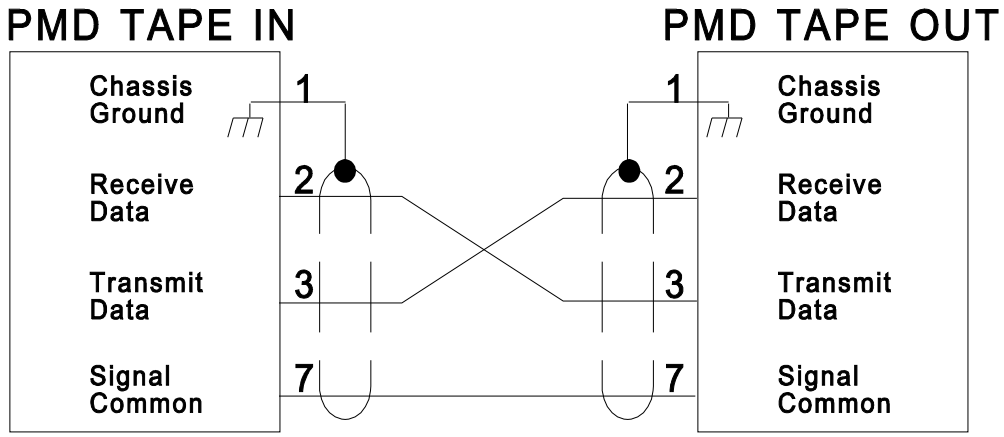
SECTION 3 - ONLINE PROGRAMMING
PART C: WORKING WITH MESSAGE FILES

If you follow this procedure, the 400 will respond with the same message as above and the terminal screen will tell you the status of the verification.

Note: When the 400 unit is set to TAPE OUT, it is set for two stop bits, no parity and eight-bit word length.

Transferring between two PMDs

Message programs can also be transferred directly from one PMD to another. This procedure does require a special cable connecting the two units at their *Terminal/Printer Ports*. Please refer to the illustration below:



Load memory ends when an error occurs. Memory loaded successfully before the error occurs is usable. A message program from one software revision PMD may or may not successfully load into a PMD of another software revision. Exercise caution when transferring message programs created with different software revisions; check the program in the unit to verify that it will operate as expected.

To transfer a program from one PMD to another:
1. Set the baud rate on both PMDs to the same value.
2. Connect the units at the *Terminal/Printer Port* referring to the illustration above.
3. Set the PMD that you are loading messages into to TAPE IN.
4. Set the PMD that you are loading messages from to TAPE OUT.
5. **Press** ENTER on the TAPE IN unit; then **press** ENTER on the TAPE OUT unit.

The TAPE IN unit will respond as described in the TAPE IN procedure and the TAPE OUT unit will respond as described in the TAPE OUT procedure.

SECTION 3 - ONLINE PROGRAMMING
PART C: WORKING WITH MESSAGE FILES

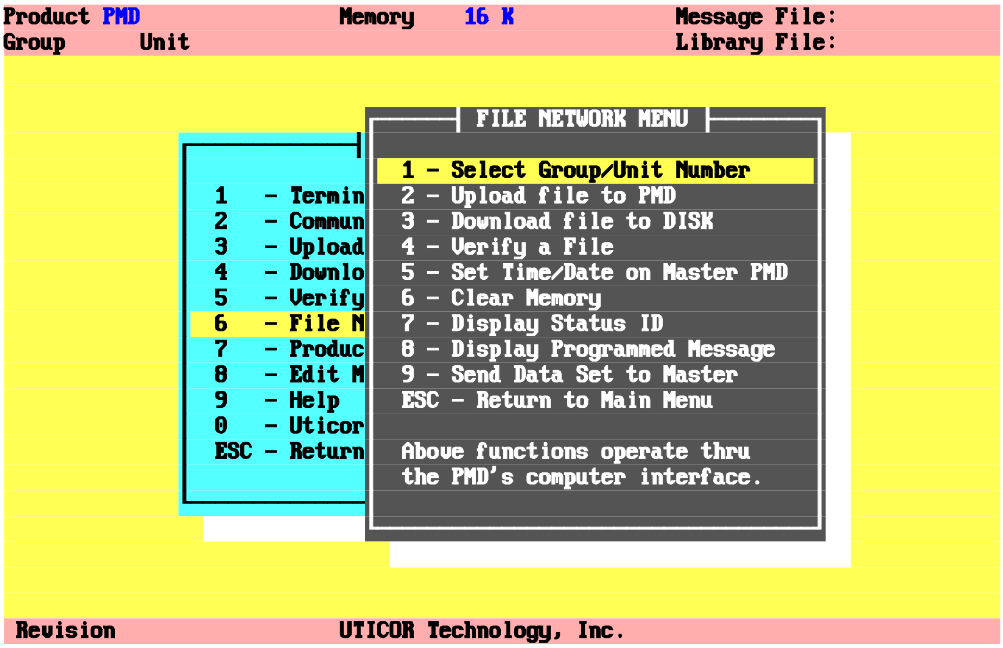
Verify a File

This option allows you to verify that all of the message file data has transferred wholly from the computer to the 400 or from the 400 to the computer. **Select `Verify a File`** from the **`MAIN MENU`**. This function does NOT load contents into any memory; it just checks the messages and parameter settings in the 400 unit against those in the computer to confirm that they are identical. If system attributes or individual messages' options are changed after the transfer was made, verification fails. You can choose to do this verification independently off the **`MAIN MENU`** or after each uploading/downloading process as described above.

Note: The 400 unit compares the "Overhead" bytes (parameter settings) first. Then the unit compares the messages.

File Network

This option only operates through the 400's *Computer Interface Port* and the unit must be in the DISPLAY MODE. Use it to upload or download message files to units that are connected on an RS-422 communication network. The selected message file can be directed to any unit on the network by selecting the proper group and unit number. **Select `File Network`** from the **`MAIN MENU`**. This menu appears on your terminal screen:



SECTION 3 - ONLINE PROGRAMMING
PART C: WORKING WITH MESSAGE FILES

Select Group/Unit Number This option lets you select the proper group and unit number for uploading and downloading message files on a communication network.

Upload File to PMD This option lets you transfer a message program stored on disk to the desired 400 unit on a communication network.

Download File to Disk This option lets you transfer a message program from a 400 unit to disk.

Verify a File This option allows you to compare selected file(s) on disk to a message program located in the 400 Series unit to make sure that the data are identical.

Set Time/Date on Master PMD This option allows the transfer of the computer's time and date to the selected master unit.

Clear Memory This option will cause all stored messages in the 400 unit to be erased and the system attributes to be reset to the factory default settings.

Display Status ID This option will display information concerning the status of the connected display, such as number of messages, revision level, type of product etc.

Display Programmed Message This option allows a message to be displayed on the connected 400 unit.

Send Data Set to Master This option allows variable data to be sent to the 400 unit.

Printing Message Files You can print your message program from the 400 unit directly to the printer. **Connect** your cable to the printer and to the 400 at the *Terminal/Printer Port*. Again, confirm that the baud rate, stop bits and parity mode on the 400 unit match those of the printer. **Set** the 400 to **`PRINT USER PROGRAM`** using the MODE and ENTER push buttons to print the messages and their attributes. The 400 unit will display **`PRINTING MESSAGES`** on its top line, and when it's done printing, it will display **`PRINTING COMPLETE - PRESS RESTART`**.

Note: Parity Enabled = 7 Bit Word Length; Parity Disabled = 8 Bit Word Length.

SECTION 3 - ONLINE PROGRAMMING
PART C: WORKING WITH MESSAGE FILES

Printing Problems

!If the printout is made up of "garbage" characters, verify that the 400 unit and the printer are set for the same baud rate, parity and stop bits.

!If characters are missing from the printouts, the 400 is probably sending characters faster than the printer can print them.

You can correct this by either going to a lower baud rate on both the 400 unit and the printer, or by using some combination of hand shaking or null characters. Determining the number of nulls will require some trial and error on your part.

!Connect the printer's RTS (Request To Send) line to the 400 unit's CTS (Clear To Send) line. The 400 also supports the X-on/X-off function found on some printers.

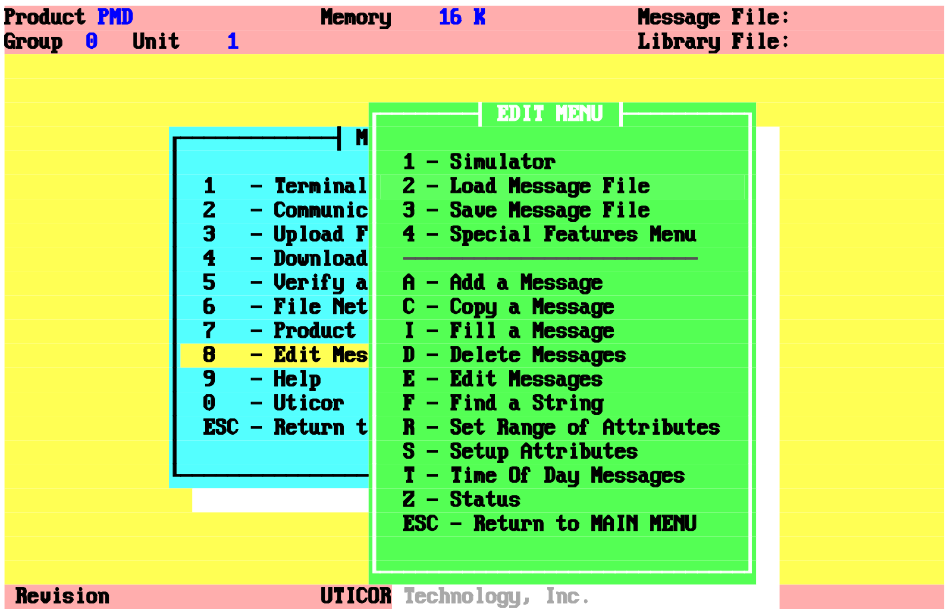
This completes **ONLINE PROGRAMMING** development. You have been introduced to all of the tools needed for developing, editing and viewing message programs. Of course, the best way to become more knowledgeable about the versatility of your 400 unit is to practice with all of the menus and sub-menus and refer to the help menus.

SECTION 3 - ONLINE PROGRAMMING
PART C: WORKING WITH MESSAGE FILES

SECTION 4 - OFFLINE PROGRAMMING

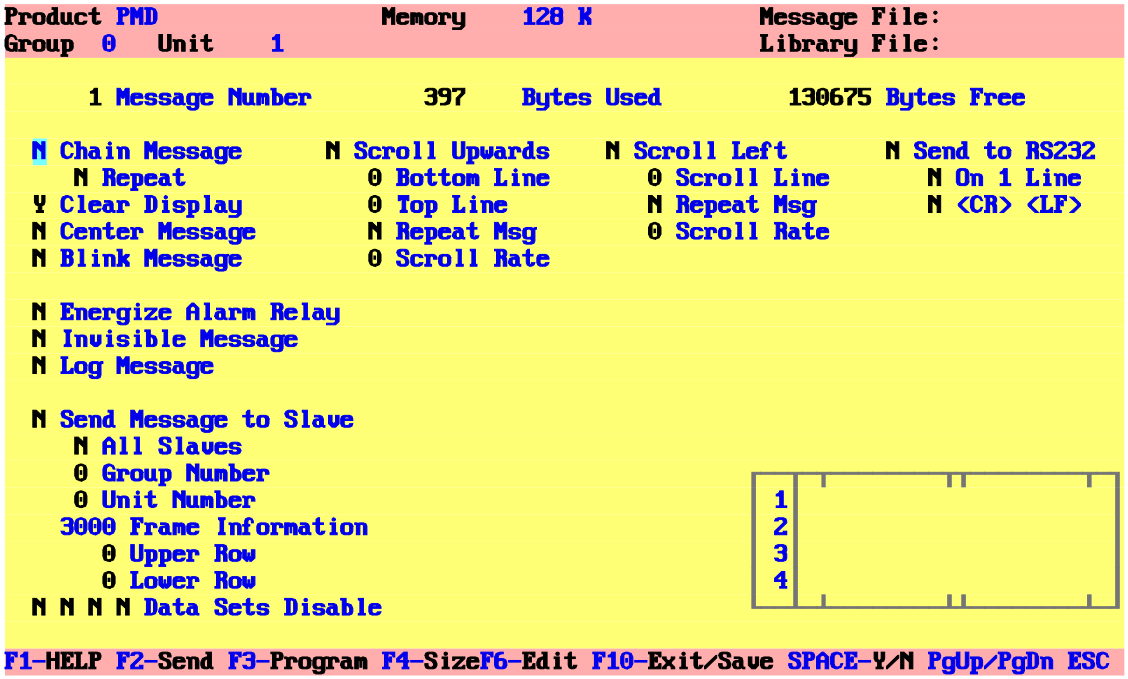
In this section you will learn how to use each option on the **EDIT MENU**, the offline programming "main menu." Right away, you'll be introduced to the actual hands-on programming with the **Add a Message** sub-menu. Then we'll continue down the **EDIT MENU** with **Copy a Message** through **Status**. Finally, we'll address the top half of the **EDIT MENU** starting with **Simulator** and ending the section with **Special Features Menu**.

To continue from the **GETTING STARTED** section, you have selected **Edit Message Program** from the **MAIN MENU**. Instructions for each option follow. In addition to the manual explanations, remember that you can always refer to the bottom of your main screen for help. When you complete **OFFLINE PROGRAMMING**, you will know how to program offline and put your unit to work for you!



Add a Message

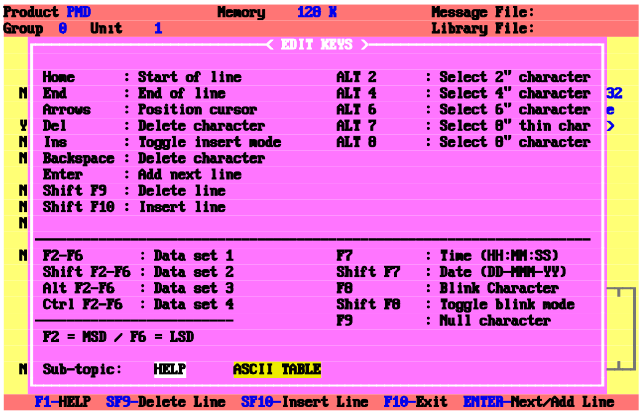
Use this command to program new messages. **Press** A to add a message. The program responds with the menu on the following page which lets you format your messages to blink, scroll up or left, chain together, and overlay each other to create multiple messages. You can also designate various "outputs" such as energizing the *Relay Output Terminals* to cause an alarm to sound, sending messages to a printer, slaves or the data log.



- **Enter** a number between 0 and 65,534 (you designate each of your messages with a number) and **hit** Enter. Next, you move through each of the message attributes. Up/Down arrow keys move the cursor through attribute selections. The space bar toggles through the available options for each cursor location. **F6 MOVES THE CURSOR TO THE MESSAGE TEXT AREA.**

Add a Message Screen Assistance

Note that this **Add a Message** or menu provides the following services using the function keys at the bottom of the screen:



F1-HELP: gives you individual help on each option. When you're in the message text area (F6), F1 gives you an extensive list of useful keyboard keys, embedded codes, function keys and additional commands.

F2 - Send: sends the message to the PMD's computer interface in DISPLAY MODE.
F3 - Program: will program the current message into the PMD's memory through the *RS-422 Computer Port*.
F4 - Size: will select the size of the message text area (20, 40 or 80 characters).

Chain Message

A chained message contains a list of message numbers (up to 115) and causes the unit to sequence through this list of programmed messages whenever the chained message number is triggered (Let's say you're programming message 33 for this example). **Toggle** Y to make this message 33 a chain message. You will notice that if you do choose to make this a chain message, the only other option you can toggle is the **`Repeat`** (chain message) (explained next). So from here, **enter** F6 to start entering the first message number to be in the chain, then **press** Enter to add another (up to 115 entries). Example:

```
1>11
2>23
3>92
4>
```

!With this example, the chain would appear on your display with messages in this order: 11, 23, 92 whenever you **select** message 33. To end the chain, **press** F10.

Notes: You can repeat message numbers. The unit accepts any valid message number for the chain list as long as it is currently programmed in the unit and it is not another chained message. Unlike other kinds of messages, a chained message contains no text of its own and therefore ignores all message options except for **`REPEAT`** (chain message); selecting **`Chain Message`** toggles all other message options to their defaults. Individual messages in a chain message maintain their own options while part of a chain (blinking, logging etc).

Repeat (chain message) - Enter Y and the chain message repeats until a new message is selected. **Enter** N and the chain message sequences through only once.

Clear Display

This option lets you designate your message as one that will clear the rest of the display screen when it displays. **Enter** Y to **clear text and reset data set values to zero** before this message is displayed. **Enter** N to allow previously displayed characters to remain in locations on the display screen where this message does not overlay them.

Center Message This option will center your message on all lines of the display. **Enter** Y to center message text on the display lines. **Enter** N and the message will not be centered; no "blanks" are placed before or after the text.

Note: This option will also reset data sets to zero and clear the display.

Note: For the 450, 460, 470, 475 and 480 units, the variable data sets will **not** be cleared when the display is cleared.

Blink Message This option blinks the entire message on and off when displayed (including time, date and variable data). **Enter** Y to select this option or N so that the message blinking behaves only according to use of the blink delimiters. (See the **`Main Attribute Setup Menu`** of the **`System Attribute Setup menu`** later in this section).

Note: To blink individual characters, **hit** F8 on the character that you want to blink. To blink one word, **hit** Shift F8 before you type the word. Characters that remain on the display from previous messages will blink also if you have chosen to blink the entire message on and off.

Scroll Upwards **Enter** Y to have your message scroll up from a lower line of the display to an upper line of the display. The message can scroll up starting from and ending on any of the unit's lines (described next). **Enter** N if you do not want your message to scroll up. (If **`Scroll Left`**, explained next, is selected Y, **`Scroll Upwards`** is automatically set to N).

Note: The maximum number of characters in a scroll up message is 235. **Bottom Line** - This number indicates the lowest line on which the **`Scroll Upwards`** message begins. **Enter** a number between 1 and 4 and **press** Enter, or just Enter to select the number shown within brackets.

Top Line - This number indicates the top line on which the **`Scroll Upwards`** message ends. **Enter** a number between 1 and 4 and **press** Enter, or just Enter to select the number shown within brackets. This number must be less than or equal to the number indicated in **`Bottom Line`**.

Repeat Msg. - This option determines if the **`Scroll Upwards`** message will repeat. **Enter** Y if you want it to repeat or N if you do not want it to repeat.

Scroll Rate - This option lets you set an individually assigned **`Scroll Upwards`** rate/speed for this message. **Enter** a number between 1 and 99 (0.1 to 9.9 seconds) and **press** Enter, or just Enter to select the setting shown within brackets. A setting of 00 indicates no scroll rate is stored with the message and the system scroll rate is used.

Scroll Left

Enter Y to have your message scroll left. The message can scroll right to left on any one line of the display. **Enter** N if you do not want your message to scroll left. (If **`Scroll Upwards`**, explained above, is selected Y, **`Scroll Left`** is automatically set to N).

Scroll Line - This option indicates on which display line a **`Scroll Left`** message is displayed. **Enter** a number between 1 and 4 and **press** Enter, or just Enter to select the setting shown within brackets.

Repeat Msg. - This option determines if the **`Scroll Left`** will repeat. **Enter** Y if you want it to repeat or N if you do not want it to repeat.

Scroll Rate - This option lets you set an individually assigned **`Scroll Left`** rate/speed for this message. **Enter** a number between 1 and 99 (0.1 to 9.9 seconds) and **press** Enter, or just Enter to select the setting shown within brackets. A setting of 00 indicates no scroll rate is stored with the message and the system scroll rate is used.

Send to RS232

This option lets you send your message to the RS-232 *Terminal/Printer Port* to be printed each time it is displayed. **Press** Y to implement this option or N to deny the output via the RS-232 *Terminal/Printer Port*.

Note: Characters that remain on the display from previous (non-scrolling) messages are also printed. ! If the message to be printed has imbedded data sets, then these data sets must be updated before the message text will be printed.

On 1 Line - This option lets you select how you would like your message text to appear on the printed out sheet if you opt to print. **Enter** Y to have your message printed on one line (If it is not a **`Scroll Left`** message, one space will appear between each line of your message). Line wrap behavior is printer dependent. Each Enter pressed during programming inserts an end-of-line marker for the printer. This marker advances the printing element to the beginning of the next line.

<CR><LF> - This option lets you designate your message as a marker to tell the printer when to advance to the next line. **Enter** Y to tell the printer to advance to the beginning of the next line after this message has printed. If you **enter** N, the next message will begin printing on the same line as this one.

Energize Alarm Relay This option lets you designate your message as one that will energize the *Relay Output Terminals* on the back of the unit every time the message is selected. **Enter** Y to select this option or N so that the *Relay Output Terminals* will not activate with this message.

Note: Energizing this terminal stops when a new message that does not energize the relay is selected. ! The timed relay works independently and takes precedence over this prompt.

Invisible Message This option lets you print, log or display your message on slave units without displaying the message on the master. **Enter** Y to implement this option or N so that the message will be displayed on the master unit.

Log Message This option tells the master to store the message (every time it is selected) as well as the time and date it was triggered. Variable data values displayed within the message are also stored. Note that only the original data set value and the first change to it will be logged. All changes in the data that may occur while the message is being displayed will not be logged. **Enter** Y to implement this option. If you **enter** N, the message is not stored in the data log.

Note: The **`Data Log`** must be enabled (See the **`Main Attribute Setup Menu`** of the **`System Attribute Setup`** menu later in this section).

Send Message to Slave This option lets you determine whether or not your message is displayed on slave units (Group and unit numbering slaves is described below). **Enter** Y to send the message to the *Slave Port* and see it displayed on specified slave(s). If you **enter** N, message display options and message text are not sent to the *Slave Port* when the message is displayed.

All Slaves - This option allows you to broadcast the message to all slave displays connected to the master's *Slave Port*. **Enter** Y to send the message to all slaves (Setting **`Group Number`** and **`Unit Number`** options to zero toggles this option to Y).

Enter N so that only messages addressed in the **`Group Number`** and **`Unit Number`** options will receive and display the message (Setting **`Group Number`** and **`Unit Number`** options to non-zero toggles this option to N).

Group Number - This option indicates the group number for the slave(s) that will accept the message. **Enter** a number between 0 and 15 and **press** Enter to select the group number, or just Enter to select the current setting.

Unit Number - This option indicates the unit number within the above group number for the slave(s) that will accept the message. **Enter** a number between 0 and 4095 and **press** Enter to select the unit number, or just Enter to select the current setting.

Note: A zero group and zero unit number addresses all slaves. A set group number with a zero unit number displays the messages on all slaves within the group. A specified group and unit number displays the message on one slave only. Slaves are described in more detail in **APPENDIX D - ACCESSORIES**.

PMD 3000 Frame Definition

Frame definition lets you specify where to display the message on the LED field by determining the **`Upper/Lower Rows`**. The frame definition is defined by "sticks." A stick is 20 characters, two inches high and is denoted in the PMD 3000 part number with 1W2H number designator. The number preceding the W in the part number tells you how many sticks wide this PMD 3000 is. The number preceding the H in the part number tells you how many sticks high this PMD 3000 is. For example:

!1 stick wide and 2 sticks high (3000-N1W2H)= 2 lines of 20, two-inch characters

!2 sticks wide and 4 sticks high (3000-N2W4H)= 4 lines of 40, two inch characters

!2 sticks wide and 8 sticks high (3000-N2W8H)= 8 lines of 40, two-inch characters

SECTION 4 - OFFLINE PROGRAMMING

You need at least a 1W2H to display four-inch characters; and at least a 1W4H to display six- and eight-inch characters. The full width of the display is enabled. Define your display by designating the **`Upper/Lower Rows`**. If frame specifiers are not used, the unit will use its default value.

To display four-inch characters, you need to have at least two rows open.

Example:	Upper Row = 1	or:	Upper Row = 2
	Lower Row = 2		Lower Row = 3

To display six-inch characters, you need to have at least three rows open.

Example:	Upper Row = 1	or:	Upper Row = 2
	Lower Row = 3		Lower Row = 4

To display eight-inch characters, you need to have at least four rows open.

Example:	Upper Row = 1	or:	Upper Row = 3
	Lower Row = 4		Lower Row = 7

After you designate the upper and lower numbers and **press** F6, a message text area unique to the 3000 appears on your screen. Type in text as you would in the regular message text area and use the function keys to change character heights (F1 lists these commands as well as many others).

SECTION 4 - OFFLINE PROGRAMMING

Character Height

Character height can be changed at any time in a message. Character height is changed with commands inserted into the message text exactly as follows:

Enter Alt 2 to designate 2-inch characters.
Enter Alt 4 to designate 4-inch characters.
Enter Alt 6 to designate 6-inch characters.
Enter Alt 7 to designate thin 8-inch characters (same width as 4" characters).
Enter Alt 8 to designate 8-inch characters.

Frame definition can be larger than the character size. However, if the frame definition is smaller than the character size selected, the largest character size that will fit in the frame definition will be displayed.

NNNN Data Sets Disable

This option protects the data sets from being changed by the operator.

Toggle the N to Y to implement this option and protect the selected data set(s). When a data set is protected, it can be displayed but not changed by the user.

Variable Data Sets

As introduced at the beginning of this section with the part on **Add a Message Screen Assistance**, you can put data sets in your messages. Each message can have four separate data sets of up to five significant digits. To program variable data digits, use the commands shown below:

	MSD Most Significant Digit				LSD Least Significant Digit
Data Set 1	F2	F3	F4	F5	F6
Data Set 2	Shift F2	Shift F3	Shift F4	Shift F5	Shift F6
Data Set 3	Alt F2	Alt F3	Alt F4	Alt F5	Alt F6
Data Set 4	Ctrl F2	Ctrl F3	Ctrl F4	Ctrl F5	Ctrl F6

For data sets with less than five digits, simply omit the commands from the MSD side of the chart. The PMD will display leading zeros e.g. 0004. Data sets can be combined to form up to a 20-digit number.

You may place ASCII characters between digits of the variable data sets or on either side of the entire data set.

For example, let's say you want to show the variable cost of gold in data set one. You would enter: $ F3 F4 F5 F6 # . This code might then represent $4000# to report that it is four thousand dollars per pound.

As another example, let's say you want to show the variable percentage of metal scrapped for one day in data set two. You would enter: Shift F5 Shift F6 % .

This code might then represent **12%**.

Embedded Codes Some features are handled by codes embedded within the message itself. This segment covers these options.

!**Time** - To have the time displayed in a message, **enter `F7`**. The text box displays: **`HH:MM:SS`** to indicate that the time code is programmed. Time occupies eight character locations of the message.

!**Date** - To have the date displayed in a message, **enter `Shift F7`**. The text box displays: **`DD-MMM-YY`** to indicate that the date code is programmed. Date occupies nine character locations of the message.

!**Null Characters** - Null characters are useful when overlaying messages on the display. Overlaid messages do not clear the display and allow previous messages to remain in non-programmed or null areas. To program a null character in the message, **enter `F9`** (for each null location).

Notes: If this message does not clear the display, characters from previous messages can remain in null locations.

!**Control Characters** - Standard ASCII control characters can be embedded in messages. To program ASCII control characters, **enter `^`**, followed by the conventional control code character.

SECTION 4 - OFFLINE PROGRAMMING

!**Functional Commands** - Specific embedded commands in message text cause the unit to perform particular duties. These programmed codes are displayed along with the message text when triggered. The unit is instructed with the text string **#$** followed by an additional letter. These commands are listed below and must be entered in the first three character positions of the first line.

#$A Enable Data Log - This turns data logging function on. (This message can be logged). This option does not change "programmed" data log setting. If you restart the unit, Data Log Enable will return to the software-controlled setting.

#$B Disable Data Log - This turns data logging function off. (This message cannot be logged). This option does not change "programmed" data log setting. If you restart the unit, Data Log Enable will return to the software-controlled setting.

#$C Clear Data Log - This clears data log memory. (This message can be logged. It will be the first logged message in the new data log).

#$D Print Data Log - This causes the data log contents to be sent to the *Terminal/Printer Port*. (Normal PMD operation is suspended while data log is being printed). Data log is printed from first logged message to most recently logged message. The unit outputs one logged message per scan.

#$E Stop Data Log Print - This terminates data log printing and allows message logging to resume. (This message can be logged).

#$F Increment Clock 1 hour (for daylight-savings time) - This does not affect the date setting. The hours will roll over.

#$G Decrement Clock 1 Hour (for daylight-savings time) - This does not affect the date setting. The hours will roll over.

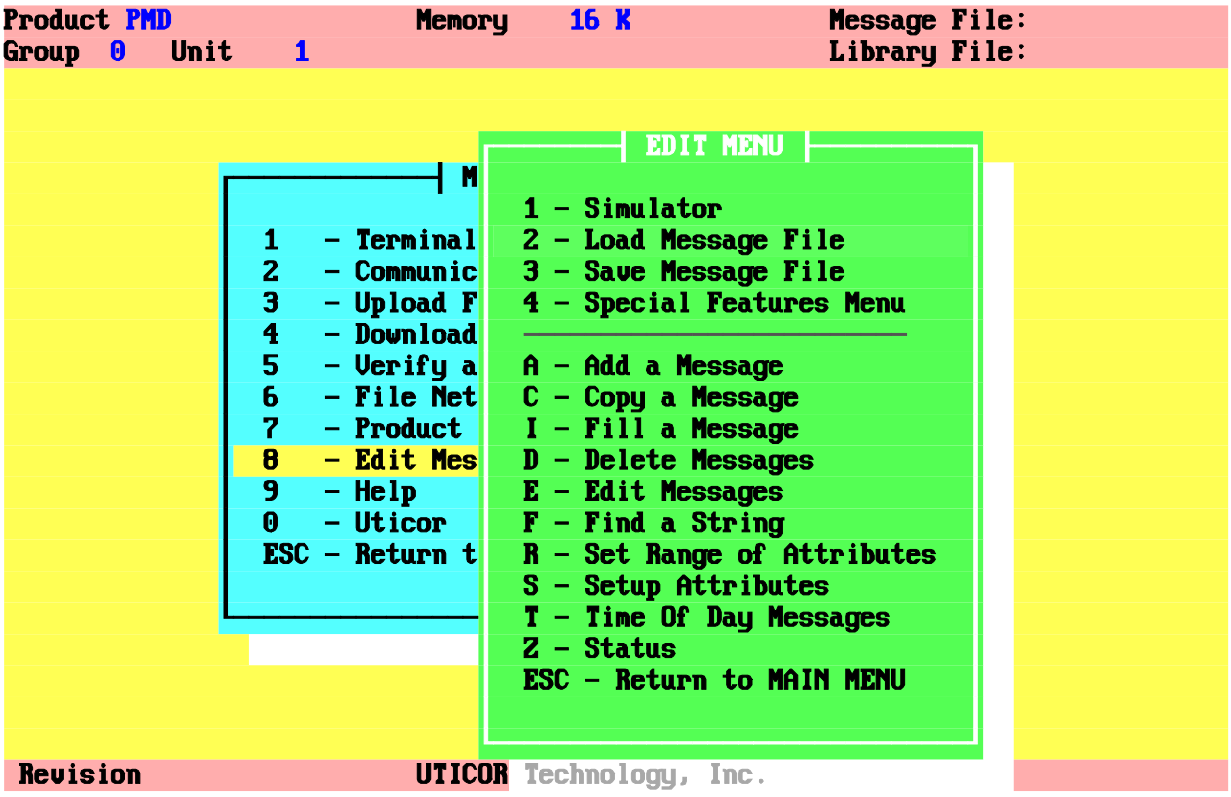
#$H Enable Displaying of Time of Day Messages - When this is embedded at the beginning of a message (the first characters on line one of the message text), this code turns on the time of day message display function. The unit powers up with this option enabled. This function is used to turn the time of day option back on after it has been disabled but has no effect on the display if the option is already enabled.

#$I Disable Displaying of Time of Day Messages - When embedded at the beginning of a message (the first characters on line one of the message text), this code turns off the time of day message display function. No date-triggered messages will be displayed when this option is disabled. This code has no effect on the display if the option is already disabled.

Note: These **Functional Commands** cannot be seen using the **`Simulator`** command (described later in this section) from the main **`EDIT MENU`**. They will simply appear in their code form e.g. #$I, #$G etc. You must be in the DISPLAY MODE, described in **SECTION 6 - DISPLAY MODE** to see their commands applied.

Finally, hit F10 to save your message and then exit the Add Message screen. You are returned to the offline programming main menu, EDIT MENU.

Note: The F10 only saves your previous keystrokes to your computer's RAM. You must still save your changes to disk.



SECTION 4 - OFFLINE PROGRAMMING

Now that you know your way around the message attributes menu, or the **Add a Message** section, we will proceed down the offline programming **EDIT MENU** starting up again with **Copy a Message**, going through **Status**, and then up to complete the top half of the **EDIT MENU**.

Copy a Message

Use this command to copy existing messages to other message number location(s). **Press** C to copy a message. The **COPY MESSAGE**S sub-menu will pop up and give you the following prompts:

Start Source:
End Source:
Start Target:
Confirm:

!**Enter** the message number you want to copy after the **Start Source/End Source** options. Then **enter** the **Start Target**, where you want the message copied. If you only want to copy one message into one other message number location, make the **Start Source** and the **End Source** the same number.

!If you answer the **Confirm** option Y, the software will prevent you from copying over a message that already exists in the **Target** range. When you've answered all four prompts, **press** F2 to copy the message. The program will tell you it is copying the messages and will then ask:
Overwrite Message x : if you answered the **Force Confirmation:** option Y, and will only overwrite it if you tell it to.

Fill a Message

Use this command to fill one message into another message or range of message locations. **Press** I to fill a message. The **FILL MESSAGES** sub-menu will pop up and give you the following prompts:

Source Message #:
Target Start Range:
Target End Range:
Force Confirmation:

!**Enter** the message number you want to fill to somewhere else after the **Source Message #**. Then **enter** the **Target Start Range** and **Target End Range**. If you only want to fill this message into one other message number location, make the **Target Start Range** and the **Target End Range** the same number.

!If you answer the **Force Confirmation** option Y, the software will prevent you from filling into a message number that already has a message in it. When you've answered all four prompts, **press** F2 to fill the message. The program will tell you it's filling the message and will then ask: **Overwrite Message x** if you answered the **Force Confirmation** option Y, and will only overwrite it if you tell it to.

Delete Messages

Use this command to delete an existing message. **Press** D to delete a message. The **DELETE MESSAGES** sub-menu will pop up and give you the following prompts:

```
Start of Range:
End of Range:
Force Confirmation:
```

!**Enter** the message number you want to delete after the **Start of Range:** option. Then **enter** the **End of Range:**, the range of messages that you want to delete. If you only want to delete one message number, make the **Start of Range:** and the **End of Range:** numbers the same number.

!If you answer the **Force Confirmation:** option Y, the software will make you confirm the deletion of each message in the target range. When you've answered all three prompts, **press** F2 to delete the message(s). The program will either ask you to confirm each deletion (if you answered the **Force Confirmation:** option Y) or it will simply tell you that the messages have been deleted (if you answered the **Force Confirmation:** option N).

Edit Messages

Use this command to edit an existing message. **Press** E to edit a message. The program responds with a screen that looks exactly like the one you used for the **Add a Message** command. **Enter** the message number you want to edit and the contents will appear in the message box. Make attribute and/or message content changes and **press** F10 to save the changes.

Find a String

Use this to find messages containing a particular string. **Enter** F; **enter** the word to be searched for exactly as it appears in the message; then **press** Enter. The message numbers that contain your string request will appear on the screen or you will receive the response: **Search String Not Found**.

SECTION 4 - OFFLINE PROGRAMMING

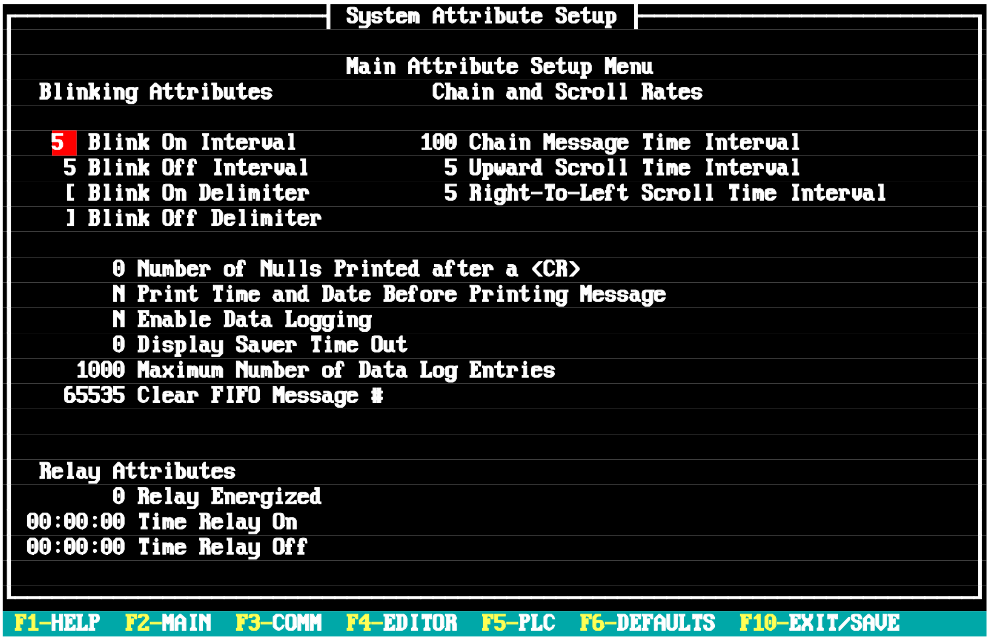
Note: As the note says on the bottom of the **FIND STRING** sub-menu, this command is blink and case sensitive, which reminds you to be conscious of upper or lower case letters in the actual message when you try to locate them as well as whether or not the actual message blinks. (Shift F8 makes the message blink).

Set Range of Attributes

Use this option to set or change the message attributes of all messages within a specified range of message numbers. **Select** **Set Range of Attributes** from the **EDIT MENU** and then **enter** your **Start Range** and **End Range** numbers. Then proceed to **select** the attributes as you would in adding or editing messages.

Setup Attributes

Use this to display the **System Attribute Setup** sub-menu and implement its commands. **Press** S. The program will display the following sub-menu.



SECTION 4 - OFFLINE PROGRAMMING

Blinking Attributes

These commands let you control the blinking attributes. **Blink On Interval/Blink Off Interval** - Use these commands to set the rate at which blinking characters and messages blink on and off in tenths of seconds. **Enter** a number between 1 (0.1 second) and 99 (9.9 seconds) and **press** Enter for each prompt, or just Enter to leave them at the current setting shown within brackets. The program does not accept non-digits or numbers above the range specified.

Blink On Delimiter/Blink Off Delimiter - Use these commands to change character definition for the **`Blink On`** and **`Blink Off`** delimiters. The defaults are [for the **`Blink On delimiter`** and] for the **`Blink Off delimiter`**. If these characters are needed for display in a message, however, either or both can be exchanged for other characters. **Enter** your new choice delimiter at each prompt, or just **press** Enter to leave them at the current setting shown within brackets.

Note: Delimiters can be any printable characters except control characters and the ^ character. Characters used as delimiters cannot be used for any other purpose.

Chain and Scroll Rates

Use these commands to set up time rates for chained and scrolling messages.

Chain Message Time Interval - **Enter** a number between 1 (0.1 second) and 255 (25.5 seconds) to set the amount of time each message within a chained message will remain on the display, or just **press** Enter to leave it at the current setting shown within brackets. If a scrolling message is in the chain, however, it will display until it is done scrolling.

Upward Scroll Time Interval/Right-to-Left Scroll Time Interval - **Enter** a number between 1 (0.1 second) and 99 (9.9 seconds) to set the amount of time each line will remain on each line of the display (for **`Upward Scroll Time Interval`**) or the amount of time each character will remain at each character location (for **`Right-to-Left Scroll Time Interval`**). The terminal does not accept non-digits or numbers above the specified range.

Number of Nulls Printed After a <CR>

Use this command to program a specific number of null characters to be sent to the *RS-232 port* after each <CR>. **Enter** a number between 0 and 255 and **press** Enter, or Enter only to leave it in its current state, shown within brackets. The program does not accept non-digits or a number higher than 255.

Note: Some printers require a delay time after receiving a carriage return to prevent loss of characters at the beginning of lines. This command allows the creation of a delay (after <CR>) which may be adjusted to the required value.

Print Time and Date Before Printing Messages Use this command to enable/disable printing the time and date before each message. **Enter** Y or N.

Enable Data Logging Use this command to enable/disable the data log. If you **enter** Y, logged messages will be stored in the data log memory when selected. If you **enter** N, no messages will be stored in the data log memory when selected.

Display Saver Timeout Use this command to clear the display of the same message after a set number of hour(s). **Enter** a number between 1 and 24 to select the number of hours you wish to leave the message unchanged on the display if no other messages are selected, or **press** Enter to leave this option undefined or at a prior setting.

Note: If you **select** 0, you will **disable** this option.

Maximum Number of Data Log Entries Use this option to determine the number of entries (message numbers and variable data values) that the data log will hold. When enabled, the data log will store this number of readings and discard the oldest records.

Note: The largest value allowed is 3000.

Clear FIFO Message # (First-In-First-Out) Use this command to select a message number to be used on the PLC Port to clear the incoming message FIFO (First-In-First-Out). **Enter** a number between 0 and 65535 and **press** Enter, or just Enter to leave it at its current setting shown within brackets. When selected in the DISPLAY MODE, this message number will cause the incoming message FIFO to empty (clear memory). The FIFO must contain less than 32 entries for it to accept the clear FIFO number.

Relay (Output Terminals) Attributes Use these commands to determine the length of time that the *Relay Output Terminals* will be energized when activated by a selected message, or to set the internal timer for the *Relay Output Terminals*.

Relay Energized Time Interval - Enter 0 to have the *Relay Output Terminals* energized until a new message is selected. **Enter** a number between 1 and 255 for the number of seconds that you wish to have the *Relay Output Terminals* energized each time that a message programmed to energize the relay output is selected.

Timed Relay On Time/Timed Relay Off Time - You can also set up the timed *Relay Output Terminal* activation with the clock settings. **Set** the time you want the relay energized and when you want it turned off.

System Attribute Setup Screen Assistance

Note that this **`System Attribute Setup`** sub-menu provides the following services using the function keys at the bottom of the screen:

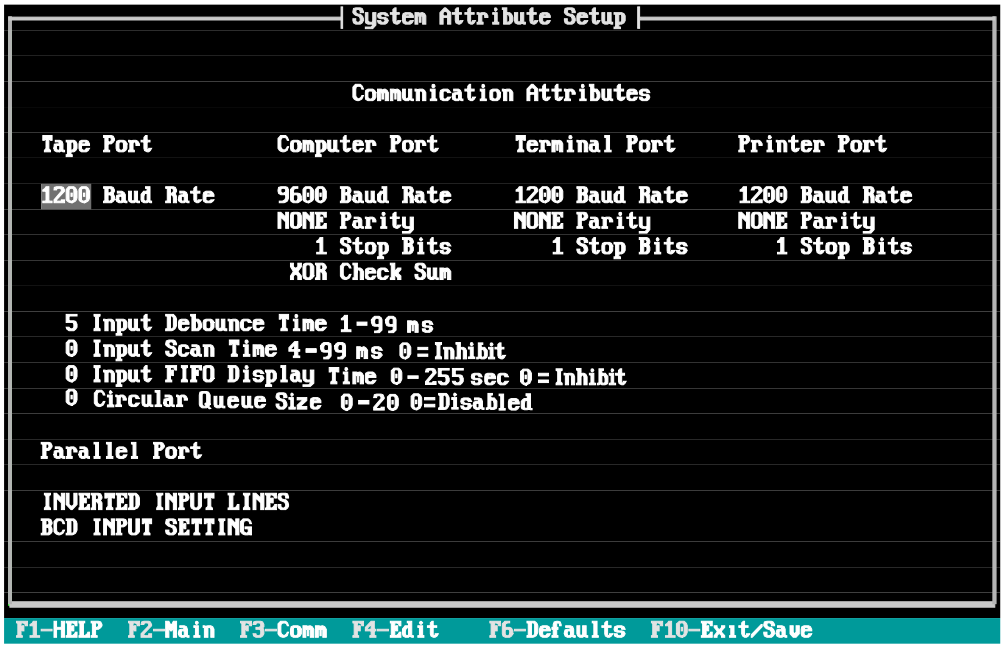
F1-HELP : tells you the time range for each attribute e.g. For the Blink On/Blink Off times, you can choose any number between 1 and 99.

F2-Main : lets you go back to the **`Main Attribute Setup Menu`** of the **`System Attribute Setup menu`** after you've been using one of the other function keys.

Communication Attributes

F3-Comm.: lets you set up the PMD communication attributes. Use the arrow keys and space bar as you did on the **`Add a Message`** menu to change the values on the sub-menu that pops up.

WARNING! Baud rates are stored with the message file.



!**Input Debounce Time** - Does not apply to PMD 400 Series. Leave the setting at the default .

!**Input Scan Time** - Does not apply to PMD 400 Series. Leave the setting at the default .

!**Input FIFO Display Time** is an especially useful option when your system calls up a series of alarm messages at virtually the same time. You can set this option (0-255 sec) so that each message in the FIFO list will be displayed for your set time duration before the next message in the FIFO is displayed. If you set it to 0, it is disabled. And if the circular queue (described next) is enabled, the FIFO is disabled and no variable data can be s

!**Circular Queue Size** is only applicable to the direct PLC interface units. Use this option to set how many messages can be contained and circled through indefinitely. When the queue size is set to 0, the queue is disabled; if it is greater than 0, it is enabled. The circular message queue is discussed in greater detail in **SECTION 6 - THE DISPLAY MODE** and **SECTION 7 - PLC INTERFACING**.

e
n
t
t
h
r
o
ugh the message/data word.

!**INVERTED/NON-INVERTED INPUT LINES** lets you determine how the *PLC port* looks at the message/data lines: the way they come in as 1s or 0s (non-inverted) **or** the complements of these (inverted). If you have a direct PLC interface unit, make sure it is set for non-inverted.

!**BCD/BINARY INPUT SETTING** lets you determine whether the 400 unit will consider the input data lines as a Binary or Binary Coded Decimal number.

System Attribute F4-Edit: lets you change your default editor attributes
Setup Screen
Assistance

cont.

F5-PLC: lets you set up attributes unique to your PLC interface unit. Refer to the following parts of **SECTION 7 - PLC INTERFACING**.

PART A: 350 A-B INTERFACE
PART B: 360 SIEMENS/TI INTERFACE
PART C: 370 MODICON INTERFACE
PART D: 375 MODICON MODBUS PLUS INTERFACE
PART E: 380 GE GENIUS I/O INTERFACE

SECTION 4 - OFFLINE PROGRAMMING

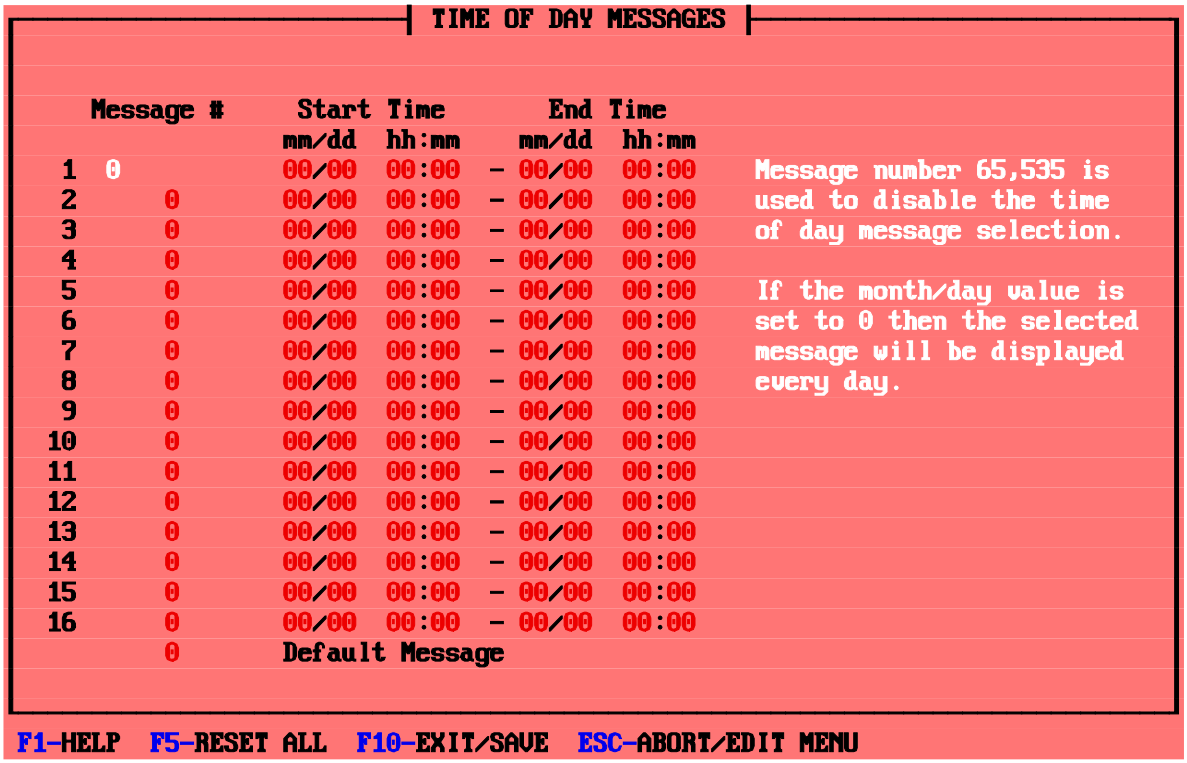
F6-Defaults: changes settings to factory defaults
F10-Exit/Save: saves the changes you've made and exits the **`System Attribute Setup menu`**.

Time of Day Messages

Use this option to program up to 16 time- and date-triggered messages. When you select it off the **`EDIT MENU`**, the following screen appears. Insert the message(s) you want to be displayed, and then designate start times and dates as well as end times and dates. (If the month/day columns are set to 0, the selected message will be displayed every day). Use the arrow keys and the Enter key to move from column to column. This option only works if you download your messages through the RS-422 *Computer Port*; it will not work if your message file is downloaded through the *tape port*.

Status

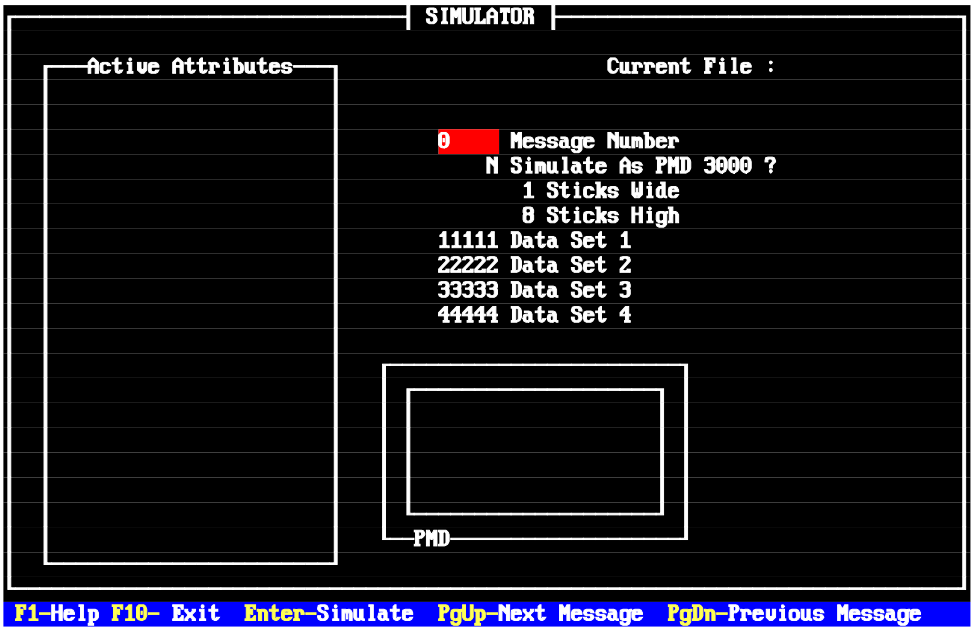
This option simply gives you your program status: **`Largest Message, Smallest Message, Total Messages, Bytes Used, Bytes Free`** and **`Message Numbers in Use`**.



SECTION 4 - OFFLINE PROGRAMMING

Simulator

This option allows you to see a simulated version of your programmed messages as well as their active attributes (blinking, centering etc.) as they would appear on the display unit. This option may be particularly helpful to you as you experiment with the **`Add a Message`** option explained in the beginning of this section. When you select **`Simulator`**, the following screen will appear.



Save Message File

Use this option to save the messages (and their attributes) you've been creating. They can be saved to your hard drive or on disk. **Select `Save Message File`** from the **`EDIT MENU`**. **Select** F3 to change the directory if you would like. **Type** in your directory choice (e.g. A:\). **Type** in a name for your message file (e.g. TEST). This file name will appear at the bottom of your screen after **`Select Filename:`** as it is typed. The program will then quickly respond with **`Saving Message x`** for each message saved before returning you to the **`EDIT MENU`**.

Load Message File

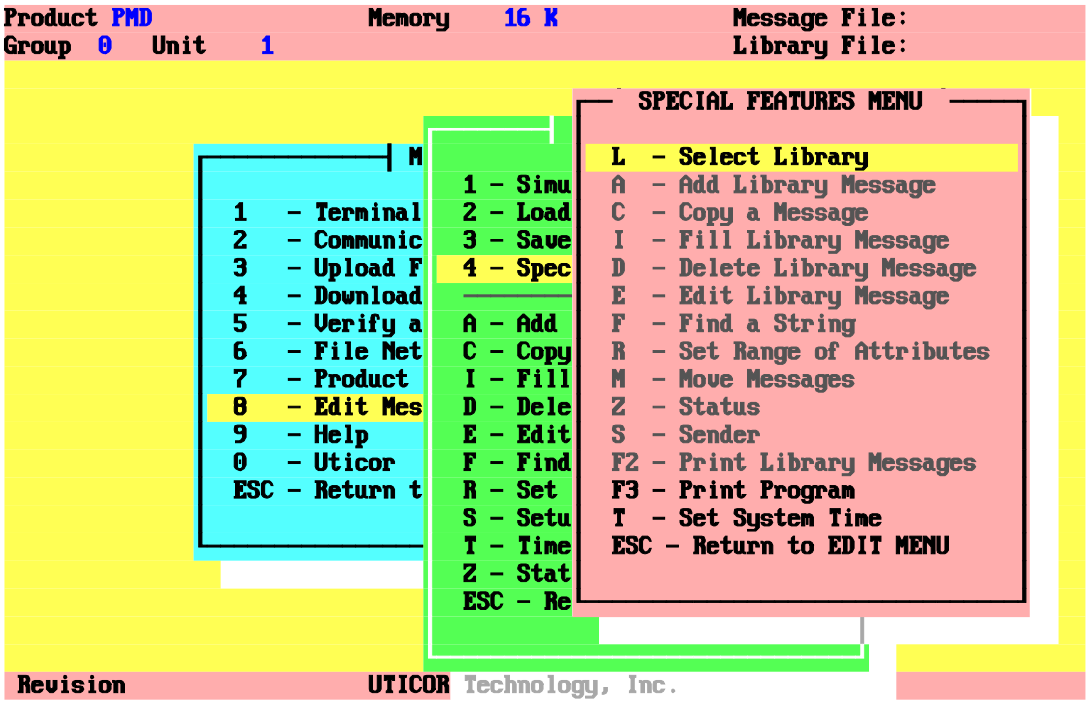
Use this option to load your message file saved on disk to user memory in the computer. **Select `Load Message File`** from the **`EDIT MENU`**. Make sure that **`Directory:`** at the top of your screen represents your disk directory (e.g. C:\). **Hit** Enter to load each message file. The program will then quickly respond with **`Loading Message x`** for each message that is part of that file before returning you to the **`EDIT MENU`**.

SECTION 4 - OFFLINE PROGRAMMING

WARNING! Baud rates are stored with the message file.

Special Features Menu

This command is especially useful to operators who use many of the same messages for a variety of applications. You'll notice that the **SPECIAL FEATURES MENU** closely resembles the **EDIT MENU** as so many of its options are the same. **Select** **Special Features Menu** from the **EDIT MENU**. The **SPECIAL FEATURES MENU** appears on your screen:



Select Library

Use this command to create a library. **Select** **Select Library** from the **SPECIAL FEATURES MENU**. **Type** in a name for a new library e.g. your name. You now have a library "file" from which messages can be transferred to message files and message files can be transferred to your library file(s).

Menu Options

For **Add Library Message**, **Copy a Message**, **Fill Library Message**, **Delete Library Message, Edit Library Message, Find a String, Set Range of Attributes,** and **Status**, these options all behave as they do in the main **EDIT MENU**. Refer to the beginning of this section.

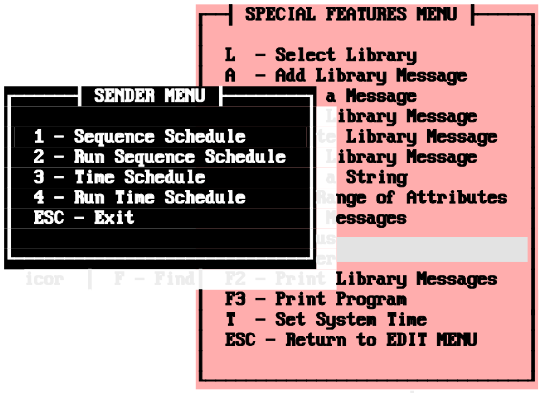
Move Messages

This is the key option for the **SPECIAL FEATURES MENU**. Use this option to move a message(s) from a library file to a message file or message(s) from a message file to a library file. **Enter Move Messages** from the **SPECIAL FEATURES MENU**. You will then need to respond to the following prompts:

```
Move to Library:          N/Y
Start of Source Range:    0
End of Source Range:      0
Start of Target Range:    0
Force Confirmation:       Y
```

!For **Move to Library**, Y moves messages from the message file to the library. N moves messages from the library to the message file. For **Start of Source Range/End of Source Range**, you insert your message numbers to be moved; and for **Start of Target Range**, you insert the number where the messages should be moved to. **Force Confirmation** will have you confirm each move.

Sender (software)



This is UTICOR software that will allow messages stored in a file to be automatically sent to a display unit and displayed (They are not programmed into the PMD). It is intended for intra-plant employee communication. The message can be sent in a predetermined sequence or based on the time of day.

Communication to the unit is through its RS-422 *Computer Interface Port*. **Select Sender** from the **SPECIAL FEATURES MENU** and this sub-menu will appear on your screen.

!**Sequence Schedule** allows a sequence of messages to be entered. **Enter** the message number, time duration (0-99 seconds) that the message will be displayed, and the group and unit numbers of the display products.

!**Run Sequence Schedule** allows a sequence of messages to be sent to display unit(s). To start the sequence, **press** F9; the sequence will keep repeating until F3 is pressed. F4 is a toggle key which halts and restarts the sequencing. Your displayed message(s) and a decrementing timer will appear at the top of your screen as they are sent to the display(s).

!**Time Schedule** allows messages to be sent to displays based on the time of day. **Enter** a message number and the time/date it is to be sent. Group and unit numbers can also be entered.

!**Run Time Schedule** allows messages to be sent to displays based on the time of day. **Press** F9 to start the time of day sequence. The program must be left in this mode to operate properly.

Printing You can send a message file or library file to a printer, the screen, or a file. You print to a printer connected directly to the computer. **Select** the **Special Features Menu**, and from it, either the **Print Library Messages** or the **Print Program option**. **Set** the print attributes and **hit** Enter.

F2 Print Library Messages Use this command to print the current library file. You can print the entire file or a range of messages with their corresponding attributes.

F3 Print Program Use this command to print the current message file while using online or offline applications. You can print the entire file or a range of messages with their corresponding attributes.

T Set System Time Use this option to reset the computer clock. **Select Set System Time** from the **SPECIAL FEATURES MENU** and respond to the prompts.

This completes **OFFLINE PROGRAMMING** development. You have been introduced to all of the tools needed for developing, editing and viewing message programs. Of course, the best way to become more knowledgeable about the versatility of your 400 unit is to practice with all of the menus and sub-menus and refer to the help menus.

SECTION 5 - PROGRAMMING EXAMPLES

This section is intended to help you apply much of the knowledge you have acquired in your programming section. As this section is broken into two parts: **Part A: Online Programming Examples** and **Part B: Offline Programming Examples**, you can continue implementing your programming skills via the programming method you have chosen.

You will be lead step-by-step from the beginning of the desired result to the end. The three examples in each part include the system and message attributes which most of UTICOR's users have indicated they use often, or with which they encounter difficulty.

Before you begin, make sure you have followed the "Getting Started with On/Offline Programming" procedure in **SECTION 2 - GETTING STARTED**.

PART A: ONLINE PROGRAMMING EXAMPLES

Example 1: Includes
! **Center Message**
! **Clear Display**
! **Overlay a Message**

Let's say you want to program three messages into the master 400 unit. You want two of the three messages to be centered on the display and clear the previous message's text from the screen before it is displayed. You want the third message to be uncentered and "overlay" the message displayed before it.

1. To start programming your messages, **put** the 400 unit into PROGRAM MODE by using the MODE and ENTER push buttons.

2. **Select** **`Terminal Emulator`** from the **`MAIN MENU`** on the computer.

3. **Press** Enter on the computer, then H to list the **`COMMANDS`**.

Note: All commands are coming from the 400. Your computer is now just a dumb ASCII terminal. All data entered is going directly to the 400 unit, not the disk or RAM memory in the computer.

4. **Press** A to add your first message. The terminal responds: **`ADD MESSAGE #`**. **Enter** the number 1 and **hit** Enter. The terminal responds: **`REPLACE MESSAGE CONTROL OPTIONS (Y/N)?`**

5. **Enter** Y. The terminal shows the **`OPTION MENU`**.

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

```
                          OPTION MENU

1. CHAIN MESSAGE              <N>  15. CLEARS DISPLAY       <Y>
2. REPEAT CHAIN MESSAGE       <N>  16. CENTER MESSAGE       <N>
                                   17. STARTING DATA SET    <1>
3. SCROLLS UPWARD             <N>
4.   BOTTOM LINE              <1>  18. SEND TO SLAVE        <N>
5.   TOP LINE                 <1>  19.   TO ALL SLAVES      <Y>
6.   REPEAT                   <N>  20.   GROUP              <0>
7.   SCROLL TIME INTERVAL     <0>  21.   UNIT               <0>

8. SCROLLS LEFT               <N>  22. ALARM RELAY          <N>
9.   SCROLL LINE              <1>  23. INVISIBLE MESSAGE    <N>
10.  REPEATS                  <N>  24. LOG MESSAGE          <N>
11.  SCROLL TIME INTERVAL     <0>  25. BLINK ENTIRE MSG     <N>

12.SEND MSG TO RS232 PORT     <N>
13.  ON ONE LINE              <N>  26. DS PROTECT N N N N
14.  <CR><LF> AT END OF MSG   <N>  98. Help
                                   99. Exit

Enter Selection:
```

6. **Enter** option **16. CENTER MESSAGE** (if it is **not** currently set at <Y>). The **OPTION MENU** will disappear from your terminal screen and then reappear with the **16. CENTER MESSAGE** option changed to <Y>, meaning all lines of message 1 will be centered on the display.

7. From this same menu, **enter** option **15. CLEARS DISPLAY** (if it is **not** currently set at <Y>). The **OPTION MENU** will disappear from your terminal screen and then reappear with the **15. CLEARS DISPLAY** option changed to <Y>, meaning message 1 will clear the display of any other text before it is displayed.

8. Now that you have set these options, **select** 99. Exit to start entering your text for message 1.

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

```
@Oh, say can you
@see, by the dawn's
@early light...
```

9. When the terminal displays @, you may start entering message text (Please **enter** the text shown at the left). When the twentieth character is entered on one line, the program terminal will sound a bell or beep to indicate that a PMD 400 standard line has been programmed. **Press** Enter to advance to the next line and continue entering text, or **press** Ctrl Z to end the message. The terminal now asks: **`ADD MESSAGE #2 (Y/N) <Y>?`**

10. **Press** Y and proceed with steps 4-9 to add your second message in the same manner. Please put **`Hello!`** on the first line, **`Bonjour!`** on the second line, and **`Hola!`** on the third line. **Press** Ctrl Z to end message. The terminal now asks: **`ADD MESSAGE 3 (Y/N) <Y>?`**

11. **Press** Y and **hit** Enter. The terminal responds: **`REPLACE MESSAGE CONTROL OPTIONS (Y/N)?`**

12. **Enter** Y. The terminal again shows the **`OPTION MENU`**.

13. **Enter** option **`15. CLEARS DISPLAY`** to change it to N, and option **`16. CENTER MESSAGE`** to also change it to N.

14. Now that you have set these options, **select `99. Exit`** to start entering text for message 3. Please leave the first three lines blank by just **hitting** the Enter key for each, and **type** in **`Welcome Visitors!`** for the fourth line.

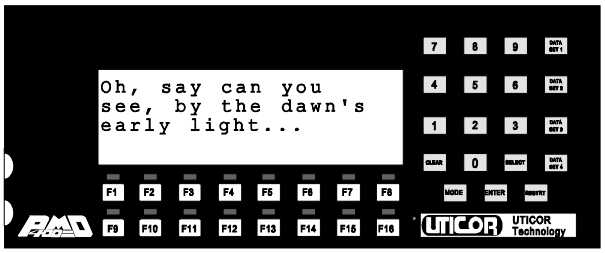
15. **Enter** Ctrl Z to end the message. After you have added this third message, answer the **`ADD MESSAGE #4 (Y/N)`** as N to be returned to the **`MAIN>COMMANDS`** menu.

16. Finally, you can view the messages you have just programmed by **selecting `VIEW MESSAGES`** from the **`MAIN>COMMANDS`** menu and **typing** in message 1, 2 and 3 independently.

Message 1

!Message 1 will appear on the unit as shown: a centered message that will clear any message displayed before it. If you had **not** selected the **`CENTER MESSAGE`** option, the message would be displayed as follows.

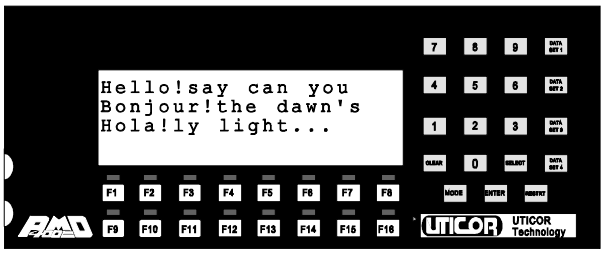
SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES



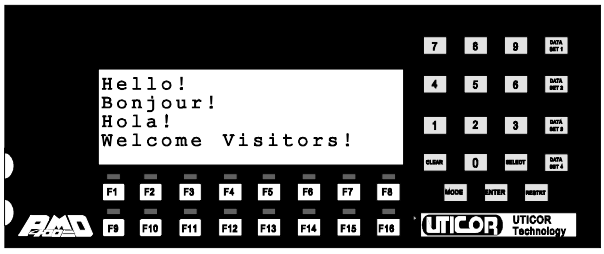
uncentered message

Message 2

!Message 2 will appear on the display as shown: a centered message that will clear any message displayed before it. If you had **not** chosen the **CLEARS DISPLAY** and **CENTERS MESSAGE** options for message 2, part of message 1 would remain on the display while message 2 is displayed, as shown below.



Message does not center or clear display.



Message 3

!Message 3 will appear as shown: an uncentered message that does not clear message 2 before it is displayed; it "overlays" message 2. While the display directly above this paragraph exemplifies a mistake, this example shows how you can choose **not** to clear the display and overlay messages to your advantage.

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

Example 2: Includes
! **Scrolls Left**
(Repeating)
(Change scroll rate)
! **Variable Data Sets**
! **Time/Date**
Triggered Messages

Let's say you want to program a message that repeatedly scrolls left, has variable data in it, and will be displayed on a specific date at a specific time.

1. **Put** the 400 into PROGRAM MODE by using the MODE and ENTER push buttons.

2. **Select** **`Terminal Emulator`** from the **`MAIN MENU`** on the computer.

3. **Press** Enter on the computer, then H to list the **`COMMANDS`**.

4. **Press** A to add your message. The terminal responds: **`ADD MESSAGE #`**. **Enter** the number 10 and **hit** Enter. The terminal responds: **`REPLACE MESSAGE CONTROL OPTIONS (Y/N)?`**

5. **Enter** Y. The terminal shows the **`OPTION MENU`**:

```
                                OPTION MENU

1. CHAIN MESSAGE              <N>  15. CLEARS DISPLAY       <Y>
2. REPEAT CHAIN MESSAGE       <N>  16. CENTER MESSAGE       <N>
                                   17. STARTING DATA SET    <1>
3. SCROLLS UPWARD             <N>
4.   BOTTOM LINE              <1>  18. SEND TO SLAVE        <N>
5.   TOP LINE                 <1>  19.   TO ALL SLAVES      <Y>
6.   REPEAT                   <N>  20.   GROUP              <0>
7.   SCROLL TIME INTERVAL     <0>  21.   UNIT               <0>

8. SCROLLS LEFT               <N>  22. ALARM RELAY          <N>
9.   SCROLL LINE              <1>  23. INVISIBLE MESSAGE    <N>
10.  REPEATS                  <N>  24. LOG MESSAGE          <N>
11.  SCROLL TIME INTERVAL     <0>  25. BLINK ENTIRE MSG     <N>

12.SEND MSG TO RS232 PORT     <N>
13.  ON ONE LINE              <N>  26. DS PROTECT N N N N
14.  <CR><LF> AT END OF MSG   <N>  98. Help
                                   99. Exit

Enter Selection:
```

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

6. **Enter** option **`8. SCROLLS LEFT`**. The **`OPTION MENU`** will disappear from your terminal screen and then reappear with the **`8. SCROLLS LEFT`** option changed to <Y>, meaning this message number 10 will scroll from the right of the display to the left.

7. From this same menu, **enter** option **`9. SCROLL LINE`** and **type** in 2 so that your message will scroll across the second line of the display.

8. Finally, **enter** option **`10. REPEATS`** and make sure it is <Y> so that the scrolling message will repeat several times.

9. Now that you have set these options, **select `99. EXIT`** to start entering your text for message 10.

```
@ The temperature
@ today is 345
@ Sales increased
@ 45%!
```

10. When the terminal displays @ you may start entering message text. **Type** in **`The temperature`** for the first line. **Press** Enter to advance to the next line, **press** the space bar once, and **type** in **`today is`**. Since we want to program variable data in this message, press **`CTRL W  CTRL X CTRL Y`** (they will appear on your screen as **`345`**) to form a three digit number for the temperature, and then **type** in a period before **pressing** Enter.

11. On the third line, **press** the space bar once, **type** in **`Sales increased`**, and **press** Enter to advance to the fourth line. **Press** the space bar once and type in **`CTRL X  CTRL Y`** (they will appear on your screen as **`45`**), followed by a % and an ! to form a two digit number for a percentage rate. The terminal now asks: **`ADD MESSAGE #11 (Y/N) <Y>?`**

12. **Press** N.

13. Now to program message 10 to be displayed on a specific date at a specific time, **select `Set Up Parameters`** from the **`MAIN>COMMANDS`** menu and **press `15) TIME OF DAY MESSAGES`**. The terminal responds with the following sub-menu.

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

```
                              Date Triggered Messages
```

```
Default Time Of Day Message:  XXXXX

   MSG#   On           Off            MSG#     ON           OFF
1.   MMM DD HH:MM MMM DD HH:MM 9. MMM DD HH:MM  MMM DD HH:MM
2.                                  10.
3.                                  11.
4.                                  12.
5.                                  13.
6.                                  14.
7.                                  15.
8.                                  16.

1. Modify An Entry
2. Initialize List
3. Modify Default Time Of Day Message
4. Exit
```

14. **Select** option **`1. Modify an Entry`**. You are prompted with: **`ENTRY NUMBER`**.

15. **Enter** 1 and **press** Enter. The terminal responds: **`DELETE (Y/N) <N>`**:

16. **Press** N. The terminal responds: **`MESSAGE NUMBER <xxxxx>`**:

17. **Enter** message 10 and **press** Enter. Now the terminal responds: Every Day <Y>:

18. **Press** N. Now you must answer the next four prompts to set the time that message 10 will start to be displayed:

```
ON MONTH (1-12) <MMM>:
ON DAY (1-31) <DD>:
ON HOUR (0-23) <HH>:
ON MINUTE (0-59) <MM>:
```

19. Answer these prompts with today's month and date; and set it to display within this hour and a few minutes so you can see the results.

20. The same four prompts above will also ask you to designate a time to have the message stop being displayed. Again, answer these prompts with today's month and date; and set it to turn off within fifteen minutes.

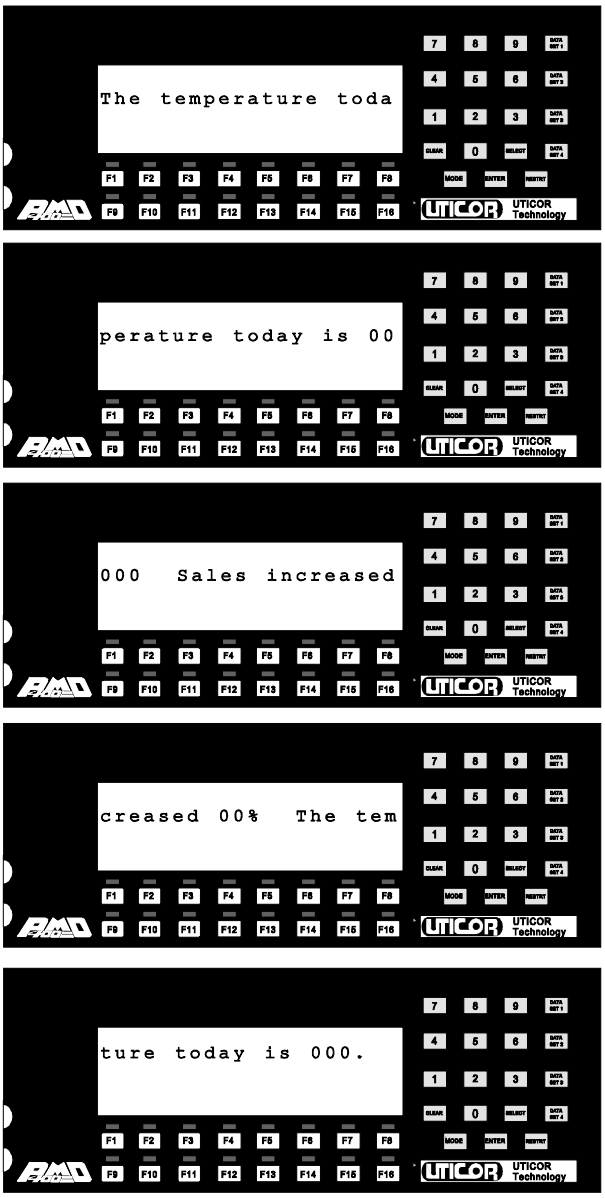
SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

21. **Press** 4. Exit to be returned to the **MAIN>COMMANDS** menu.

22. For this example, slow down the scroll rate so that you can easily see message 10 as it is displayed. From the **Set Up Parameters** menu again, **select 6) RATE FOR SCROLL AND CHAIN MESSAGE**. From the prompts that appear, **select** option **3) DEFAULT TIME RATE FOR SCROLL LEFT <5>** and **enter 20.**

23. **Put** the unit in DISPLAY MODE by using the MODE and ENTER push buttons.

Message 10 will be displayed during the time period you have just assigned with zeroes holding the digit places for the variable data (since it is not connected to a PLC) as it scrolls from the right of the display to the left on the second line.



SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

Example 3: Includes
! **Edit a Message**
! **Print a Message**
! **Log a Message**
! **Embedded Codes**
! **Chain Message**

Let's say you want to edit the three messages (1, 2 and 3) you programmed in **Example 1**. In addition to the original message attributes, you want each message to be logged and printed; you want to add an embedded code to message 1; and you want to chain all three messages together.

1. To resume programming, **put** the 400 unit into the PROGRAM MODE by using the MODE and ENTER push buttons.

2. **Select** **`Terminal Emulator`** from the **`MAIN MENU`** on the computer.

3. **Press** Enter on the computer, then H to list the **`COMMANDS`**.

4. **Select** S to see the **`SET UP PARAMETERS`** sub-menu. From it, **press** 10 and Enter to enable the data log. The terminal responds with the prompts:

```
DATA LOGGING <DISABLED> *
1) ENABLE
2) DISABLE
3) CLEAR
4) SIZE OF LOG <1000>
```

* The current state of options 1) and 2) is shown within these brackets.

5. **Select** option **`1) ENABLE`** (if it is **not** already selected) so that these three messages can be stored in the data log buffer when programmed to do so and when they are selected.

6. **Press** H again to list the **`COMMANDS`** menu.

7. **Press** E to edit message 1. The terminal responds: **`EDIT MESSAGE #`**.

8. **Enter** 1 and **press** Enter. The terminal responds with the programmed options followed by: **`REPLACE MESSAGE CONTROL OPTIONS (Y/N)?`**

9. **Type** in Y. The terminal shows the **`OPTIONS MENU`**.

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

```
                              OPTION MENU

1. CHAIN MESSAGE              <N>  15. CLEARS DISPLAY       <Y>
2. REPEAT CHAIN MESSAGE       <N>  16. CENTER MESSAGE       <Y>
                                   17. STARTING DATA SET    <1>
3. SCROLLS UPWARD             <N>
4.   BOTTOM LINE              <1>  18. SEND TO SLAVE        <N>
5.   TOP LINE                 <1>  19.   TO ALL SLAVES      <Y>
6.   REPEAT                   <N>  20.   GROUP              <0>
7.   SCROLL TIME INTERVAL     <0>  21.   UNIT               <0>

8. SCROLLS LEFT               <N>  22. ALARM RELAY          <N>
9.   SCROLL LINE              <1>  23. INVISIBLE MESSAGE    <N>
10.  REPEATS                  <N>  24. LOG MESSAGE          <N>
11.  SCROLL TIME INTERVAL     <0>  25. BLINK ENTIRE MSG     <N>

12.SEND MSG TO RS232 PORT     <N>
13.  ON ONE LINE              <N>  26. DS PROTECT N N N N
14.  <CR><LF> AT END OF MSG   <N>  98. Help
                                   99. Exit

Enter Selection:
```

10. **Enter** option **12. SEND MSG TO RS232 PORT** (if it is **not** currently set at <Y>) and Enter to program message 1 to print each time it is selected.

```
REPLACE LINE (Y/N)? N
@see, by the dawn's
REPLACE LINE (Y/N)? N
@early light...
REPLACE LINE (Y/N)? N
@HH:MM:SS
```

11. Now **enter** option **24. LOG MESSAGE** (if it is **not** currently set at <Y>) and Enter to store message 1, as well as the time and date it was triggered, every time it is selected.

12. Now that you have set these options, **select 99. Exit** to start editing the text of message 1.

13. The terminal will ask you if you want to **REPLACE LINE (Y/N)?** for each of the four lines. **Enter** N for the first three lines.

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

14. When the @ appears for the last line, we're going to program an **Embedded Code** to have the current time be displayed in message 1. (More **Embedded Codes** are described in **SECTION 3-ONLINE PROGRAMMING, PART B: MESSAGE ATTRIBUTES**). **Enter** Ctrl T. The terminal displays **`HH:MM:SS`** to indicate that the time code is programmed.

15. **Enter** Ctrl Z to complete the message.

16. Now, to program messages 2 and 3 to print and log also, follow steps 6-12. When the terminal asks you if you want to **`REPLACE LINE (Y/N)?`** for each of the four lines, simply respond N and **enter** Ctrl Z after you have responded N to the last line.

17. Finally, we will create a **Chain Message** which chains messages 1, 2 and 3 together as one message.

18. **Press** H to list the **`COMMANDS`** again.

19. **Select** A to add a message. The terminal responds: **`ADD MESSAGE #`**.

20. **Enter** 23. The terminal responds with the message options and asks: **`REPLACE MESSAGE CONTROL OPTIONS (Y/N)?`**.

21. **Hit** Y. **Select** 1. **`CHAIN MESSAGE`** and **press** Enter.

22. Also **select `2. REPEAT CHAIN MESSAGE`** (if it is **not** already set to <Y>).

```
2>1
3>3

ADD MESSAGE #24 (Y/N)
<Y>?
```

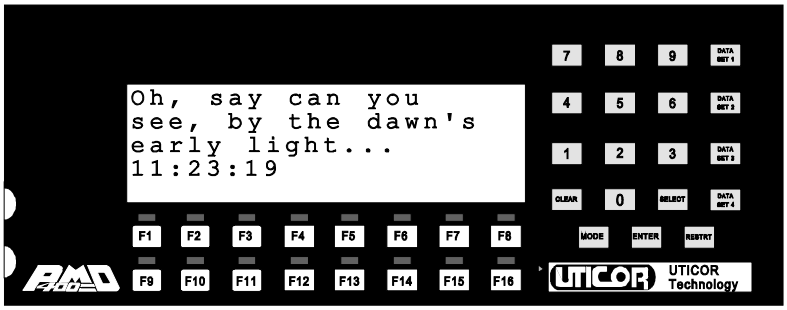
23. **Enter** 99. Exit to start making the chain. The terminal responds: **`1>`**, prompting you for the first message of the chain.

24. **Type** in 1 and Enter, then 2 and Enter, then 3 and Enter. Finally, **hit** Enter twice to end the chain. The terminal will ask: **`ADD MESSAGE #24 (Y/N) <Y>?`**

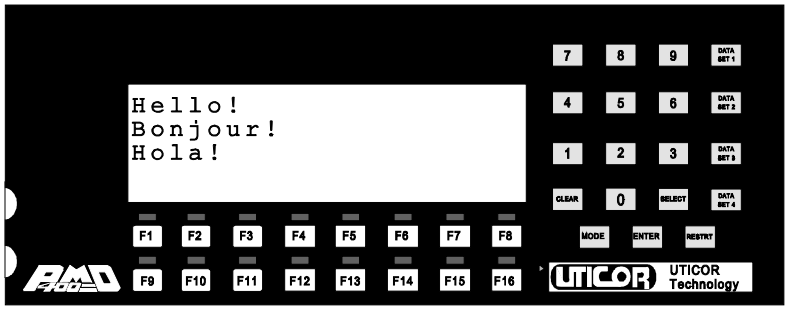
25. **Enter** N.

SECTION 5 - PROGRAMMING EXAMPLES
PART A: ONLINE PROGRAMMING EXAMPLES

26. You are finished programming, so to view this chained message 23 which includes messages 1, 2 and 3 and all of their message attributes, **select** V from the **MAIN>** menu and **type** in 23. A simulation of what your unit displays is illustrated below:



Message 1 of Chained Message 23



Message 2 of Chained Message 23

Message 3 of Chained Message 23

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES

Example 1: Includes
! **Center Message**
! **Clear Display**
! **Overlay a Message**
! **Save Message File**

Let's say you want to program three messages. You want two of the three messages to be centered on the display and clear the previous message's text from the screen before they are displayed. You want the third message to be uncentered and "overlay" the message displayed before it. You will save all three as a message file to disk to be displayed on a standard PMD 400.

1. To start programming your messages, **select** **`Edit Message Program`** from the **`MAIN MENU`** on the computer.

2. **Press** A to add your first message. The program responds with the following menu:

```
Product PMD                Memory    128 K          Message File:
Group  0   Unit    1                                 Library File:

     1 Message Number        397    Bytes Used        130675 Bytes Free

 N Chain Message      N Scroll Upwards    N Scroll Left       N Send to RS232
   N Repeat             0 Bottom Line       0 Scroll Line       N On 1 Line
 Y Clear Display        0 Top Line          N Repeat Msg        N <CR> <LF>
 N Center Message       N Repeat Msg        0 Scroll Rate
 N Blink Message        0 Scroll Rate

 N Energize Alarm Relay
 N Invisible Message
 N Log Message

 N Send Message to Slave
   N All Slaves
   0 Group Number
   0 Unit Number                                    1
  3000 Frame Information                            2
     0 Upper Row                                    3
     0 Lower Row                                    4
 N N N N Data Sets Disable

F1-HELP F2-Send F3-Program F4-SizeF6-Edit F10-Exit/Save SPACE-Y/N PgUp/PgDn ESC
```

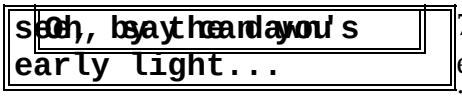
3. **Enter** the number 1 and **hit** Enter.

4. Use the arrow keys to move down to the **`Clear Display`** option and use the space bar to toggle it to Y, meaning message 1 will clear the display of any other text before it is displayed.

5. Again, use the arrow keys to move to the **`Center Message`** option and use the space bar to toggle it to Y, meaning all lines of message 1 will be centered on the display.

6. Now that you have set these options, **press** F6 to move the cursor to the message text area and start entering your text for message 1.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES



7. **Enter** the text shown at the left. When the twentieth character is entered on one line, the program terminal will sound a bell or beep to indicate that a PMD 400 standard line has been programmed. **Press** Enter to advance to the next line and continue entering text.

8. **Press** F10 twice to exit and save this message.

9. **Press** A to add your second message in the same manner. Please put **`Hello!`** on the first line, **`Bonjour!`** on the second line, and **`Hola!`** on the third line.

10. Again, **press** F10 to exit and save this message.

11. **Press** A to add the third message. The program again shows the **`Add a Message`** menu.

12. This time, **toggle `Clear Display`** and **`Center Message`** to N.

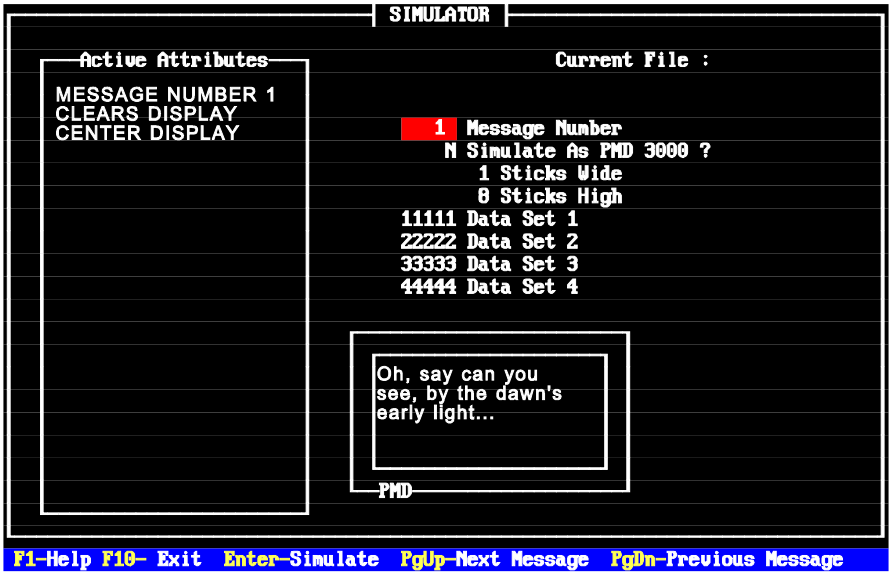
13. Now that you have set these options, **press** F6 to start entering your text for message 3. Please leave the first three lines blank by just **hitting** the Enter key for each, and **type** in **`Welcome Visitors!`** for the fourth line.

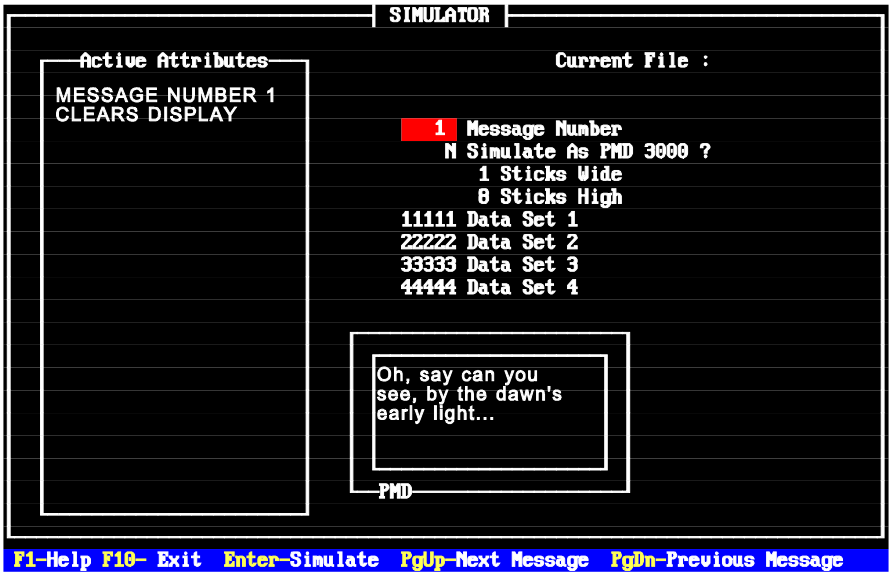
14. **Press** F10 to exit and save.

15. Finally, you can view the messages you have just programmed as well as their active attributes as they would appear on a display unit by **selecting `Simulator`** from the main **`EDIT MENU`** and **typing** in messages 1, 2 and 3 independently.

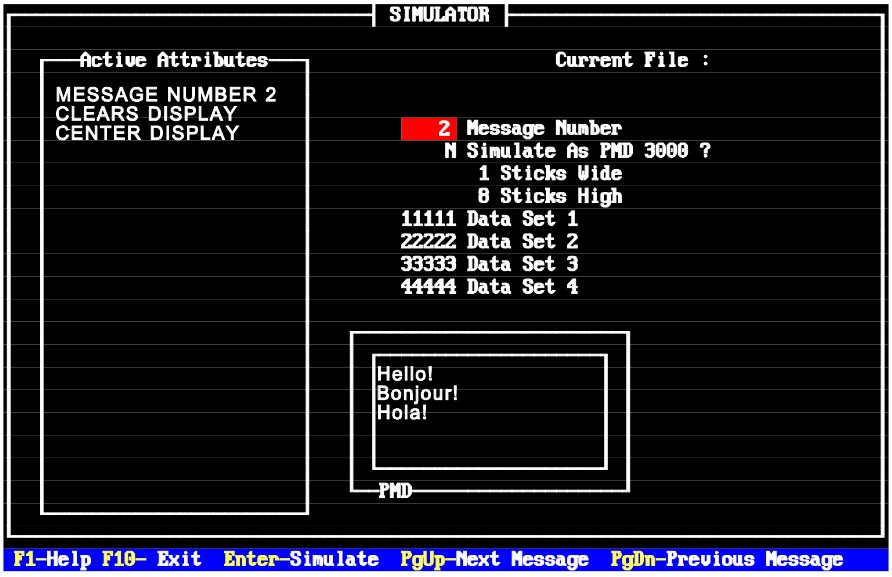
!Message 1 would appear as shown: a centered message that will clear any message displayed before it. If you had **not** selected the **`Center Message`** option, the message would be displayed as shown in the second screen on the following page.

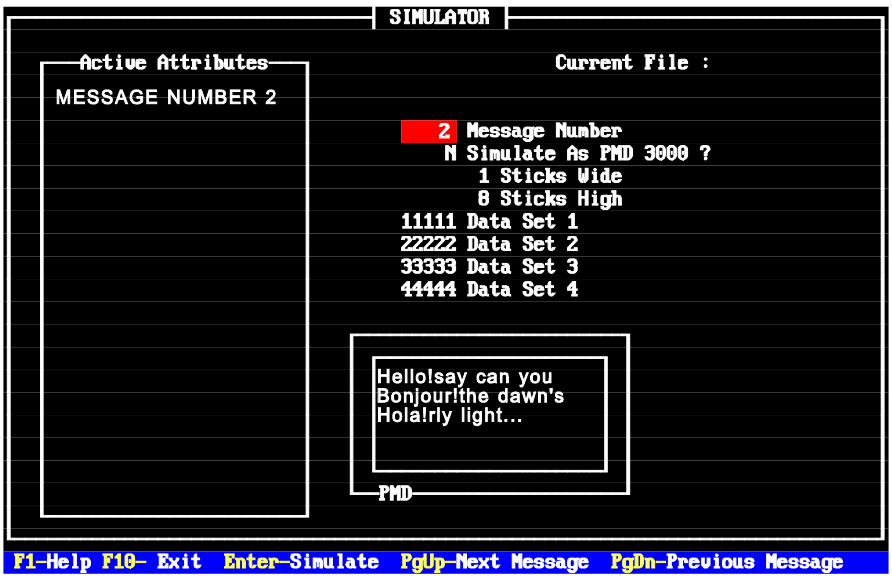
SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES



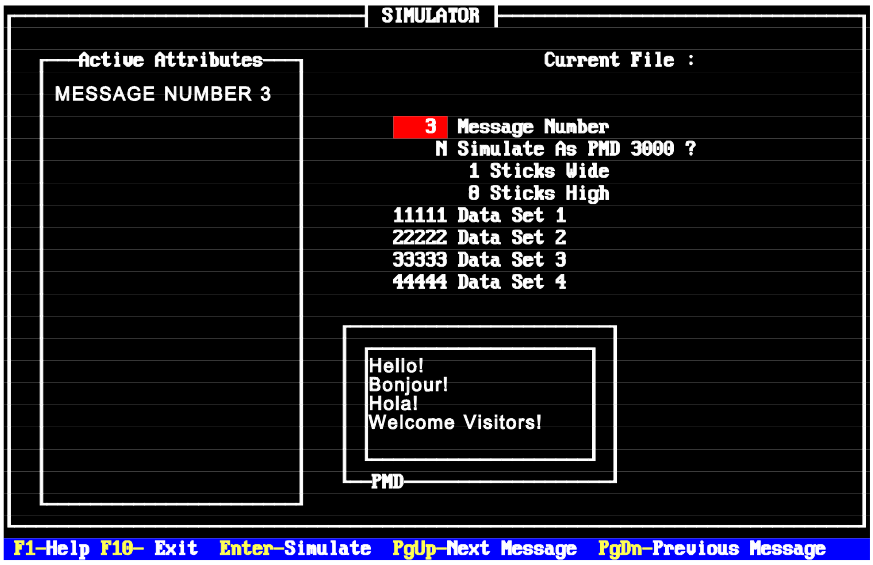

SIMULATOR
Active Attributes
MESSAGE NUMBER 1
CLEARS DISPLAY
Current File :
1 Message Number
N Simulate As PMD 3000 ?
1 Sticks Wide
8 Sticks High
11111 Data Set 1
22222 Data Set 2
33333 Data Set 3
44444 Data Set 4
Oh, say can you
see, by the dawn's
early light...
PMD
F1-Help F10- Exit Enter-Simulate PgUp-Next Message PgDn-Previous Message

!Message 2 would appear as shown: a centered message that will clear any message displayed before it. If you had **not** chosen the **`Clear Display`** and **`Center Message`** options for message 2, part of message 1 would remain on the display while message 2 is displayed, as shown in the second screen below.




SIMULATOR
Active Attributes
MESSAGE NUMBER 2
Current File :
2 Message Number
N Simulate As PMD 3000 ?
1 Sticks Wide
8 Sticks High
11111 Data Set 1
22222 Data Set 2
33333 Data Set 3
44444 Data Set 4
Hello!say can you
Bonjour!the dawn's
Hola!rly light...
PMD
F1-Help F10- Exit Enter-Simulate PgUp-Next Message PgDn-Previous Message

!Message 3 would appear as shown: an uncentered message that will not clear message 2 before it would be displayed; it "overlays" message 2. While the screen on the previous page exemplifies a mistake, this example shows how you can choose **not** to clear the display and overlay messages to your advantage.



16. To save the messages you have created onto disk, **exit** the **`Simulator`**

and **select** **`Save Message File`** from the **`EDIT MENU`**. **Select** F3 to change the directory if you would like. **Type** in your directory choice (e.g. A:\). **Type** in a name for your message file (e.g. TEST). This file name will appear at the bottom of the screen after **`Select Filename:`** as it is typed. The program will then quickly respond with **`Saving Message x`** for each message saved before returning you to the **`EDIT MENU`**.

Example 2: Includes
! **Scroll Left**
(Repeating)
(Change scroll rate)
! **Variable Data Sets**
! **Time/Date**
Triggered Messages

Let's say you want to program a message that repeatedly scrolls left, has variable data in it, and will be displayed on a specific date at a specific time on a standard PMD 400.

1. To start programming your messages, **select** **`Edit Message Program`** from the **`MAIN MENU`** on the computer.

2. **Press** A to add your message. The program again responds with the following menu.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES

```
Product PMD                Memory    128 K          Message File:
Group  0   Unit    1                                 Library File:

     1 Message Number        397    Bytes Used        130675 Bytes Free

  N Chain Message      N Scroll Upwards    N Scroll Left       N Send to RS232
    N Repeat              0 Bottom Line       0 Scroll Line       N On 1 Line
  Y Clear Display         0 Top Line          N Repeat Msg        N <CR> <LF>
  N Center Message        N Repeat Msg        0 Scroll Rate
  N Blink Message         0 Scroll Rate

  N Energize Alarm Relay
  N Invisible Message
  N Log Message

  N Send Message to Slave
    N All Slaves
    0 Group Number                                    1
    0 Unit Number                                     2
   3000 Frame Information                             3
      0 Upper Row                                     4
      0 Lower Row
  N N N N Data Sets Disable

F1-HELP F2-Send F3-Program F4-SizeF6-Edit F10-Exit/Save SPACE-Y/N PgUp/PgDn ESC
```

3. **Enter** the number 10 and **hit** Enter.

4. Use the arrow keys to move to the **`Scroll Left`** option and use the space bar to toggle it to Y, meaning this message number 10 will scroll from

the right of the display to the left.

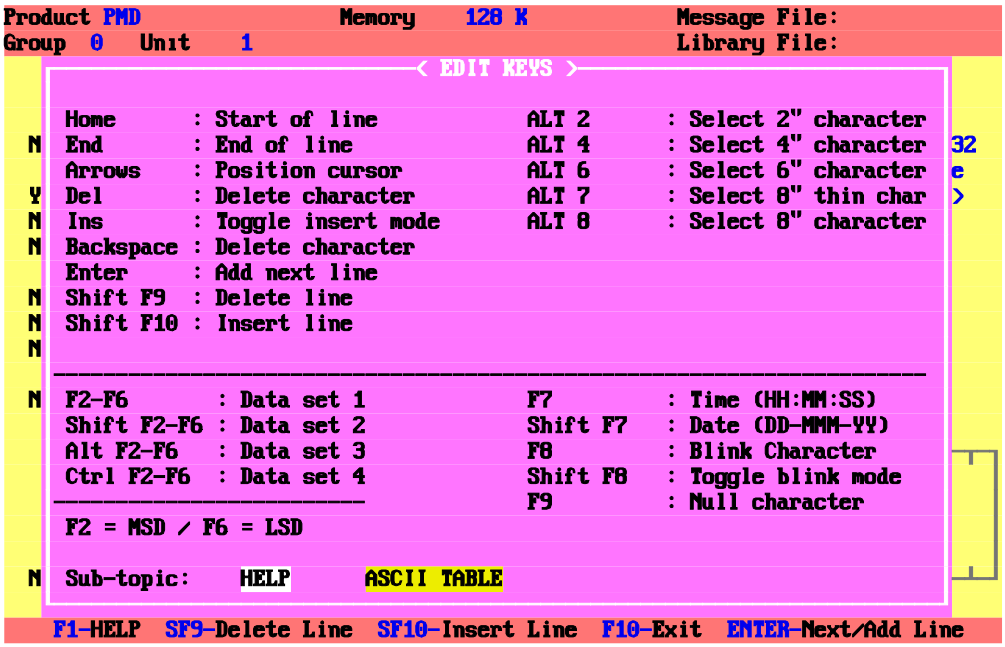
5. **Hit** Enter. Continue using the arrow keys and space bar to set the **`Scroll Line`** as 2, the **`Repeat Msg.`** as Y, and the **`Scroll Rate`** as 10.

6. Now that you have set these options, **press** F6 to move the cursor to the message text area and start entering your text for message 10.

7. For the first line, **type** in **`The temperature`** and **press** Enter to advance to the next line. **Press** the space bar once and **type** in **`today is`**.

8. Since we want to program variable data in this message, **hit** F1 to be reminded of the function keys which insert variable data.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES



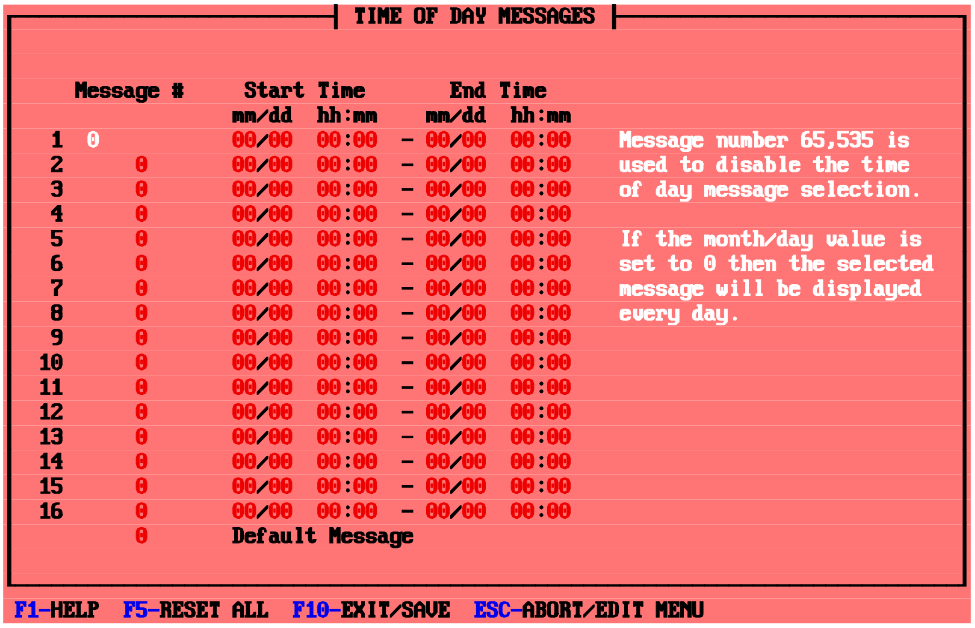
9. **Press** Esc to start entering the variable data place holders. **Enter** F4 F5 F6 (they will appear on your screen as 111) to form a three digit number for the temperature, and then **type** a period before **pressing** Enter.

10. On the third line, **press** the space bar once, **type** in Sales increased, and **press** Enter to advance to the fourth line.

11. **Press** the space bar once and then **enter** Shift F5 Shift F6 (they will appear on your screen as **22**), to form a two digit number for a percentage rate followed by a % and an ! . **Press** F10 twice to exit and save.

12. Now, to program message 10 to be displayed on a specific date at a specific time, **select** **Time of Day Messages** from the main **EDIT MENU**. The terminal responds with the following menu.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES



13. **Insert** message 10 under the **`Message #`** column. Use the right arrow key to move to the next column and **insert** today's month and date under the **`Start Time mm/dd`** column.

14. **Set** the **hh:mm** section as this hour and within a few minutes from now.

15. **Set** the **End Time mm/dd** as today's month and date again and set the **hh:mm** to turn off within approximately fifteen minutes.

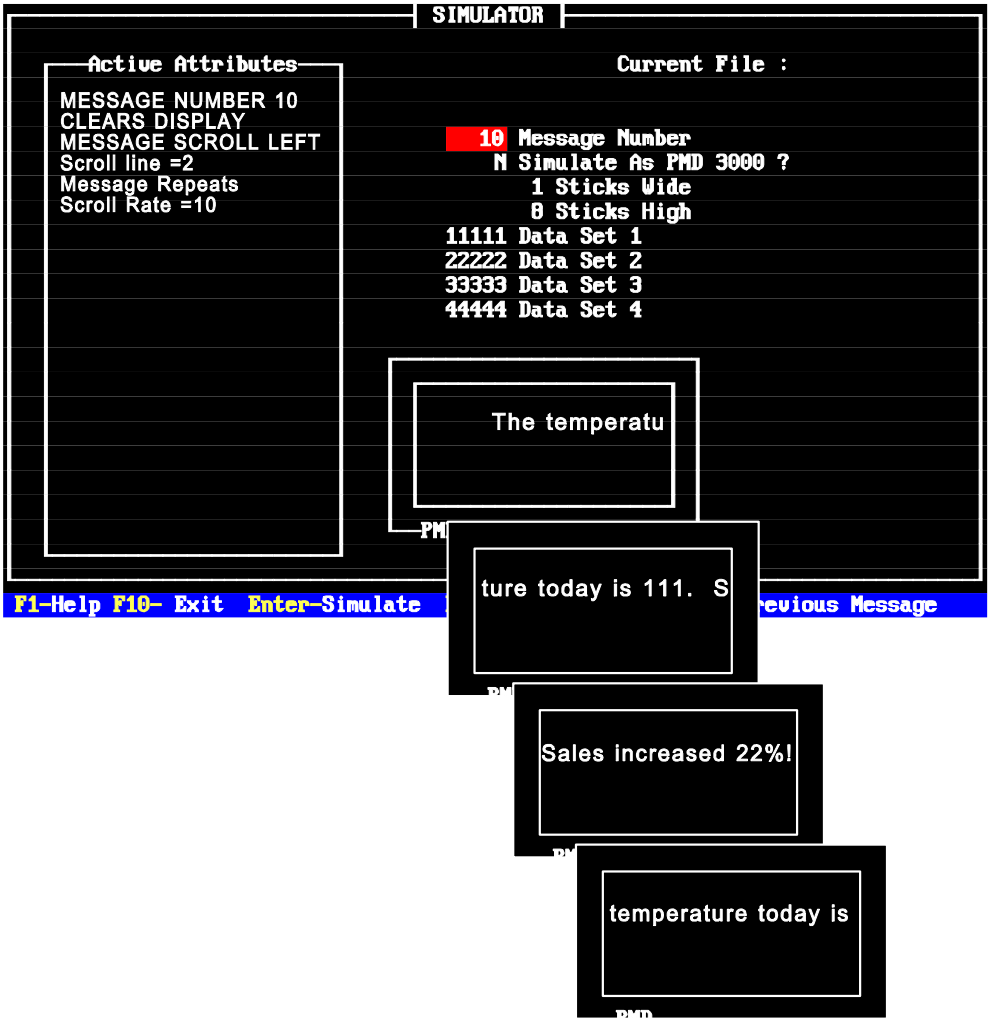
16. **Press** F10 to exit and save.

17. Message 10 will be displayed during the time period you have just assigned as it scrolls from the right of the display to the left on the second line, with the accurate variable data, when it is connected to the PLC.

Note: Time of Day messages can only be uploaded to DISPLAY through the *RS-422 Computer Interface Port*, not through the *RS-232 Tape Port*.

For now, you can see a simulation using the **Simulator** selection from the main **EDIT MENU**.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES



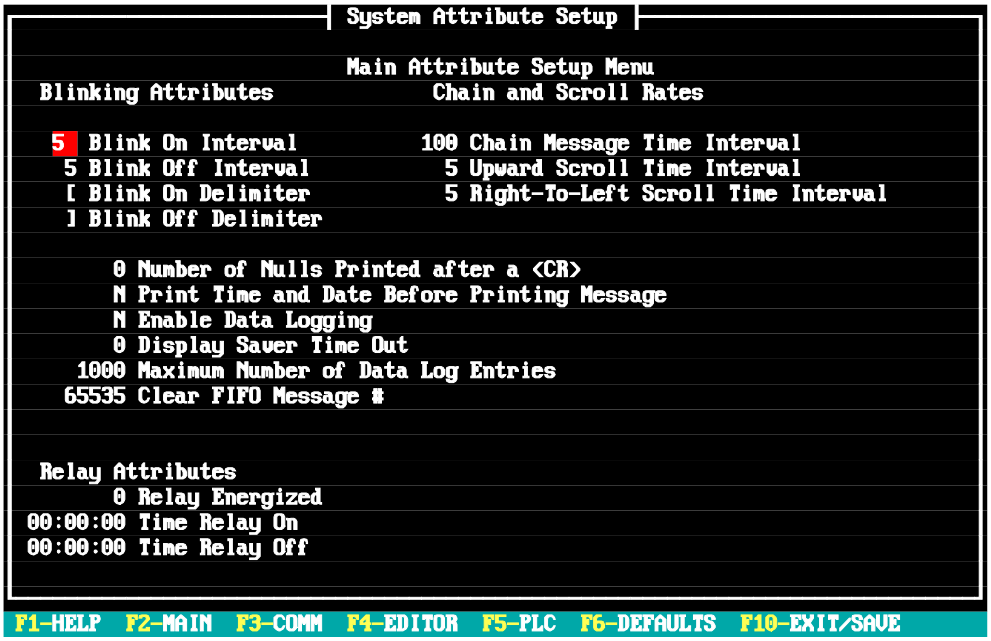
SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES

Example 3: Includes
! **Edit a Message**
! **Print a Message**
! **Log a Message**
! **Embedded Codes**
! **Chain Message**
! **Send to 3000 Slave**
! **Four-inch, Six-inch, eight-inch thin characters**

Let's say you want to edit messages 1, 2 and 3. In addition to the original message attributes, you want each message to be logged and printed. You want to add an embedded code to message 1, and you want to chain all three messages together and send them to a PMD 3000 Slave.

1. To start programming your messages, **select** **`Edit Message Program`** from the **`MAIN MENU`** on the computer.

2. **Press S** to see the **`Setup Attributes`** sub-menu:

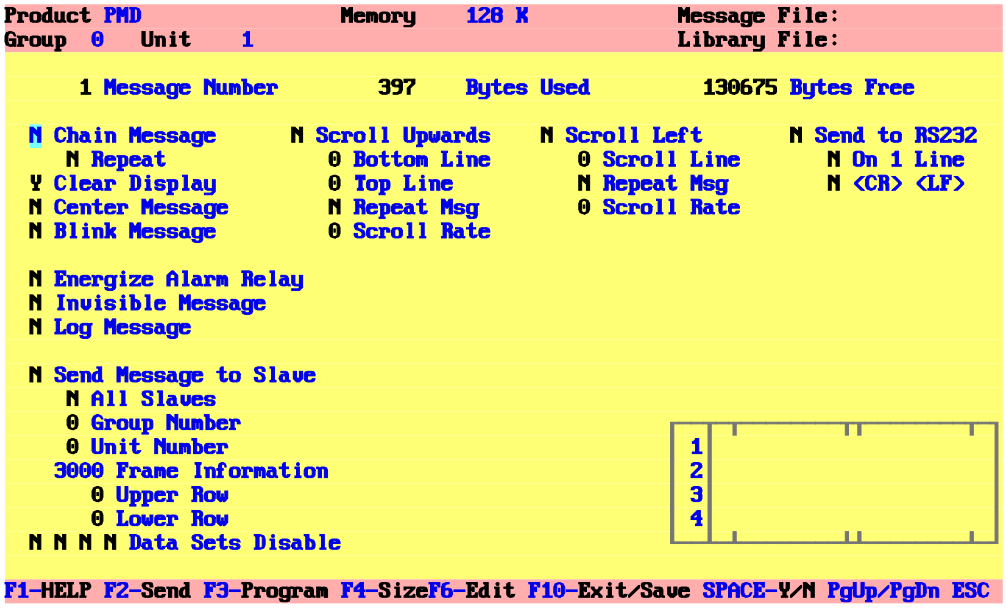


3. Use the arrow keys and space bar to change the **Enable Data Logging** option to Y.

4. **Press** F10 to exit and save.

5. From the main **EDIT MENU**, **press** E to edit message 1. The program responds with a menu that looks exactly like the **Add a Message** menu.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES



6. **Enter** the number 1 and **hit** Enter.

7. Use the arrow keys to move down to the **`Send to RS232`** option and use the space bar to toggle it to Y, meaning message 1 will be sent to the *RS-232*

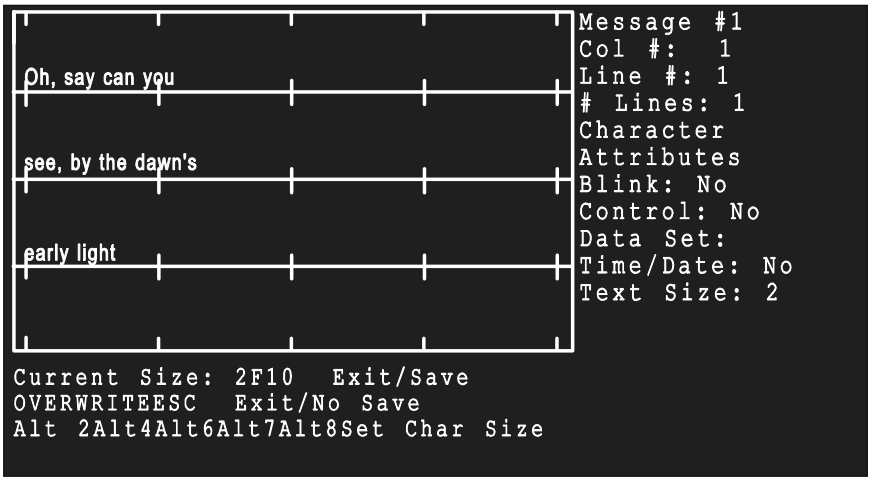
Terminal/Printer Port to be printed each time it is selected.

8. Again, use the arrow keys to move to **Log Message** and use the space bar to toggle it to Y so that message 1, as well as the time and date it was triggered, will be stored every time it is selected.

9. Now move to the **Send Message to Slave** option and toggle it to Y. Then move to the **3000 Frame Information** and designate the **Upper Row** as 1 and the **Lower Row** as 8.

10. Now that you have set these options, **press** F6 to move the cursor to the message text area and start editing your text for message 1. Because you are going to send this message to a PMD 3000 Slave, the "**Graphics Editor**" screen appears on your screen.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES



11. Let's delete the first line and make the characters bigger. **Press** Alt4 to make four-inch characters. **Retype** `Oh, say` and **hit** Enter.

12. **Delete** line 2. **Press** Alt4 again to make four-inch characters. **Retype** `can you` and **hit** Enter.

13. **Delete** line 3. **Press** Alt4 again to make four-inch characters. **Retype** `see...` and **hit** Enter.

14. For line 4, **press** F7 to insert the **embedded code** to have the current time displayed in message 1. The program displays: `HH:MM:SS` to indicate that

the time code is programmed.

15. **Press** F10 twice to exit and save.

16. **Press** E to edit message 2. The program again responds with the menu.

17. **Enter** message 2, **hit** Enter, and follow steps 7-10.

18. **Delete** line 1 and **press** Alt6 to make six-inch characters. **Retype** **`Hello!`** and **hit** Enter.

19. **Delete** line 2 and **press** Alt6 to make six-inch characters again. **Type** **`Hi!`**

20. **Delete** line 3 and **press** F10 twice to exit and save.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING EXAMPLES

21. To edit the final message (3), **press** E and follow the same procedure, steps 7-10.

22. Move to the fourth line and delete the current text. **Press** Alt7 to make thin eight-inch characters. **Type** **`good-bye!`** and **press** F10 to exit and save.

23. Finally, we will create a **Chain Message** which chains messages 1, 2 and 3 together as one message.

24. **Select** **`Add a Message`** from the main **`EDIT MENU`**.

25. **Type** 23 and Enter.

26. Use the arrow keys and space bar to change **`Chain Message`** to Y and **`Repeat`** to Y.

27. **Press** F6 to move the cursor to the message text area. The program responds: **`1>`**, prompting you for the first message of the chain.

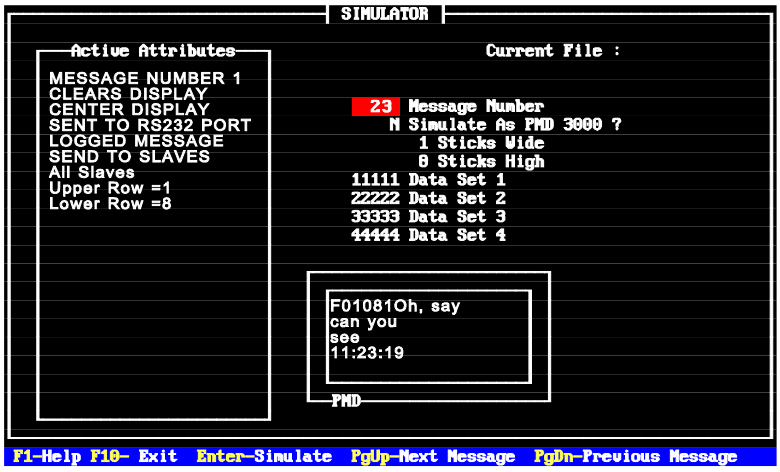
28. **Type** 1 and Enter, then 2 and Enter, then 3.

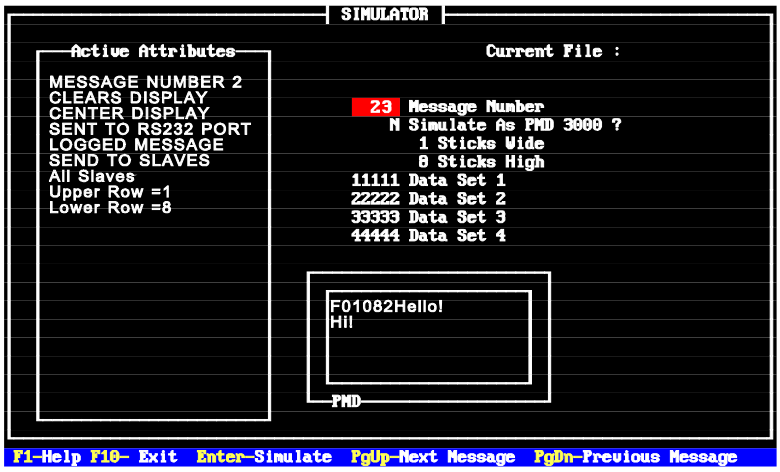
```
Product PMD                Memory    128 K          Message File:
Group  0   Unit    1                                 Library File:

   23  Message Number        397    Bytes Used        130675 Bytes Free

 Y Chain Message      N Scroll Upwards    N Scroll Left       N Send to RS23
    Y Repeat             0 Bottom Line       0 Scroll Line       N On 1 Line
 Y Clear Display         0 Top Line          N Repeat Msg        N <CR> <LF>
 N Center Message        N Repeat Msg        0 Scroll Rate
 N Blink Message         0 Scroll Rate

 N Energize Alarm Relay
 N Invisible Message
 N Log Message

 N Send Message to Slave
    N All Slaves
    0 Group Number
    0 Unit Number                                  1 1>1
  3000 Frame Information                           2 2>2
     0 Upper Row                                   3 3>3
     0 Lower Row                                   4
 N N N N Data Sets Disable

F1-HELP F2-Send F3-Program F4-SizeF6-Edit F10-Exit/Save SPACE-Y/N PgUp/PgDn ESC
```

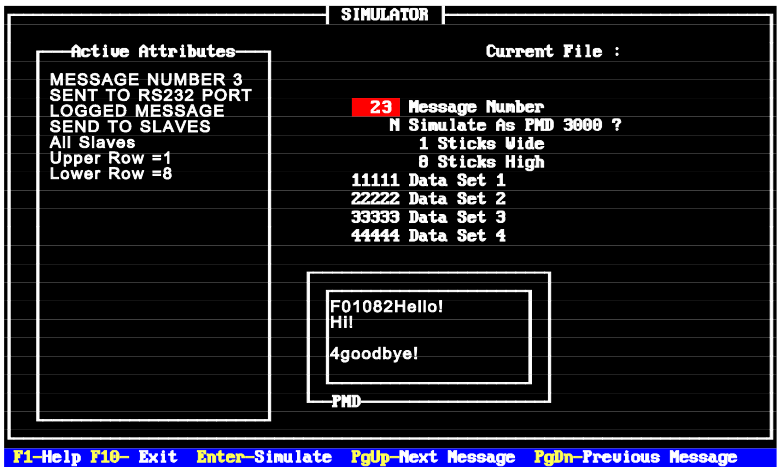
29. **Press** F10 twice to exit and save.

SECTION 5 - PROGRAMMING EXAMPLES
PART B: OFFLINE PROGRAMMING

30. You are finished programming. Message 23 will be displayed on your PMD 3000 Slave and logged and printed when it is selected by the PLC. For now, you can see a simulation of the messages as well as their frame/character designators and message attributes using the **`Simulator`** selection from the main **`EDIT MENU`**.







SECTION 6 - THE DISPLAY MODE

The DISPLAY MODE is the operating mode for the 400 unit. While in the DISPLAY MODE, the PMD is controlled solely by a programmable controller and/or other intelligent device and requires no user input.

In normal PMD master DISPLAY MODE, the PMD accepts information from the *Computer Port* and/or dedicated *PLC Port* and displays messages directed from them. With the 450, 460, 470, 475 and 480, the *PLC Port* uses what appears as a 16-input *Parallel Port* that is read as a collective value and is interpreted as either Binary or BCD (Binary Coded Decimal). This number is then translated as either a message number or a variable data value to be inserted into the displayed message. This is done by the four message control bits which first determine whether the port actually contains valid information.

The PMD master also displays date-triggered messages according to the **`Program Time of Day`** messages option of the **`Set Up Parameters`** menu. In turn, the master sends information to slave displays, a printer, an alarm device, and/or its internal data log. Other than occasionally changing the batteries for the data log and real-time clock, you may never have to touch the unit again!

Display Mode Preparation

Use the following procedure to prepare the PMD for DISPLAY MODE:

1. **Connect** AC and DC power lines to the PMD. (For 230 VAC operation, you **must** make adjustments to the unit.)

2. **Enter** the message program into the PMD memory.

3. **Set up** the ports used (*Printer Port, Relay Output Timer, PLC Interface* etc.) **Set** the scan time. **Enable/Disable** the Data Log using the MODE and ENTER push buttons on the front of the display.

4. **Set** the Group and Unit Number of the display.

5. **Connect** the controller and other devices to the various connectors.

6. **Press** RESTART. The PMD self-tests then enters the DISPLAY MODE.

SECTION 6 - THE DISPLAY MODE

What Happens During the Display Mode

The following sequence of events takes place when the PMD enters the DISPLAY MODE:

1. The PMD displays **SELF TESTING** for a minimum of five seconds then displays:

```
IN DISPLAY MODE
REVISION x
GROUP x UNIT x
DATA LOG ENABLED
```

The message identifies the unit and shows the software revision number, the programmed unit address, and whether the data log is enabled or disabled.

2. Then the display clears and the unit enters the DISPLAY MODE.

3. The display powers up with the time- and date-triggered message option enabled. Any time- and date-triggered message in which the current time is inclusive will be displayed.

4. If your PMD is enabled to display messages by the PLC, it will display the message called for by the data bits.

Note: Neither message information from the dedicated *PLC Port*, *Computer Ports*, nor the time- and date-triggered message option takes precedence over the other. Information from the time- and date-triggered list and from the *Computer Port* is processed immediately. *PLC Port* readings are either displayed immediately or placed on the incoming message FIFO.

5. The unit remains in the DISPLAY MODE indefinitely. If power is lost and then restored, the unit self-tests and re-enters the DISPLAY MODE.

The PMD provides two methods for reading the *PLC Port* and two methods for displaying messages:

- !PMD Scan versus Programmed Scan Time
- !Non-programmed Display Time versus Programmed Display Time

Scan Time

In the DISPLAY MODE, the PMD's software operates in a repetitive loop: it reads (the PLC inputs if it is a 450, 460, 470, 475, 480), the serial inputs, and then updates the display. This loop is the PMD's "scan" and its length may vary between 1 and 180 msec, depending on how much of the PMD's display must be updated.

SECTION 6 - THE DISPLAY MODE

System Scan

When using system scan rate, the unit operates as follows:
For a new message number to be accepted from the *PLC Port*, the PMD must read the same new value on two successive scans as well as every scan after that for the amount of time specified by the **`Timed Scan`** (described next). If the inputs don't change for that period of time, the message whose number is read will be displayed.

Note: With this process, the inputs to the PMD must be held constant for at least one PMD scan (up to 180 msec). ! The largest valid message number that can be read from the *PLC Port* is 9999 BCD or 65534 Binary. The largest valid data set value is 9999 BCD or 65535 Binary.

Timed Scan

When the PMD is set for a timed scan (4-99 msec), the *PLC Port* is read at this programmed time interval. Information read during a timed scan is considered valid regardless of the system scan. Therefore, the state of the Data Hold input is critical while the PLC data is in a state of change. The PMD will accept the reading of the PLC data as valid message/data in each scan that the Data Hold input reads logic '1'.

Display Times

Display time allows messages to be displayed for up to 255 seconds. A non-programmed display time (zero seconds) displays a message until it is replaced by another message that either clears the display or overlays that message. A programmed display time (1 to 255 seconds) displays messages for a programmed length of time. While messages can remain on the display longer than their display time, the programmed display time ensures that each message will at least be displayed for this pre-determined length of time. Once the message has timed out, it will not reappear unless it is triggered again.

While using the programmed display time, the PMD continues to scan the *PLC Port* at the selected scan rate. All valid readings from the *PLC Port* are placed in a message FIFO. This FIFO holds up to 32 valid readings (message numbers and data sets). When this FIFO is full, the PMD ignores additional readings until the FIFO contains less than its 32 maximum.

If set, the Clear FIFO message number can clear the FIFO simply by being selected on the *PLC Port*. When selected, the Clear FIFO message gets placed on the FIFO and then gets processed (i.e. it clears the FIFO). The FIFO must contain less than 32 readings before this Clear FIFO can take place because the Clear FIFO message enters the FIFO in the same manner as other messages.

Note: Messages and data read from the *Computer Port* are not placed in the message FIFO but are displayed immediately.

SECTION 6 - THE DISPLAY MODE

Time/Date-Triggered Messages

When the PMD enters DISPLAY MODE, any time-of-day message that has a current display time will be displayed. The PMD checks the time-of-day message list (in the DISPLAY MODE) once per minute. When the display time equals the time/date setting of the real-time clock, the message is displayed. When the OFF time for the message is reached, the message is removed from the display and the default time-of-day message (or no message, if you programmed that option for NONE) is displayed until replaced with new text.

If two date-triggered messages are programmed for the same ON time, a message programmed to be displayed every day is displayed before a message that is programmed for a specific date. This means that the everyday message will be displayed for a very short time before the date-specific message is displayed (until the OFF time is reached or the message is replaced with new information).

The time- and date-triggered message option is disabled when the unit receives a message with the **Disable Display of Time of Day Messages** code embedded at the beginning of it (as the first characters on the first line of the message text). Then the unit ignores all information in the time- and date-triggered message list. When a message with the **Enable Display of Time of Day Messages** code (at the beginning of the message text) is received, scanning and processing the time- and date-triggered message list resumes.

Displaying Messages

When selected and displayed, a message will output information and display itself according to how it was programmed. Exceptions to this rule follow:
Output Exceptions - Messages programmed to **not** energize the *Relay Output Terminals* will not affect operation of the timed relay.
Display Exceptions - Sometimes messages will be cleared from the displays even though the newly selected message was not programmed with the "Clear Display" option. This happens when:

- !the previous message was a centered message containing control characters or nulls.
- !an error message was previously being displayed.
- !scroll up clears all lines it uses.
- !scroll left clears the line it uses.

Note: If a message that does not clear the display is selected while a left-scrolling or upward-scrolling message is being displayed, the new message will scroll off the display (with the scrolling message) on the affected line.

SECTION 6 - THE DISPLAY MODE

Variable Data Sets

For the 400 unit, the variable data is cleared before the message is displayed. After the message is triggered, the variable data must be sent to appear in the message.

Variable data received through the *Computer Interface* is displayed as soon as it is received. It will overwrite any data currently in the variable data sets. Variable data should be sent after the message because when the message is displayed, it will clear the data sets.

To enter data from the front panel into messages with data sets, **press** the DATA SET x button, **enter** the data using the numeric keypad, and then **press** SELECT to complete the entry for each of the data sets. If you make a mistake, use the CLEAR button and then re-enter the correct number(s).

Note: If the data does not seem to be affected by your manipulating the buttons on the front panel, check to see if you have the **"DS Protect"/"Data Sets Disable"** options described in the programming sections set to "protect" the data sets from being changed by the operator.

For the PLC interface units, the 450, 460, 470, 475 and 480, data sent directly from the PLC is put into the variable data sets each time new data is received from the PLC. PLC data is not cleared when a new message is triggered. When a new message is triggered, it will be displayed initially with the data currently in the variable data sets (This is different than the standard 400 unit).

The Display Mode and Slave Message Displays

When a message is selected, the master PMD displays the message on its display. If the message's parameters indicate it is also to be sent to slave(s), the master will send the message to the specified slave(s). Any changes to any of the data sets will cause the master PMD to send all four data sets to all slaves.

The PMD *Slave Port* is always active when the PMD is in the (PMD master) DISPLAY MODE. Slave units interfaced to a master display will not affect the operation of the PROGRAM, TAPE IN, or TAPE OUT modes.

Most slaves display messages identically to the master. All message options pertaining to the message appearance affect the message on the slave's display (blinking, scrolling etc.) as well. This means that when you set up message appearance using the message options, these options are also sent to the slave with the message text.

SECTION 6 - THE DISPLAY MODE

Data Log

The PMD data log can store approximately 3050 entries of messages and data; however, the data log can be made smaller using the **`Size of Log`** option in the **`Data Logging`** command of the **`Set Up Parameters`** menu. When the PMD data log is enabled, it stores either 3050 or the programmed "size" number of entries until it is cleared. When the data log becomes full, it discards the oldest entries to make room for the new information. This way, a full data log of 1000 size contains the 1000 most recent readings of the *PLC Port* or the *Computer Interface*.

Daily Relay Alarm

When the relay output timer is enabled, these terminals are energized at the program-determined time of day. Incoming messages have no effect on these terminals while the timer is energizing them. They remain active until you exit the DISPLAY MODE or the timer turns them off (at which time they return to message control).

Circular Message Queue

A circular message queue is available on all 400 units equipped with a PLC interface. The circular queue is an option that can be enabled or disabled, and as long as the circular queue is disabled, all PMD 400s will maintain their other capabilities. This feature keeps newer units compatible with older ones.

The circular queue is adjustable in size from 0 to 20 messages. When the queue size attribute is set to 0, the queue is disabled. The queue is only enabled when the queue size is greater than 0. The unit must be restarted each time the queue size is changed. If the queue mode is disabled (queue size = 0), the FIFO will be used according to how it is set up. The standard 32 entry FIFO is disabled when the queue is enabled and no variable data can be sent through the message/data word.

Data sets can only be sent to the PMD from the PLC by mapping the data sets directly to the PMD. With the queue enabled, the PMD will cycle through the messages in the queue according to the display time associated with each message in the DISPLAY MODE. Each message in the queue will be displayed according to its associated display time. Any function that affects the queue (add, delete, clear) will be ignored when the queue is disabled. As long as the queue size is greater than 0, the queue can be enabled or disabled from the PLC interface. When the queue mode is disabled, the last message from the queue that was displayed will remain on the display. When the queue is reenabled, the next message on the queue will be displayed. Each message entry in the queue will contain the message number, message display time and four data sets.

If no message display time is specified when the message is triggered, the FIFO display time will be used as the default message display time (5 sec). The Clear FIFO message number will have no effect on the circular queue. The PLC interface will be able to add entries, delete entries, clear, and enable or disable the queue. If the queue size is 0, however, the PLC cannot enable it.

If the PLC tries to add an entry to the queue when it is full (as indicated by a status bit in the PLC), the entry will be thrown away.

With the queue enabled in DISPLAY MODE, the PMD will cycle through the queue according to the message display time for each entry. The data sets that are being read from the PMD will not be updated on the display. The data sets on the display will be the data sets that were stored with the message number as part of the queue entry.

If a PLC communication error occurs while the queue is enabled, the data in the control word will be interpreted as defined when communication is reestablished. This means that if a message is added to the queue and a PLC error occurs, the queue will be disabled until communication resumes. Then when the error is corrected, the PMD will read the outputs again, and if they are still valid, the message will be added to the queue again.

Specific use of the queue option is explained in greater detail at the end of **SECTION 7 - PLC INTERFACING** in **PART F: REFERENCE**.

SECTION 6 - THE DISPLAY MODE

SECTION 7 - PLC INTERFACING

The PMD uses the PLC interface to monitor data registers for controlling messages, data sets, and LED status. The PMD will also use the PLC interface to write function key, data set, and status information to registers in the PLC.

Using PLC Outputs to Control:

!**Messages** - The PMD monitors two 16-bit registers to control messages. The first register is the Message/Data register and is used to select a message number. The second register is the Control register and it is used to determine when the message is triggered and to control the use of the other registers monitored by the PMD. A message is triggered by setting the Message/Data word to the desired message number and then setting the Message/Data and Data Hold bits in the Control Word to one. **This will set the control word to nine.**

While you are using the "Map data sets to PMD" and/or "Map data sets to PLC" under normal operating conditions, the message control word will still remain at the decimal value of nine, thus eliminating any need to strobe the individual bits of the control word.

!**Data sets** - The PMD can be configured to monitor up to four 16-bit registers to be used for variable data information in the PMD. Each word of PLC data is mapped directly into one data set. Each time the PLC data changes, the new PLC data will be put into the corresponding data set in the PMD. Note: Data sets are not mapped onto the display when the queue feature is enabled.

!**LEDs** - The PMD can be configured to monitor up to three 16-bit registers to be used for LED status in the PMD 400 Series (with a 400E expander module). Each word of PLC data is mapped directly into one set of 16 LEDs. Each time the PLC data changes, the new PLC data will be put into the corresponding LED status in the PMD.

Controlling PLC Register With:

!**Function Keys** - The PMD 400 Series can be configured to map up to three sets of 16 function keys to up to three 16-bit registers in the PLC. Each time a function key is pressed, it sets the corresponding bit in the PLC. When the key is released, the bit is cleared.

Data Sets - The PMD can be configured to map up to four data set words directly into four registers in the PLC. Each time a data set changes in the PMD, it will be written to the corresponding register in the PLC.

SECTION 7 - PLC INTERFACING

!**PMD Status** - The PMD uses one 16-bit register in the PLC to indicate the status of the PMD. Each time the register changes, it is rewritten to the PLC.

Circular Message Queue

A circular message queue is available on all 400 units equipped with a PLC interface. The circular queue is a feature that can be enabled or disabled, and as long as the circular queue is disabled, all PMD 400s will maintain their other capabilities. This feature keeps newer units compatible with the older ones.

The circular queue is adjustable in size from 0 to 20 messages. When the queue size attribute is set to 0, the queue is disabled. The queue is only enabled when the queue size is greater than 0. The unit must be restarted each time the queue size is changed. If the queue mode is disabled (queue size = 0), the FIFO will be used according to how it is set up. The standard 32 entry FIFO is disabled when the queue is enabled and no variable data can be sent through the message/data word.

With the queue enabled, the PMD will cycle through the messages in the queue according to the display time associated with each message in the DISPLAY MODE. Each message in the queue will be displayed according to its associated display time. Any function that affects the queue (add, delete, clear) will be ignored when the queue is disabled.

As long as the queue size is greater than 0, the queue can be enabled or disabled from the PLC interface. When the queue mode is disabled, the last message from the queue that was displayed will remain on the display. When the queue is reenabled, the next message on the queue will be displayed. Each message entry in the queue will contain the message number, message display time and four data sets. If no message display time is specified when the message is triggered, the FIFO display time will be used as the default message display time (5 sec). The Clear FIFO message number will have no effect on the circular queue. The PLC interface will be able to add entries, delete entries, clear, and enable or disable the queue. If the queue size is 0, however, the PLC cannot enable it. If the PLC tries to add an entry to the queue when it is full (as indicated by a status bit in the PLC), the entry will be thrown away.

With the Queue enabled in DISPLAY MODE, the PMD will cycle through the queue according to the message display time for each entry. The data sets that are being read from the PMD will not be updated on the display. The data sets on the display will be the data sets that were stored with the message number as part of the queue entry.

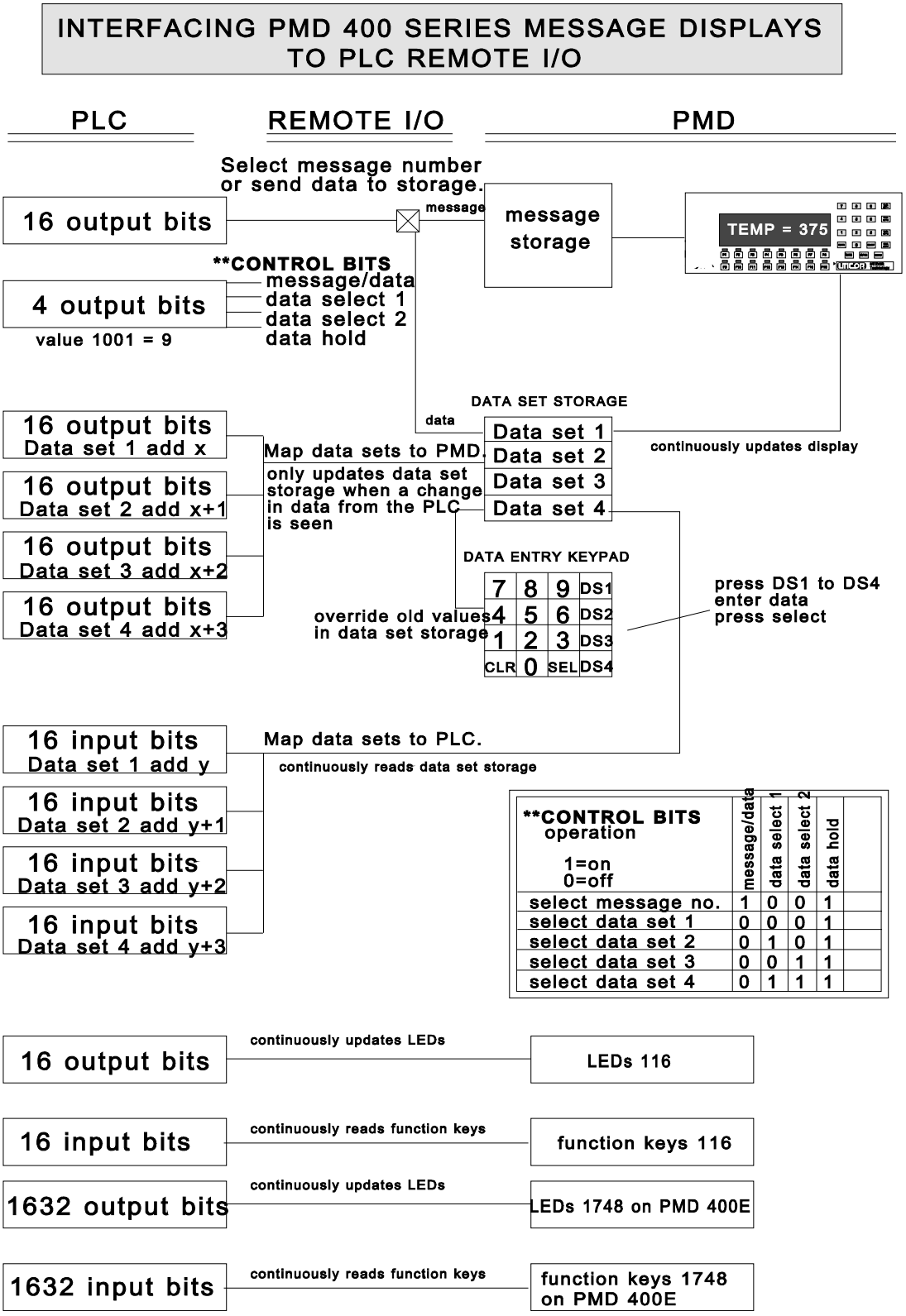
If a PLC communication error occurs while the queue is enabled, the data in the control word will be interpreted as defined when communication is reestablished. This means that if a message is added to the queue and a PLC error occurs, the queue will be disabled until communication resumes. Then when the error is corrected, the PMD will read the outputs again, and if they are still valid, the message will be added to the queue again. When the queue is enabled, the data entry keys on the front panel of the PMD 400 Series are disabled.

Specific use of the queue option is explained in greater detail at the end of this **SECTION 7 - PLC INTERFACING** in **PART F: REFERENCE**.

PLC Set up Overview

If you are using a 400 Series product with a PLC interface, you need to be aware of the unit's scan time. The 400 unit has a **maximum** scan time of 185 milliseconds. The typical scan time is less than 185 milliseconds, but it is possible that the PLC will wait the 185 msec between each triggered message to guarantee that the 400 product will see the information. All four data sets can be changed within a single scan but a data set cannot be changed twice within a scan of the 400.

SECTION 7 - PLC INTERFACING



SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

This section is intended to give you specific information about the 450 A-B Interface unit that makes it unique from the basic 400. It is a supplement to, and specified reference point within, the preceding manual sections.

The PMD 450 is essentially a PMD 400 which directly interfaces to an Allen-Bradley PLC through Remote I/O, Block Transfer, or Data Highway/Plus. Each of these modes operates independently from the other and the 450 can be configured to communicate using any one of them. The PMD will load the appropriate configuration from the attributes during its startup procedure.

The *A-B interface connector* located on the bottom of the PMD 450 is a right-angle, seven-position, removable terminal block. The middle position is not used; the other six connections provide dual hook up for the twin-axial cable and its shield. This allows dual connections for multi-dropping the 450 in the main controller/remote I/O rack system configuration.

The information received through this connector is collected in the PMD's registers and from that point, is manipulated similar to a *parallel port*. The 16 bits of information can be read in either Binary or BCD (Binary Coded Decimal) and inverted or non-inverted. These options are software selectable, as described on the following pages.

In configuring your 450 for operation, the prompt will always start with which type of A-B PLC will be used. You must select a particular mode of operation, with the default mode being Remote I/O. The PMD will prompt for enabling Data Highway, Data Highway Plus, or Block Transfer. If these are answered as **N** or 'no', the unit will prompt for the set ups required to process Remote I/O data. If Data Highway, Data Highway Plus, or Block Transfer is selected, the unit will then prompt for the required data to process that selected PLC interface mode. The prompts and accompanying explanations for each A-B Interface follow.

Please refer to **PART F: REFERENCE** of this section for input/output table definitions.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

REMOTE I/O:
Setting up 450 A-B Communication

To configure your 450 for Remote I/O online, **select** **SETUP PARAMETERS** from the **MAIN>COMMANDS** menu and then **select** option **5 - AB PLC COMMUNICATION** and option **17 - FUNCTION KEYS/LED CONTROL**. If you opt to configure your 450 using offline programming, you access your menu starting from the **MAIN MENU**, then selecting **EDIT MESSAGE PROGRAM**, then **SETUP ATTRIBUTES**, and finally, **F5 - PLC**. Illustrations of the offline menus follow this segment on the "online" prompts. The following explanations of the options, however, are universal for both programming methods.

Set up Parameters Online Programming Prompts

When you select option **5**, the terminal responds with the following prompts.

Option 5

!**PLC TYPE (1=PLC2, 0=PLC3-PLC5) <0>**
This option lets you select the system with which the 450 will be operating: PLC2 or PLC3/PLC5. Your response will set the 450 for the correct Allen-Bradley system so that the 450 will specify the current valid range for the rack.

!**ENABLE DATA HIGHWAY (Y/N) <N>**
This prompt determines whether or not the Data Highway interface will be loaded and used for PLC communication. Therefore, if you want to use Remote I/O, you must answer this prompt as **N**.

!**ENABLE DATA HIGHWAY PLUS (Y/N) <N>**
This prompt determines whether or not the Data Highway Plus interface will be loaded and used for PLC communication. Therefore, if you want to use Remote I/O, you must answer this prompt as **N**.

!**ENABLE BLOCK TRANSFER (Y/N) <N>**
The PMD 450 Allen-Bradley PLC interface supports block transfers. When block transfers are enabled, however, your 450 will no longer process Remote I/O data. Therefore, if you want to use Remote I/O, you must answer this prompt as **N**.

!**ACTIVE RACK (Y/N) <N>**
The unit can either monitor the PLC via two 16-bit words of a PLC output group (nonactive mode) or emulate the Remote I/O rack (active mode).

This prompt treats the 450 as a single slot Remote I/O rack with its own unique rack address, receiving information through two I/O slots. Nonactive Mode causes the 450 to monitor two I/O slots of an existing Remote I/O rack (addressed with the same rack number) through the output image table. The rack being monitored can be a PLC, an actual I/O rack, or an active PMD. For example, your system has three I/O racks and one rack has all of the inputs. Therefore, set the PMD to the same rack address as the inputs.

!ENABLE PARTIAL RACK (Y/N) <N>
This prompt will appear if the preceding prompt was answered as **Y**. It determines if the message input rack will be enabled as a partial or full rack (A full rack is eight 16-point module positions). If it is chosen as a partial rack, the following two prompts will appear.

!SIZE OF PARTIAL RACK (1=1/4 2=1/2 3=3/4 4=FULL) <1>
This prompt determines the size of the partial rack to enable. Each quarter of a rack represents two I/O groups. This allows you to enable only as much of the rack as needed.

!STARTING QUARTER OF PARTIAL RACK (0-3) <0>
This prompt defines where the partial rack will start.

Note: A PLC5 will auto-configure a 450 (in the active mode) as a full rack.

!PLC I/O RACK # (1-7 OCTAL) <1> or PLC I/O RACK # (0-37) <0>
This prompt asks you to set the rack address, with rack number limits dependent on the type of PLC you chose. If you selected a PLC2, the limits are 1-7 and will default to the value currently in the attributes, and to zero after a clear memory. For a PLC3/PLC5, the limits will be 0-37 (octal) with a default of 0.

For active mode, the 450 must be set to a rack number which is unique to the PLC with which the 450 is communicating. The 450 is assigned a rack address independent of the rack addresses that are already assigned in the system. For nonactive mode, the 450 is assigned one rack address number of those already in use in the system, depending on which rack the 450 is supposed to monitor.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

!PLC I/O GROUP # DATA BITS (0-7) <0>
This option sets which I/O output group (slot) to read as the message data word. A BCD or Binary number in this address causes the corresponding message number to be displayed.

!PLC I/O GROUP # CONTROL BITS <1>
This option determines which output group will be read as the message control word. Note: This should be set to a value of 9...1001 at the PLC.

!NUMBER OF DATA SETS TO MAP TO PLC (0=NONE, 1-4) <0>
This prompt determines how many data sets will be mapped from the 450 to the PLC. If you choose 0, then none of the four following prompts will appear.

!STARTING DATA SET NUMBER (1-4) <1>
This prompt allows you to choose which data set to start with. As many data sets as are specified will be mapped into the PLC beginning with the starting data set. The 450 will only map as many sets as allowed by the address range of the PLC I/O rack. For example, you specify that four data sets should be mapped to the PLC starting at data set 1 and mapping them to I/O rack 1, I/O group 6. In this one, only two data sets will be mapped because the valid address range for I/O groups is 0-7. It will map data set 1 to I/O group 6, and data set two to I/O group 7. Data sets 3 and 4 will not be mapped.

!PLC I/O RACK # (1-7 OCTAL) <1> or PLC I/O RACK # (0-37 OCTAL) <0>
This prompts you to select to which I/O rack number you want the data sets sent. The first prompt will appear if you have a PLC2 rack. The second prompt will appear if you have a PLC3 or PLC5 rack.

!PLC I/O GROUP # (0-7) <0>
After the PLC I/O rack number has been selected, you are asked to choose the PLC I/O group number. This option is set according to which module position(s) you want the data to be sent. In order for data to be sent from the PMD to the PLC, the PMD must be configured as an active rack. As a result, when function keys or data sets are mapped to the PLC, the racks selected for them will be made active. All other racks are only made active when you specify them as active racks.

!NUMBER OF DATA SETS TO MAP TO PMD (0=NONE, 1-4) <0>
Another option you are given is to map the data sets from the PLC to the 450. This allows variable data to be read by the PMD from any existing or non-existing I/O rack.

If **0=NONE** is the response to this prompt, the next four prompts will not be given. Otherwise, they will proceed as follows.

!STARTING DATA SET NUMBER (1-4) <1>
This prompt asks you which data set to start with. It is followed by a prompt for you to indicate the PLC I/O rack number. You intend for the 450 to read variable data.

!PLC I/O RACK NUMBER (1-7) <1> or PLC I/O RACK NUMBER (0-37) <0>
This prompt lets you select from which rack number the PMD will read the data set. It can be any existing or non-existing rack.

!ACTIVE RACK (Y/N) <N>
This prompt lets you decide whether or not this rack will be active. When mapping data sets to the 450, the 450 unit must either monitor an existing rack or emulate an active rack. If there is no existing active rack to monitor, the 450 can be made active. If there is an existing active rack for the 450 unit to monitor, however, then the 450 can be made inactive.

!PLC I/O GROUP # (0-7) <0>
This prompt determines the starting I/O group to map into the data set. Subsequent data sets will be mapped from succeeding I/O groups until all data sets have been mapped or the end of the rack is reached.

!PLC BAUD RATE (0=57.6KB 1=115.2KB 3=230.4KB) <0>
This option sets the communication baud rate.

!BLOCK TRANSFER COMMAND MODE(0=DISABLED,1=ENABLED)<0>
The block transfer command mode is used to interface a computer on the Data Highway that is running UTICOR's 10F50 programming software with a PMD 350 on the network. It is required that the computer have an Allen-Bradley 1784-KT card in it.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

BLOCK TRANSFER WRITE COMMANDS

!ENABLE PARTIAL RACK (0=NO, 1=YES)<0>
This prompt determines if the rack specified will be enabled as a whole or partial rack. The rack specified will be made active as a whole rack if a partial rack is not specified.

!PLC I/O RACK # (1-7 OCTAL)<1> OR PLC I/O RACK # (0-37 OCTAL)<0>
This prompt determines which rack number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O GROUP # (0-7)<0>
This prompt determines which group number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O MODULE # (0-1)<1>
This prompt determines which module number will be made active and where the PMD will look for the block transfer commands.

BLOCK TRANSFER READ COMMANDS

!ENABLE PARTIAL RACK (0=NO, 1=YES)<0>
This prompt determines if the rack specified will be enabled as a whole or partial rack. The rack specified will be made active as a whole rack if a partial rack is not specified.

!PLC I/O RACK # (1-7 OCTAL)<1> OR PLC I/O RACK # (0-37 OCTAL)<0>
This prompt determines which rack number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O GROUP # (0-7)<0>
This prompt determines which group number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O MODULE # (0-1)<1>
This prompt determines which module number will be made active and where the PMD will look for the block transfer commands.

!INVERTED/NON-INVERTED (0=INVERTED, 1=NON-INVERTED) <1>
This prompt lets you determine how the PLC port looks at the message/data lines: the way they come in as 1s or 0s (non-inverted)...**or** the complements of these (inverted).

!BCD/BINARY (0=BCD, 1=BINARY) <0>
This prompt lets you determine whether the 450 will consider the message/data lines as a Binary or Binary Coded Decimal.

!TIMED SCAN (1=ENABLED, 2=DISABLED) <2>
This prompt is for setting a specified scan time for how often the *PLC port* is scanned at the "system" scan rate (between 1 and 180 msec).

!QUEUE SIZE (0-20, 0=DISABLED) <0>
The queue is automatically disabled and it cannot be enabled from the PLC interface when the size is specified as 0. If the size is specified as 0, the unit will use the 32 entry FIFO if a nonzero message display time is specified. The queue is configured to the size specified by this prompt and then it must be enabled by the PLC interface before it will accept any add, delete, or clear functions.

!MESSAGE DISPLAY TIME (0-255 SEC.) <0>
This prompt determines if the messages will remain on the display until replaced, or if they will be displayed for a specified time.

When the display time is set to zero, all messages will remain on the display until they are replaced. If a specified time is entered (in seconds ranging from 1 to 255), the messages will each display for that time period before being replaced by the next.

Set up Parameters Online Programming Prompts Option 17

Select SETUP PARAMETERS from the **MAIN>COMMANDS** menu and then **select** option **17-FUNCTION KEYS/LED CONTROL**. The terminal responds with these prompts:

!LED CONTROL (SELECT 1=PMD, 2=CI, 3=PLC) <1>
lets you choose what you want to use as the input source for the LEDs. If **1=PMD** is selected, the LEDs will be controlled by the function keys. If **2=CI** (*computer interface*) is selected, the LEDs will be controlled by computer interface commands. If **3=PLC** is selected, the LEDs will be controlled by the PLC outputs. A word of outputs will control 16 LEDs on a bit map basis. Bit 0 of the word controls LED 1 and bit 15 of the word controls LED 16.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

!NUMBER OF LEDS CONTROLLED BY PLC (0=NONE, 1=16, 2=32, 3=48) <0>-- (only if 3=PLC on the previous prompt)
is used to specify how many LEDs will be controlled by the outputs of the PLC. The prompt will reflect how many LEDs are available on the unit. The LEDs are controlled by output words 3, 4, and 5 of the PLC output data being monitored by the 400 product. If the output word is not controlling LEDs, it will be ignored. Each output word can control 16 LEDs on a bit map basis.

!ACTIVE RACK (1=ACTIVE, 0=NOT ACTIVE) <0>
When mapping LEDs to the 450, the unit must either monitor an existing active rack or emulate an active rack. If there is no existing active rack to monitor, the 450 must be made active. If there is an existing active rack for the 450 to monitor, however, then the 450 must be made inactive.

!PLC I/O RACK # (0-7 OCTAL) <1> or (0-37 OCTAL) <0>
You need to indicate the PLC I/O rack number from which you intend the 450 to read variable data. Your prompt depends on what kind of PLC you have indicated that you are using.

!STARTING PLC I/O GROUP # (0-7) <0>
This prompt determines the starting I/O group to map into the LEDs. Subsequent LEDs will be mapped from succeeding I/O groups until all LEDs have been mapped or the end of the rack is reached.

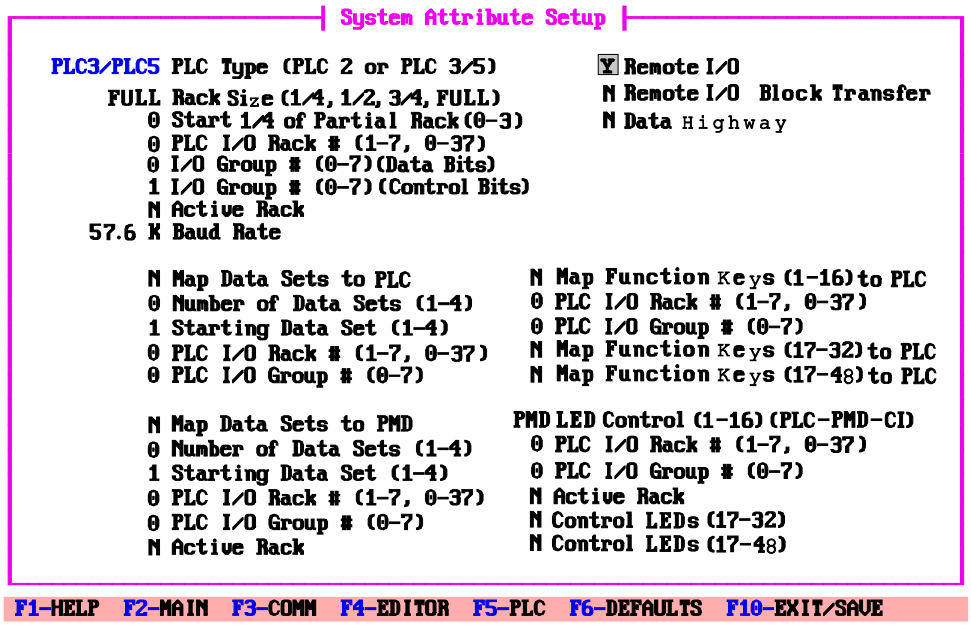
!NUMBER OF FUNCTION KEYS TO MAP TO PLC (0=NONE, 1=16, 2=32, 3=48) <0>
allows you to map the function keys to set inputs in the PLC. This prompt will also be dependent on the number of function keys that are available on the unit. Each set of 16 keys is bit mapped to one word of input. Function key 1 sets bit 0 and function key 16 sets bit 15. When the function key is pressed, the input bit is set, and when the function key is released, the bit is cleared. The function keys are mapped to input words 5, 6 and 7.

!PLC I/O RACK # (1-7 OCTAL) <1> or (0-37 OCTAL) <0>
This prompts you to select to which I/O rack number you want the 450 to send function keys. The first prompt will appear if you have a PLC2 rack. The second prompt will appear if you have a PLC3 or PLC5 rack.

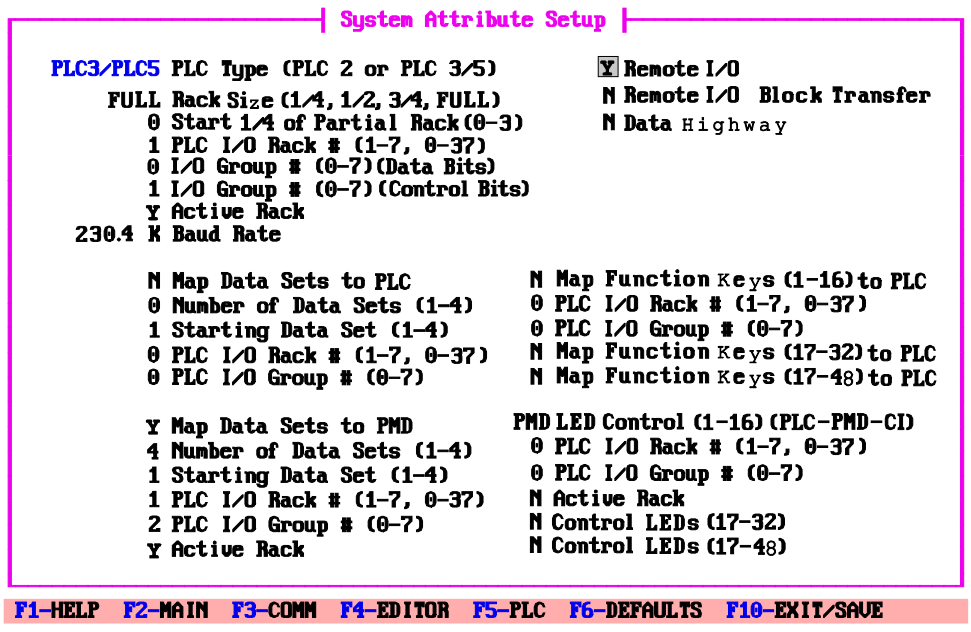
!STARTING PLC I/O GROUP # (0-7) <0>
This prompt determines the starting I/O group to map the first 16 function keys into. Subsequent sets of function keys are mapped into succeeding I/O groups until all function keys are mapped or the end of the rack is reached.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

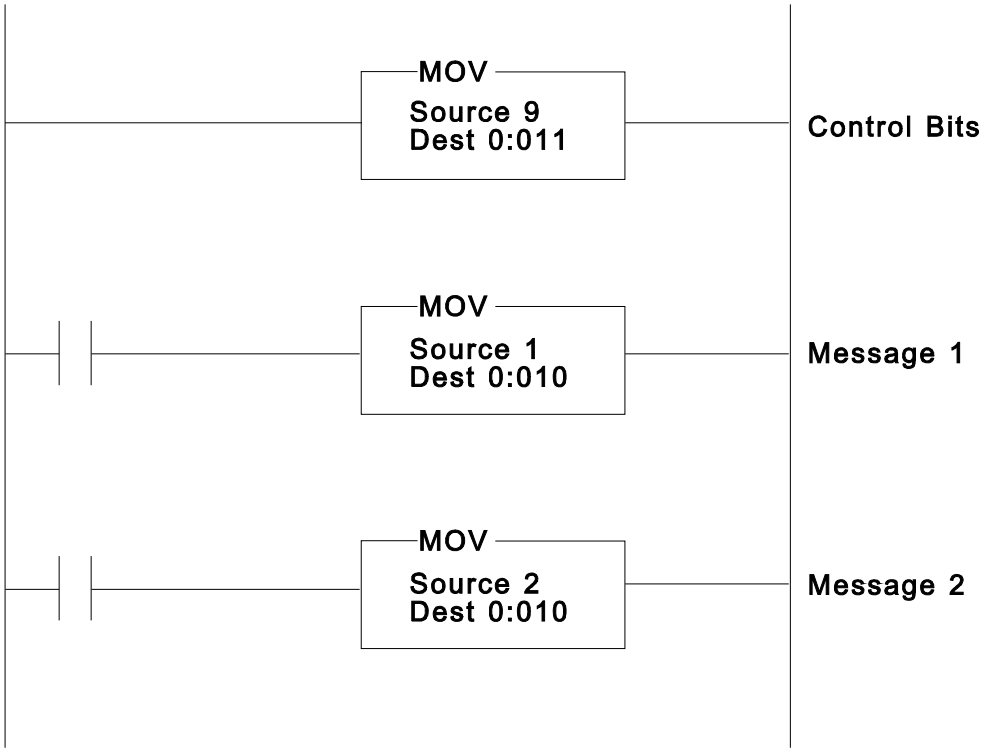
Offline programming menu:



Remote I/O Example


System Attribute Setup
PLC3/PLC5 PLC Type (PLC 2 or PLC 3/5)
FULL Rack Size (1/4, 1/2, 3/4, FULL)
0 Start 1/4 of Partial Rack (0-3)
1 PLC I/O Rack # (1-7, 0-37)
0 I/O Group # (0-7) (Data Bits)
1 I/O Group # (0-7) (Control Bits)
Y Active Rack
230.4 K Baud Rate
N Map Data Sets to PLC
0 Number of Data Sets (1-4)
1 Starting Data Set (1-4)
0 PLC I/O Rack # (1-7, 0-37)
0 PLC I/O Group # (0-7)
Y Map Data Sets to PMD
4 Number of Data Sets (1-4)
1 Starting Data Set (1-4)
1 PLC I/O Rack # (1-7, 0-37)
2 PLC I/O Group # (0-7)
Y Active Rack
Y Remote I/O
N Remote I/O Block Transfer
N Data Highway
N Map Function Keys (1-16) to PLC
0 PLC I/O Rack # (1-7, 0-37)
0 PLC I/O Group # (0-7)
N Map Function Keys (17-32) to PLC
N Map Function Keys (17-48) to PLC
PMD LED Control (1-16) (PLC-PMD-CI)
0 PLC I/O Rack # (1-7, 0-37)
0 PLC I/O Group # (0-7)
N Active Rack
N Control LEDs (17-32)
N Control LEDs (17-48)
F1-HELP F2-MAIN F3-COMM F4-EDITOR F5-PLC F6-DEFAULTS F10-EXIT/SAVE

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE



SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

DATA HIGHWAY

The Allen-Bradley Data Highway interface can be configured to communicate using either Data Highway or Data Highway Plus (from here-on the Data Highway/Data Highway Plus interface will be referred to as Data Highway unless otherwise specified). The PMD cannot process remote addressing. It can only communicate on the local Data Highway network. Up to 64 nodes can be connected to a Data Highway Plus network and up to 254 nodes can be connected to a Data Highway network. The PMD can operate in two modes. It can poll a specific PLC for a data table and send an input table to this PLC each time the input data changes. It can also wait on the Data Highway until another node sends a table of data to the PMD. This method minimizes the required traffic on the data highway communication link.

The PLC data tables used by the Data Highway interface are described in **PART F: REFERENCE** in this **SECTION 7 - PLC INTERFACING**. These are the same data tables that are used by the other PLC interfaces. Each data table is 16 words long. The PLC output data table is data that is read from the PLC by the PMD or is sent to the PMD by the PLC. The PLC input data table is data that is read from the PMD by the PLC or is sent to the PLC by the PMD.

Polling a Specific PLC

The Data Highway interface operates much like the other PLC interfaces. It can operate in two different modes. In one mode, the 450 constantly polls a specific range of addresses in a specific PLC. This mode will continually send commands to the specified PLC to read 16 words of data that will be used for the PLC output table. As soon as the PMD gets a reply to one command, it will issue another read command. The PLC output table being read by the PMD is monitored for changes, and the PLC does not have to initiate any transactions to send the data to the PMD on the data highway. In this mode, the PMD will also send a copy of the PLC input table to the PLC each time the data is changed.

The PMD will monitor a table of 16 registers from an integer file. The integer file is of type 'N' and the file number can range from 0-999. The table of registers begins at the element number within the integer file. The configuration procedure will only prompt for the file number and element number (0-9999). A typical address would look like N11:0. This would be integer file number 11, element 0.

Receiving Data from any PLC

In the other mode of operation, the PMD does not initiate any transactions on the Data Highway. All transactions must be initiated by another station on the data highway network. This mode of operation minimizes the number of transactions on the network.

Each time the data is changed in the PLC output table, the PLC must send this data to the 450. The PLC can also issue read commands to the 450 to obtain a copy of the current input table in the PMD. Any station on the network can issue commands to the PMD. This allows multiple stations to display messages on a single PMD.

Note: Commands that are sent to a PMD must specify a specific address to read data from or write data to. If a different address is used, the command is rejected. The address needed by each command accepted by the PMD is shown below.

Write Commands

The PMD 450 can process three different write commands. These are the Write commands issued by the PLC when processing the MSG Instruction. The data must be sent to a specific address within the PMD. The address for each instruction within the PMD is specified below. If the address does not match the address shown for each instruction below, the PMD will respond with the error code shown below. The PMD uses this address to determine what the data will be used for.

PLC 5 Typed Write

Length to Read	16 words
Address to Read	N10:1

Error code STS = 0xF0 EXTSTS = 0x12 invalid parameter

PLC 2 Unprotected Write

Length to Read	16 words
Address to Read	10 (PLC2 word address)

Error code STS = 0x50 addressing problem

PLC 3 Word Range Write

Length to Read	16 words
Address to Read	N10:1

Error code STS = 0xF0 EXTSTS = 0x12 invalid parameter

Read Commands

The PMD 450 can process three different read commands. These are the Read commands issued by the PLC when processing the MSG Instruction. The data must be read from a specific address within the PMD. If the address does not match the address shown for each instruction below, the PMD will respond with the error code shown below. This address is used to determine what data will be returned in the reply to a read command.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

PLC 5 Typed Read

Length to Read 16 words
Address to Read N10:1
Error code STS = 0xF0 EXTSTS = 0x12 invalid parameter

PLC 2 Unprotected Read

Length to Read 16 words
Address to Read 10 (PLC2 word address)
Error code STS = 0x50 addressing problem

PLC 3 Word Range Read

Length to Read 16 words
Address to Read N10:1
Error code STS = 0xF0 EXTSTS = 0x12 invalid parameter

Wiring Changes

The wiring connection for data highway is different than the connection for Remote I/O. The 1 and 2 wires are reversed when they are connected to terminals 1 and 2 and communicating with data highway. The terminals should be connected according to the diagram shown.

REMOTE I/O	**DATA HIGHWAY/PLUS**
Terminal 1--------1	Terminal 1--------2
Terminal SH--------Shield	Terminal SH--------Shield
Terminal 2--------2	Terminal 2--------1

DATA HIGHWAY Setting up 450 A-B Communication

To configure the 450 for Data Highway online, **select** **SETUP PARAMETERSfrom the MAIN>COMMANDS** menu and then **select** option **5- AB PLC COMMUNICATION** and option **17 - FUNCTION KEYS/LED CONTROL** If you opt to configure the 450 using offline programming, access the menu starting from the **MAIN MENU**, then select **EDIT MESSAGE PROGRAM**, then **SETUP ATTRIBUTES**, and finally, **F5 - PLC**. Illustrations of the offline menus follow this segment on the "online" prompts. The explanations of the options, however, are universal for both programming methods.

Set up Parameters Online Programming Prompts Option 5

!PLC TYPE (1=PLC2, 0=PLC3/PLC5) <0>

This prompt determines which type of data highway commands will be used when the PMD polls for data on the Data Highway/Plus interface. The PLC2 commands used are the unprotected read/write commands. The PLC 3/5 commands are the word range read/write.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

!ENABLE DATA HIGHWAY (Y/N) <N>
This prompt determines whether or not the Data Highway interface will be loaded and used for PLC communication. Therefore, if you want to use Data Highway, you must answer this prompt as **Y**.

!ENABLE DATA HIGHWAY PLUS (Y/N) <N>
This prompt determines whether or not the Data Highway Plus interface will be loaded and used for PLC communication. Therefore, if you want to use Data Highway Plus, you must answer this prompt as **Y**.

!DATA HIGHWAY PLUS @ 115.2 K BAUD (Y/N) <N>
This prompt determines if the Data Highway Plus interface will be configured to run at 115.2 K baud instead of the default 57.6 K baud.

!ENABLE DATA HIGHWAY READ (Y/N) <N>
This prompt determines if the **PMD** should poll a specific PLC for a PLC output table. If this is answered as **N**, then the PLC must send the data to the PMD. If it is answered as **Y**, the PMD will continually read the data from the specified PLC address.

!ENABLE DATA HIGHWAY WRITE (Y/N) <N>
This prompt determines if the **PMD** will send its PLC input table to a specific PLC each time the inputs change. If this is answered as **N**, then the data must be read by the PLC. If it is answered as **Y**, the PMD will send the inputs to the PLC each time any value in the input data changes.

!PROCESS ASYNCHRONOUS DATA (Y/N) <N>
This prompt determines whether or not the unit will process read/write commands that are sent to it by the **PLC**. If this is answered as **N**, the PMD will reject all commands sent from a PLC to read or write data. If this is answered as **Y**, the PMD will process the commands that it recognizes when they are sent from a PLC.

!DATA HIGHWAY STATION ADDRESS (0-376 OCTAL) <1> or DATA HIGHWAY PLUS STATION ADDRESS (0-77 OCTAL) <1>
This is the station number of the PMD on the Data Highway/Plus network. The station address depends on which type of network will be used. Data Highway Plus only supports 64 stations. Data Highway supports 254 stations. These numbers are represented in octal when presented in the prompt for station number.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

!DATA HIGHWAY DESTINATION STATION ADDRESS (0-376 OCTAL) <1> or DATA HIGHWAY PLUS DESTINATION STATION ADDRESS (0-77 OCTAL) <1>
This is the station address of the PLC that the PMD will poll for output data and/or send its input table to.

!DATA HIGHWAY READ PLC2 WORD ADDRESS (0-65535) <0> or DATA HIGHWAY READ FILE NUMBER (0-999) <0> DATA HIGHWAY READ FILE ELEMENT (0-9999) <0>
This is the starting address in the PLC of the output data table that will be read by the PMD. The output data table will start with the word specified here and take the next 16 words. This address depends on which type of PLC will be polled. The PLC 2 address is a word address into the PLC2's memory. The PLC3/5 address is an integer file number and an element number within the file. The PMD will display the proper prompt depending on which type of PLC is chosen.

!DATA HIGHWAY WRITE PLC2 WORD ADDRESS (0-65535) <0> or DATA HIGHWAY WRITE FILE NUMBER (0-999) <0> DATA HIGHWAY WRITE FILE ELEMENT (0-9999) <0>
This is the starting address in the PLC of the input data table that will be written by the PMD. The input data table will start with the word specified here and take the next 16 words. This address depends on which type of PLC will be written to. The PLC2 address is a word address into the PLC2's memory. The PLC3/5 address is an integer file number and an element number within the file. The PMD will display the proper prompt depending on which type of PLC is chosen.

!NUMBER OF DATA SETS TO MAP TO PLC (0=NONE, 1-4) <0>
This prompt determines how many data sets will be mapped from the 450 to the PLC. As many data sets as are specified will be mapped into the PLC beginning with the starting data set.

!STARTING DATA SET NUMBER (1-4) <1>
This prompt allows you to choose which data set to start with. As many data sets as are specified will be mapped into the PLC beginning with the starting data set.

!NUMBER OF DATA SETS TO MAP TO PMD (0=NONE, 1-4) <0>
Another option you are given is to map the data sets from the PLC to the 450.

!STARTING DATA SET NUMBER (1-4) <1>
This prompt asks you which data set to start with.

!INVERTED/NON-INVERTED (0=INVERTED, 1=NON-INVERTED) <0>
This prompt lets you determine how the *PLC port* looks at the message/data lines: the way they come in as 1s or 0s (non-inverted)...**or** the complements of these (inverted).

!BCD/BINARY (0=BCD, 1=BINARY) <0>
This prompt lets you determine whether the 450 will consider the message/ data lines as a Binary or Binary Coded Decimal.

!TIMED SCAN (1=ENABLED, 2=DISABLED) <2>
This prompt is for setting a specified scan time for how often the *PLC port* is scanned at the "system" scan rate (between 1 and 180 msec).

!QUEUE SIZE (0-20, 0=DISABLED) <0>
The queue is automatically disabled and it cannot be enabled from the PLC interface when the size is specified as 0. If the size is specified as 0, the unit will use the 32 entry FIFO if a nonzero message display time is specified. The queue is configured to the size specified by this prompt and then it must be enabled by the PLC interface before it will accept any add, delete, or clear functions.

!MESSAGE DISPLAY TIME (0-255 SEC.) <0>
This prompt determines if the messages will remain on the display until replaced, or if they will be displayed for a specified time.

When the display time is set to zero, all messages will remain on the display until they are replaced. If a specified time is entered (in seconds ranging from 1 to 255), the messages will each display for that time period before being replaced by the next.

Set up Parameters Online Programming Prompts Option 17

Select SETUP PARAMETERS from the **MAIN>COMMANDS** menu and then **select** option **17 - FUNCTION KEYS/LED CONTROL**. The terminal responds with these prompts:

!LED CONTROL (SELECT 1=PMD, 2=CI, 3=PLC) <1>
lets you choose what you want to use as the input source for the LEDs. If **1=PMD** is selected, the LEDs will be controlled by the function keys. If **2=CI** (*computer interface*) is selected, the LEDs will be controlled by computer interface commands. If **3=PLC** is selected, the LEDs will be controlled by the PLC outputs. A word of outputs will control 16 LEDs on a bit map basis. Bit 0 of the word controls LED 1 and bit 15 of the word controls LED 16.

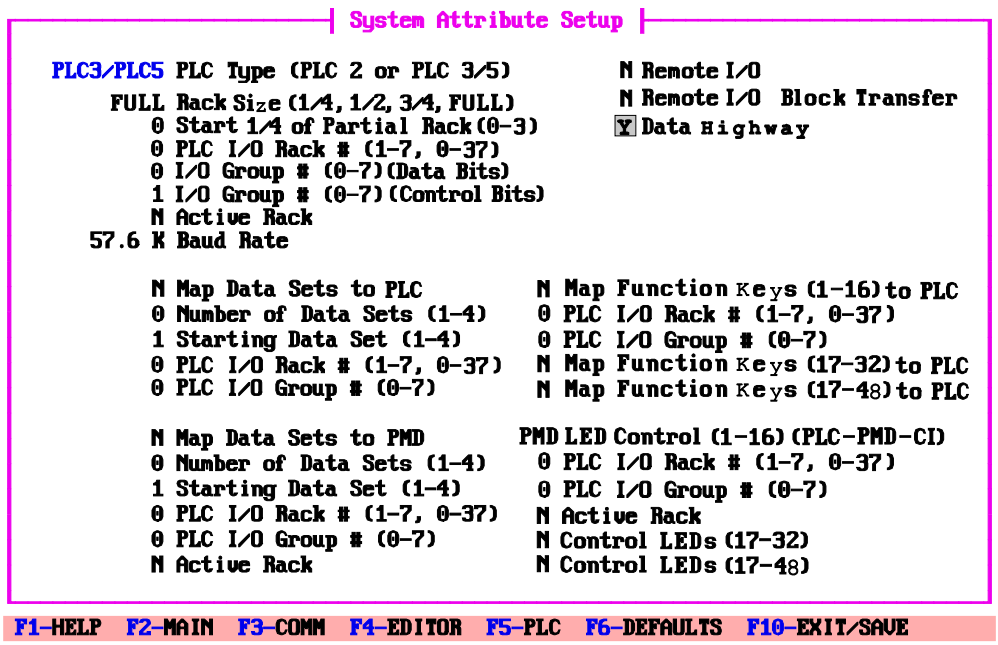
!NUMBER OF LEDS CONTROLLED BY PLC (0=NONE, 1=16, 2=32, 3=48) <0>-- (only if 3=PLC on the previous prompt)

is used to specify how many LEDs will be controlled by the outputs of the PLC. The prompt will reflect how many LEDs are available on the unit. The LEDs are controlled by output words 3, 4, and 5 of the PLC output data being monitored by the 400 product. If the output word is not controlling LEDs, it will be ignored. Each output word can control 16 LEDs on a bit map basis.

!NUMBER OF FUNCTION KEYS TO MAP TO PLC (0=NONE, 1=16, 2=32, 3=48) <0>

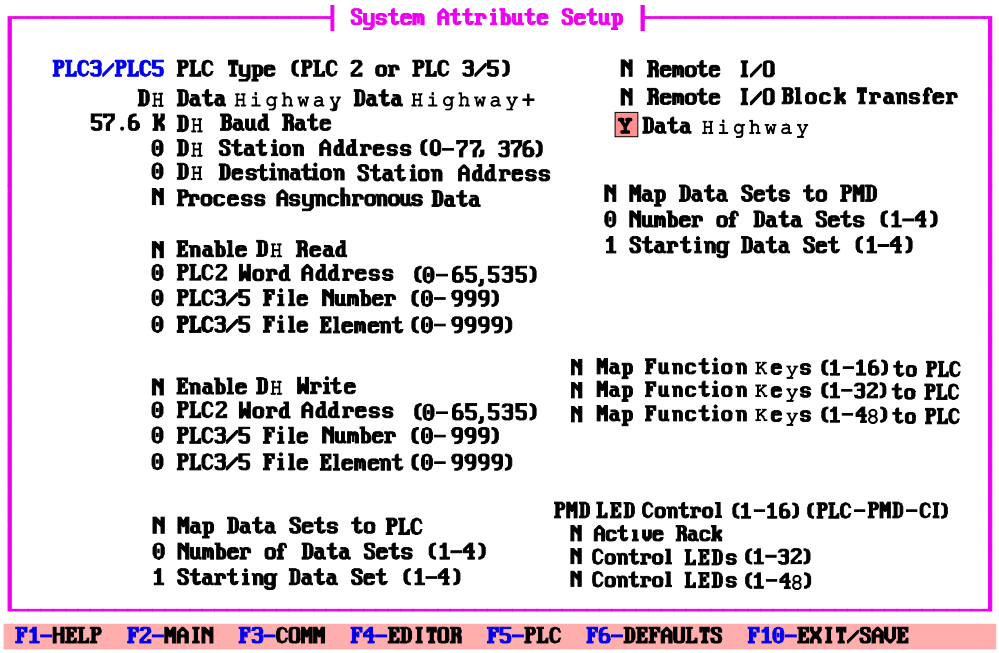
allows you to map the function keys to set inputs in the PLC. This prompt will also be dependent on the number of function keys that are available on the unit. Each set of 16 keys is bit mapped to one word of input. Function key 1 sets bit 0 and function key 16 sets bit 15. When the function key is pressed, the input bit is set, and when the function key is released, the bit is cleared. The function keys are mapped to input words 5, 6 and 7.

Offline programming menus:



SECTION 7 - PLC INTERFACING

PART A: 450 A-B INTERFACE



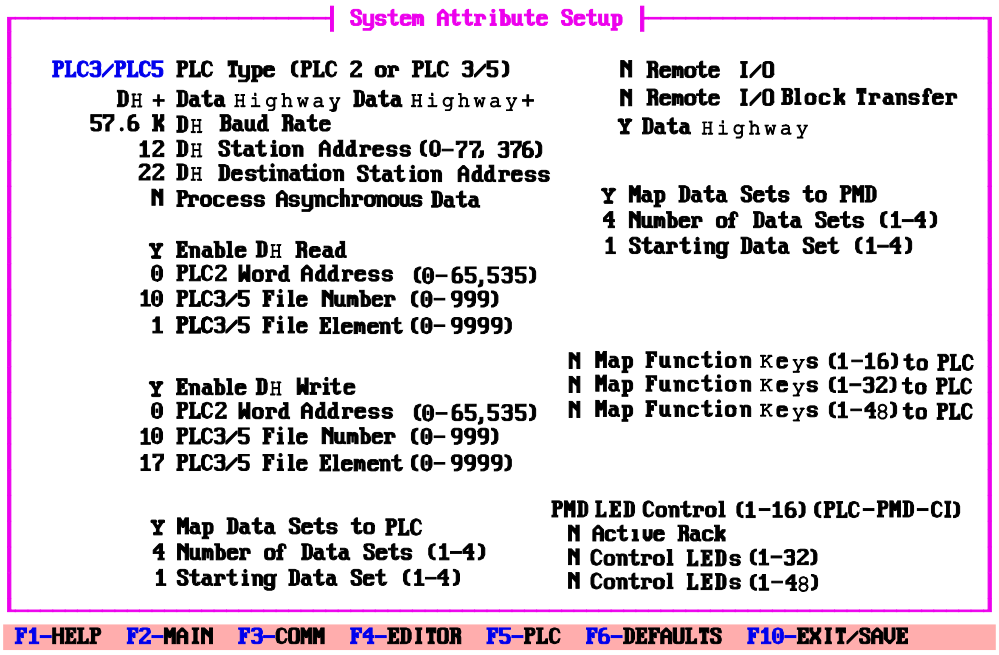
Data Highway Example

Polling a Specific PLC - This example shows how to set up a PMD 450 to poll a PLC connected to a Data Highway Plus interface. This configuration

defines the PMD to be node 12 and communicates to a PLC at node 22. The PMD will read 16 words from integer file N10:1. This data will be used for the PLC output table in the PMD. When any data from the PLC input table changes, the PMD will send a copy of the PLC input table to N10:17 in the PLC. With a configuration like this, the PLC can trigger a message by setting the Control Word of the output table to 9 and putting a valid message number in the Message/Data word.

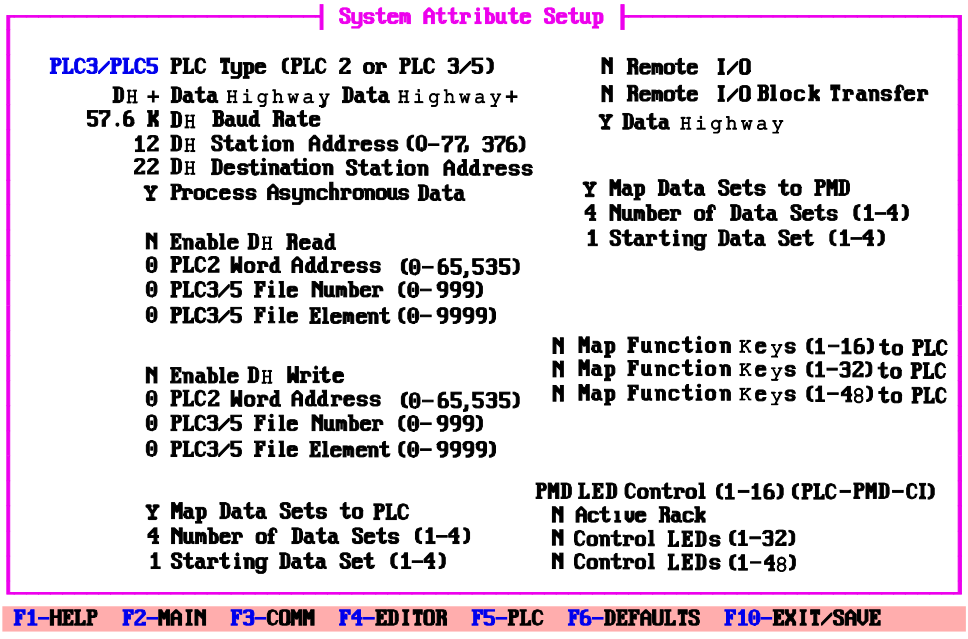
SECTION 7 - PLC INTERFACING

PART A: 450 A-B INTERFACE



Receiving data asynchronously - This example shows the configuration required for the PMD to accept commands from a PLC on a Data Highway Plus Network. The PMD will be configured as node 12. The MSG instructions in the PLC should be set

up as shown.


System Attribute Setup
PLC3/PLC5 PLC Type (PLC 2 or PLC 3/5)
DH + Data Highway Data Highway+
57.6 K DH Baud Rate
12 DH Station Address (0-77, 376)
22 DH Destination Station Address
Y Process Asynchronous Data
N Enable DH Read
0 PLC2 Word Address (0-65,535)
0 PLC3/5 File Number (0-999)
0 PLC3/5 File Element (0-9999)
N Enable DH Write
0 PLC2 Word Address (0-65,535)
0 PLC3/5 File Number (0-999)
0 PLC3/5 File Element (0-9999)
Y Map Data Sets to PLC
4 Number of Data Sets (1-4)
1 Starting Data Set (1-4)
N Remote I/O
N Remote I/O Block Transfer
Y Data Highway
Y Map Data Sets to PMD
4 Number of Data Sets (1-4)
1 Starting Data Set (1-4)
N Map Function Keys (1-16) to PLC
N Map Function Keys (1-32) to PLC
N Map Function Keys (1-48) to PLC
PMD LED Control (1-16) (PLC-PMD-CI)
N Active Rack
N Control LEDs (1-32)
N Control LEDs (1-48)
F1-HELP F2-MAIN F3-COMM F4-EDITOR F5-PLC F6-DEFAULTS F10-EXIT/SAVE

This MSG instruction will write 16 words from address N25:0 to the PMD. The destination data table address must be N10:1. Any other address will result in an error.

MESSAGE INSTRUCTION DATA ENTRY FOR CONTROL BLOCK N7:10

Communication Command:	PLC-5 Typed Write
PLC-5 Data Table Address:	N25:0
Size in Elements:	16
Local/Remote:	LOCAL
Remote Station:	N/A
Link ID:	N/A
Remote Link Type:	N/A
Local Node Address:	12
Destination Data Table Address:	N10:1

This MSG instruction will read 16 words from address N10:1 in the PMD and place the data received at address N25:16. The destination data table address must be N10:1. Any other address will result in an error.

MESSAGE INSTRUCTION DATA ENTRY FOR CONTROL BLOCK N7:30

Communication Command:	PLC-5 Typed Read
PLC-5 Data Table Address:	N25:16
Size in Elements:	16
Local/Remote:	LOCAL
Remote Station:	N/A
Link ID:	N/A
Remote Link Type:	N/A
Local Node Address:	12
Destination Data Table Address:	N10:1

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

BLOCK TRANSFER

The PMD 450 Allen-Bradley PLC interface supports block transfers. When block transfers are enabled, your 450 will no longer process Remote I/O data. The PMD 450 supports block transfer read and write functions from the PLC. The block transfer write function can either monitor a real block transfer from an active rack or it can emulate an active rack and process the block transfer itself. It accepts up to 16 words of output from the PLC. If more than 16 words are sent, only the first 16 are used. If fewer than 16 are sent from the PLC, it is possible that some of the data in the output table will be 0s. The block transfer read function emulates an active rack and processes the block transfer data. These functions can be enabled independently.

If a block transfer is specified as active, it will be active as either a quarter rack on the rack and group specified in the block transfer or as a full rack on the rack specified.

Setting up 450 A-B Communication

To configure the 450 for Block Transfer online, **select** **`SETUP PARAMETERS`** from the **`MAIN COMMANDS`** menu and then **select** option **`5 - AB PLC COMMUNICATION`** and option **`17 - FUNCTION KEYS/LED CONTROL`**. If you opt to configure your 450 using offline programming, you access your menu starting from the **`MAIN MENU`**, then selecting **`EDIT MESSAGE PROGRAM`**, then **`SETUP ATTRIBUTES`**, and finally, **`F5 - PLC`**. Illustrations of the offline menus follow this segment on the "online" prompts. The following explanations of the options, however, are universal for both programming methods.

Set up Parameters Online Programming Prompts

When you select option **`5`**, the terminal responds with the following prompts.

Option 5

`!PLC TYPE (1=PLC2, 0=PLC3-PLC5) <0>`

This option lets you select the system with which the 450 will be operating: PLC2 or PLC3/PLC5. Your response will set the 450 for the correct Allen-Bradley system so that the 450 will specify the current valid range for the rack.

`!ENABLE DATA HIGHWAY (Y/N) <N>`

This prompt determines whether or not the Data Highway interface will be loaded and used for PLC communication. Therefore, if you want to use Block Transfer, you must answer this prompt as **`N`**.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

!ENABLE DATA HIGHWAY PLUS (Y/N) <N>
This prompt determines whether or not the Data Highway Plus interface will be loaded and used for PLC communication. Therefore, if you want to use Block Transfer, you must answer this prompt as **N**.

!ENABLE BLOCK TRANSFER (Y/N) <N>
This prompt allows the unit to enable block transfer functions. This prompt must be answered as **Y** to access the remaining prompts required to enable block transfer. When block transfers are enabled, none of the prompts pertaining to Remote I/O will be presented.

!ENABLE BLOCK TRANSFER WRITE (0=NONE, 1=ACTIVE, 2=NOT ACTIVE) <0>
The PMD 450 can either monitor an active block transfer write (inactive mode) or emulate an active rack and reply to the block transfer write (active mode). The block transfer write function will monitor up to 16 words of data. The block transfer write function must send at least ten words of information to fill the output table in the PMD. If the write function sends fewer than ten words of data, the output words at the end of the table will remain 0s.

This prompt determines which type of block transfer write should be enabled. If you select **0=NONE**, no block transfer write will be enabled. If you choose **1=ACTIVE**, a block transfer write will be enabled on the rack and group specified in the following prompts. The rack and group specified will be made active either as a full rack or as a partial rack depending on how you answer the next prompt.

If you respond **2=NOT ACTIVE**, a block transfer write function that is currently active will be monitored according to the rack and group specified in the following prompts.

!ENABLE PARTIAL RACK (0=NO, 1=YES) <0>
This prompt determines whether an active rack will be enabled as a full rack or as a partial rack. A partial rack will be enabled on the rack and group specified for the block transfer. If this is going to be a nonactive block transfer, this prompt determines which portion of the rack status will be monitored. If this is a nonactive block transfer that will be monitoring an active partial rack, this should be enabled as a partial rack to ensure that the proper rack status is monitored.

SECTION 7 - PLC INTERFACING
PART A: 450 A-B INTERFACE

!PLC I/O RACK # (0-37 OCTAL) <0>
This prompt specifies which rack the PMD will emulate (active rack) or monitor (nonactive rack).

!PLC I/O GROUP # (0-7) <0>
This prompt specifies the I/O group within the rack that the PMD will emulate or monitor (single-slot addressing only).

!PLC I/O MODULE # (0-1) <0>
This prompt specifies the module within the I/O group.

!ENABLE BLOCK TRANSFER READ (0=NONE, 1=ACTIVE) <1>
The PMD 450 must be an active rack to respond to block transfer read functions. The block transfer read function can request up to 16 words of information from the PMD. The PMD will return as many words as are requested, up to 16 words. The words that are returned will come from the beginning of the input table. If five words are requested, the PMD will send back the first five words of the input table.

This prompt determines which type of block transfer read should be enabled. If you choose **0=NONE**, no block transfer read will be enabled. If you select **1=ACTIVE**, a block transfer read will be enabled on the rack and group specified in the following prompts. (The rack and group specified will be made active as a quarter rack).

!ENABLE PARTIAL RACK (0=NO 1=YES) <0>
This prompt determines whether an active rack will be enabled as a full rack or as a partial rack. A partial rack will be enabled on the rack and group specified for the block transfer.

!PLC I/O RACK # (0-37) <0>
This prompt specifies which rack the PMD will emulate (active rack) for the active block transfer read.

!PLC I/O GROUP # (0-7) <0>
This option specifies the I/O group within the rack that the PMD will emulate.

!PLC I/O MODULE # (0-1) <0>
This option specifies the module within the I/O group.

!NUMBER OF DATA SETS TO MAP TO PLC (0=NONE, 1-4) <0>
This prompt determines how many data sets will be mapped from the 450 to the PLC. If you choose 0, then none of the four following prompts will appear. As many data sets as are specified will be mapped into the PLC beginning with the starting data set.

!STARTING DATA SET NUMBER (1-4) <1>
This prompt allows you to choose which data set to start with. As many data sets as are specified will be mapped into the PLC beginning with the starting data set.

!NUMBER OF DATA SETS TO MAP TO PMD (0=NONE, 1-4) <0>
Another option you are given is to map the data sets from the PLC to the 450. This allows variable data to be read by the PMD from any existing or non-existing I/O rack.

!STARTING DATA SET NUMBER (1-4) <1>
This prompt asks you which data set to start with. It is followed by a prompt for you to indicate the PLC I/O rack number. You intend for the 450 to read variable data.

!PLC BAUD RATE (0=57.6KB 1=115.2KB 3=230.4KB) <0>
This option sets the communication baud rate.

!BLOCK TRANSFER COMMAND MODE (0=DISABLED,1=ENABLED)<0>
The block transfer command mode is used to interface a computer on the Data Highway that is running UTICOR's 10F50 programming software with a PMD 350 on the network. It is required that the computer have an Allen-Bradley 1784-KT card in it.

BLOCK TRANSFER WRITE COMMANDS
!ENABLE PARTIAL RACK (0=NO, 1=YES)<0>
This prompt determines if the rack specified will be enabled as a whole or partial rack. The rack specified will be made active as a whole rack if a partial rack is not specified.

!PLC I/O RACK # (1-7 OCTAL)<1> OR PLC I/O RACK # (0-37 OCTAL)<0>
This prompt determines which rack number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O GROUP # (0-7)<0>
This prompt determines which group number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O MODULE # (0-1)<1>
This prompt determines which module number will be made active and where the PMD will look for the block transfer commands.

BLOCK TRANSFER READ COMMANDS

!ENABLE PARTIAL RACK (0=NO, 1=YES)<0>
This prompt determines if the rack specified will be enabled as a whole or partial rack. The rack specified will be made active as a whole rack if a partial rack is not specified.

!PLC I/O RACK # (1-7 OCTAL)<1> OR PLC I/O RACK # (0-37 OCTAL)<0>
This prompt determines which rack number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O GROUP # (0-7)<0>
This prompt determines which group number will be made active and where the PMD will look for the block transfer commands.

!PLC I/O MODULE # (0-1)<1>
This prompt determines which module number will be made active and where the PMD will look for the block transfer commands.

!INVERTED/NON-INVERTED (0=INVERTED, 1=NON-INVERTED) <1>
This prompt lets you determine how the PLC port looks at the message/data lines: the way they come in as 1s or 0s (non-inverted)**...or** the complements of these (inverted).

!BCD/BINARY (0=BCD, 1=BINARY) <0>
This prompt lets you determine whether the 450 will consider the message/data lines as a Binary or Binary Coded Decimal.

!TIMED SCAN (1=ENABLED, 2=DISABLED) <2>
This prompt is for setting a specified scan time for how often the *PLC port* is scanned at the "system" scan rate (between 1 and 180 msec).

!**QUEUE SIZE (0-20, 0=DISABLED) <0>**
The queue is automatically disabled and it cannot be enabled from the PLC interface when the size is specified as 0. If the size is specified as 0, the unit will use the 32 entry FIFO if a nonzero message display time is specified. The queue is configured to the size specified by this prompt and then it must be enabled by the PLC interface before it will accept any add, delete, or clear functions.

!**MESSAGE DISPLAY TIME (0-255 SEC.) <0>**
This prompt determines if the messages will remain on the display until replaced, or if they will be displayed for a specified time. When the display time is set to zero, all messages will remain on the display until they are replaced.

If a specified time is entered (in seconds ranging from 1 to 255), the messages will each display for that time period before being replaced by the next.

Set up Parameters Online Programming

Select SETUP PARAMETERS from the **MAIN>COMMANDS** menu and then **select** option **17 - FUNCTION KEYS/LED CONTROL**. The **Prompts** terminal responds with these prompts:

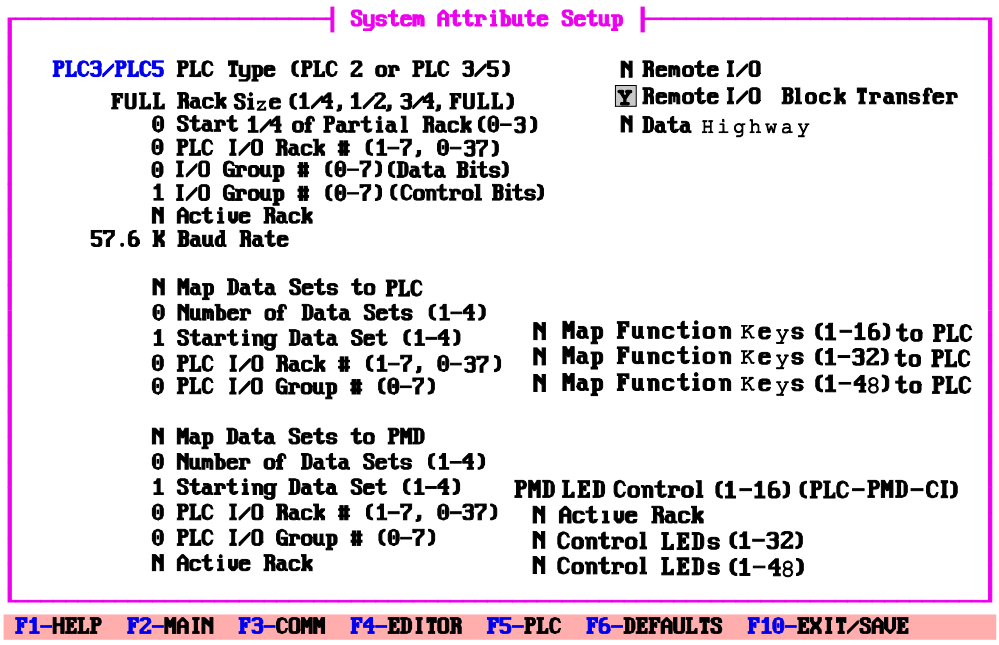
Option 17

!**LED CONTROL (SELECT 1=PMD, 2=CI, 3=PLC) <1>**
lets you choose what you want to use as the input source for the LEDs. If **1=PMD** is selected, the LEDs will be controlled by the function keys. If **2=CI** (*computer interface*) is selected, the LEDs will be controlled by computer interface commands. If **3=PLC** is selected, the LEDs will be controlled by the PLC outputs. A word of outputs will control 16 LEDs on a bit map basis. Bit 0 of the word controls LED 1 and bit 15 of the word controls LED 16.

!**NUMBER OF LEDS CONTROLLED BY PLC (0=NONE, 1=16, 2=32, 3=48) <0>-- (only if 3=PLC on the previous prompt)** is used to specify how many LEDs will be controlled by the outputs of the PLC. The prompt will reflect how many LEDs are available on the unit. The LEDs are controlled by output words 3, 4, and 5 of the PLC output data being monitored by the 400 product. If the output word is not controlling LEDs, it will be ignored. Each output word can control 16 LEDs on a bit map basis.

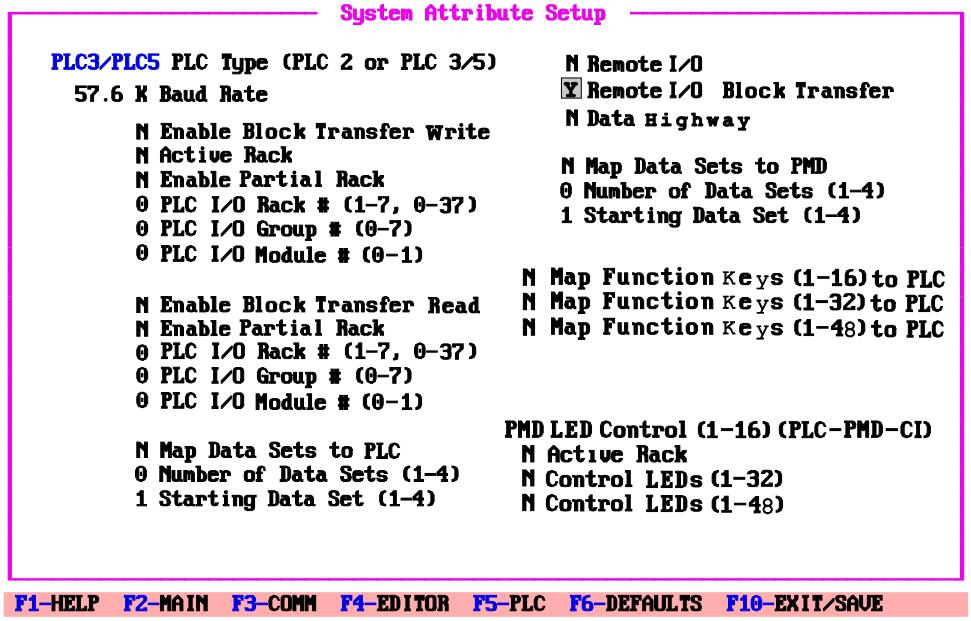
!NUMBER OF FUNCTION KEYS TO MAP TO PLC (0=NONE, 1=16, 2=32, 3=48) <0> allows you to map the function keys to set inputs in the PLC. This prompt will also be dependent on the number of function keys that are available on the unit. Each set of 16 keys is bit mapped to one word of input. Function key 1 sets bit 0 and function key 16 sets bit 15. When the function key is pressed, the input bit is set, and when the function key is released, the bit is cleared. The function keys are mapped to input words 5, 6 and 7.

Offline programming menus:

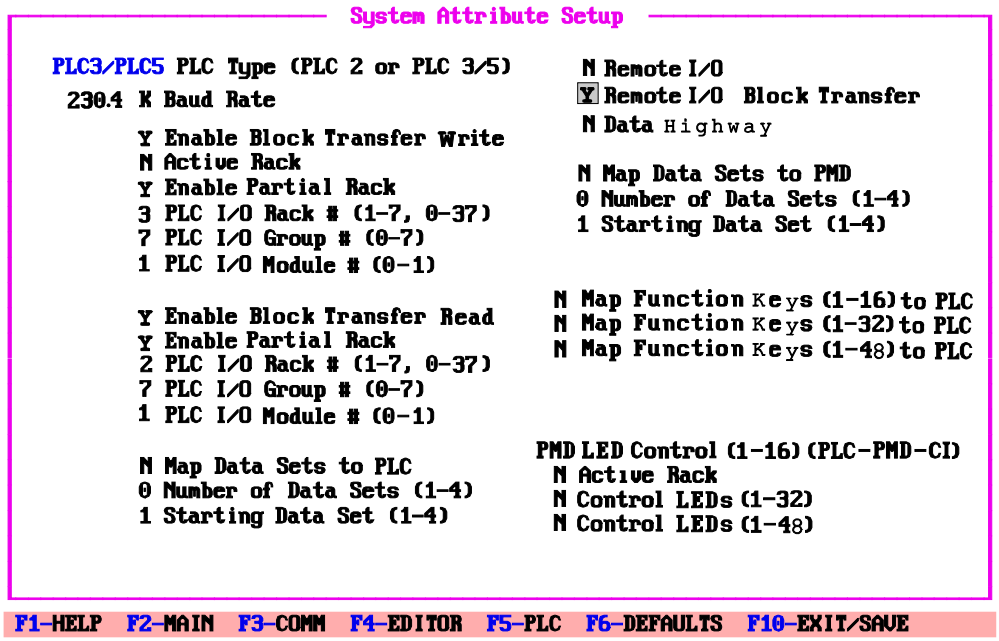


SECTION 7 - PLC INTERFACING

PART A: 450 A-B INTERFACE



Block Transfer Example



SECTION 7 - PLC INTERFACING
PART B: 450 A-B INTERFACE

Block Transfer Write - This example uses an active block transfer write function to transfer 16 words of data to the PMD. The PMD will be active as a quarter rack on rack 3, I/O groups 6 and 7. The data that is going to be sent to the PMD will be in the Integer File N7 at element 50. The first word of the output table (Message/Data Word) will be N7:50. The second word of the output table (Control Word) will be N7:51. The last word of the output table will be N7:65.

The PLC setups for the block transfer write function are as follows:

Rack	3
Group	7
Module	1
Control Block	N7:10
Data File	N7:50
Length	16
Continuous	Y

If desired subsequent PMDs could monitor this same block transfer by setting them up identical to this one and enabling them as not active, this would permit multiple PMDs to monitor one block transfer function.

Block Transfer Read - This example also uses an active block transfer read function to reply to a block transfer read function issued by the PLC. The block transfer must be active in order to return input data back to the PLC. The first word of the input data (data set 1) will be seen in N7:90. The input data will be placed in the next 16 words through N7:105.

The PLC setups for the block transfer read function are as follows:

Rack	2
Group	7
Module	1
Control Block	N7:20
Data File	N7:90
Length	16
Continuous	Y

SECTION 7 - PLC INTERFACING
PART B: 450 A-B INTERFACE

Pass-Through Programming

The PMD 450 allows pass-through programming for units that are operating in either remote I/O or block transfer mode. Pass-through programming is available through 10F50 programming software under File Network of the Main Menu. Pass-through programming allows the display to receive computer interface commands from a source node through the PLC that is communicating with the PMD.

In order to use pass-through programming, there must be an Allen-Bradley 1784-KT interface in the computer. In order to communicate to the PMD 450, the following setups must be followed.

1. Take the appropriate steps to initialize the KT card and load the TSRs.

2. Configure the computer interface port group, unit and checksum setting in the PMD to match the settings in 10F50.

3. Configure the Block Transfer Command Mode attributes in the PMD 450.

4. Connect the PMD to the PLC and configure the PLC to support the appropriate amount of remote I/O. This can be done using AutoConfig.

5. Start 10F50 (**type PMDP**) and use the File Network option to access the Network Interface Setup Screen. Configure the 1784-KT card and connect the 1784-KT card to the Data Highway network.

6. Use the File Network menu to communicate with the specified PMD.

Remote I/O

The remote I/O interface requires an additional quarter rack to perform the block transfer command mode. This extra quarter rack cannot be used for anything else besides the pass-through commands. Pass-through programming mode operates independent of the remote I/O interface.

The PMD receives the data from a block transfer write. If the data in the block transfer write completes the command, the PMD will send the computer interface reply in the next block transfer read it receives. The source node has to issue a block transfer read command to the PMD to obtain the reply. The PMD cannot send its reply to the command until it receives a block transfer read from the PLC. Block transfers can only exchange 64 words (128 bytes) of data per block transfer. Therefore, each computer interface command should take three block transfers at the most.

SECTION 7 - PLC INTERFACING
PART B: 450 A-B INTERFACE

Block Transfer

The block transfer interface requires an additional quarter rack to perform the pass-through command mode. This extra quarter rack cannot be used for anything else besides the pass-through commands. Pass-through programming mode operates independent of the block transfer interface.

The PMD receives the data from the block transfer write. If the data in the block transfer write completes the command, the PMD will send the computer interface reply in the next block transfer read it receives. The source node has to issue a block transfer read command to the PMD to obtain the reply. The PMD cannot send its reply to the command until it receives a block transfer read from the PLC. Block transfers can only exchange 64 words (128 bytes) of data per block transfer. Therefore, each computer interface command should take three block transfers at the most.

Pass-Through Packet Definitions

The data sent in the PLC command will be the complete computer interface packet without the leading 0xAA word.

SECTION 7 - PLC INTERFACING
PART B: 460 SIEMENS/TI INTERFACE

This section is intended to give you specific information about the 460 SIEMENS/TI Interface product that makes it unique from the basic 400. It is a supplement to, and a specified reference point within, the preceding manual sections.

The PMD 460 is similar to the PMD 400 and contains an interface to Siemens/Texas Instruments Series 545, 560 and 565 PLCs which have the RS-485 Remote I/O module. This interface will support many features, including triggering messages and transferring data directly from the PLC to the 460 data sets. The 460 will appear as an RBC (Remote Base Controller) to the SIEMENS/TI PLC. The 460 can listen to an existing RBC and use the information from an actual RBC (non-active mode), or it can imitate an RBC and communicate with the PLC (active mode). If the 460 is selected to be a non-active RBC, it will rely on information from an existing active RBC. If it is selected to be active, it will mimic an RBC by establishing communication with the SIEMENS/TI PLC.

The interface connector is a nine-position "D"-style connector located on the bottom of the unit. It is this connector which allows the unit to appear as an RBC to the PLC.

The pin assignments are as follows:

Pin#	Purpose
1	not used
2	not used
3	transceiver signal (-) *
4	not used
5	signal ground *
6	not used
7	not used
8	transceiver signal (+) *
9	not used

*connections that are required

Please refer to **PART F: REFERENCE** of this section for input/output table definitions.

SECTION 7 - PLC INTERFACING
PART B: 460 SIEMENS/TI INTERFACE

Setting up 460 SIEMENS/TI Communication

To configure your 460 online, **select** **SETUP PARAMETERS** from the **MAIN>COMMANDS** menu and then **select** option **5 - SIEMENS/TI PLC COMMUNICATION** and option **17 - FUNCTION KEYS/LED CONTROL**. If you opt to configure your 460 using offline programming, you access your menu starting from the **MAIN MENU**, then **select EDIT MESSAGE PROGRAM**, then **SETUP ATTRIBUTES**, and finally, **F5 - PLC**. An illustration of the offline menu follows this segment on the "online" prompts. The following explanations of the options, however, are universal for both programming methods.

Set up Parameters Online Programming Prompts Option 5

!**BASE NUMBER (0-15) <0>**
allows you to select the number of the remote base that will be connected to the 460. Choose a number between 1 and 15 and **hit** Enter, or just Enter to select the default value. Do not use base number zero.

!**ACTIVE RBC (Y/N) <N>**
You will specify whether the 460 unit will respond to the RBC or merely listen in on the communication with this selection. If the 460 is configured as a non-active base, it will not be able to send data to the PLC; it will only be monitoring an active base to read outputs. **Type** **Y** or **N** to choose and **press** Enter, or **press** Enter to select the default.

!**INPUT SLOT NUMBER (1-16) <1>** and **OUTPUT SLOT NUMBER 1-16) <1>**
These prompts will appear when the unit is configured for regular I/O modules. The slots specified will be configured for 16 words of input and 16 words of output. Therefore, you should use two different slot numbers.

!**NUMBER OF DATA SETS TO MAP TO PLC (0=NONE, 1-4) <0>** asks if any data sets would be mapped from the 460 to the PLC. If **0** is the response, no data sets will be mapped to the PLC.

Note: Mapping of data sets to the PLC is accomplished through the *PLC interface port* on the 460 unit.

!**STARTING DATA SET (1-4) <1>** is used to get the starting data set number when mapping the data sets to the PLC. If no data sets are to be mapped to the PLC, then this prompt will not appear.

SECTION 7 - PLC INTERFACING
PART B: 460 SIEMENS/TI INTERFACE

!NUMBER OF DATA SETS TO MAP TO PMD (0=NONE, 1-4) <0>
asks if any of the data sets should be mapped from the PLC outputs to the 460. If **0** is the response, then no data sets will be mapped to the 460. Otherwise, the specified number of data sets will be mapped to it.

!STARTING DATA SET NUMBER (1-4) <1>
asks for the starting data set number to be used when mapping data sets to the 460 from the PLC outputs. If no data sets are to be mapped to the 460, this prompt will not appear.

!INVERTED/NON-INVERTED (0=INVERTED 1=NON-INVERTED) <1>
This prompt lets you determine how the PLC port looks at the message/data lines: the way they come in as 1s or 0s (non-inverted)...**or** the complements of these (inverted).

!BCD/BINARY (0=BCD, 1=BINARY) <0>
This prompt lets you determine whether the 460 will consider the message/data lines as a Binary or Binary Coded Decimal.

!TIMED SCAN (1=ENABLED, 2=DISABLED) <2>
This prompt is for setting a specified scan time for how often the *PLC port* is scanned at the "system" scan rate (between 1 and 180 msec).

!QUEUE SIZE (0-20, 0=DISABLED) <0>
The queue is automatically disabled and it cannot be enabled from the PLC interface when the size is specified as **0**. If the size is specified as **0**, the unit will use the 32 entry FIFO if a nonzero message display time is specified. The queue is configured to the size specified by this prompt and then it must be enabled by the PLC interface before it will accept any add, delete, or clear functions.

!MESSAGE DISPLAY TIME (0-255 SEC.) <0>
This prompt determines if the messages will remain on the display until replaced, or if they will be displayed for a specified time. When the display time is set to zero, all messages will remain on the display until they are replaced. If a specified time is entered (in seconds ranging from 1 to 255), the messages will each display for that time period before being replaced by the next.

SECTION 7 - PLC INTERFACING
PART B: 460 SIEMENS/TI INTERFACE

Set up Parameters Online Programming Prompts Option 17

Select `SETUP PARAMETERS` from the **`MAIN>COMMANDS`** menu and then **select** option **`17 - FUNCTION KEYS/LED CONTROL`**. The terminal responds with these prompts:

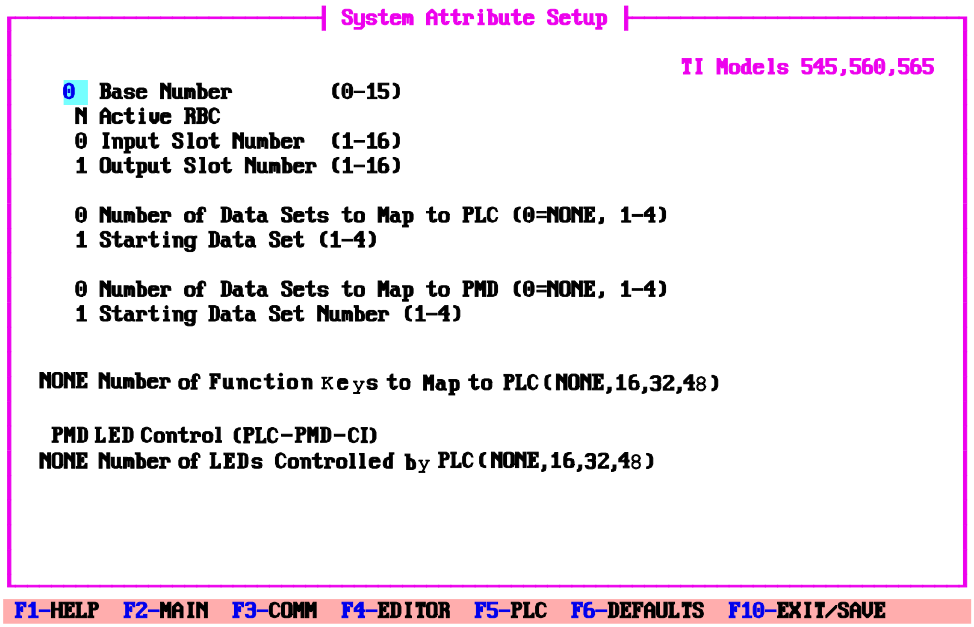
!**`LED CONTROL (SELECT 1=PMD, 2=CI, 3=PLC) <1>`**
lets you choose what you want to use as the input source for the LEDs. If **`1=PMD`** is selected, the LEDs will be controlled by the function keys. If **`2=CI`** (*computer interface*) is selected, the LEDs will be controlled by computer interface commands. If **`3=PLC`** is selected, the LEDs will be controlled by the PLC outputs. A word of outputs will control 16 LEDs on a bit map basis. Bit 0 of the word controls LED 1 and bit 15 of the word controls LED 16.

!**`NUMBER OF LEDS CONTROLLED BY PLC (0=NONE, 1=16, 2=32, 3=48) <0>-- (only if 3=PLC on the previous prompt)`** is used to specify how many LEDs will be controlled by the outputs of the PLC. The prompt will reflect how many LEDs are available on the unit. The LEDs are controlled by output words 3, 4, and 5 of the PLC output data being monitored by the 400 product. If the output word is not controlling LEDs, it will be ignored. Each output word can control 16 LEDs on a bit map basis.

!**`NUMBER OF FUNCTION KEYS TO MAP TO PLC (0=NONE, 1=16, 2=32, 3=48) <0>`** allows you to map the function keys to set inputs in the PLC. This prompt will also be dependent on the number of function keys that are available on the unit. Each set of 16 keys is bit mapped to one word of input. Function key 1 sets bit 0 and function key 16 sets bit 15. When the function key is pressed, the input bit is set, and when the function key is released, the bit is cleared. The function keys are mapped to input words 5, 6 and 7.

SECTION 7 - PLC INTERFACING
PART B: 460 SIEMENS/TI INTERFACE

Offline programming menu:



SECTION 7 - PLC INTERFACING
PART C: 470 MODICON INTERFACE

This section is intended to give you specific information about the 470 Modicon Interface product that makes it unique from the basic PMD 400. It is a supplement to, and a specified reference point within, the preceding manual sections.

The 470 is essentially a PMD 400 with an optional Modicon interface board. The interface board allows the unit to directly interface to a Modicon S908 RIO processor. The interfacing is accomplished by the 470 emulating a D908 module in the remote I/O communicating system. This allows the unit to communicate directly with the "supervisor" over the high-speed communication link. The 470's female BNC coaxial connector is located on the bottom of the unit.

The unit is able to set inputs based on keypad and data set information. The 470 monitors PLC outputs for message control and data set information. These capabilities can be enabled or disabled when the unit is programmed.

The maximum allowable attenuation between the controller and PMD 470 drop is 30db (which is 5db less than the Modicon remote I/O specification). You should calculate the expected loss based on the length and type of trunk cable, the number of drops in the system, and whether or not a splitter has been used. The expected loss should not exceed 30db to ensure good communication. (Reference Modicon manual #GM-0984-RIO).

The 470 is set up to exchange 16 words of input and 16 words of output with the Modicon S908 module. The Modicon PLC must be set up to use these inputs and outputs and the drop specified when programming the 470. Sixteen words of input and 16 words of output will be exchanged with the remote I/O Master.

If no data sets are mapped to the PLC, the four input words will still be exchanged but will have no meaning. (See the input/output tables in **PART F: REFERENCE** of this section).

The Modicon PLC can be configured using a P190 panel, MODSOFT panel software, or IBM P190 emulation software. The 470 is a D908 drop and the PLC must be configured for one block of 16 words of input and one block of 16 words of output at the drop specified on the 470.

SECTION 7 - PLC INTERFACING
PART C: 470 MODICON INTERFACE

The following is a description of configuring the PLC using MODSOFT PLC programming software package: The PLC is set up using the Config menu under the Offline option. When entering this menu, you must define a PLC type if one is not already defined. The PLC type can be changed using the PLC Type option in the OvrView menu (an option under the Config menu). The OvrView option is also used to configure the I/O ranges and modules.

The I/O configuration must be defined in the OvrView menu. Use the Ranges option to define enough I/O to send 16 words of outputs to a D908 drop and that needed for the ladder program and other I/O modules. You will also need to define enough I/O for 16 words of input. Use the I/O option to set the number of segments, drops, and modules needed. The number of segments must be greater than or equal to the highest drop number being used. The number of drops must be greater than or equal to the highest drop number to be used.

The number of modules must be greater than or equal to the number of modules to be used. It takes one module for each line used to configure the D908 plus some overhead. Making this larger than needed will use more memory so make sure there is enough available for use. The value for both these Ranges and I/O affect how much memory is used.

After defining the I/O, the drop must be defined. This is done from the Tcop option in the Config menu. Use the GetDrop option in the Tcop menu to configure the drop that will be used to communicate with the 470. The drop must be defined as a D908. The D908 definition is under the Special option in the menu. If this drop was previously undefined, then use the Special option to define it as a D908. The Special option can be accessed by pressing the escape key to return to the menu. If the drop has already been defined as something other than a D908, it can be deleted using the DelDrop option first. Once it has been deleted, it can be configured as a D908.

After defining the drop as a D908, there are eight lines (or slots) for defining the outputs and inputs for the drop. The outputs and inputs for the drop must be defined as two 16 word blocks. The outputs must be defined as a block of 16 words at 4xxxx. The inputs must be defined as a block of 16 words at 3xxxx. The first word of output sent must correspond to the first word in the output data definition for the 470. The first word of input received will be the first word of the input data sent by the 470.

SECTION 7 - PLC INTERFACING
PART C: 470 MODICON INTERFACING

SAMPLE TCOP SCREEN FROM MODSOFT

DISTRIBUTED CONTROL PROCESSOR

Drop : 3 of 3 Rack 1
Drop Hold Up Time : 3 (x 100ms)

Slot	Block Length	Master References	Slave References	Data Group Health Bit
101				
102				30001-OPTION STATE
103				30002-WRITE COUNT
104	16	40001 - 40016	30003 - 30018	1, 4
105	16	30001 - 30016	40001 - 40016	1, 5
106				
108				
109				
110				
111				

Using the configuration shown above, the 470 is configured as drop 3. The 16

input words, starting at 30001, will be controlled by the 470. The 16 output words, starting at 40001, will be sent to the 470 and interpreted according to the output data definition.

Controlling Variable Data

Modicon Interface units contain four 16-bit words of variable data that can be used within a message. This allows a variable data field within message text. These variable data sets can be controlled by the PLC outputs. When the 470 has been programmed to map data sets to the 470 product, the appropriate PLC outputs will be directly mapped into the variable data according to the PLC output data definition. The data will be mapped as shown below:

OUTPUT WORD 6 MAPS TO DATA SET 1
OUTPUT WORD 7 MAPS TO DATA SET 2
OUTPUT WORD 8 MAPS TO DATA SET 3
OUTPUT WORD 9 MAPS TO DATA SET 4

SECTION 7 - PLC INTERFACING
PART C: 470 MODICON INTERFACING

Control PLC Inputs With PMD Variable Data

The Modicon Interface units can map the values contained within its variable data sets directly to the input words in the PLC input table. The 470 must be programmed to map data sets to the PLC. The data will be mapped as shown below:

DATA SET 1 MAPS TO INPUT WORD 1
DATA SET 2 MAPS TO INPUT WORD 2
DATA SET 3 MAPS TO INPUT WORD 3
DATA SET 4 MAPS TO INPUT WORD 4

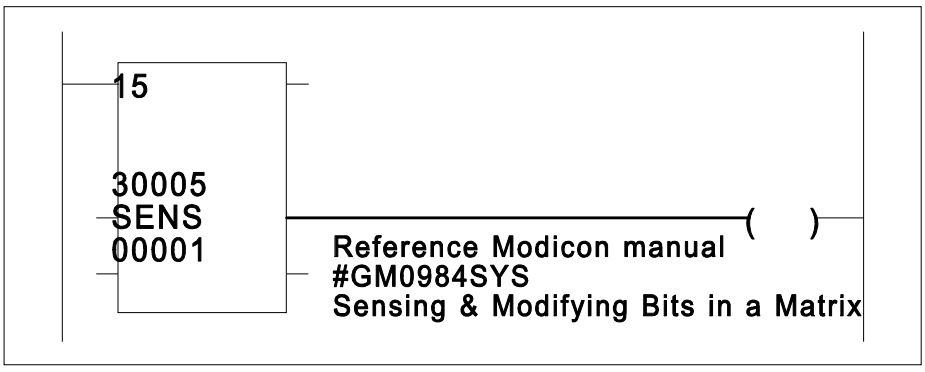
Using Function Keys to Set PLC Inputs

Function keys can be used to set bits in the input words sent to the PLC. Each function key sets one bit. The function keys must be programmed to be mapped to the PLC. When a function key is pressed, the bit will be set and when the function key is released, the bit will be cleared. The bits are set/cleared according to the INPUT data map. (Function Keys/LED programming is explained toward the end of part **C**).

Example: Using that traffic cop screen example, input register 30005 holds the function key input bits. If Function Key #2 is pressed, input register 30005 would have the following values.

Modicon LSB Modicon MSB
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0

Note: When using the SENS function to monitor bits, the bit numbering will be reversed. Function Key 1 will appear as bit 16 and Function Key 16 will appear as bit 1.



SECTION 7 - PLC INTERFACING
PART C: 470 MODICON INTERFACING

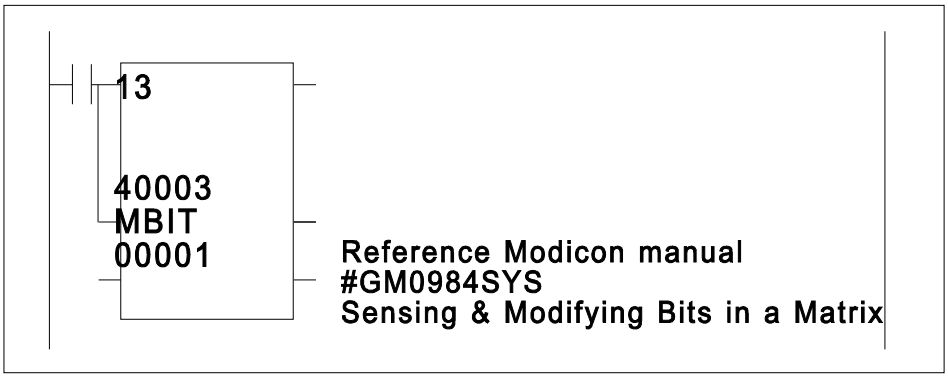
LEDS

The LEDs above each function key can be controlled by the PLC output data. Each 16 bit word controls 16 LEDs on a bit basis. When the bit is set, the LED is on and when the bit is clear, the LED is off. The 470 must be programmed to have the LEDs controlled by the PLC. The LEDs are set based on the OUTPUT data map.

Example: Using the traffic screen example again, output register 40003 would be for the first 16 LEDs. To turn on the LED above Function Key #4, the following values would need to be sent from the PLC.

Modicon LSB															Modicon MSB
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Note: When using the Modicon MBIT function to control status LEDs, the bit numbering will be reversed. LED 1 will be controlled by bit 16 and LED 16 will be controlled by bit 1.



Setting up 470 Modicon Communication

To configure your 470 online, **select** **SETUP PARAMETERS** from the **MAIN>COMMANDS** menu and then **select** option **5 - MODICON PLC COMMUNICATION** and option **17 - FUNCTION KEYS/LED CONTROL**. If you opt to configure your 470 using offline programming, access the menu starting from the **MAIN MENU**, then select **EDIT MESSAGE PROGRAM**,

then **SETUP ATTRIBUTES**, and finally, **F5 - PLC**. An illustration of the offline menu follows this segment on the "online" prompts. The explanations of the options, however, are universal for both programming methods.

Set up Parameters Online Programming Prompts Option 5

!**DROP NUMBER (1-32) <1>**
specifies where the 470 will be addressed during communication with the MODICON PLC.

!NUMBER OF DATA SETS TO MAP TO PLC (0=NONE, 1-4) <0>
specifies how many data sets will be mapped to the inputs of the PLC. A zero indicates no data sets will be mapped and a number from one to four gives the number of data sets mapped to the PLC.

!STARTING DATA SET (1-4) <1>
is the number used when mapping data sets to the PLC. Only the data sets specified by the **NUMBER OF DATA SETS TO MAP TO PLC** and the **STARTING DATA SET** will be mapped. For example, if the number of data sets to map is 2 and the starting data set is 2, then only data sets 2 and 3 will be mapped to the PLC.

!NUMBER OF DATA SETS TO MAP TO PMD (0=NONE, 1-4) <0>
specifies how many data sets will be mapped from the outputs of the PLC directly to the variable data of the 470. A zero indicates no data sets will be mapped and a number from one to four gives the number of data sets mapped from the PLC to the 470.

!STARTING DATA SET NUMBER (1-4) <1>
is the number used when mapping data sets from the PLC to the 470. Only the data sets specified by the **NUMBER OF DATA SETS TO MAP TO PMD** and the **STARTING DATA SET** will be mapped. For example, if the number of data sets to map is 3 and the starting data set is 2, then data sets 2, 3, and 4 will be mapped to the 470.

!INVERTED/NON-INVERTED (0=INVERTED 1=NON-INVERTED) <1> This prompt lets you determine how the PLC port looks at the message/data lines: the way they come in as 1s or 0s (non-inverted)...**or** the complements of these (inverted).

!BCD/BINARY (0=BCD, 1=BINARY) <0>
This prompt lets you determine whether the 470 will consider the message/data lines as a Binary or Binary Coded Decimal.

!TIMED SCAN (1=ENABLED, 2=DISABLED) <2>
This prompt is for setting a specified scan time for how often the *PLC port* is scanned at the "system" scan rate (between 1 and 180 msec).

!QUEUE SIZE (0-20, 0=DISABLED) <0>
The queue is automatically disabled and it cannot be enabled from the PLC interface when the size is specified as **0**.

If the size is specified as **0**, the unit will use the 32 entry FIFO if a nonzero message display time is specified. The queue is configured to the size specified by this prompt and then it must be enabled by the PLC interface before it will accept any add, delete, or clear functions.

!MESSAGE DISPLAY TIME (0-255 SEC.) <0>
This prompt determines if the messages will remain on the display until replaced, or if they will be displayed for a specified time.

Set up Parameters Online Programming Prompts Option 17

Select SETUP PARAMETERS from the **MAIN>COMMANDS** menu and then **select** option **17 - FUNCTION KEYS/LED CONTROL**. The terminal responds with these prompts:

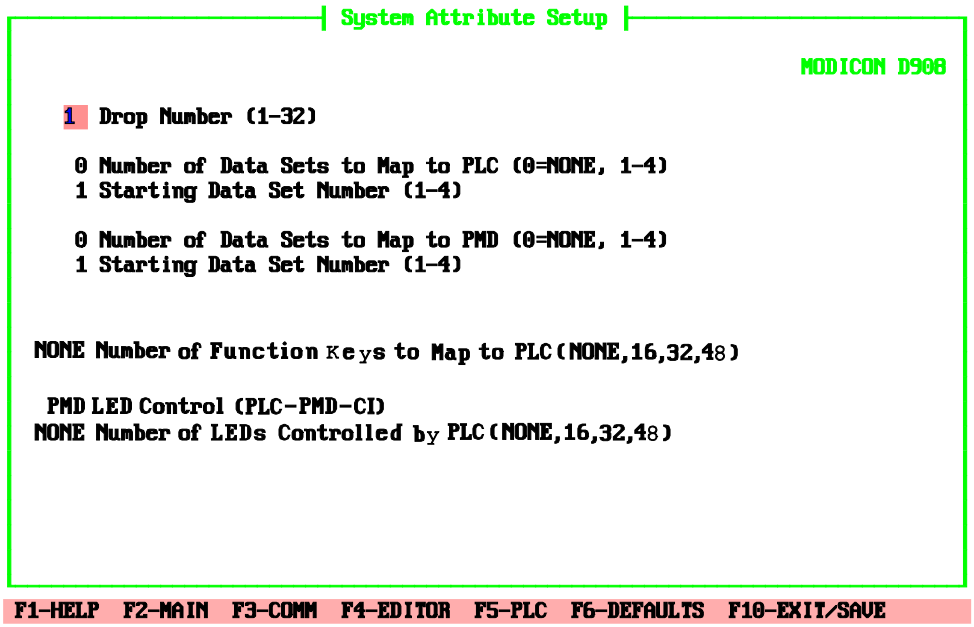
!LED CONTROL (SELECT 1=PMD, 2=CI, 3=PLC) <1>
lets you choose what you want to use as the input source for the LEDs. If **1=PMD** is selected, the LEDs will be controlled by the function keys. If **2=CI** (*computer interface*) is selected, the LEDs will be controlled by computer interface commands. If **3=PLC** is selected, the LEDs will be controlled by the PLC outputs. A word of outputs will control 16 LEDs on a bit map basis. Bit 0 of the word controls LED 1 and bit 15 of the word controls LED 16.

!NUMBER OF LEDS CONTROLLED BY PLC (0=NONE, 1=16, 2=32, 3=48) <0>-- (only if 3=PLC on the previous prompt)
is used to specify how many LEDs will be controlled by the outputs of the PLC. The prompt will reflect how many LEDs are available on the unit. The LEDs are controlled by output words 3, 4, and 5 of the PLC output data being monitored by the 400 product. If the output word is not controlling LEDs, it will be ignored. Each output word can control 16 LEDs on a bit map basis.

!NUMBER OF FUNCTION KEYS TO MAP TO PLC (0=NONE, 1=16, 2=32, 3=48) <0>
allows you to map the function keys to set inputs in the PLC. This prompt will also be dependent on the number of function keys that are available on the unit. Each set of 16 keys is bit mapped to one word of input. Function key 1 sets bit 0 and function key 16 sets bit 15. When the function key is pressed, the input bit is set, and when the function key is released, the bit is cleared. The function keys are mapped to input words 5, 6 and 7.

SECTION 7 - PLC INTERFACING
PART C: 470 MODICON INTERFACING

Offline programming menu:



SECTION 7 - PLC INTERFACING
PART D: 475 MODICON MODBUS PLUS INTERFACE

This section is intended to give you specific information about the 475 Modicon MODBUS PLUS product that makes it unique from the basic PMD 400. It is a supplement to, and a specified reference point within, the preceding manual sections.

The 475 is a PMD 400 that interfaces to Modicon MODBUS PLUS local area network via a nine-position, "D"-style connector on the bottom of the unit.

Setting the MODBUS PLUS Address

The MODBUS PLUS' node address can be set using the first six positions of the dip switches on the bottom of the unit. For the setting of the switches, when the switch is in the ON position, it is at a logic zero level. Likewise, the OFF position is at a logic one level. Position 1 is the most significant bit of the number and the actual node address value will be one more than the selected value on the switches. For example, if the switches are set to a value of 5, then the node address will be 6. Valid node addresses are 1-64. Please refer to the table.

SWITCH #						Switch Settings
6	5	4	3	2	1	NODE ADDRESS
0	0	0	0	0	0	= 1 (0+1)
0	0	0	0	0	1	= 2 (1+1)
0	0	0	0	1	0	= 3 (2+1)
0	0	0	0	1	1	= 4 (3+1)
!	!	!	!	!	!	! !
1	1	1	1	1	1	= 64 (63+1)
MSD					LSD	

The 475 is a node on the network with its own network address. The MODBUS PLUS interface board will have a board identification code of 5. This is the code that will be returned in the Computer Interface Identify PMD message. This board will have a revision code of 1.

PLC Interface Test

The PLC interface test that can be entered using the push buttons on the front of the unit will display a decimal representation of the unit's current MODBUS PLUS node address. It may take up to five seconds for the node address to be displayed and it will only be displayed for five seconds before entering the PLC interface test.

SECTION 7 - PLC INTERFACING
PART D: 475 MODICON MODBUS PLUS INTERFACE

Two types of connectors are available from Modicon for connecting devices to the network. Each inline drop requires a line connector, Modicon part number AS-MBKT-085. This part number contains one connector. The drops at the two ends of the cable each require a terminating connector, Modicon part number AS-MBKT-185. This part number contains two connectors.

The PMD 475 MODBUS PLUS PLC interface functions in four modes. These modes include polling a specific PLC, receiving MODBUS commands to read data from and write data to the PMD, monitoring and sending global data, and ASCII. These modes can be used in any combination but it is very important to understand that the PMD will use the last new data that it receives. It is up to you to control the order in which the data is sent to the PMD. This is vital in making sure that all data is displayed when using more than one method to send data to the PMD.

First Mode: Polling a Specific PLC

The first mode polls a specific range of holding registers in a specific PLC and sets a specific range of registers in the same PLC. In this mode, the PMD constantly reads the range of registers and monitors the values for a change. This mode causes the PMD to increase the traffic on the MODBUS PLUS network.

The 475 is set up to exchange 16 words of input and 16 words of output with the MODBUS PLUS destination node. The Modicon PLC must be set up to use these inputs and outputs at the node specified when programming the unit. Note that there will always be 16 input words and 16 output words exchanged. If the unit is not programmed to map data to the inputs or use the PLC outputs, the data will still be exchanged. For example, if no data sets are being mapped to the PLC, the four input words used for this would still be exchanged but without meaning.

Second Mode: Receiving MODBUS Commands to Read/Write Data to PMD

The second mode puts the PMD in the position of being a slave device on the MODBUS PLUS network. In this mode, the PMD will only reply to commands that it receives. The PMD will not initiate any transactions on the MODBUS PLUS network. Any node can send data to the PMD using the MODBUS command to set holding registers or the MSTR function to write data. The input table from the PMD can be read by any node with the MODBUS command to read holding registers or a MSTR function to read data.

SECTION 7 - PLC INTERFACING
PART D: 475 MODICON MODBUS PLUS INTERFACE

Master Write Data - Data can be sent to the PMD asynchronously using the MODBUS command "Preset Multiple Registers." This is the same command as the write data command of the MSTR function. The data will follow the format of the current PMD output table described at the end of this section in **PART F: REFERENCE**. The command sent to the PMD must specify that the data be written to register 40002. The data will be monitored for changes from the current values in the PMD. Each time new data is received by the PMD it will be processed according to the control word.

An example of an MSTR block to send data to a PMD 475 follows. Note that the starting 4xxxx register must be set to 3 to specify to the PMD to write to register 40002.

Control Block

register 0	1=write function
register 1	0=MODBUS PLUS error status
register 2	n=number of registers that will be sent to the PMD
register 3	3=starting 4xxxx register in the slave to write to (must be 3)
register 4	routing path byte 1
register 5	routing path byte 2
register 6	routing path byte 3
register 7	routing path byte 4
register 8	routing path byte 5

Data Area

register 0	The data area should contain the data that will be sent to the PMD.
register n	Last register of data that will be sent to the PMD

This example shows how to send 16 registers of data (32 bytes) to a PMD that is configured as node 4. The MSTR block will have its control block at 40010 and its data area will start at 40020. The registers should be set as follows to display the message 1.

SECTION 7 - PLC INTERFACING
PART D: 475 MODICON MODBUS PLUS INTERFACE

CONTROL BLOCK

REG		
40010	1	MSTR write function
40011	0	MODBUS PLUS status
40012	16	length of data area to send
40013	3	starting 4xxxx register in the slave to write to (must be 3)
40014	4	1st routing byte (node address)
40015	1	2nd routing address (data slave transaction path #)
40016	0	3rd routing byte
40017	0	4th routing byte
40018	0	5th routing byte
40019	0	

DATA AREA

REG		
40020	1	Message number
40021	9	Control word
40022	0	LEDs 1-16
40023	0	LEDs 17-32
40024	0	LEDs 33-48
40025	111	Data Set 1
40026	222	Data Set 2
40027	0	Data Set 3
40028	0	Data Set 4
40029	0	Queue delete message number
40030	0	Queue message display time
40031	0	not used
40032	0	not used
40033	0	not used
40034	0	not used
40035	0	not used

MSTR Read Data - The input table of the PLC interface can be read by any node on the network by reading address 40002 from the PMD. This can be done using the MODBUS command "Read Holding Registers." This is the same as the read data command of the MSTR function. When this command is received, the PMD will return a copy of its current input table as described at the end of this section in **PART F: REFERENCE**. An example of an MSTR block to read the input table of a 475 follows. Note that the starting 4xxxx register must be set to three to specify to the PMD to read register 40002.

SECTION 7 - PLC INTERFACING
PART D: 475 MODICON MODBUS PLUS INTERFACE

Control Block

register 0	2=read function
register 1	0=MODBUS PLUS error status
register 2	n=number of registers that will be read from the PMD
register 3	3=starting 4xxxx register in the slave to read from (must be 3)
register 4	routing path byte 1
register 5	routing path byte 2
register 6	routing path byte 3
register 7	routing path byte 4
register 8	routing path byte 5

Data Area

register 0	The data area contains the data received from the PMD.
register n	Last register of data that was received from the PMD.

This example shows how to read 16 registers of data (32 bytes) from a PMD that is configured as node 4. The MSTR block will have its control block at 40010 and its data area will start at 40020.

CONTROL BLOCK

REG		
40010	2	MSTR read function
40011	0	MODBUS PLUS status
40012	16	length of data are to put reply
40013	3	starting 4xxxx register in the slave to read from (must be 3)
40014	4	1st routing byte (Node address)
40015	1	2nd routing address (data slave transaction path #)
40016	0	3rd routing byte
40017	0	4th routing byte
40018	0	5th routing byte
40019	0	

DATA AREA

REG		
40020	0	Data set 1
40021	0	Data set 2
40022	0	Data set 3
40023	0	Data set 4

40024	0	Function keys 1-16
40025	0	Function keys 17-32
40026	0	Function keys 33-48
40027	0	Data set status
40028	0	Status word
40029	0	not used
40030	0	not used
40031	0	not used
40032	0	not used
40033	0	not used
40034	0	not used
40035	0	not used

Third Mode: Global Data

The third mode uses the global data that is passed with the token on the MODBUS PLUS network. The PMD will constantly monitor the global data from every node on the network. Whenever a node sends global data, the PMD will monitor the data for changes from the last global data received from that node. The global data will only be used if it is different from the last global data from that specific node.

The PMD can also send its input table out as global data. It should be noted that if there are multiple nodes sending global data that the performance of the PMD will degrade because the PMD must examine the global data from each node sending global data on each scan of the PMD.

The PMD can be configured to monitor any 16 contiguous words of global data. This data will be used as the output table within the PMD. The PMD can also be configured to send its input data to the 16 contiguous words in the global data table. The PMD will monitor all global data that it receives so care must be taken when selecting which words of the global data to monitor. If the PLC system is already using global data, it may be better to send the data to the PMD using the MODBUS commands.

Note: A large number of nodes passing global data will degrade the performance of the PMD as it monitors data from a large number of nodes.

The PMD can be configured to process global data from the MODBUS PLUS network. The data must follow the format of the output table at the end of this section in **PART F: REFERENCE**. Each time global data is received, the PMD will compare it to the last data received from that node. If the data is the same as the last global data from that node, it will not be used. If the data has changed, it will be used as the current data from the PLC.

It will override data from other sources including MODBUS Commands and data that the PMD read from the PLC. The input table maintained by the PMD will be sent out as global data each time it changes. Please refer to the input table at the end of this section in **PART F: REFERENCE**.

Fourth Mode: ASCII Protocol

The fourth mode supports a MODBUS PLUS node sending ASCII data to the PMD and receiving replies to the ASCII commands that it sends. Any MODBUS PLUS node can send data to the PMD 475 using the PMD 400 ASCII protocol. The PMD follows the MODBUS PLUS protocol and uses MODBUS commands to accept data from a master and send replies. In these commands, the PMD will act like a MODBUS slave to the MODBUS PLUS master.

The data is sent to the PMD using the MODBUS Preset Multiple Registers (Function 10 Hex) command. Replies to the ASCII commands to read data from the PMD can be read back using the Read Holding Registers (Function 03) command. When the PMD is configured to accept ASCII data from the MODBUS PLUS interface, it will not process any ASCII data from the computer interface. The computer interface will only process PMD commands. The PMD will not process ASCII data from both the computer interface and the MODBUS PLUS interface. Therefore, if the PMD is configured to process ASCII data from both the computer interface and the MODBUS PLUS interface, it will only process the ASCII data from the computer interface.

As introduced previously, the ASCII data is sent to the PMD using a MODBUS command to Preset Multiple Registers (function 10 HEX). This command allows you to send up to 100 registers (200 bytes) at one time. In this command, you must specify the first register number, the number of registers to set, and the number of bytes in the data buffer. The other values will be passed back to the master MODBUS PLUS node from the PMD in the MODBUS response. The data sent to the PMD is treated as an ASCII string and processed one character at a time. ASCII commands sent to the PMD can be spread over more than one MODBUS command. Any data bytes that are set to 0 will not be processed. The data received by the PMD will be processed according to the ASCII computer interface protocol.

Any ASCII commands that will issue a reply must be followed with a Read Holding Registers command to read the reply to the command. The PMD will not send the reply to the ASCII command until the MODBUS Master issues a read command to the PMD. Then the response to the last ASCII command received by the PMD will be returned in the reply to the read holding registers command.

The Read Holding registers command must specify to read register 40001 and how many registers to read. The longest possible reply from the PMD is currently 87 bytes long (read function keys as an ordered list). The data in the reply will not include a leading carriage return and line feed.

If the read command does not request enough bytes to include the complete reply, the portion of the reply that is not returned will be lost. The first character of the reply will always be the '^'. The reply buffer will be padded with 0s to fill the buffer up to the length specified in the MODBUS read command.

The capability to receive and send ASCII data is in addition to the current MODBUS PLUS functionality that reads and writes inputs and outputs. The PMD can monitor outputs and set input registers at the same time that it receives and sends ASCII data.

Some exceptions must be noted. The replies from the PMD will not be preceded by a carriage return and line feed. Commands that are not applicable to a unit will return a reply with a length of 0. Data sent to the PMD will be processed based on the length of the data received in the MODBUS command. Any bytes that are 0 will not be processed. Therefore, if the MODBUS command sending the data indicates that it contains 50 bytes of data and only the first ten bytes are ASCII data/commands for the PMD, the remaining 40 bytes should be set to 0. All replies to ASCII commands will be padded with 0s to fill the buffer requested by the read command. If the MODBUS command to read a reply from the PMD requests 50 bytes and there are only 20 bytes in the ASCII reply, the remaining 30 bytes will be set to 0 in the reply data.

The ASCII commands to set the LEDs on a 475 with a 400E expander module or a 475 will be able to control the LEDs regardless of which LED control mode is selected. Care must be taken when controlling the LEDs from the MODBUS PLUS ASCII mode so that one mode of control does not overwrite another. If the PMD is controlling the LEDs, the MODBUS PLUS ASCII mode can potentially overwrite the LED status set by the PMD.

Sending/Reading ASCII Data Using the MSTR Function

ASCII data can be sent to the PMD from a MODBUS PLUS PLC using the MODBUS PLUS Master function (MSTR). The MSTR function allows the PLC to initiate transactions with other MODBUS PLUS nodes. The PMD will accept and respond to the write data and the read functions. All other MSTR functions will be ignored and the PMD will respond with the command rejected error.

SECTION 7 - PLC INTERFACING
PART D: 475 MODICON MODBUS PLUS INTERFACE

Three things must be programmed into the MSTR block: the address of the control block, the address of the data area, and the maximum size of the data area. The MSTR block should be programmed as shown below. Note that the starting 4xxxx register is specified as 2. This will cause the MSTR function to write the data to register 40001 as required by the PMD.

Write Function

Control Block

register 0	1=write function
register 1	0=MODBUS PLUS error status
register 2	n=number of registers that will be sent to the PMD
register 3	2=starting 4xxxx register in the slave to write to must be 2
register 4	routing path byte 1
register 5	routing path byte 2
register 6	routing path byte 3
register 7	routing path byte 4
register 8	routing path byte 5

Data Area

register 0	The data area should contain the ASCII string of commands and data that will be sent to the PMD. Any bytes in the data area that should be ignored by the PMD must be set to 0.
register n	Last register of ASCII data to be sent to the PMD.

This example shows how to send 26 registers of ASCII data (50 bytes) to a PMD that is configured as node 4 with a group and unit of 1 and 7 respectively. The MSTR block has its control block at 40010 and its data area starts at 40020. The registers should be set as shown below to display the message shown below.

TESTING ASCII DATA
ON THE PMD 475

CONTROL BLOCK

40010	1	write function
40011	0	MODBUS PLUS error status
40012	26	number of registers to be sent
40013	2	starting 4xxxx register in the slave to write to

40014	4	destination node address
40015	1	data slave transaction path
40016	0	routing byte set to 0
40017	0	routing byte set to 0
40018	0	routing byte set to 0
40019	0	

DATA AREA

REG	ASCII	HEX
40020	^A	5E41
40021	01	3031
40022	00	3030
40023	07	3037
40024	^E	5E45
40025	0T	3054
40026	ES	4553
40027	TI	5449
40028	NG	4E47
40029	A	2041
40030	SC	5343
40031	II	4949
40032	D	2044
40033	AT	4154
40034	A<cr>	410D
40035	ON	4F4E
40036	T	2054
40037	HE	4845
40038	P	2050
40039	MD	4D44
40040	47	3437
40041	5	4500
40042		0000
40043		0000
40044		0000
40045		0000

Read Function **Control Block**

register 0	2=read function
register 1	0=MODBUS PLUS error status
register 2	n=number of registers that will be read from the PMD

register 3	2=starting 4xxxx register in the PMD to read from must be 2
register 4	routing path byte 1
register 5	routing path byte 2
register 6	routing path byte 3
register 7	routing path byte 4
register 8	routing path byte 5

Data Area

register 0	The data will contain the ASCII string that is the response to the last ASCII command that requires a response from the PMD. If there is no pending response in the PMD, the length in the command returned from the PMD will be 0. When the PMD does have a reply to return, it will pad the buffer to the length requested by the read function with 0s.
register n	Last register of ASCII data that was read from the PMD

This example shows how to read 26 registers of ASCII data (50 bytes) from a PMD that is configured as node 4 with a group and unit of 1 and 7 respectively. The MSTR block will have its control block at 40050 and its data area will start at 40060.

The data read by this command will be the reply to a read command that was sent previously. For this example, we will assume that a read command of ^R11 has already been sent to the PMD.

The registers should be set as shown below.

CONTROL BLOCK

40050	2	read function
40051	0	MODBUS PLUS error status
40052	26	number of registers to be sent
40053	2	starting 4xxxx register in the slave to read from
40054	4	destination node address
40055	1	data transaction path
40056	0	routing byte set to 0
40057	0	routing byte set to 0
40058	0	routing byte set to 0

DATA AREA

REG	ASCII	HEX
40060	^R	5E52
40061	FK	464B
40062	01	3031
40063	,F	2C46
40064	K0	4B30
40065	5,	352C
40066	FK	464B
40067	14	3134
40068	,F	2C46
40069	K4	4B34
40070	2^	325E
40071	Z	5A00
40072		0000
40073		0000
40074		0000
40075		0000
40076		0000
40077		0000
40078		0000
40079		0000
40080		0000
40081		0000
40082		0000
40083		0000
40084		0000
40085		0000

Sending ASCII

Write function - Data is sent to the PMD using the Preset Multiple Registers Function (10 hex). This command allows up to 100 registers (200 bytes) of ASCII data to be sent at a time.

Command Data		Response Data
byte 0	0x10 = function code	0x10 = function code
byte 1	1st register # high byte	1st register # high byte
byte 2	1st register # low byte	1st register # low byte
byte 3	# of registers to be set high byte	# of registers set high byte
byte 4	# of registers to be set low byte	# of registers set low byte
byte 5	# of bytes in the data buffer	
byte 6	1st byte of data buffer	

Read function - Data is read from the PMD using the Read Holding Registers Command (function 03). This command can read up to 125 4xxxx registers (250 bytes) at one time. The longest possible response length from the PMD 475 is 87 bytes using the read function keys with the ordered list option.

	Command Data	Response Data
byte 0	0x03 = function code	0x03 = function code
byte 1	1st 4xxxx register to read # high byte	# of data bytes read
byte 2	1st 4xxxx register to read # low byte	1st register high byte
byte 3	set to 00	1st register low byte
byte 4	# of registers to read	2nd register high byte
byte 5	# of bytes in the data buffer	2nd register low byte
byte 6	1st byte of data buffer	

Setting up 475 Modicon MODBUS PLUS COMMUNICATION

!To configure your 475 online, **select** **SETUP PARAMETERS** from the **MAIN>COMMANDS** menu and then **select** option **5 - MODBUS PLUS PLC COMMUNICATION** and option **17 - FUNCTION KEYS/LED CONTROL**. If you opt to configure your 475 using offline programming, you access your menu starting from the **MAIN MENU**, then **select** **EDIT MESSAGE PROGRAM**, then **SETUP ATTRIBUTES**, and finally, **F5 - PLC**. An illustration of the offline menu follows this segment on the "online" prompts. The following explanations of the options, however, are universal for both programming methods. The interface requires the user to enter the network routing path and the starting input and output registers.

Set up Parameters Online Programming Prompts Option 5

!**MODBUS PLUS (0=BOTH 1=READ 2=SET INPUTS 3=NEITHER)**
This prompt configures the polling mode of the PMD 475. The unit can be configured to continually read a specified set of registers from a specific node and/or send the input table to a specified set of registers in a specific node. If the unit is reading a set of registers, it will continuously issue read commands to that node to monitor the data. If the unit is sending its input table, it will only issue a write command when the data changes. When neither is selected, the unit will not read outputs from a PLC or send its input table to the specified node.

If you respond **0=BOTH**, it indicates the normal mode of reading 16 output words and writing 16 input words. A **1=READ OUTPUTS** response indicates that the unit should only read the 16 output words. A unit configured to only read outputs will not attempt to set any input words. A **2=SET INPUTS** indicates that the unit should only set the 16 input words.

A unit configured to only set the input words will not attempt to read the output words. A **3=NEITHER** indicates that the unit should not attempt to read outputs or write inputs.

Note: Even though the reading outputs and writing inputs may have been disabled, the unit will maintain the characteristics of a PMD 475 i.e. It will not clear data sets each time a new message is triggered.

!PROCESS SLAVE COMMANDS (Y/N) <N>
This prompt defines whether the PMD 475 will accept or reject MODBUS commands to read the PLC input table or write to the PLC output table. When the unit is configured to process slave commands, any node on the network can write to the PLC output table in the PMD.

This is accomplished using the MSTR write data function or the MODBUS command to set holding registers. The data must be written to register 40002 to be processed as the PLC output table. The unit will also process the MSTR read data function by returning a copy of the PLC input table maintained by the PMD. The command must read from register 40002 to read the PLC input table.

When the **PROCESS SLAVE COMMANDS** prompt is answered **Y**, the PMD 475 will be configured to accept slave commands from any node on the network. If the PMD is not configured to process slave commands, a reply indicating the command was rejected will be returned. The PMD can process two slave commands, the MSTR Write command (Preset Multiple Registers) and the MSTR Read command (Read holding registers). The MSTR write function is used to send a table of 16 words to the PMD. The MSTR read function is used to read the 16 word input table from the PMD.

!SEND GLOBAL DATA (Y/N) <N>
This prompt enables the PMD to send the PLC input table to the Global data for the PMD 475 node address. Each time the PLC input table changes new global data will be sent.

!INPUT TABLE OFFSET IN GLOBAL DATA (0-16) <0>
An offset can be programmed that will allow the unit to place the PLC input table into the global data at any location. The PLC input table will take up 16 words of global data. If an offset of six is specified, word 6 of the global data will be word 1 of the PLC input table.

SECTION 7 - PLC INTERFACING
PART D: 475 MODICON MODBUS PLUS INTERFACE

!RECEIVE GLOBAL DATA (Y/N) <N>
This prompt enables monitoring global data. The unit will constantly monitor new global data from the MODBUS PLUS network. If the unit is not configured to receive global data, it will ignore all global data.

!OUTPUT TABLE OFFSET IN GLOBAL DATA (0-16) <0>
When the unit is configured to monitor global data, an offset into the global data can be programmed to allow the unit to monitor any 16 contiguous words of global data. If the offset is set to 5, the fifth word in the global data will be the first word in the PLC output table.

!MODBUS PLUS ADDRESS BYTES
specify a list of router nodes and a final destination node. The destination node contains the holding registers that will contain the data to be used by the 475. The routing path can contain up to four router nodes and a final destination node. The first node after the destination is specified as a zero to signal the end of the routing path.

If there is an error in the routing path specified by you, then a **ROUTING FAILED ERROR** message will be displayed after the unit enters DISPLAY MODE. Refer to the MODICON documentation for more information on the network routing path.

!STARTING INPUT REGISTER <1>
allows the user to specify the first 4xxxx register to be used by the interface when sending input information to the PLC. The first 4xxxx register is 40001 and is specified as 1 to the unit.

The valid range for the first input holding register is 1-65535. i.e. A value of 1 for the first input holding register will specify register 40001 and a value of 400 will specify register 40400. The 475 will use 16 input registers starting with the first input holding register. If the first register is specified as 40001, then registers 40001-40016 will be used as input registers by the 475.

!STARTING OUTPUT REGISTER <1>
specifies which 4xxxx holding register will be used by the unit when reading data from the PLC. The first 4xxxx register is 40001 and is specified as 1 to the 475 unit. The valid range for the first output holding register is 1-65535.

The 475 will read 16 registers starting with the first output holding register and then will read 16 registers starting at 40017. This means the 475 will read registers 40017-40032.

!**NUMBER OF DATA SETS TO MAP TO PLC (0=NONE, 1-4) <0>**
specifies how many data sets will be mapped to the inputs of the PLC. A 0 indicates no data sets will be mapped, and a number from 1 to 4 gives the number of data sets mapped to the PLC.

!**STARTING DATA SET NUMBER (1-4) <1>**
is the starting data set used when mapping data sets to the PLC. Only the data sets specified by the **NUMBER OF DATA SETS TO MAP TO PLC** and the **STARTING DATA SET NUMBER** will be mapped. For example, if the number of data sets to map is 2 and the starting data set is 2, then only data sets 2 and 3 will be mapped to the PLC.

!**NUMBER OF DATA SETS TO MAP TO PMD (0=NONE, 1-4) <0>**
specifies how many data sets will be mapped from the outputs of the PLC directly to the variable data of the 475. A 0 indicates no data sets will be mapped and a number from 1 to 4 gives the number of data sets mapped from the PLC to the 475.

!**STARTING DATA SET NUMBER (1-4) <1>**
is the starting data set used when mapping data sets from the PLC to the 475. Only the data sets specified by the **NUMBER OF DATA SETS TO MAP TO PMD** and the **STARTING DATA SET NUMBER** will be mapped. For example, if the number of data sets to map is 3 and the starting data set is 2, then data sets 2, 3 and 4 will be mapped to the 475.

!**PROCESS ASCII DATA (Y/N) <N>**
configures the unit for MODBUS PLUS ASCII data. This prompt must be answered **1=YES** for the unit to process ASCII data from the MODBUS PLUS interface. If the PMD is not configured to process ASCII data from the MODBUS PLUS interface, the data will be received, but it will not be processed.

If the data was sent from a 984 using a MSTR block to read or write registers, the MSTR block will show an error of 0x6210 which indicates that the transaction was rejected by the slave.

!INVERTED/NON-INVERTED (0=INVERTED 1=NON-INVERTED) <1> This prompt lets you determine how the *PLC port* looks at the message/data lines: the way they come in as 1s or 0s (non-inverted)...**or** the complements of these (inverted).

!BCD/BINARY (0=BCD 1=BINARY) <0>
This prompt lets you determine whether the 475 will consider the message/data lines as a Binary or Binary Coded Decimal.

!TIMED SCAN <2> DISABLED
This prompt is for setting a specified scan time for how often the *PLC port* is scanned at the "system" scan rate (between 1 and 180 msec).

!QUEUE SIZE (0-20, 0=DISABLED) <0>
The queue is automatically disabled and it cannot be enabled from the PLC interface when the size is specified as **0**. If the size is specified as **0**, the unit will use the 32 entry FIFO if a nonzero message display time is specified.

The queue is configured to the size specified by this prompt and then it must be enabled by the PLC interface before it will accept any add, delete, or clear functions.

!MESSAGE DISPLAY TIME (0-255 SEC.) <0>
This prompt determines if the messages will remain on the display until replaced, or if they will be displayed for a specified time. When the display time is set to zero, all messages will remain on the display until they are replaced. If a specified time is entered (in seconds ranging from 1 to 255), the messages will each display for that time period before being replaced by the next.

Set up Parameters Online Programming Prompts Option 17

Select SETUP PARAMETERS from the **MAIN>COMMANDS** menu and then **select** option **17 - FUNCTION KEYS/LED CONTROL**. The terminal responds with these prompts:

!LED CONTROL (SELECT 1=PMD, 2=CI, 3=PLC) <1>
lets you choose what you want to use as the input source for the LEDs. If **1=PMD** is selected, the LEDs will be controlled by the function keys. If **2=CI** (*computer interface*) is selected, the LEDs will be controlled by computer interface commands. If **3=PLC** is selected, the LEDs will be controlled by the PLC outputs.

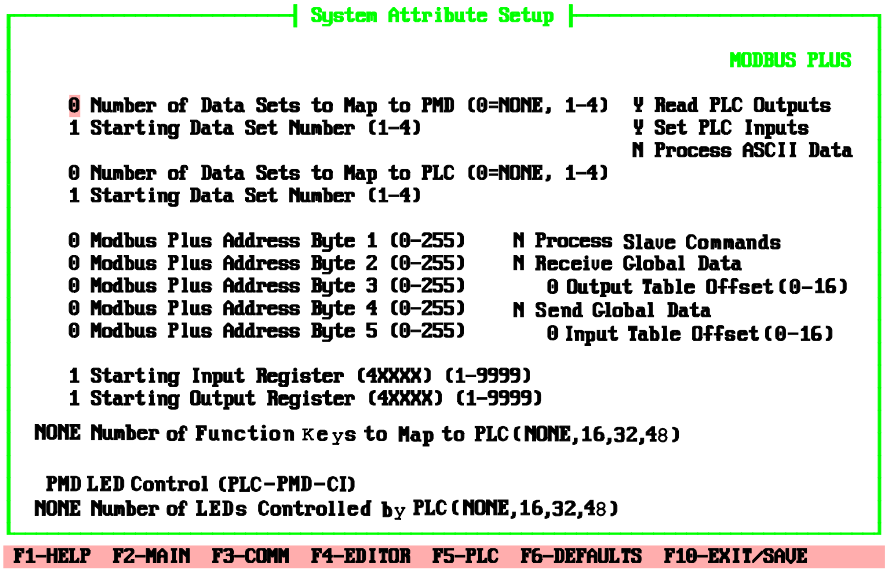
A word of outputs will control 16 LEDs on a bit map basis. Bit 0 of the word controls LED 1 and bit 15 of the word controls LED 16.

!NUMBER OF LEDS CONTROLLED BY PLC (0=NONE, 1=16, 2=32, 3=48) <0>-- (only if 3=PLC on the previous prompt)
is used to specify how many LEDs will be controlled by the outputs of the PLC.

The prompt will reflect how many LEDs are available on the unit. The LEDs are controlled by output words 3, 4, and 5 of the PLC output data being monitored by the 400 product. If the output word is not controlling LEDs, it will be ignored. Each output word can control 16 LEDs on a bit map basis.

!NUMBER OF FUNCTION KEYS TO MAP TO PLC (0=NONE, 1=16, 2=32, 3=48) <0>
allows you to map the function keys to set inputs in the PLC. This prompt will also be dependent on the number of function keys that are available on the unit. Each set of 16 keys is bit mapped to one word of input. Function key 1 sets bit 0 and function key 16 sets bit 15. When the function key is pressed, the input bit is set, and when the function key is released, the bit is cleared. The function keys are mapped to input words 5, 6 and 7.

Offline programming menu:



SECTION 7 - PLC INTERFACING
PART E: 480 GE GENIUS I/O

This section is intended to give you specific information about the 480 GE Genius I/O product that makes it unique from the basic PMD 400. It is a supplement to, and a specified reference point within, the preceding manual sections.

The PMD 480 has all of the PMD 400 capabilities, but the 480 contains support for a Genius Network Adapter ("GENA") board which allows the PMD to be configured as a node on the Genius I/O system for the GE 90-70 Series PLC. The 480 can be configured as an I/O block on a Genius I/O system and will receive data from a bus interface module. A bus interface module is typically a PLC with a Genius bus controller module or a PCIM card installed in a personal computer. The PMD 480 will exist on the Genius I/O network as an I/O block broadcasting its inputs to the bus and reading the outputs sent to it by the bus controller.

The PMD will monitor 32 bytes of output data that will be used to make up the 16 words of the PLC output table in the PMD. The PMD will also use the Genius I/O interface to broadcast 32 bytes of input data that will contain the 16 word PLC input table from the PMD. Please refer to **PART F: REFERENCE** of this section for input/output table definitions.

The information received through the right-angle, eight-position, removable terminal block located on the bottom of the 480 is collected in the PMD's registers and from that point, is manipulated similar to a *parallel port*. The 16 bits of information can be read in either Binary or BCD (Binary Coded Decimal) and inverted or non-inverted. These options are software selectable, as described on the following pages.

Setting up 480 GE GENIUS I/O Communication

The PMD 480 must be configured to fit into the Genius network. The PMD must have a unique serial bus address and it must be configured to use the same baud rate that is used by the bus controller module and the rest of the devices on the network.

To configure your 480 online, **select** **`SETUP PARAMETERS`** from the **`MAIN>COMMANDS`** menu and then **select** option **`5 - GE GENIUS I/O COMMUNICATION`**. If you opt to configure your 480 using offline programming, you access your menu starting from the **`MAIN MENU`**, then selecting **`EDIT MESSAGE PROGRAM`**, then **`SETUP ATTRIBUTES`**, and finally, **`F5 - PLC`**. Illustrations of the offline menus follow this segment on the "online" prompts. The following explanations of the options, however, are universal for both programming methods.

SECTION 7 - PLC INTERFACING
PART E: 480 GE GENIUS I/O

Set up Parameters
Online Programming Prompts
Option 5

!SERIAL BUS ADDRESS (1-30) <1>

The PMD must be assigned a serial bus address that is unique for this Genius I/O bus. No other node on the same bus as the PMD can use this bus address. Bus address 0 is reserved for the hand-held controller and bus address 31 is reserved for the bus controller.

!BAUD RATE (0=153.6E 1=153.6S 2=76.8 3=38.4) <1>

The PMD must be configured for the same baud rate as the other devices on the I/O bus.

0=153.6 K baud extended
1=153.6 K baud standard
2=76.8 K baud
3=38.4 K baud

!BROADCAST CONTROL DATA LENGTH (0-16) <0>

The broadcast control data length specifies how many bytes of analog input data will be put on the bus by the PMD during each serial bus scan. If this is set to 0, no data will be sent to the PMD. If the PMD is to be configured as an I/O block, the PMD should be configured to send 16 words of analog input (%AI) data to the bus controller. The data sent will be the 16 words of the PMD input table. This must match the length of the analog input data expected by the bus controller.

!DIRECTED CONTROL DATA LENGTH (0-16) <0>

The directed control data length determines how many words of data the bus controller will send to the PMD during each serial bus scan. If this value is set to 0, no data will be sent to the PMD. If the PMD is to be configured as an I/O block, the PLC should be configured to send 16 words of analog output (%AQ) data to the PMD. This data will be used to make up the PMD output table. This must match the length of the analog output data that is configured in the bus controller for this I/O block.

SECTION 7 - PLC INTERFACING
PART E: 480 GE GENIUS I/O

!GENA LABEL PMD ID NUMBER (PMDXX 1-30) <1>
The PMD will put a string to identify itself in the GENA setup table. The string will be "PMDXX" where XX is the number entered at this prompt. This is the string that will be displayed on the hand-held monitor when monitoring the I/O on the PMD.

!DESTINATION SERIAL BUS ADDRESS (1-31) <1>

This is the address of the PLC that will be polled if the PMD is configured to poll.

!READ DATAGRAM POLL TIME (0-255) <0>

This is the interval between read datagrams that will be issued by the PMD. The time interval is expressed in tenths of a second (10 = 1 second). This should be adjusted to prevent the PMD from slowing down the Genius Network. If this interval is set to 0, the PMD will issue a new read datagram request to the destination PLC as soon as the last request is complete.

!INPUT REGISTER OFFSET (0-65535) <0>

This is the starting register number (%Rxxxxx) that the PMD will send its input table to using a write device datagram. The PMD will send 16 words of data to be placed starting at this register number. If this number is 0, no data will be sent.

!OUTPUT REGISTER OFFSET (0-65535) <0>

This is the starting register number (%Rxxxxx) that the PMD will read from the PLC. The PMD will read 16 words of data starting from this register. If this number is 0, no data will be read.

!NUMBER OF DATA SETS TO MAP TO PLC (0=NONE, 1-4) <0>
asks if any data sets would be mapped from the 480 to the PLC. If **0** is the response, no data sets will be mapped to the PLC.

Note: Mapping of data sets to the PLC is accomplished through the *PLC interface port* on the 480 unit. This can only be done by sending data to the PMD through the *computer interface port* and mapping it to the PLC.

!**STARTING DATA SET (1-4) <1>** is used to get the starting data set number when mapping the data sets to the PLC. If no data sets are to be mapped to the PLC, then this prompt will not appear.

!**NUMBER OF DATA SETS TO MAP TO PMD (0=NONE, 1-4) <0>** asks if any of the data sets should be mapped from the PLC outputs to the 480. If **0** is the response, then no data sets will be mapped to the 480. Otherwise, the specified number of data sets will be mapped to it.

!**STARTING DATA SET NUMBER (1-4) <1>**
asks for the starting data set number to be used when mapping data sets to the 480 from the PLC outputs. If no data sets are to be mapped to the 480, this prompt will not appear.

!**INVERTED/NON-INVERTED (0=INVERTED 1=NON-INVERTED) <1>** This prompt lets you determine how the PLC port looks at the message/data lines: the way they come in as 1s or 0s (non-inverted)...**or** the complements of these (inverted).

!**BCD/BINARY (0=BCD, 1=BINARY) <0>**
This prompt lets you determine whether the 480 will consider the message/data lines as a Binary or Binary Coded Decimal.

!**TIMED SCAN (1=ENABLED, 2=DISABLED) <2>**
This prompt is for setting a specified scan time for how often the PLC port is scanned at the "system" scan rate (between 1 and 180 msec).

!**QUEUE SIZE (0-20, 0=DISABLED) <0>**
The queue is automatically disabled and it cannot be enabled from the PLC interface when the size is specified as **0**. If the size is specified as **0**, the unit will use the 32 entry FIFO if a nonzero message display time is specified.

The queue is configured to the size specified by this prompt and then it must be enabled by the PLC interface before it will accept any add, delete, or clear functions.

!**MESSAGE DISPLAY TIME (0-255 SEC.) <0>**
This prompt determines if the messages will remain on the display until replaced, or if they will be displayed for a specified time. When the display time is set to zero, all messages will remain on the display until they are replaced.

If a specified time is entered (in seconds ranging from 1 to 255), the messages will each display for that time period before being replaced by the next.

PLC Configuration

Analog I/O

The PMD should be configured as a generic I/O device on the Genius I/O bus. The PMD will use analog inputs (%AI) and analog outputs (%AQ). This data is exchanged in the foreground on the Genius I/O bus. The length of analog input data must match the broadcast control length in the PMD. The length of analog output data must match the directed control length in the PMD. If these are set to 0 and a configuration error occurs, set the PMD for one word of input and one word of output and configure the PLC to match. Depending on the PLC, it may have to be configured for at least one word of inputs and one word of outputs before the PMD can be placed on the bus. The PMD must be on the bus before it can send or receive datagrams. The PMD will only send the input table when the broadcast length is set to 16 and the PMD will only use the output data when the length is set to 16.

Sending Datagrams

The PMD can also be configured to send write device datagrams and read device datagrams to a 90-70 series PLC to read and write register(%R) memory. This is done by programming the input and output register offsets. If the input register offset is non-zero, the PMD will send a write device datagram to the PLC each time the data in the PLC input table changes. If the output register offset is non-zero, the PMD will send read device datagrams based on the poll time interval to read the data that will make up the PLC output table. The PMD must be active as a device on the Genius I/O bus before it can send datagrams.

Receiving Datagrams

The PMD can also receive datagrams sent by the PLC. These datagrams can be used to send the output table to the PMD and read a copy of the PLC input table from the PMD. The PMD must be active on the Genius I/O bus before it can receive datagrams. The PMD can process Read Device datagrams and Write Device datagrams that will allow any node on the Genius I/O Bus to read the PLC and input table from the PMD and write a PLC output table to the PMD. These datagrams can be sent from a PCIM card in a personal computer, a PLC, or any other intelligent node on the Genius I/O bus. This section discusses sending these datagrams using a PCIM card in a personal computer and using the COMREQ instruction in a PLC.

A copy of the PMD's input table can be read by sending a read device datagram using the SendMsgReply function on a PCIM card installed in a personal computer. The datagram should be programmed in 'C' as follows.

```
Msg.Destination=node;
Msg.Function=0x20;            /*Function Code */
Msg.T_SumFunction=0x1E;  /*Read Device Sub Function Code */
Msg.R_SubFunction=0x1F;   /*Read Device Reply Sub Function Code */
Msg.Priority=NORMALP;        /*Normal Priority */
Msg.T_Length=6;                 /*Length of datagram to be sent */

Msg.Data[0]=0;                  /*reserved always 0 */
Msg.Data[1]=1;                  *LSB of address */
Msg.Data[2]=0;
Msg.Data[3]=0;
Msg.Data[4]=0;                  /*MSB of address */
Msg.Data[5]=32;                 /*length of data to read (bytes) */

status = SendMsgReply(1, &Msg);
```

If the call to SendMsgReply was successful, you must issue a call to GetMsg to read the received reply to the Read Device.

```
status = GetMsg(1,&Msg);
if (status==SUCCESS)
    {
    if(Msg.SubFunction==0x1F)
        {
        /*The data will be found at Msg.Data[6] */
        /*Msg.Data[6] LSB word 1 Msg.Data[7] MSB word 1 */
        /*Msg.Data[8] LSB word 2 Msg.Data[9] MSB word 2 */

        for(i=0;i<16;i++)
        inputs[i]=Msg.Data[(i*2)+6]+(Msg.Data[(i*2)+7]<<8);
        }
        }
```

COMREQ 15 Request Datagram Reply

The Request Datagram Reply command can be used to send any datagram that causes the target device to return a reply. In this case it will be used to send a Read Device datagram. The Bus Controller will automatically dequeue the re reply datagram and pass it to the CPU. The Command Block for a read device datagram follows.

Address	13	Length of command + length of datagram
Address+1	0	No Wait
Address+2	8	Status block memory type
Address+3	180	Status block offset
Address+4	0	Idle timeout value
Address+5	0	Max. comm. time
Address+6	15	Command number
Address+7	5	Device number to receive the message
Address+8	0x20	Function code
Address+9	0x1E	Subfunction code
Address+10	0	Priority
Address+11	6	Length of datagram
Address+12	0x1F	Subfunction code of reply
Address+13	8	Reply memory type
Address+14	260	Reply offset
Address+15	32	Maximum size of reply
Address+16	0x0100	LSB datagram byte 1 reserved always 0 MSB datagram byte 2 LSB of address
Address+17	0x0000	LSB datagram byte3 Address MSB datagram byte 4 Address
Address+18	0x2000	LSB datagram byte 5 MSB of address MSB datagram byte 6 Read data length (bytes)

PCIM SendMsg

The PMD's PLC output table can be updated by sending a write device datagram using the SendMsg function on a PCIM card installed in a personal computer. The datagram should be programmed in 'C' as follows.

```
Msg.Destination=node;
Msg.Function=0x20;  /*Function code */
Msg.Subfunction=0x20;      /*Write device subfunction code */
Msg.Priority=NORMALP;      /*Normal priority */
Msg.Length=6;              /*Length of datagram to be sent*/

Msg.Data[0]=0;             /*Reserved always 0 */
Msg.Data[1]=1;             /*LSB of address */
Msg.Data[2]=0;
Msg.Data[3]=0;
Msg.Data[4]=0;             /*MSB of address */
Msg.Data[5]=32;            /*length of data to read (bytes) */
```

```
Msg.Data[6]=LSB word 1;
Msg.Data[7]=MSB word 1;
Msg.Data[8]=LSB word 2;
Msg.Data[9]=MSB word 2;
--------------------
Msg.Data[36]=LSB word 16;
Msg.Data[37]=MSB word 16;

status=SendMsgReply(1,&Msg);
```

COMREQ 14 Send Datagram

The Send Datagram command can be used to send any datagram . In this case, it will be used to send a Write Device datagram. The Command Block for a write device datagram is shown below.

Address	29	Length of command + length of datagram	
Address+1	0	No wait	
Address+2	8	Status block memory type	
Address+3	280	Status block offset	
Address+4	0	Idle timeout value	
Address+5	0	Max. comm. time	
Address+6	14	Command number	
Address+7	5	Device number to receive the message	
Address+8	0x20	Function code	
Address+9	0x20	Subfunction code	
Address+10	0	Priority	
Address+11	38	Length of datagram	
Address+12	0x0100	LSB datagram byte 1	Always 0
		MSB datagram byte 2	LSB of address
Address+13	0x0000	LSB datagram byte 3	Address
		MSB datagram byte 4	Address
Address+14	0x2000	LSB datagram byte 5	MSB of address
		MSB datagram byte 6	Read data length (bytes)
Address+15	0	Word 1	Output table word 1
Address+16	0	Word 2	Output table word 2
Address+17	0	Word 3	
Address+18	0	Word 4	

Address+31	0	Word 16	

SECTION 7 - PLC INTERFACING
PART E: 480 GE GENIUS I/O

Operation The following is a brief description of the input and output tables used for the PLC I/O.

PLC Input Table The entire PLC input table is copied to the GENA input table each time any of the words in the PLC input table is changed. GENA will broadcast the inputs onto the Genius I/O system on its next scan. The PMD does not wait until the inputs are broadcast before it updates the input table with new inputs. It is possible that inputs can change and be changed again before GENA can broadcast them onto the network.

Function Keys Each time a function key is pressed, the corresponding bit in the PLC input table will be set. When the key is released, the bit will be cleared. Only one key at a time can be pressed.

Data Sets Each time a data set in the PMD is changed, it will be copied to the PLC input table. A data set can be changed in the PMD by the computer interface, the PLC interface, or the data entry keypad (PMD 480 only).

Status Word Any change in the status word will be copied to the PLC input table.

PLC Output Table The PMD will read its entire table of outputs from the GENA card on each scan of the PMD. Each time the outputs are read, they will be monitored for a change and the PMD will take the appropriate action for each output table entry.

Message Word The message word contains the message number used to trigger a message or add the message to the queue.

Control Word The control word determines how the output data table will be interpreted. It contains bits to control messages, data sets, and LEDs.

LED Status The LED status words are used to control the state of the 16 LEDs on the front panel of the PMD 480.

Data Sets Four words of output are used to control the four variable data sets in the PMD.

Queue Display Time The display time for a message added to the queue is entered here.

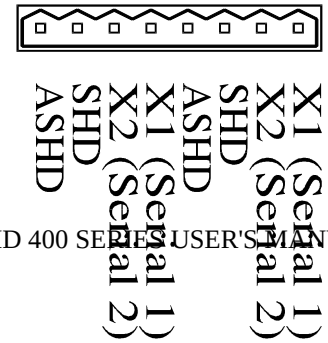
Queue Delete Msg. The message number to be deleted from the queue is entered here.

SECTION 7 - PLC INTERFACING
PART E: 480 GE GENIUS I/O

Offline Programming Menu:

```
                    System Attribute Setup

                                                          GE GENIUS I/O

   1 Serial Bus Address (1-30)
   0 Baud Rate (0=153.6E 1=153.6S 2=76.8 3=38.4)
  32 Broadcast Control Data Length (0-255)
  32 Directed Control Data Length (0-255)
   1 Gena PMD ID Number (PMDXX 1-30)

   0 Number of Data Sets to Map to PLC (0=NONE, 1-4)
   0 Starting Data Set Number (1-4)

   0 Number of Data Sets to Map to PMD (0=NONE, 1-4)
   1 Starting Data Set Number (1-4)

NONE # of Function Keys to Map to PLC (NONE, 16, 32, 48)

 ??? LED Control (PLC-PMD-CI)
NONE # of LEDs Controlled by PLC (NONE, 16, 32, 48)

F1-HELP  F2-MAIN  F3-COMM  F4-EDITOR  F5-PLC  F6-DEFAULTS  F10-EXIT/SAVE
```

Bottom view of GENA connector



SECTION 7 - PLC INTERFACING
PART F: REFERENCE

PLC Output Table (Data received by the PMD)

WORD #	DESCRIPTION
1	Message/Data Word
2	Control Word: defined below
3	1st 16 LEDs Bit 0 = LED 1 (0=off, 1=on) Bit 1 = LED 2 ---------- bit 15 = LED 16 (most significant bit)
4	2nd 16 LEDs (with 400E expander module except 480 unit) Bit 0 = LED 17 (0=off, 1=on) Bit 1 = LED 18 ---------- Bit 15 = LED 32 (most significant bit)
5	3rd 16 LEDs (with 400E expander module except 480 unit) Bit 0 = LED 33 (0=off, 1=on) Bit 1 = LED 34 ---------- Bit 15 = LED 48 (most significant bit)
6	Data Set 1
7	Data Set 2
8	Data Set 3
9	Data Set 4
10	Message to Clear from Queue (0-65535, 65535 = clear queue)
11	Message Display Time in Seconds (0-255)
12	Variable Data Slave Address
13-16	reserved; do not use

Control Word Description

Bit 0 =	Message/Data (0=data; 1=message)
Bit 1 =	Data Select 1
Bit 2 =	Data Select 2
Bit 3 =	Data Hold (0=not set; 1=set)
Bit 4 =	Variable Data Slave Address Valid (0=valid; 1=invalid)
Bits 5-7 =	reserved; do not use
Bit 8 =	Queue Enable (1=enabled; 0=disabled)
Bit 9 =	Data Sets Valid (0=valid; 1=invalid)
Bit 10 =	LEDs valid (0=valid, 1=invalid)
Bit 11 =	Delete Message Valid
Bit 12 =	Data set 1 enable (0=enabled; 1=disabled)
Bit 13 =	Data set 2 enable (0=enabled; 1=disabled)
Bit 14 =	Data set 3 enable (0=enabled; 1=disabled)
Bit 15 =	Data set 4 enable (0=enabled; 1=disabled)

Message/Data Word - The 16 bits of message/data word are used similar to the 16 message/data bits of a *parallel port* on a PMD 300. This 16-bit word is interpreted as either a Binary or Binary Coded Decimal (BCD) number and then translated as either a message number or variable data set, depending on the value of the Control Word.

Control Word - The least significant four bits of the Control Word are also used like the control bits on the *parallel port*. The Message/Data and Data Hold bits in the Control Word should normally be set to one and the Data Select bits should be zero. This will make the Message/Data word always represent a message number. By using the input bits, the PLC can trigger messages and send new data set information. When the variable data slave address bit is valid, the variable data is sent to the slave matching the address found in output word 12. The LEDs valid status bit works in the same manner. When the LED status bit is set to 1 the outputs will not be mapped into the LEDs. The PLC will only control the LEDs when this bit is clear.

The **Data Select 1 and 2** bits determine which data set is represented by the Message/Data Word when the Message/Data bit is set to 0.

Data Select 1	Data Select 2	Data Set
0	0	1
1	0	2
0	1	3
1	1	4

The **Data Hold** bit determines whether or not the data in the Message/Data Word is valid. The data in the Message/Data Word will not be interpreted as valid until the Message/Data bit is set to "Logic 0" and the Data Hold bit is set to "Logic 1." When triggering a message, however, the Message/Data and Data Hold bits should both be set to "Logic 1."

The **Queue Enable** bit of the Control Word controls whether the queue is enabled or disabled. When the bit is set to one, the queue is enabled; when the bit is set to zero, the queue is disabled.

The **Data Sets Valid** status bit determines when the PMD will read the outputs being used for data set information. When this bit is set to 1 (invalid), the PMD will not read the outputs being used for data set information. The outputs for data sets will only be used when the data sets valid bit is clear (set to 0).

Up to four output words can be used to directly map information from the PLC outputs to the variable data in the PMD. One word of output information will be mapped directly into one data set. Each time the information in the PLC changes, the new data will be put into the corresponding data set in the PMD. PLC data is only mapped into the PMD when it changes. The variable data is also sent to the slave display addressed in output word 12 and validated by bit 4 of the Control Word.

A message is triggered by setting the Message/Data Word to the desired message number. Then set the Message/Data bit in the Message Control Word to a one to indicate that this is a message number. After the Message/Data bit is set, the Data Hold bit should be set in the Message Control Word to indicate that the Message/Data Word contains valid information.

Up to three output words can be used to control three sets of 16 LEDs available with the addition of the 400E expander module. Each bit in each word will control one LED. When the bit is set to 1, the LED will be on, and when the bit is set to 0, the LED will be off. The LEDs will be updated each time the PMD receives new data from the PLC. Bit 0 is the least significant bit of the word.

PLC Input Table (Data Sent by the PMD)

WORD #	DESCRIPTION
1	Data set 1
2	Data set 2
3	Data set 3
4	Data set 4
5	1st 16 function keys function key 1 = bit 0 function key 2 = bit 1 ---------- function key 16 = bit 15
6	2nd 16 function keys (with 400E expander module except 480 unit) function key 17 = bit 0 function key 18 = bit 1 ---------- function key 32 = bit 15 (most significant bit)

	7	3rd 16 function keys (with 400E expander module except 480 unit) function key 33 = bit 0 function key 34 = bit 1 ---------- function key 48 = bit 15 (most significant bit)
PMD 460, 470	8	Handshaking Bit 0 = inputs Bit 1 = outputs
PMD 450, 475, 480	8	Data Ready Word Bits 0-3 = unused
PMD 460, 470	9	Bit 4 = operator entering data set 1 (0=no, 1=yes) Bit 5 = operator entering data set 2 (0=no, 1=yes) Bit 6 = operator entering data set 3 (0=no, 1=yes) Bit 7 = operator entering data set 4 (0=no, 1=yes) Bit 8-15 = unused
PMD 450, 475, 480	9	Status Word Bit 0 = DISPLAY MODE (1=yes; 0=no)
PMD 460, 470	10	Bit 1 = queue full (1=yes; 0=no) Bit 2 = message valid Bit 3 = delete message valid Bit 4 = data sets valid Bit 5 = LEDs valid
	10 or 11-16	reserved; do not use

The **Data Set** information consists of up to four 16-bit words. The PMD can be configured to map up to four data set words directly into four registers in the PLC. Each time a data set changes in the PMD, it will be written to the corresponding register in the PLC. A data set in the PMD can be changed by the computer interface or PLC outputs.

The **Function Key** information consists of up to three 16-bit words. There will be one word for each set of 16 function keys. The number of function keys that can be mapped to the PLC is dependent on the number of function keys installed. Each time a function key is pressed, the corresponding bit in the PLC inputs will be set. When the function key is released, the bit in the PLC inputs will be cleared. The function key inputs are only set as long as the function key is pressed.

The **Handshaking** word is only present on the PMD 460 and 470 units. This will change the offset of the Data Ready Word and the Status Word for these units, as shown above. The input bit is set when the inputs being sent to the PLC have changed.

The input bit is cleared after the new inputs have actually been sent to the PLC. The output bit is set by the PMD when new outputs are received from the PLC. The output bit is cleared by the PMD when it has processed new outputs.

The **Data Ready Word** of the input information has four bits that tell the PLC when a user has selected a data set for entry from the keypad. If a bit is set, the user has selected that data set using the keypad. After the data is entered and the data set is deselected, the bit is cleared to indicate that the user has finished entering data.

The **Status Word** of the input data shows the status of the PMD. The Status Word is only valid if there are no errors communicating with the PMD. If there are errors communicating, the bits of the Status Word may be incorrect.

The **Display Mode bit** (Bit 0) is set to one when the PMD is in DISPLAY MODE. **Queue Full bit** (Bit 1) is set to one when the Queue is full and no more entries can be added. **Message Valid bit** (Bit 2) is set as a reflection of the Message/Data bit in the Control Word of the output table. When the Message/Data bit is valid, this bit will be set to the same value. If the Message/Data bit is 0, this bit will be 0; if the Message/Data bit is 1, this bit will be 1.

Delete Message Valid bit (Bit 3) is set as a reflection of the Delete Message bit of the Control Word (bit 11). When the Delete Message Valid bit is set, this bit will be set; when it's clear, this bit will be clear. **Data Sets Valid** (Bit 4) is set when the Data Sets Valid bit of the Control Word (bit 9) is set, and clear when the Data Sets Valid bit is clear. **LEDs Valid bit** Bit 5 is set when the LEDs valid bit of the control word (bit 10) is set, and clear when the LEDs valid bit is clear.

The Queue

As described previously, Bit 8 of the Control Word of the PLC interface output table controls whether the queue is enabled or disabled. When the bit is set to one, the queue is enabled; when the bit is set to zero, the queue is disabled. When the queue makes a transition from enabled to disabled, the last message displayed from the queue will remain on the display until it is replaced by another message.

When the queue makes a transition from disabled to enabled, it will display the next message in the queue. An error in the PLC communication will automatically disable the queue. It will return to its prior state when error-free communication is regained.

When the queue is enabled, the data sets that are being read from the PLC will not be updated on the display. Each message entry in the queue contains a copy of the data sets at the time the entry was added to the queue. These values will be used each time the message in the queue is displayed and the data sets on the display will not be overwritten with data from the PLC when the queue is enabled.

Add a Message to the Queue

Messages can only be added to the queue when it is enabled. A message is added to the queue when the Message/Data bit and the Data Hold bit of the Control Word are set to one. If these two bits are set, the PMD will add a message to the queue each time a different message number is read from the Message/Data Word. Any time one of these bits makes a 0 to 1 transition while the other bit is set to one, a message will be added to the queue.

The table below shows when a message is added to the queue. Please refer to the "key" that follows.

MSG/DATA	DATA HOLD	MSG #	ACTION TAKEN
0	X	XXXXX	No message added to the queue.
X	0	XXXXX	No message added to the queue.
1	0->1	XXXXX	Message XXXXX added to the queue.
0->1	1	XXXXX	Message XXXXX added to the queue.
1	1	XXXXX->YYYYY	Message YYYYY added to the queue.

X = don't care (bit can be 1 or 0)
0->1 = bit goes from 0 to 1
0 = bit is clear
1 = bit is set

Delete a Message from the Queue

Messages can only be deleted from the queue when the queue is enabled. A message is deleted from the queue when a message number is placed in word 10 of the PLC output table and the Delete Message Valid bit is set. If the Delete Message Valid bit is set and the message number in word 10 of the output table changes, the new message number will be deleted. When the Delete Message Valid bit makes a transition from 0 to 1, the message in word 10 of the output table will be deleted from the queue.

The PMD 450 Allen-Bradley Remote I/O interface will delete messages from the queue in a different manner. Since the A-B Remote I/O interface does not have a table of outputs, it will delete the message in the Message/Data Word when the Delete Message Valid bit is set. When a message is deleted from the queue in this manner, the Data Hold and Message/Data bits of the Control Word must be clear.

Clear the Queue

The queue can only be cleared when it is enabled. The queue is cleared when message number 65535 is placed in the Delete Message Number of the PLC output table (word 10) and the Delete Message Valid bit is set. The procedure for clearing the queue is the same as deleting a message from the queue with the message number specified as 65535.

Data Sets and the Queue

Each time a message is added to the queue, the current values of the data sets from the PLC output table are placed on the queue at the time the message is added. These data set values are used regardless of the state of the Data Sets Valid bit.

Each time a message is pulled off the queue to be displayed, the data sets for the message are placed in the variable data sets in the PMD. If the data sets from the PLC change while a message is on the display, they will not be updated on the display. The data sets from the queue entry will also be mapped back to the PLC if the PMD is configured properly.

Display Time and the Queue

Each time a message is added to the queue, the display time for that message will be taken from word 11 (Message Display Time) of the PLC output table. If this display time is zero, the unit will use the Message Display Time (default is 5) from the attributes of the PMD.

PMD 450 Error Messages

PLC INTERFACE ERROR - 450 interface board fault detected during self-test upon unit reset.

PLC COMM ERROR - This is usually the result of conflicting setups or wiring. Check baud rate setting and other system configurations.

PLC RACK DOES NOT EXIST - The PLC and the 450s are in conflict with each other or another Remote I/O rack. Check 450 rack number and communication between the PLC and the rack selected by the 450. This error is triggered when the 450 does not receive a response from the PLC.

PLC IN PROGRAM OR TEST MODE - Set the PLC to the run mode.

PLC RACK ERROR RACK NOT RESPONDING - Monitored rack incurs an error. Check rack.

PMD 450 Data Highway Error Messages

DATA HIGHWAY ERROR NO RESPONSE TO READ/WRITE COMMAND - This error message will be displayed when there is no response to the Read or Write command. This is usually caused by configuring the PMD as a Data Highway Station and then connecting the unit to a Data Highway Plus Network. It can be caused by bad wiring or incorrect configuration in the PMD or PLC.

DATA HIGHWAY ERROR STS=XX EXTSTS=YY - This error message is displayed when the PMD receives a reply with an error code in it. The error code is displayed with this message. The STS code is the status byte of the message. The EXTSTS is only included when the STS indicates it is required.

The status codes are defined in the Allen-Bradley Data Highway Protocol and Command set manual.

DATA HIGHWAY ERROR RESET FAILURE - This error message is displayed if the interface does not successfully complete its reset procedure. This indicates a hardware failure or a software problem within the unit and should be reported to UTICOR.

DATA HIGHWAY ERROR UNKNOWN COMMAND CMD=X SOURCE=Y - This error message is displayed if the interface does not successfully complete its start up procedure. This indicates a hardware failure or a software problem within the unit and should be reported to UTICOR.

DATA HIGHWAY ERROR UNKNOWN COMMAND CMD=X SOURCE=Y - This error is displayed when the PMD receives a command that is not recognized. The PMD only recognizes the commands that are defined in the read/write command portion of this document.

DATA HIGHWAY ERROR BAD ADDRESS - This error is displayed when the PMD receives a command to read or write from an address other than the addresses specified by the read/write commands that the PMD will process.

SECTION 7 - PLC INTERFACING
PART F: REFERENCE

PMD 460 Error Messages

MODULE FAILURE BASE XX - The PLC module indicated has failed.

MODULE NOT ENABLED - The PLC module has not been enabled by the PLC. A module not enabled by the PLC will not monitor any communication.

PLC COMM ERROR - No communication is occurring between the 460 and the PLC. The cables and communication parameters should be checked.

PLC SETUP ERROR INPUT/OUTPUT MISMATCH - This error occurs when the PLC is configured for a different number of inputs and outputs other than what the 460 is configured for. The 460 is configured for 16 words of input and 16 words of output. If the PLC has the same modules configured differently, this error will occur.

PLC INTERFACE HARDWARE FAILURE - This error occurs when the PLC interface detects a failure in its hardware. The status code identifies the specific failure through a bit mapped value. Bit 0-Status code 1= Ram failure, Bit 3-Status code 8 = Serial Communication failure.

PLC INTERFACE FAILURE - This error occurs when the PLC interface fails to successfully complete its configuration process when it is reset.

PMD 470 Error Messages

Several PLC error messages can be displayed when the 470 detects an error in the PLC communication or setups. These errors are shown below. When one of these errors is displayed, it will remain on the display until another message is displayed. An error message may still be on the display even though the error has been corrected. If another message is not triggered, the error message will remain on the display.

PLC INTERFACE ERROR - This error occurs when there is a hardware failure on the Modicon PLC interface board. The error code is a bit map of the cause of the error. Bit 0 = RAM failure, bit 1 = PROM failure, Bit 3 = Communication chip failure, Bit 4 = illegal interrupt.

PLC SETUP MISMATCH - This error occurs when the setup parameters in the PLC do not match the setup parameters in the 470. The 470 expects the PLC to be set up for 16 input words and 16 output words.

MODULE NOT ENABLED - This error occurs when the 470 has not received the required messages to begin communication. The setup parameters on the 470 and the PLC (particularly the traffic cop) should be checked and the unit restarted.

PLC HANDSHAKING ERROR - This error occurs when the handshaking parameters on the PMD PLC interface are set wrong. If this error occurs, there is a problem in the 470.

PLC COM ERROR - This message occurs when the communication is interrupted for any reason. If this message occurs, then the user should check the setup parameters and the hardware connections between the PLC and the 470.

PMD 475 Error Messages

PLC error messages are displayed when they occur to inform you of any problems. A PLC error message will remain on the display until another message from any source removes the error from the display. If a new message is not triggered, the PLC error message will remain on the display indefinitely.

TIMEOUT ERROR - A timeout error occurs when the interface board does not respond to the command issued by the 475. This is an internal error and indicates that there is a hardware problem in the unit. The unit will function after this error occurs but the PLC interface will be inoperable. The command that failed is indicated by the command code displayed.

COMMAND	DESCRIPTION
0	GET SERVICE REQUEST
1	READ OUTPUTS
2	WRITE INPUTS
4	GET RESPONSE
5	ABORT TRANSACTION
6	GET INTERFACE STATUS
7	PERFORM DIAGNOSTICS
8	SOFTWARE RESET

ROUTING FAILED - This error occurs when the PLC interface cannot find the destination node on the network. The code that gives the reason why the routing failed can be found under the MODBUS PLUS command errors in the MODICON documentation. The "error node" is the node that could not be found.

INTERFACE ERROR - This error occurs when there is a problem with the interface completing a command. The "CMD" tells you which command failed and the "CC" tells you whey the error occurred.

COMMAND	DESCRIPTION
0	GET SERVICE REQUEST
1	READ OUTPUTS
2	WRITE INPUTS
4	GET RESPONSE
5	ABORT TRANSACTION
6	GET INTERFACE STATUS
7	PERFORM DIAGNOSTICS
8	SOFTWARE RESET

CRASH CODE	DESCRIPTION
1	INTERFACE TIMEOUT
2	BAD COMMAND CODE

RESTART ERROR - This error occurs when the unit cannot properly perform the startup procedure for the MODBUS PLUS network interface. This indicates that there is a malfunction within the 475 unit.

EXCEPTION RESPONSE - This error occurs when either the starting input register or the starting output register does not exist in the specified controller. To correct this error, choose a starting input or output register such that sixteen contiguous registers can be used.

The command that had an error appears after "CMD;" the error code comes next; then the register that caused the error is shown after "INPUT/OUTPUT REG."

COMMAND	DESCRIPTION
0	GET SERVICE REQUEST
1	READ OUTPUTS
2	WRITE INPUTS
4	GET RESPONSE
5	ABORT TRANSACTION
6	GET INTERFACE STATUS
7	PERFORM DIAGNOSTICS
8	SOFTWARE RESET

ERROR CODE	DESCRIPTION
2	REGISTER DOES NOT EXIST
1	OPERATION NOT SUPPORTED

UNKNOWN ERROR - This message is displayed when an error occurs that is not recognized by the PMD 475. This error should be reported to UTICOR Customer service.

COMMAND	DESCRIPTION
0	GET SERVICE REQUEST
1	READ OUTPUTS
2	WRITE INPUTS
4	GET RESPONSE
5	ABORT TRANSACTION
6	GET INTERFACE STATUS
7	PERFORM DIAGNOSTICS
8	SOFTWARE RESET

PMD 480 Error Messages

GENIUS I/O ERROR
LOCK RELEASE TIMEOUT
FUNCTION = OUTPUTS

GENIUS I/O ERROR
LOCK RELEASE TIMEOUT
FUNCTION = INPUTS

GENIUS I/O ERROR
LOCK RELEASE TIMEOUT
FUNCTION = X

GENIUS I/O ERROR
LOCK REQUEST TIMEOUT
FUNCTION = OUTPUTS

GENIUS I/O ERROR
LOCK REQUEST TIMEOUT
FUNCTION = INPUTS

GENIUS I/O ERROR
LOCK REQUEST TIMEOUT
FUNCTION = X

GENIUS I/O ERROR
TIMEOUT CMD = X

These errors occur when the GENA board does not respond to a command issued by the PMD. This error indicates an internal problem on the GENA board.

GENIUS I/O ERROR DUPLICATE ADDRESS BUS ADDRESS = X - This error occurs when the PMD is given the same bus address as another node on the Genius I/O bus. Check the address and either change the PMD bus address or the other node with the same address to correct the problem.

GENIUS I/O ERROR NO CONTROLLER BUS ADDRESS = X - This error occurs when there is no bus controller present on the Genius I/O bus. Check the bus controller to make sure that it is operating correctly.

GENIUS I/O ERROR COMM FAILURE BUS ADDRESS = X - This error occurs when there is a communication failure between the PMD and the Genius I/O bus. Check the baud rate set up, wiring, and termination resistors.

GENIUS I/O ERROR GENA ERROR = X - This error occurs when there is a hardware failure on the GENA board. It indicates that a component on the GENA board itself has failed.

GENIUS I/O ERROR STARTUP FAILED GENA ERROR = X - This error occurs during the startup procedure when the GENA board fails its self test procedure.

GENIUS I/O ERROR CONFIGURATION FAILED COMMAND STATUS = X - This error occurs when the configuration parameters used by the PMD on the GENA board are rejected by the set up command on the GENA board.

SECTION 7 - PLC INTERFACING
PART F: REFERENCE

SECTION 8 - ASCII PROTOCOL

The Computer Interface can operate with either normal UTICOR protocol (not documented in this manual) or ASCII protocol. When the Computer Interface has been programmed for ASCII protocol, the display will act like an ASCII terminal.

!To enter ASCII protocol, **put** the 400 Series unit into PROGRAM MODE, **select** **`Terminal Emulator`** from the **`MAIN MENU`**, then **`Set Up Parameters`**, then **choose** **`Serial Ports`**. The terminal responds with the sub-menu:

```
SELECT WHICH MODE TO SETUP
1) COMPUTER INTERFACE
2) PROGRAM TERMINAL
3) PRINTER
4) TAPE IN AND OUT
?
```

!**Select** **`1) COMPUTER INTERFACE`** to set it up then answer the following prompts:

```
COMPUTER INTERFACE MODE (0=PMD, 1=ASCII) <0>
BAUD RATE (1=300, 2=600, 3=1200, 4=2400
     5=4800, 6=9600) <9600>?
STOP BITS (SELECT 1 OR 2) <1>?
PARITY ENABLED (SELECT Y OR N) <N>?
PARITY MODE (0=ODD, 1=EVEN) <1>?
CHECKSUM TYPE (0=CRC, 1=EOR, 2=NONE) <1>?
```

!**Enter** the proper letter or digit to select each item, or Enter to retain the current setting. **Press** Esc to exit the **`Serial Ports`** sub-menu.

ASCII Functions After you have set your unit to ASCII protocol, **put** it back into the DISPLAY MODE, the normal operating mode. Now when the 400 unit receives the ^A command to select it (explained below), all text will be displayed as it is received.

All control characters (characters typed with the Ctrl key down) will be ignored. Only printable characters will be processed. The ASCII circumflex (^) will not be printed since it is used for the control functions that control the way that text is displayed. Each time a ^ is sent to the unit, it is interpreted as the beginning of a new command with the exception of ^Qnn commands which must be received as part of a ^P or ^O command (Also explained below). When the unit receives a ^ command, it will start a new command and throw out any part of a command it may have been in the process of receiving.

SECTION 8 - ASCII PROTOCOL

The ASCII mode is compatible with the current ASCII slave products with two exceptions: the ^D command for scrolling text (although the ^P command allows you to operate the unit similarly) and the PMD 3000 commands, ^K (select character size) and ^L (select number of sticks per line).

You select a unit with the ^Agguuuu command before processing any other characters or commands. If you do not select it, it will not process any input.

ASCII Commands

The following is a list of commands and their general meanings. A more detailed description of each and the definition of its variables follows:

^Agguuuu	=	Select unit
^Bn	=	Select baud rate
^Cn	=	Select center mode
^En	=	Reset display
^Frr	=	Select scroll rate
^Gbbcc	=	Select on/off rate
^Hrrcc	=	Position cursor
^In	=	Select new line
^Jn	=	Select wrap
^Knnnnn	=	Trigger programmed message
^Msnnnnn	=	New variable data
^Nnnnnn	=	Delete message
^Onnnnn	=	Program message
^P	=	Display text
^Qnn	=	Message attributes (Note that this command can only be used inside ^Onnnnn or ^P commands)
^Rxy	=	Read function keys
^Swxyz	=	Set LED status (bit map)
^T	=	Read LED status (bit map)
^Un	=	Read data set(s)
^Vxyabcd	=	Set LED on/off
^Yn	=	Turns the relay on or off
^Z	=	End of text or message

^Agguuuu Select unit - This command selects which unit(s) will process the ASCII commands that are sent out. You can address units as a whole group, a subset, or as individual units. Only units that have been selected will process commands.

gg = a 2-digit group number
uuuu = a 4-digit unit number

In both cases, the number must be padded with zeroes (e.g. For unit 23, the number would be 0023).

All units will always process the ^A commands to find out if they have been selected or deselected (a unit is deselected by not being selected in the ^A command). Setting Group x and Unit x selects only the unit with those numbers; setting Group x and Unit 0 selects all units in group x; setting Group 0 and Unit 0 selects all units.

^Bn Select blink mode - This command tells the 400 unit that all text should blink.

n = 0 means turn blink off
n = 1 means turn blink on

^Cn Select center mode - This command allows you to center text for each line of the display.

n = 0 means turn centering off
n = 1 means turn centering on

^En Reset display - This command performs clear and/or reset of the selected unit. You can specify which level of reset to perform.

n = 0 means clear the display and home the cursor
n = 1 means clear the display and reset all parameters modified by ASCII protocol
n = 2 means clear the display and leave the cursor at its current position

^Frr Select scroll rate - This command selects the rate at which the unit will scroll text from right to left.

rr = 01 to 99 (.1 second to 9.9 seconds) Values less than ten must be padded with zeroes.

^Gbbcc Select on/off rate - This command allows you to select the rate at which the display will blink on and off. The valid range is between 1 and 99. Values less than ten must be padded with zeroes.

bb = 01 to 99 blink on interval (.1 second to 9.9 seconds)
cc = 01 to 99 blink off interval (.1 second to 9.9 seconds)

^Hrrcc Position cursor - This command allows you to position the cursor anywhere on the display.

rr = 01 - 04 row
cc = 01 - 20 column

^In Select new line - This command controls how a carriage return sent to the display will affect the cursor. If this option is on, then a carriage return will advance the cursor to the beginning of the next line. If it is off, then a carriage return will advance the cursor to the beginning of the line it is currently on. The default value is ON.

n = 0 means **new line** is off
n = 1 means **new line** is on

^Jn Select wrap - This command controls how the cursor will advance past the end of a display line. When this option is on, the cursor will advance to the beginning of the next line. When this option is off, the cursor will remain at the end of the line it is currently on.

n = 0 means the **wrap** is off
n = 1 means the **wrap** is on

^Knnnnn Trigger programmed message - This command allows you to display a previously programmed message in the unit. The message number must be the number of an existing message; if the message does not exist, nothing will be displayed. The message number must be padded with zeroes for any message less than 10,000.

nnnnn = 5-digit message number

^Msnnnnn New variable data - This command allows you to send new variable data to the unit. The user must define which data set and the new value to be used when sending it.

s = 1 - 4 data set number
nnnnn = 5-digit data set value

^Nnnnnn Delete message - This command allows you to delete a message in your unit. You specify a 5-digit message number to delete.

nnnnn - 5-digit message number

^Onnnnn Program message - Messages are programmed with this command. The message number selected must be different from the other messages already programmed. The text must be followed by a ^Z (end message) or the message will be ignored. The maximum length of text, including carriage returns, is 223 characters.

nnnnn = 5-digit message number

SECTION 8 - ASCII PROTOCOL

^P Display text - This command allows you to display text using the attributes of the 400 Series unit that are not available through the other ASCII commands. This command works the same as the ^Onnnnn (Program message) command except the message is displayed right away and not programmed into the unit. The maximum length of text, including carriage returns, is 225 characters.

^Qnn Message attributes - This command is only usable with ^Onnnnn (Program message) and ^P (Display text) commands. It allows you to add attributes to the display text, some of which are not available through other ASCII protocol commands.

^Q01 = Blink entire message
^Q02 = Center message
^Q03 = Energize relay
^Q04 = Log message
^Q05r = Chain message r= repeat (0=no, 1=yes) *
^Q06gguuuu = Send to slave group and unit
^Q07n = Data set n goes here (All five digits of the indicated data set will be displayed or inserted into the message being programmed).
^Q08 = Insert time here
^Q09 = Insert date here
^Q10btr = Scroll up b=bottom row (1-4), t=top row (1-4), repeat (0=no, 1=yes)
^Q11lr = Scroll left l=line # (1-4), r=repeat (0=no, 1=yes)

* Message numbers for a chain message are to be entered on one line with no carriage return between the individual message numbers. Each message number requires five digits. If a message number has less than five digits, it must be padded with zeroes.

^Rxy Read function keys - This command will cause the 400 unit to send back the current status of the function key buffers. Depending on the options you use, it can return either a bit map of the function keys or a list of the function keys pressed in the order that they were pressed. It can also optionally clear the function key buffer.

x = 0 = bit map
x = 1 = ordered list
y = 0 = normal read
y = 1 = clear function key buffers

The standard reply format from the 400 unit, when a bit map representation is selected, will be in this form:

^Rabcd^Z with no optional keypad
^Rabcdefgh^Z with 16-point expander
^Rabcdefghijkl^Z with 32-point expander

a = function keys 16-13
b = function keys 12-9
c = function keys 8-5
d = function keys 4-1
e = function keys 32-29
f = function keys 28-25
g = function keys 24-21
h = function keys 20-17
i = function keys 48-45
j = function keys 44-41
k = function keys 40-37
l = function keys 36-33

The standard reply format, when an ordered list is selected, will be:

^RFK01,FK05,FK23, FK14, FK42^Z

Length is dependent on the number of function keys pressed. The list will not be more than 16 keystrokes. The first 16 keys will be represented by this list.

^Swxyz Set LED status (bit map) - This command is used to turn the LEDs on and off. The new status is a bit map of the LEDs with a 0 meaning turn the LED off and 1 meaning turn the LED on. If the LEDs are not controlled by the Computer Interface, this command will not affect the LEDs. Each of the digits (w, x, y, z) will be a hexadecimal representation of a group of four LEDs. The reply format is as follows:

w = LEDs 16-13
x = LEDs 12-9
y = LEDs 8-5
z = LEDs 4-1

^T Read LED status - This command will cause the 400 to send back the current status of the LEDs in a bit map form with 0 meaning the LED is off and 1 meaning the LED is on. Each of the digits in the reply will be a hexadecimal representation of the Binary states of four LEDs. The reply format is as follows:

^Tabcd^Z with no optional keypad
^Tabcdefgh^Z with 16-point expander
^Tabcdefghijkl^Z with 32-point expander

a = LEDs 16-13
b = LEDs 12-9
c = LEDs 8-5
d = LEDs 4-1
e = LEDs 32-29
f = LEDs 28-25
g = LEDs 24-21
h = LEDs 20-17
i = LEDs 48-45
j = LEDs 44-41
k = LEDs 40-37
l = LEDs 36-33

^Un Read data sets - This command will cause the 400 product to return the current value of one or all of the variable data sets.

n = 0 means send all data sets
n = 1-4 means send only the data set specified by n

The standard reply format will vary depending on whether all or only one of the data sets was selected. The format for one will be ^Uxaaaaa^Z. The format for all of the data sets is ^Uxaaaaa, bbbbb, ccccc, ddddd^Z.

x = 0 means all data is ready
x = 1 means data set 1 is not ready
x = 2 means data set 2 is not ready
x = 3 means data set 3 is not ready
x = 4 means data set 4 is not ready
aaaaa = data set 1, or specified data set for single data set selection
bbbbb = data set 2
ccccc = data set 3
ddddd = data set 4

^Vxyabcd Set LED on/off - This command is used to turn the status LEDs of the PMD 400E expander module on and off depending on the bit mask sent.

LEDs that are not specified in the bit mask will not be affected by this command. The definitions of the variables in this command are as follows:

x = 0 means turn LEDs off
x = 1 means turn LEDs on
y = 1 means LED group #1 (1-16)
y = 2 means LED group #2 (17-32)
y = 3 means LED group #3 (33-48)
a = LEDs 16-13 of this group
b = LEDs 12-9 of this group
c = LEDs 8-5 of this group
d = LEDs 4-1 of this group

^Yn Turns relay on or off - This command is used to turn the relay on or off.

n = 1 means the relay will be turned on
n = 0 means the relay will be turned off

^Z End of text or message - This command must follow all ^O (Program message) and ^P (Display text) commands for the message to be valid. It will also be the last element in all replies given by the 400 Series unit.

Examples

In this portion we will give you some examples on how messages are programmed. The first is for displaying text on a single unit.

!This first example is to be sent to group 3, unit 123

^A030123Testing ASCII mode

This will cause the text "Testing ASCII mode" to be displayed at the current cursor position on a unit set up as group 3, unit 123. All other units on the same line will ignore this text. Any further commands or text would only be processed or displayed by this unit until another ^A command is issued to select a different unit. All units will monitor all commands listening for a ^A command, whether they are selected or not. Only selected units will display text or process commands.

!The second example will trigger message #433 to be displayed by the currently selected unit.

^K00433

!The third example will program message #228 with Center, Log Message, and Energize Relay options.

^O00228^Q02^Q03^Q04Message text for msg. 228^Z

APPENDIX A - UNIT SPECIFICATIONS

Certifications	UL listed CSA certified FM Class I Division 2 (Groups A, B, C and D) CSA Class I Division 2 (Groups A, B, C and D)
NEMA Ratings	NEMA 12 (NEMA 4X Optional)
Noise Immunity	NEMA ICS 2-230 Showering Arc Test ANSI C37.90a-1974 (SWC) Surge Withstand Capability Test
Service Power (AC Model)	115/230 VAC (102-132)(194-250)(47-63 Hz) (Jumper Selectable) PMD 400/450/460/470/475/480 - 22 VA
Service Power (DC Model)	24 VDC (21.5-32 VDC) (18 W.)
Fuse Type (AC Model)	115 VAC Operation - 1.0 Amp @ 250 V (factory installed) 230 VAC Operation - 0.5 Amp @ 250 V (customer supplied) 0.177" x 0.58", 2AG, Slo-Blo
Fuse Type (DC Model)	24 VDC Operation - 1.5 Amp @ 250 V (factory installed) 0.177" x 0.58", 2AG Slo-Blo
Operating Temperature	0° C to +60°C Ambient (+32° F to +140°F)
Storage Temperature	-40°°C to +95C Ambient (-40°F to +203°F)
Humidity	10-95% RH Non-condensing
Data Log/Real-Time Clock Battery	0.5 AA Lithium 3.6 V
Battery Life	Typically 5 years (minimum 1 1/2 years) OFF continuously
Clock Accuracy	1 minute per month error (maximum)
Memory Message Type	EEPROM (16, 32, 64, 128K)
EEPROM Life	Min. 10,000 changes to a given location
Memory Usage	Approximately 175 80-character messages/16K Bytes of EEPROM memory
Display Configuration	4 lines of 20 characters

APPENDIX A - UNIT SPECIFICATIONS

Display Type	5 x 7 dot matrix vacuum fluorescent (blue filtered)
Character Height	9.1 mm
Viewing Distance	18 feet (5.5 meters) approximately
Character Set	All standard ASCII upper/lower case and symbols
Overall Dimensions	14.370" wide x 6.180" high x 4.800 deep (NEMA 12)
PMD 400 panel space	14.870" wide x 6.680" high x 4.800 deep (NEMA 4X)
PMD 400 Housing Weight	7 pounds, 10 ounces
PMD Mounting	Panel mount with built-in mounting studs
Power Input Terminal Block	Wire-Clamp screws for 12-18 AWG
Serial Ports, Relay, and Control Terminal Blocks	Wire-Clamp screws for 18-22 AWG
Relay	Form C (3 Amps @ 250 VAC max. or 30 VDC max)
450 A-B PLC connector	Plug-In, 7-position terminal block
460 TI PLC connector	9-position female "D"-style connector
470 Modicon PLC connector	Female coaxial connector
475 Modicon MODBUS PLUS	9-position female "D"-style connector
480 GE Genius I/O connector	Plug-In, 8-position terminal block
480 Hand-held monitor connector	9-position male "D"-style connector
Built in Supply (PMD 400 only)	12 VDC, 350 mA max. (AC models only)
Programming Port	Baud Rate (300, 600, 1200, 2400, 4800, 9600) Parity (None, Odd, Even) Stop Bits (1, 2) Data Bits (8)
Printer Port	Baud Rate (300, 600, 1200, 2400, 4800, 9600) Parity (None, Odd, Even) Stop Bits (1, 2) Data Bits (8)

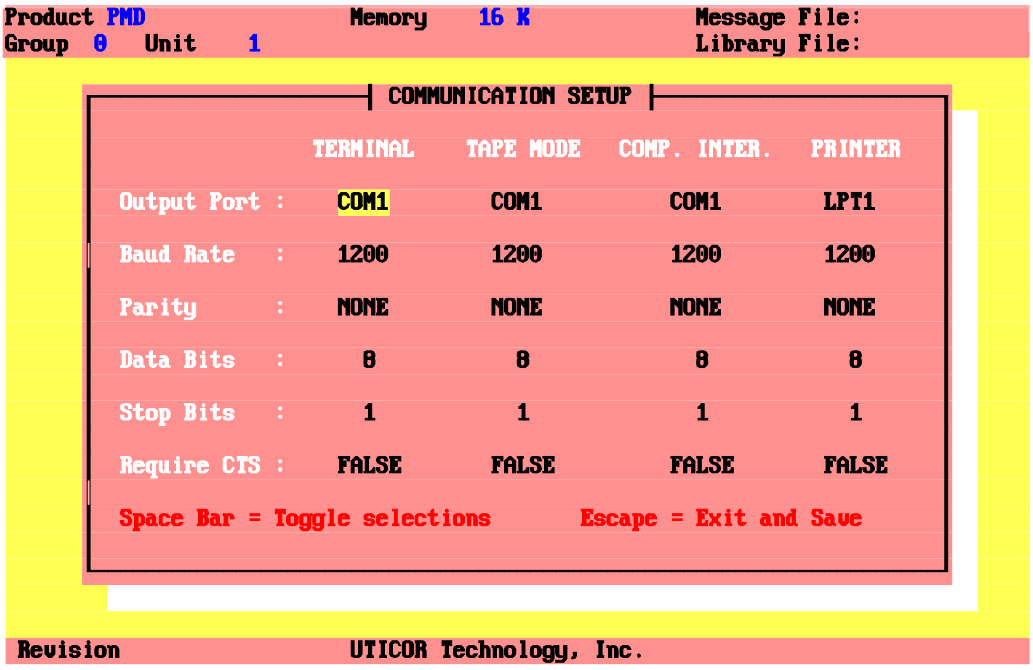
APPENDIX A - UNIT SPECIFICATIONS

Tape Port	Baud Rate (300, 600, 1200, 2400, 4800, 9600) Parity (None) Stop Bits (1) Data Bits (8) Note: The same connector is used for Programming, Printer and Tape port.
Computer Interface Port	Baud Rate (300, 600, 1200, 2400, 4800, 9600) Parity (None, Odd, Even) Stop Bits (1, 2) Data Bits (8) Checksum (None, CRC, EOR) Protocol (UTICOR, ASCII)
Slave Port	Baud Rate (9600) Parity (None) Stop Bits (1) Data Bits (8) Protocol (UTICOR)

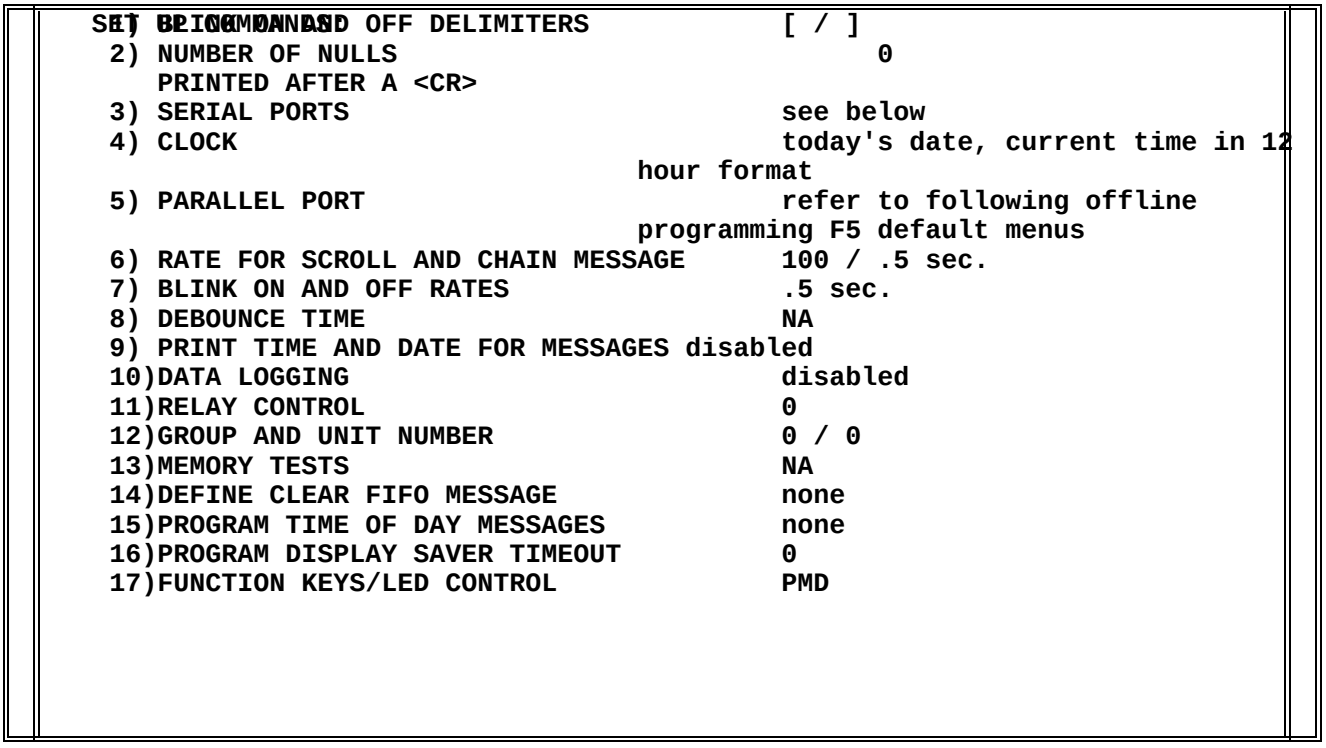
APPENDIX A - UNIT SPECIFICATIONS

APPENDIX B - DEFAULT SETTINGS

ONLINE PROGRAMMING SETUP:



ONLINE SET UP COMMANDS MENU:


SET UP COMMANDS
1) BRACKETS FOR ON AND OFF DELIMITERS [/]
2) NUMBER OF NULLS PRINTED AFTER A <CR> 0
3) SERIAL PORTS see below
4) CLOCK today's date, current time in 12 hour format
5) PARALLEL PORT refer to following offline programming F5 default menus
6) RATE FOR SCROLL AND CHAIN MESSAGE 100 / .5 sec.
7) BLINK ON AND OFF RATES .5 sec.
8) DEBOUNCE TIME NA
9) PRINT TIME AND DATE FOR MESSAGES disabled
10) DATA LOGGING disabled
11) RELAY CONTROL 0
12) GROUP AND UNIT NUMBER 0 / 0
13) MEMORY TESTS NA
14) DEFINE CLEAR FIFO MESSAGE none
15) PROGRAM TIME OF DAY MESSAGES none
16) PROGRAM DISPLAY SAVER TIMEOUT 0
17) FUNCTION KEYS/LED CONTROL PMD

APPENDIX B - DEFAULT SETTINGS

Serial Ports:

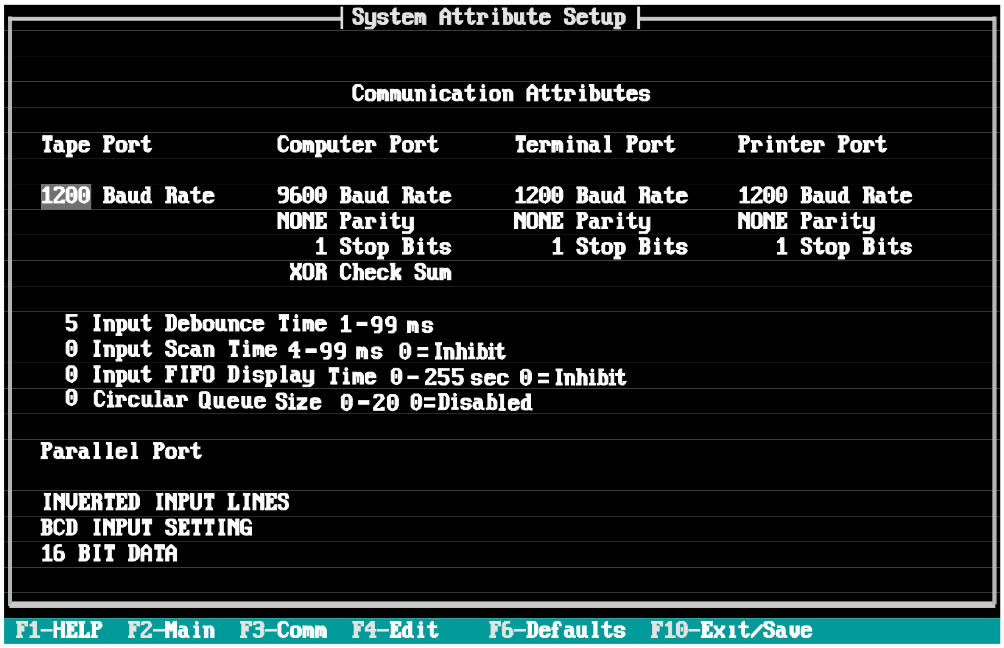
Computer Interface	PMD 9600 baud no parity 1 stop bit parity disabled XOR checksum
Program terminal	1200 baud 1 stop bit no parity
Printer	1200 baud 1 stop bit no parity
Tape in and out	1200 baud

ONLINE OPTION MENU:

```
                                OPTION MENU

1. CHAIN MESSAGE              <N>   15. CLEARS DISPLAY            <Y>
2. REPEAT CHAIN MESSAGE             <N>   16. CENTER MESSAGE            <N>
                                          17. STARTING DATA SET         <1>
3. SCROLLS UPWARD                   <N>
4.   BOTTOM LINE              <1>   18. SEND TO SLAVE       <N>
5.   TOP LINE                       <1>   19.   TO ALL SLAVES           <Y>
6.   REPEAT                         <N>   20.   GROUP               <0>
7.   SCROLL TIME INTERVAL     <0>   21.   UNIT                <0>

8. SCROLLS LEFT               <N>   22. ALARM RELAY         <N>
9.   SCROLL LINE              <1>   23. INVISIBLE MESSAGE   <N>
10.  REPEATS                        <N>   24. LOG MESSAGE              <N>
11.  SCROLL TIME INTERVAL     <0>   25. BLINK ENTIRE MSG    <N>

12.SEND MSG TO RS232 PORT     <N>
13.  ON ONE LINE              <N>   26. DS PROTECT N N N N
14.  <CR><LF> AT END OF MSG   <N>   98. Help
                                    99. Exit

Enter Selection:
```

APPENDIX B - DEFAULT SETTINGS

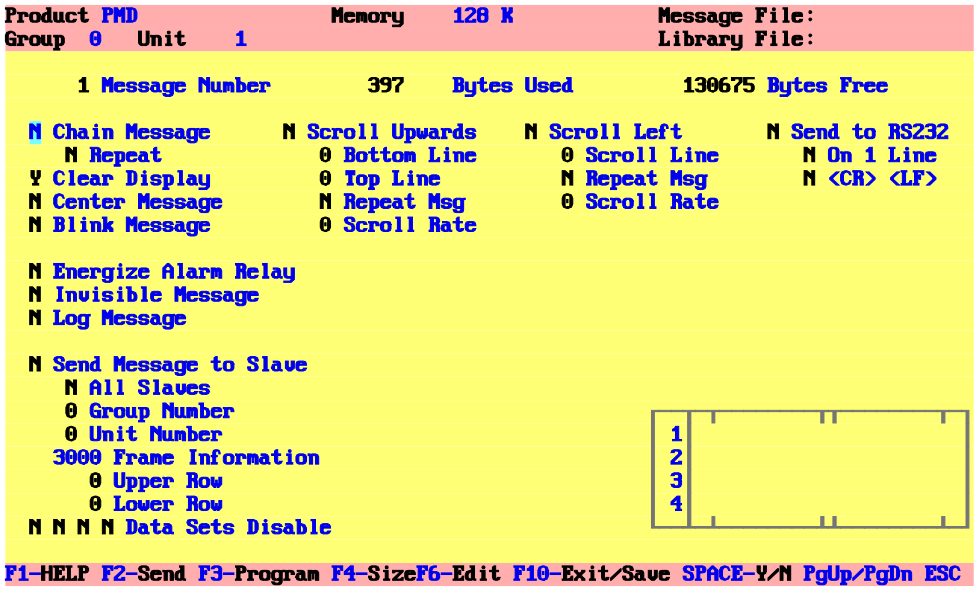
OFFLINE PROGRAMMING COMMUNICATION ATTRIBUTES: Note: This is the default screen for the PMD 400. For the 450, 460, 470, 475 and 480, the difference is that the default **INPUT LINES** are **NON-INVERTED**.



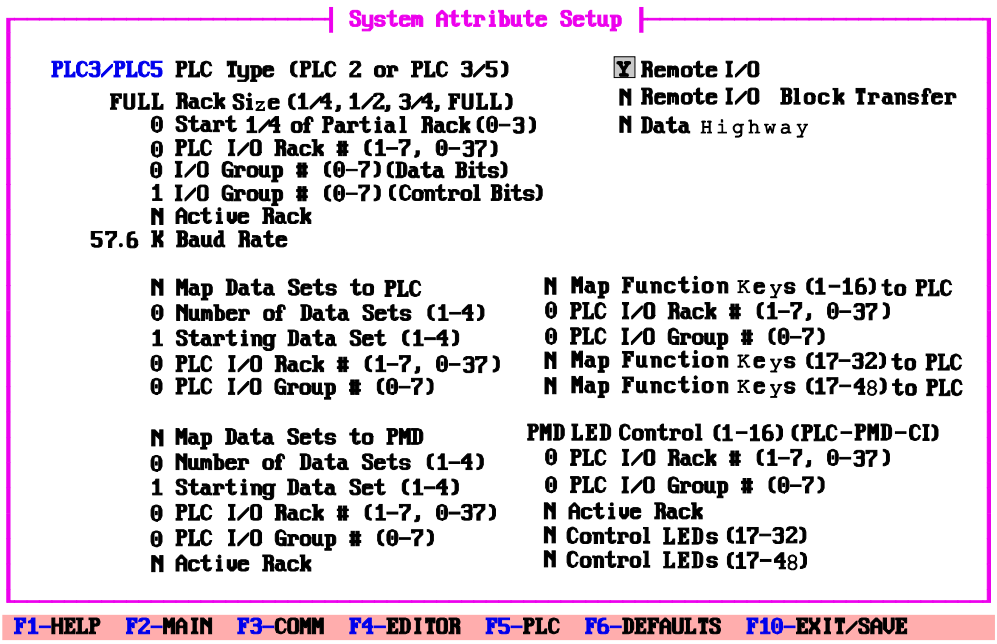
OFFLINE SYSTEM ATTRIBUTE SET UP/MAIN ATTRIBUTE SET UP MENU:

APPENDIX B - DEFAULT SETTINGS

OFFLINE ADD/EDIT A MESSAGE:

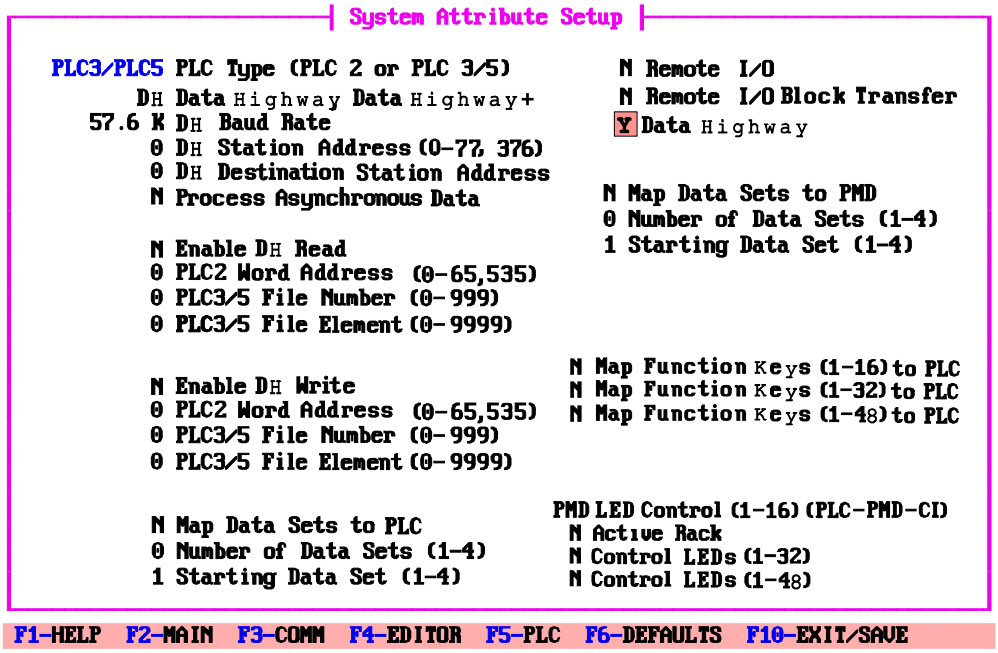


F5- PLC ALLEN-BRADLEY REMOTE I/O:

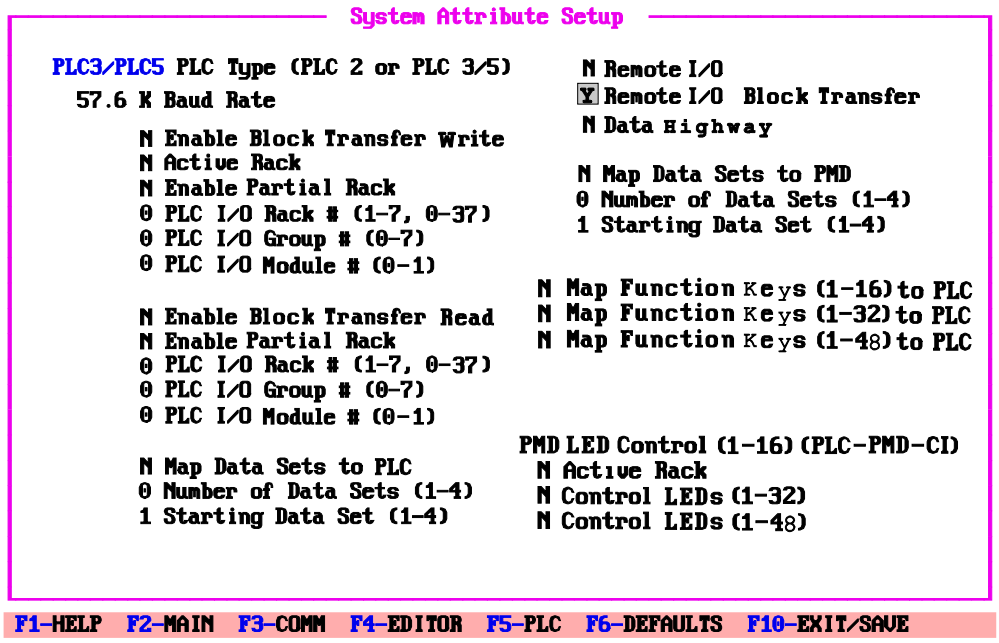

System Attribute Setup
PLC3/PLC5 PLC Type (PLC 2 or PLC 3/5)
FULL Rack Size (1/4, 1/2, 3/4, FULL)
0 Start 1/4 of Partial Rack (0-3)
0 PLC I/O Rack # (1-7, 0-37)
0 I/O Group # (0-7) (Data Bits)
1 I/O Group # (0-7) (Control Bits)
N Active Rack
57.6 K Baud Rate
N Map Data Sets to PLC
0 Number of Data Sets (1-4)
1 Starting Data Set (1-4)
0 PLC I/O Rack # (1-7, 0-37)
0 PLC I/O Group # (0-7)
N Map Data Sets to PMD
0 Number of Data Sets (1-4)
1 Starting Data Set (1-4)
0 PLC I/O Rack # (1-7, 0-37)
0 PLC I/O Group # (0-7)
N Active Rack
Y Remote I/O
N Remote I/O Block Transfer
N Data Highway
N Map Function Keys (1-16) to PLC
0 PLC I/O Rack # (1-7, 0-37)
0 PLC I/O Group # (0-7)
N Map Function Keys (17-32) to PLC
N Map Function Keys (17-48) to PLC
PMD LED Control (1-16) (PLC-PMD-CI)
0 PLC I/O Rack # (1-7, 0-37)
0 PLC I/O Group # (0-7)
N Active Rack
N Control LEDs (17-32)
N Control LEDs (17-48)
F1-HELP F2-MAIN F3-COMM F4-EDITOR F5-PLC F6-DEFAULTS F10-EXIT/SAVE

APPENDIX B - DEFAULT SETTINGS

F5- PLC ALLEN-BRADLEY DATA HIGHWAY/+ :

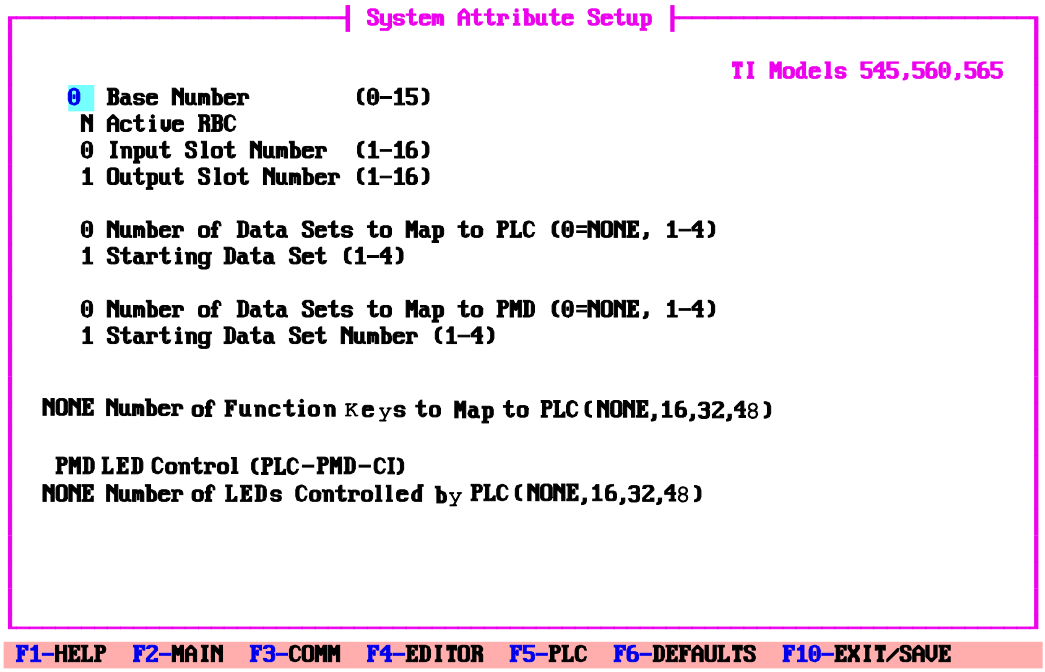


F5 - PLC ALLEN-BRADLEY BLOCK TRANSFER:

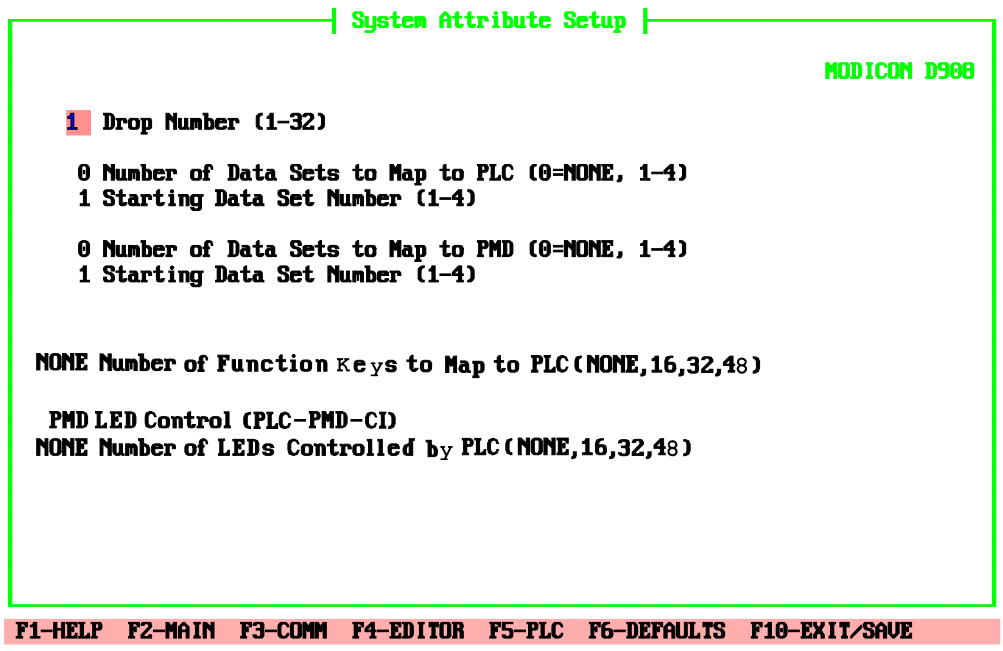


APPENDIX B - DEFAULT SETTINGS

F5 - PLC SIEMENS/TI:

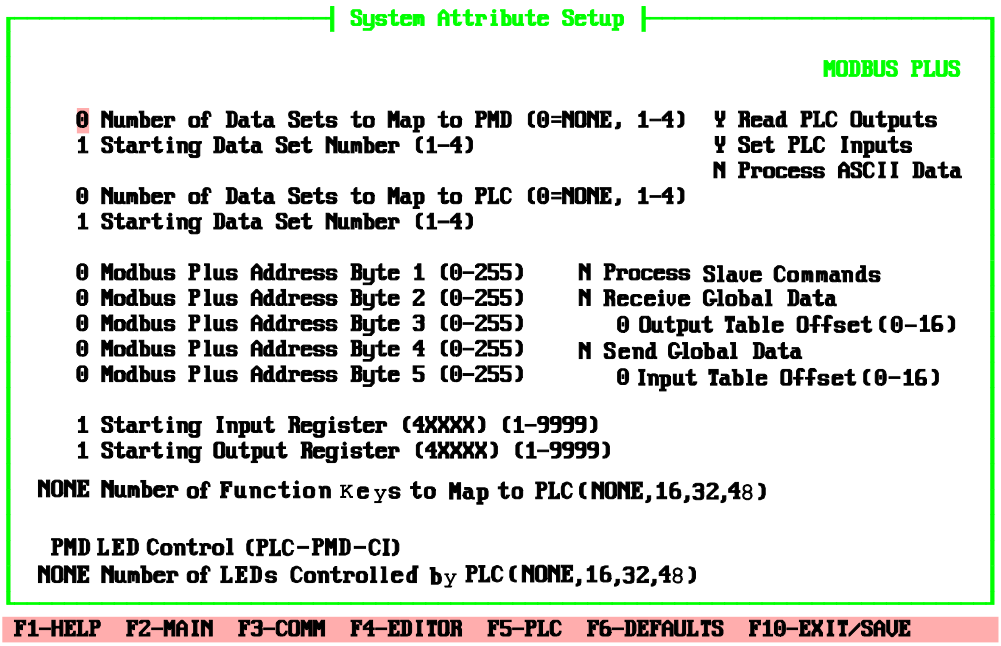


F5 - PLC MODICON:

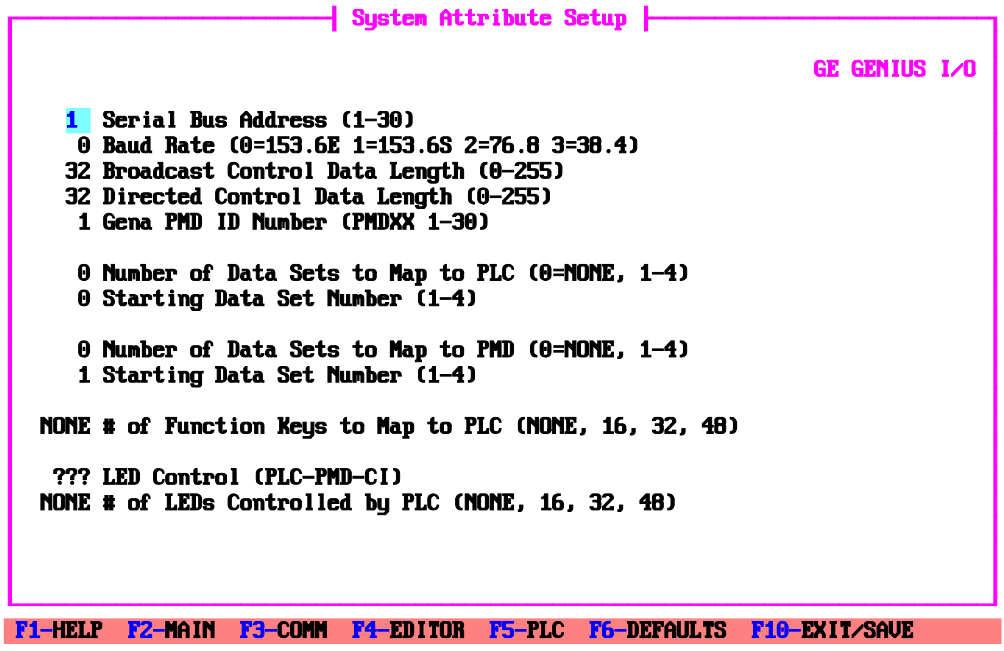


APPENDIX B - DEFAULT SETTINGS

F5 - PLC MODICON MODBUS PLUS:

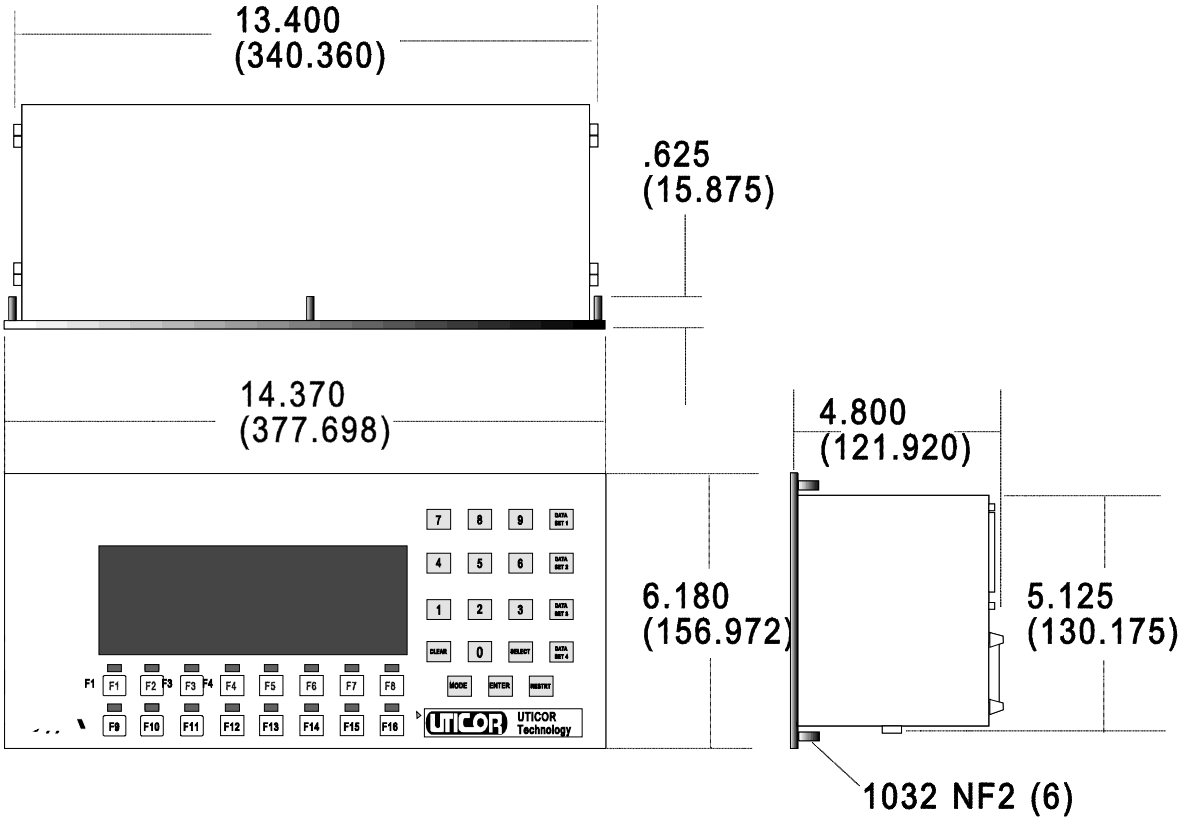


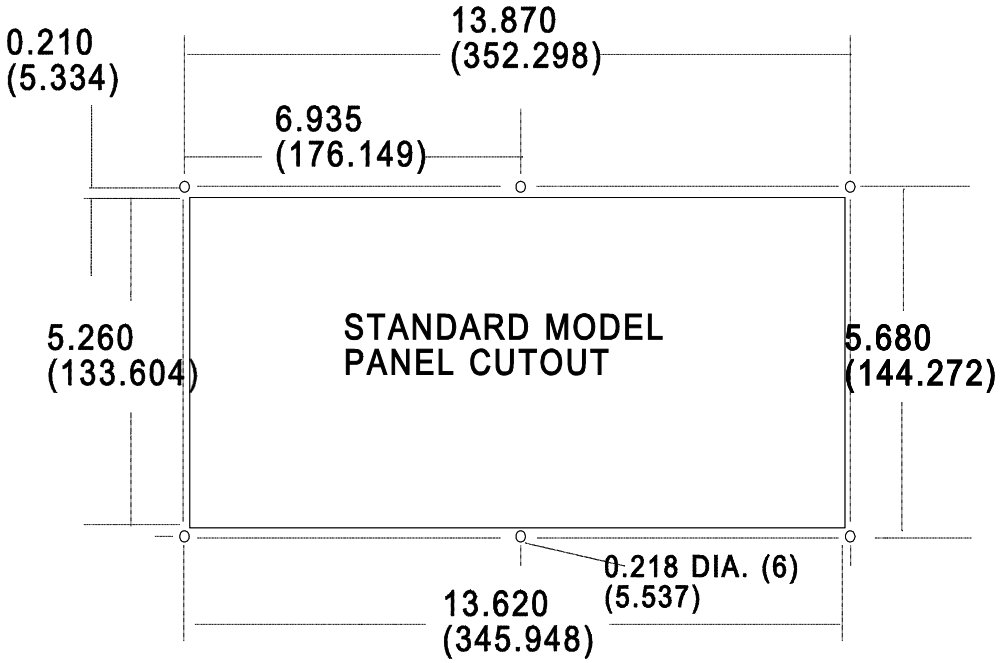
F5 - PLC GE GENIUS I/O:


System Attribute Setup
GE GENIUS I/O
1 Serial Bus Address (1-30)
0 Baud Rate (0=153.6E 1=153.6S 2=76.8 3=38.4)
32 Broadcast Control Data Length (0-255)
32 Directed Control Data Length (0-255)
1 Gena PMD ID Number (PMDXX 1-30)
0 Number of Data Sets to Map to PLC (0=NONE, 1-4)
0 Starting Data Set Number (1-4)
0 Number of Data Sets to Map to PMD (0=NONE, 1-4)
1 Starting Data Set Number (1-4)
NONE # of Function Keys to Map to PLC (NONE, 16, 32, 48)
??? LED Control (PLC-PMD-CI)
NONE # of LEDs Controlled by PLC (NONE, 16, 32, 48)
F1-HELP F2-MAIN F3-COMM F4-EDITOR F5-PLC F6-DEFAULTS F10-EXIT/SAVE

APPENDIX B - DEFAULT SETTINGS

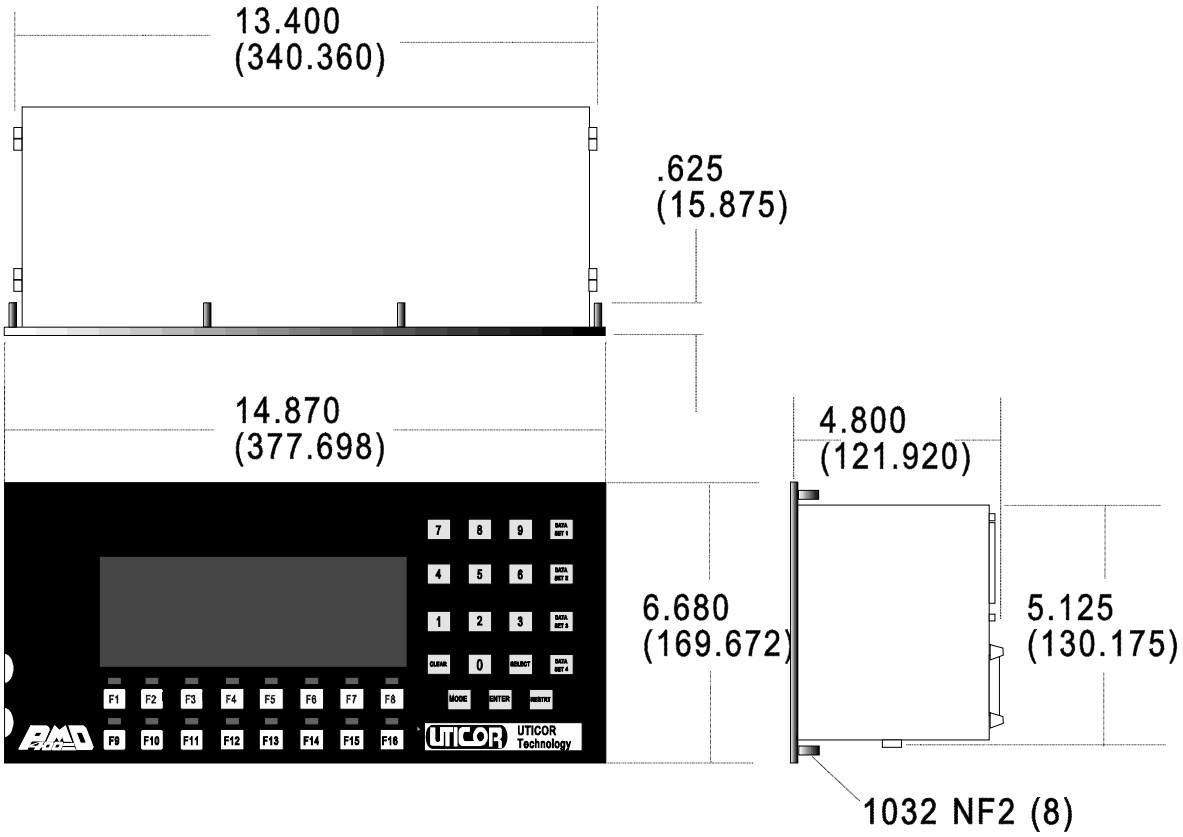
PMD 400 SERIES MODEL (NEMA 12)

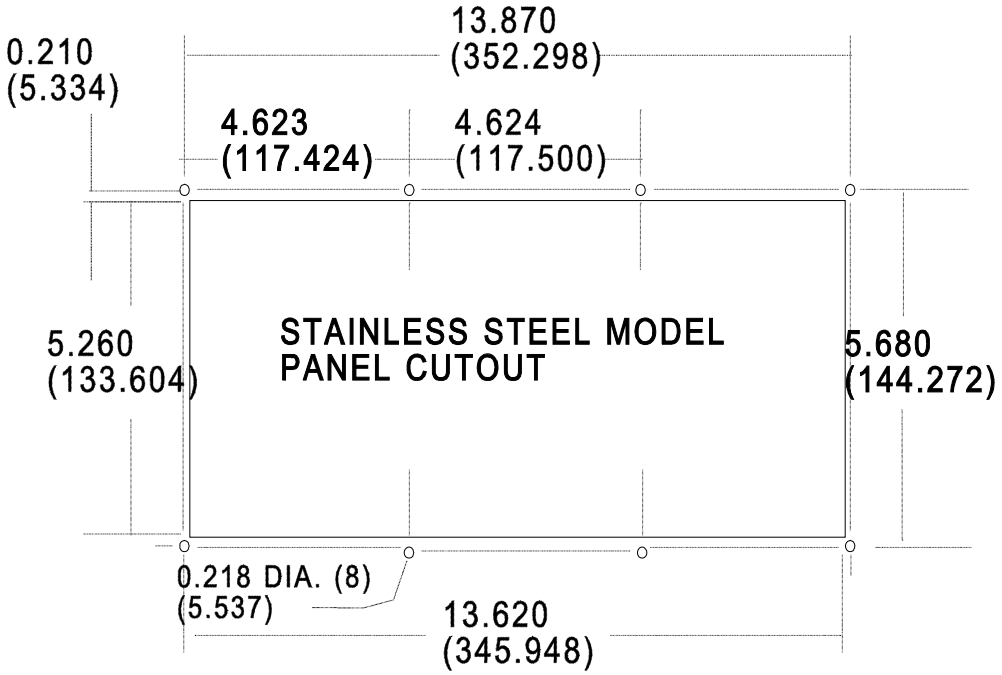




APPENDIX C - OUTLINE AND MOUNTING DIMENSIONS

PMD 400 SERIES STAINLESS STEEL MODEL (NEMA 4X)





APPENDIX D - ACCESSORIES

The accessories for the 400 Series products include:

*Slave displays
*Remote push button station
*NEMA enclosures
*PMD 400E expander modules

Slave Displays

The 400 Series products are compatible with all PMD slave displays. These slaves are available in several sizes. Character size and number differ from model to model.

PMD 150S - .2" high vacuum-fluorescent characters, one or two lines of 20 characters, compact, panel-mount display

PMD 180S - .18" high vacuum-fluorescent characters, one or two lines of 40 characters, compact, panel-mount display

PMD 200S - .43" high vacuum-fluorescent characters, two lines of 20 characters, panel-mount display

PMD 300S - .5" high vacuum-fluorescent characters, four lines of 20 characters, panel-mount display

PMD 1100 - 2" red LED stackable display that can be wall-mounted or suspended

PMD 1200 - 4" red LED stackable display that can be wall-mounted or suspended

PMD 1400 - 4" red LED numeric characters, available in four- or six-character displays, wall-mounted or suspended

PMD 3000 - 2", 4", 6", 8", and 8" narrow red LED characters, available in several size configurations

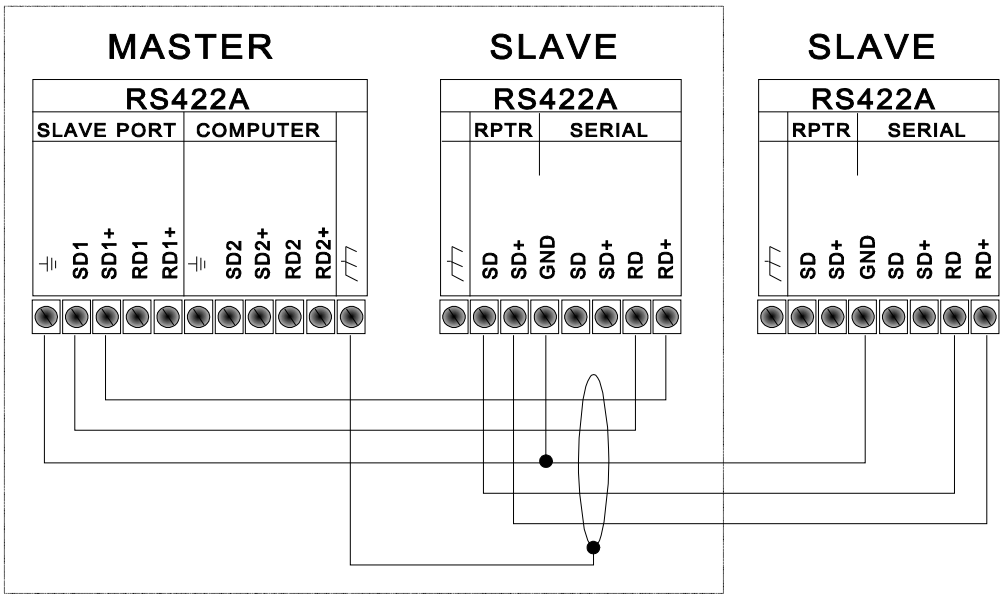
Slave displays provide a cost-effective means to network messages to various locations. Although not originating or storing messages, the slave unit displays messages that are sent from the PMD master unit, a computer, or other intelligent device. Up to 15 groups of up to 511 masters and slaves can be connected together into one system and individually addressed.

Multi-drop wiring requires signal boosting after every ten slaves and for slaves located farther than 4000 feet (1200 meters) from the controlling device as well.

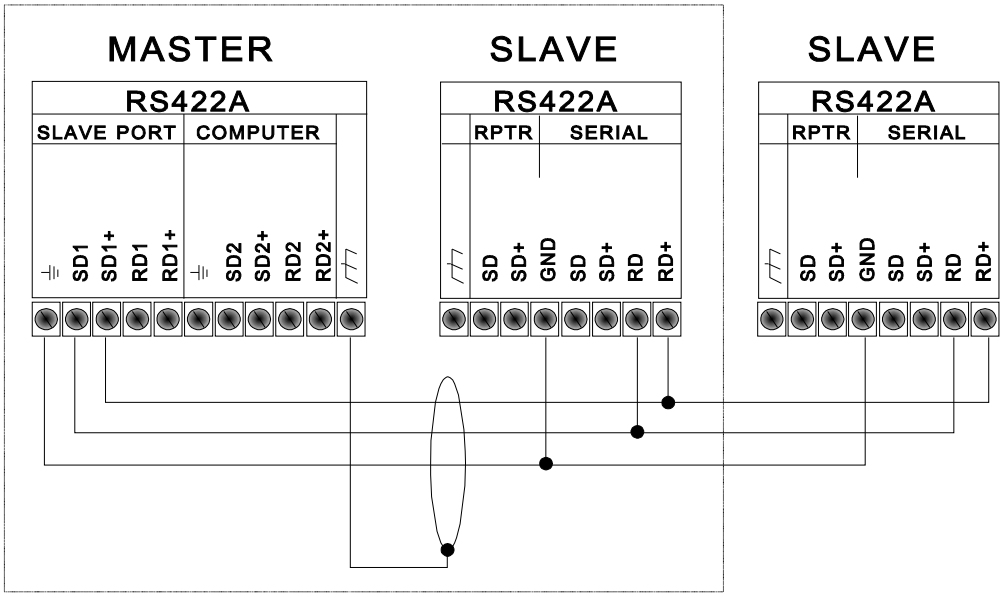
APPENDIX D - ACCESSORIES

Daisy-chain wiring provides signal boosting through the repeater circuits of each slave, and a 4000 foot distance between each device is possible. (No provision is made for boosting the reply channel, however).

DAISYCHAIN CONNECTION



MULTIDROP CONNECTION



APPENDIX D - ACCESSORIES

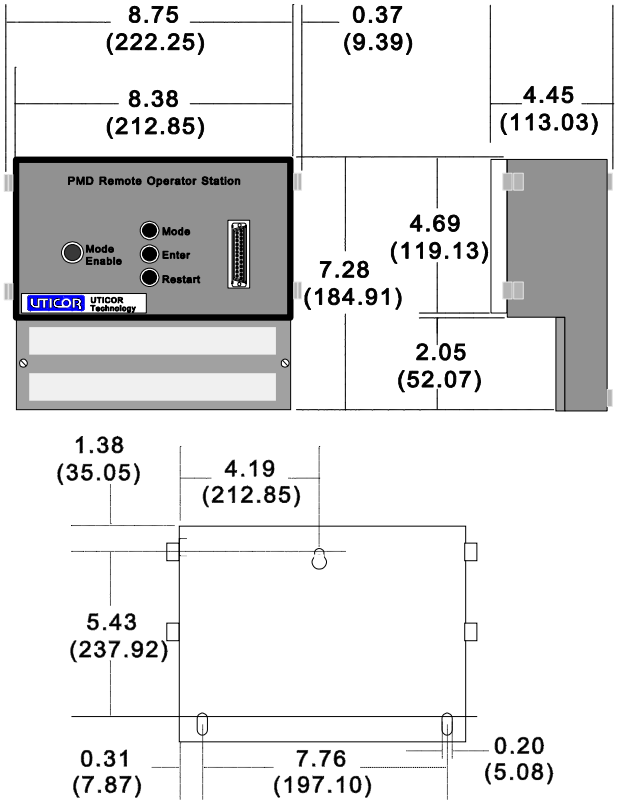
All RS-422 connections should be made with cable of similar or superior specifications and characteristics to those specified for *Belden cable number 9730.

An RS-422 "link" consists of a two-wire transmitting line, a two-wire receiving line (optional), signal common and the shield which must be terminated to safety ground. Each two-wire line should physically be implemented with the two wires of one of the twisted pairs in the cable. (The cable specified for RS-422 connections consists of three twisted pairs). Each twisted pair is individually shielded, and each shielded is brought out to a drain wire. DO NOT USE WIRES FROM DIFFERENT TWISTED PAIRS TO MAKE UP A TWO-WIRE SIGNAL LINE.

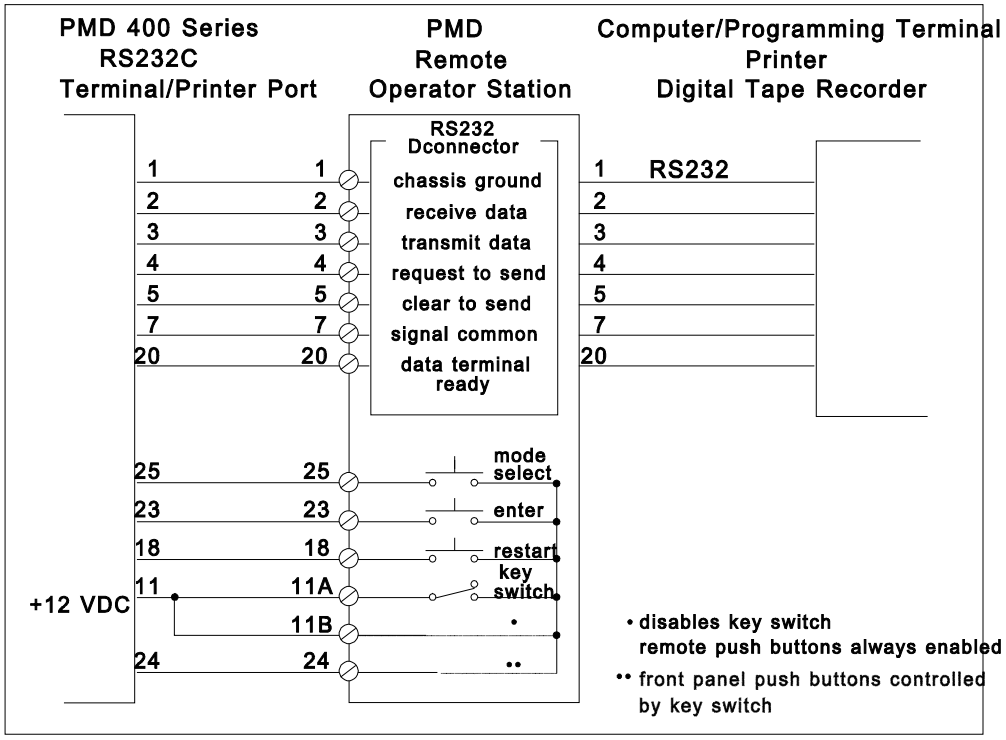
Remote Operator Station

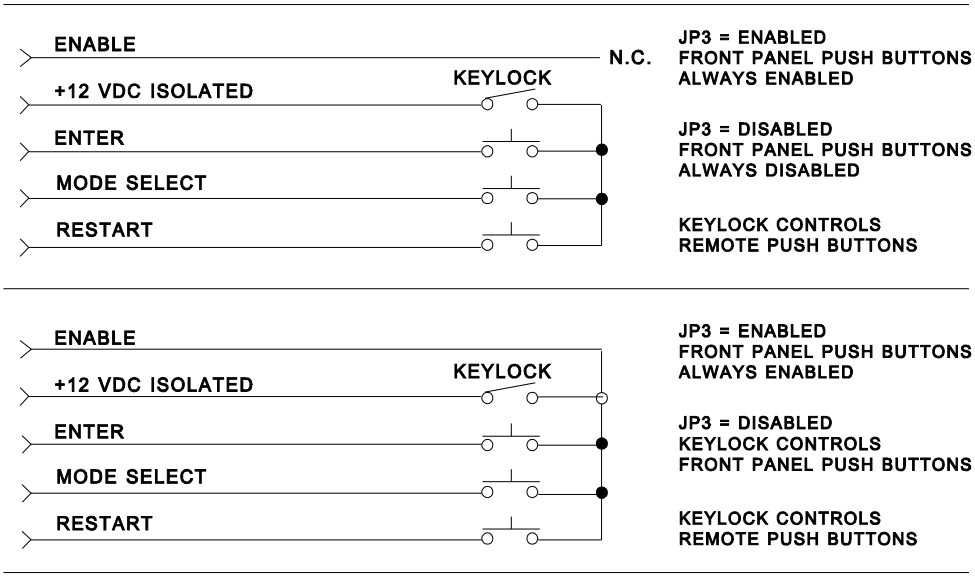
Optionally, the 400 displays can be operated via a remote operator station (Part Number 58263), shown at right. This remote station is used to activate the switch inputs and provide a convenient connection for devices that are used on the *Terminal/Printer Port*.

Wiring connections can be reconfigured at the station end. Please note that the Enable line is for the MODE, ENTER and RESTART push buttons on the front of the 400 units. The illustrations on the following pages show other possible configuration for the Enable line and the keylock and how they will operate in relation to jumper JP3 found in the 400 units. (Jumper JP3 is shown in **APPENDIX E - INTERNAL ADJUSTMENTS**.

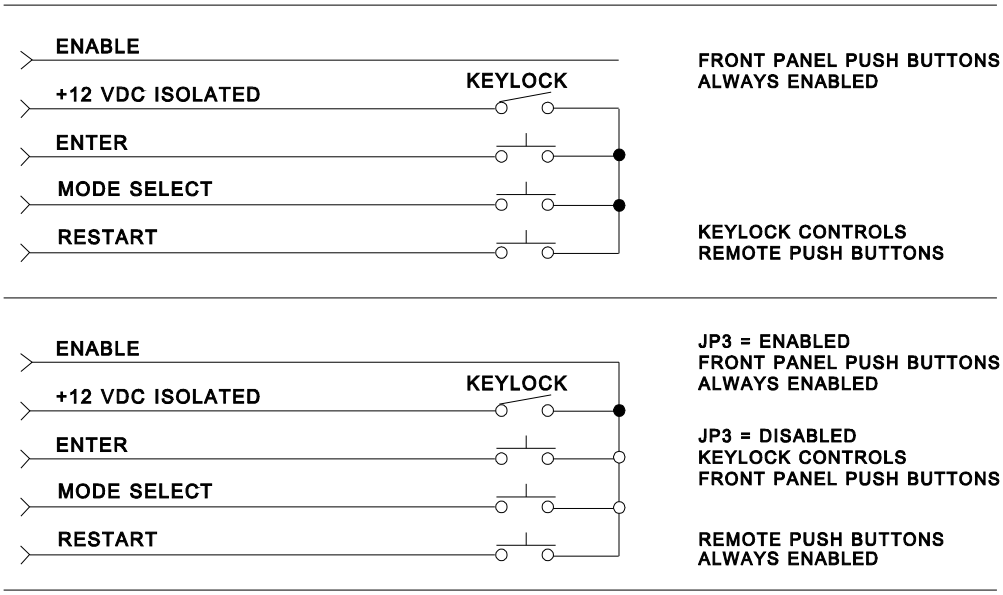


APPENDIX D - ACCESSORIES




ENABLE
+12 VDC ISOLATED
ENTER
MODE SELECT
RESTART
KEYLOCK
N.C.
JP3 = ENABLED
FRONT PANEL PUSH BUTTONS
ALWAYS ENABLED
JP3 = DISABLED
FRONT PANEL PUSH BUTTONS
ALWAYS DISABLED
KEYLOCK CONTROLS
REMOTE PUSH BUTTONS
ENABLE
+12 VDC ISOLATED
ENTER
MODE SELECT
RESTART
KEYLOCK
JP3 = ENABLED
FRONT PANEL PUSH BUTTONS
ALWAYS ENABLED
JP3 = DISABLED
KEYLOCK CONTROLS
FRONT PANEL PUSH BUTTONS
KEYLOCK CONTROLS
REMOTE PUSH BUTTONS

APPENDIX D - ACCESSORIES



You may choose to build your own station which should consist of:

*RS-232 cable from the RS-232 Terminal/Printer Port to the Remote Station

*Three push buttons for the switch input (MODE, ENTER, RESTART)

*Keylock device to enable/disable the switch input (optional)

*RS-232 cable from the Remote Station to the interface devices (printer, terminal etc.)

NEMA Enclosures

The 400 enclosures are available to provide protection for virtually any industrial environment. These enclosures are offered in different sizes to accommodate PMD adapters and/or push buttons or switches. Up to eight full-size or miniature oil-tight push buttons or switches can be mounted below the 400 on larger enclosures.

Standard construction of the enclosures is NEMA 12, with optional NEMA 4 or NEMA 4X construction available. Each enclosure consists of an extra-deep fully-gasketed lid that is continuously hinged on the right side with one or more link-lock type fasteners to secure the lid to the base. All enclosures are equipped with a rear-mounted sub-panel for mounting other components.

APPENDIX D - ACCESSORIES

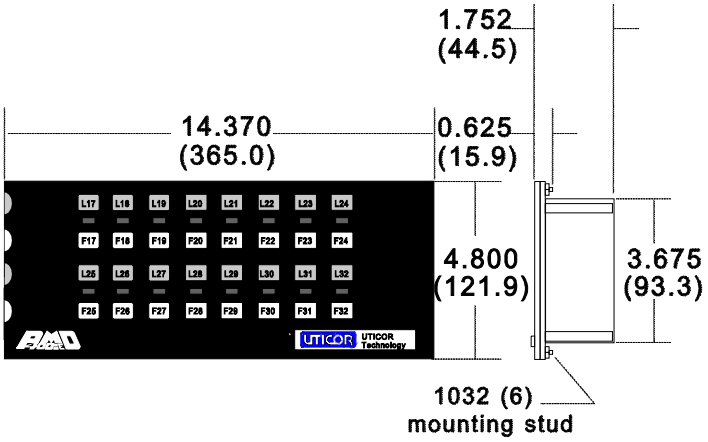
Optional removable mounting flanges are also available. Explosion-proof and dust-tight enclosures are also available. These enclosures come standardly as a NEMA 7/NEMA 9 for both Class I, Group D and Class II, Groups E, F and G. As an option, these enclosures can be suitable for NEMA 3 and 4 locations. Contact UTICOR for a product bulletin of display enclosures.

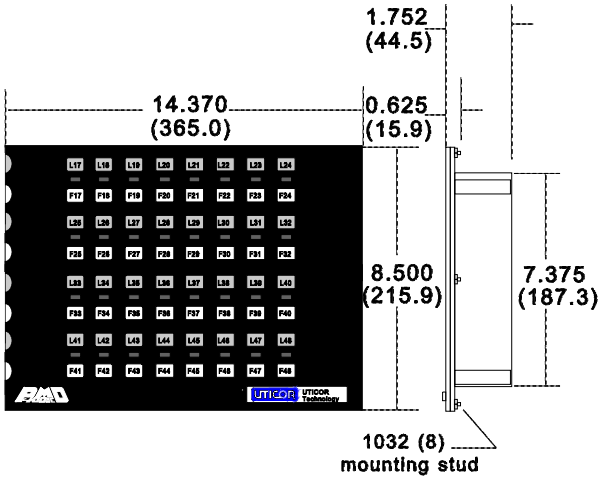
400E Expander Modules

UTICOR has developed two different modules that give the operator of a PMD 400, 450, 460, 470 or 475 additional ability to respond to messages viewed on the message display. One of the PMD 400Es has 16 function keys with associated LEDs. The other 400E has 32 function keys with associated LEDs. The function keys can be mapped to the *PLC port* or to the unit itself. They will always be mapped to the *Computer Interface Port*.

The LEDs can be used as pilot light indicators and are controlled either by commands through the *Computer Interface Port*, the PLC, or they can be lit when their associated key is pressed.

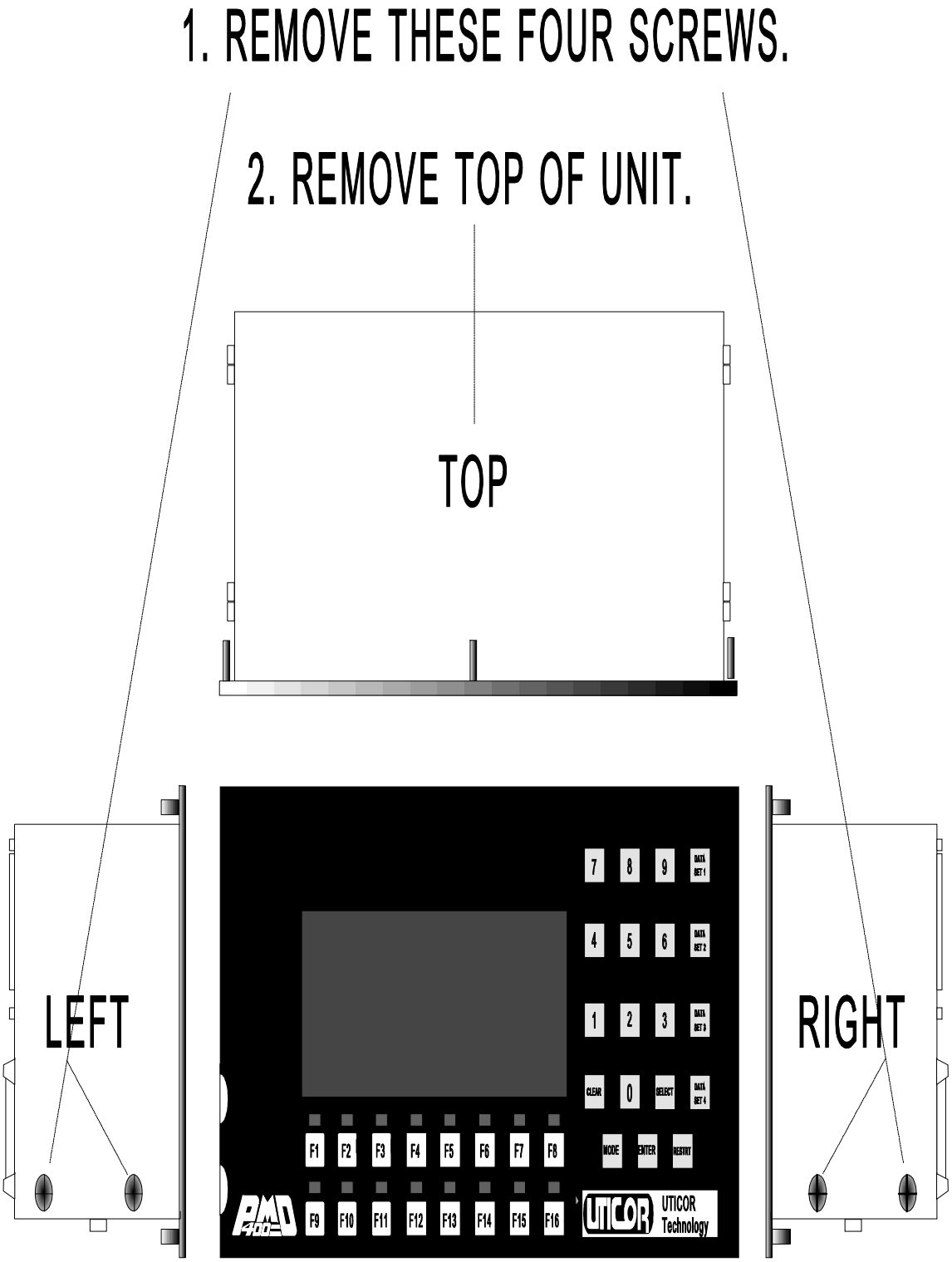
The legends for the function keys and LEDs can be customized. This is done by sliding the legend out the side of the unit, flipping it over, and writing a brief description of the function or indication. Replacement legends are also available from your local distributor.





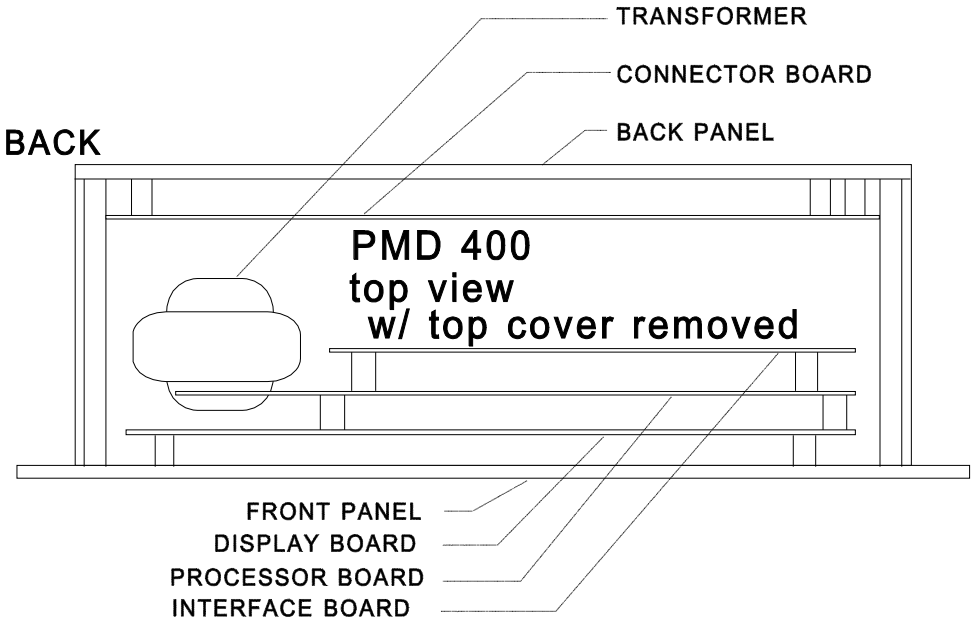
APPENDIX E - INTERNAL ADJUSTMENTS

REMOVING THE TOP COVER



APPENDIX E - INTERNAL ADJUSTMENTS

PC BOARD LOCATIONS: This drawing shows the PC board and transformer locations in the 400 units. These components will be referred to in the remainder of this appendix.



INPUT POWER To operate the 115/230 VAC 400 unit at 115 VAC:

REQUIREMENTS

1. The power input connector (from transformer to the connector board) must be plugged into the 115 VAC connector.

2. The input fuse must be a 0.177" x 0.57", 2AG, 1.0 Amp Slo-Blo 250 V (factory installed).

Units are shipped from the factory set for 115 VAC operation.

To operate the 115/230 VAC 400 unit at 230 VAC:

1. The power input connector (from transformer to the connector board) must be plugged into the 230 VAC connector.

2. The input fuse must be a 0.177" x 0.57", 2AG, 0.5 Amp Slo-Blo 250 V (customer supplied).

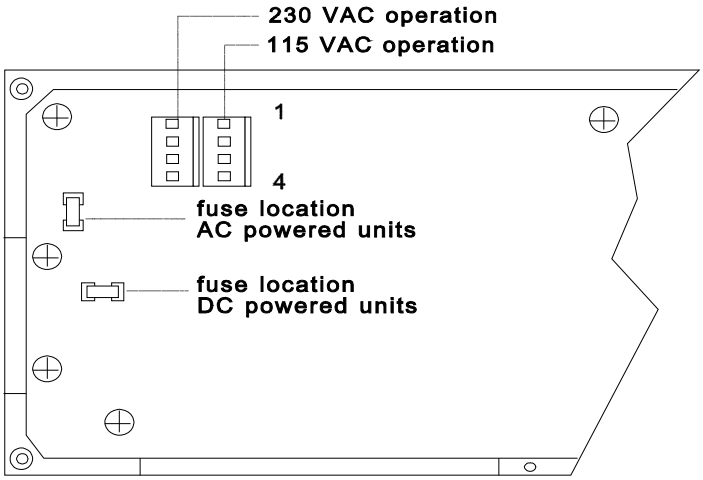
APPENDIX E - INTERNAL ADJUSTMENTS

The power input connectors and the fuse are located on the "connector board" which is mounted to the inside of the back panel. To access the connector board, remove the top cover of the unit and locate the connectors and fuse as shown below to make the necessary adjustments.

WARNING!!! DISCONNECT AC POWER FROM THE UNIT BEFORE CHANGING THE FUSE!

PMD 400 back panel inside view

connector board with power input connector and fuse locations

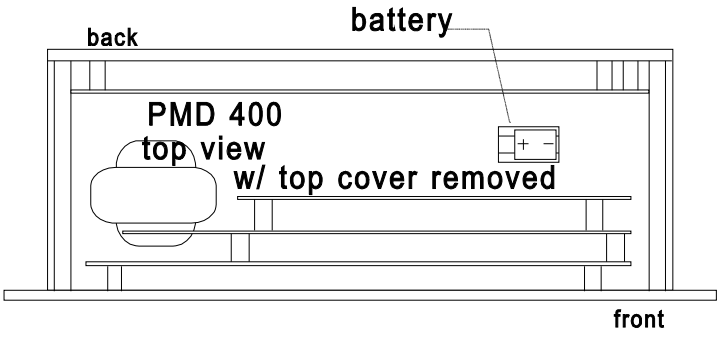


CHANGING THE BATTERY FOR THE DATA LOG AND REAL-TIME CLOCK

The 400 unit contains a lithium battery for the data log and real-time clock. To change the battery, remove the top cover (as shown in the beginning of this appendix) and refer to the drawing below for the battery location.

Use insulated tools to change the battery. Take care not to short the battery terminals to each other or to any part of the circuitry.

BATTERY WARNING!!! FIRE, EXPLOSION, AND SEVERE BURN HAZARD! DO NOT RECHARGE, DISASSEMBLE, HEAT ABOVE 212^0 F, INCINERATE, OR EXPOSE CONTENTS TO WATER.



APPENDIX E - INTERNAL ADJUSTMENTS

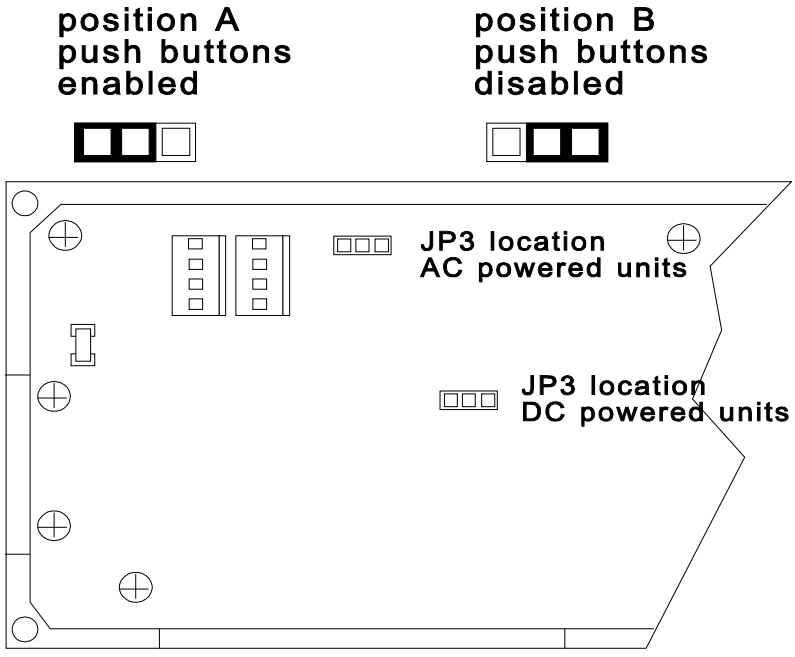
PUSH BUTTON ENABLE/DISABLE JUMPER SETTINGS

The 400 Series displays have an internal jumper JP3 that disables and enables the operation of the three front panel push buttons (MODE, ENTER, RESTART). When JP3 is in the disable position, these push buttons are either inoperable (disabled) or are controlled via the Enable line on the switch inputs of the RS-232 *Terminal/Printer Port*.

The display is shipped from the factory with the push buttons enabled. To disable the push buttons or have them controlled via the RS-232 port, remove the top cover of the display, locate JP3 on the Connector Board as shown below, and move JP3 to the desired location.

PMD 400 back panel inside view

connector board with jumper JP3 location and settings



APPENDIX F - ERROR MESSAGES

Please refer to **SECTION 7 - PLC INTERFACING, PART F: REFERENCE** for direct PLC interfacing errors.

Programming Errors

Blink on and blink off delimiters cannot be the same characters - When changing blink on and off delimiters: The definition of the two characters must be different, printable ASCII characters.

Not programmed - When editing a message: This message number was never programmed or it was deleted and never replaced.
When viewing a message: This message number is larger than the largest programmed message in the unit or the message was deleted and not replaced.

Message already exists - When adding a message: This message number is already programmed.

Source message not programmed - When copying messages: You are trying to copy a non-existent message.

xxx message(s) in copy range - When copying message: Messages already exist within the specified range that you are copying your message to. No messages will be copied when this condition exists.

Insufficient memory x messages copied - When copying messages: Not enough message program memory remains to execute the copy command you entered. The 400 unit copies as many messages as possible, beginning at the starting range for the copy command.

Escape entered x messages copied - When copying message: You pressed Esc to abort the copy command. A partial copy was already executed.

Display Errors

Not programmed - When selected in DISPLAY MODE: This message was deleted and not replaced or was never programmed.

"Tape" Errors

Invalid checksum - "Tape in": The unit has detected a checksum error on a line of the program on the tape. Retry procedure.

Did not verify - "Tape verify": Some of the message on the tape did not match the program in the unit.

Incompatible type - "Tape in": The program on the tape was made from a display whose software revision is not compatible with the software of the unit currently being loaded.

APPENDIX F - ERROR MESSAGES

Line of data lost - "Tape in": A line of data was on the tape was skipped. Retry procedure.

Insufficient memory - "Tape in": The memory capacity of the 400 unit is not large enough for the message program.

Invalid message - "Tape in": You have an invalid message length, either too short or too long.

Invalid tape format - "Tape in": The format of the tape does not match the expected format.

Memory failure - "Tape in": EEPROM memory failure (did not program).

APPENDIX G - TROUBLESHOOTING

CANNOT COMMUNICATE WITH THE PMD IN THE TERMINAL MODE:

You should be in the **`Terminal Mode`** in the programming software (10F50) before you enter PROGRAM MODE. After you enter the PROGRAM MODE, you should see the **`MAIN>COMMANDS`** menu on your computer terminal that is generated by the software contained inside the PMD. Typing an H will rewrite the menu on the terminal.

Check:

1. Is the PMD in the PROGRAM MODE?

2. Are you connected to the *Terminal/Printer Port* on the PMD?

3. Are the **`Terminal Mode`** baud rate for the PMD (accessible by the push buttons), and the **`Terminal Mode`** baud rate for the terminal (accessible from the **`MAIN MENU`**, **`Communication Setup`**), set to the same baud rate?

4. Is the RS-232 communication port that you are connected to at the computer the same one you selected in **`Communication Setup`**?

5. Is the cable you are using made to the specifications that listed in **SECTION 2 - GETTING STARTED**?

6. Exit any Memory Resident Programs and/or disconnect from any Network Systems.

IT APPEARS THAT I'M COMMUNICATING THROUGH THE TERMINAL PORT IN THE PROGRAM MODE BUT I ONLY GET ONE LETTER OF THE COMMANDS AT A TIME:

You need to make a cable that is to the specifications in this manual. You are inducing handshaking signals in the communication that is causing the software to malfunction.

CANNOT LOAD A MESSAGE FILE FROM THE COMPUTER INTO THE PMD THROUGH THE PMD'S RS-232 PORT:

As you load messages into the PMD through the PMD's RS-232 port, you should see the information scrolling across the terminal screen. At the same time, the PMD should be counting down from the total number of messages to zero. If you don't see both of these things happening, you don't have proper communication and have not loaded the message file into the PMD.

Check:

1. Is the PMD in the TAPE IN mode?

2. Are you connected to the *Terminal/Printer Port* on the PMD?

3. Are the TAPE MODE baud rate for the PMD (accessible by the push buttons), and the TAPE MODE baud rate for the terminal (accessible from the **`MAIN MENU`**, **`Communication Setup`**), set to the same baud rate?

4. Is the RS-232 communication port that you are connected to at the computer the same one you selected in **`Communication Setup`**?

5. Is the cable you are using made to the specifications that listed in **SECTION 2 - GETTING STARTED**?

6. Exit any Memory Resident Programs and/or disconnect from any Network Systems.

7. You need to set up your PMD to be ready to accept the message file before you enter the **`Upload File to PMD`** option in the programming software.

CAN NOT LOAD A MESSAGE FILE FROM THE PMD INTO THE COMPUTER THROUGH THE PMD'S RS-232 PORT:

As you load messages to the PMD, you should see the information scrolling across the terminal screen. At the same time, the PMD should be counting down from the total number of messages to zero. If you don't see both of these things happening, you don't have communication and have not downloaded the message file to the computer.

Check:

1. Is the PMD in the TAPE OUT mode?

2. Are you connected to the *Terminal/Printer Port* on the PMD?

3. Are the TAPE MODE baud rate for the PMD (accessible by the push buttons), and the TAPE MODE baud rate for the terminal (accessible from the **`MAIN MENU`**, **`Communication Setup`**), set to the same baud rate?

4. Is the RS-232 communication port that you are connected to at the computer the same one you selected in **`Communication Setup`**?

5. Is the cable you are using made to the specifications that listed in **SECTION 2 - GETTING STARTED**?

6. Exit any Memory Resident Programs and/or disconnect from any Network Systems.

7. You need to set up your computer to be ready to accept the message file from your PMD before you enter the **`Download File to Disk`** option in the programming software.

TIME OF DAY MESSAGES:

Time of Day Messages must be either programmed online or loaded through the RS-422 *computer port* via the **`File Network`** option on the **`MAIN MENU`** of the programming software. Time of Day messages will not be transferred when you load messages through the *Terminal/Printer Port* on the PMD.

DATA SETS ARE PROGRAMMED INTO THE MESSAGE BUT THE VARIABLE DATA IS NOT BEING TRANSFERRED TO THE MESSAGE:

Check Direct PLC interface units:

1. Programming the data sets online with 10F50, did you use the ^U, ^V, ^W, ^X and ^Y keys with the ^U being the most significant digit and ^Y being the least significant digit? (Pressing the Ctrl key and the letter programs the appropriate digit).

2. Programming the data sets offline with 10F50, did you use the F2, F3, F4, F5, and F6 keys with the F2 Key being the most significant digit and the F6 Key being the least significant digit for Data Set One, the SHIFT Key and F2-F6 for Data Set Two, the ALT Key and F2-F6 for Data Set Three, and the CTRL Key and F2-F6 for Data Set Four?

3. If you programmed the data sets online, did you program the correct Number of Data Sets and the correct **`Starting Data Set`** into the message?

4. Did you program the PMD to map the data sets from the PLC to the PMD?

EVERY TIME THE USER PROGRAM IS LOADED INTO THE PMD, COMMUNICATION BETWEEN A PERIPHERAL DEVICE OR THE PLC IS LOST:

Every time you download messages, the communication attributes in the PMD are updated with what is programmed in the message file. As a result, it is very important that if you program offline, set the system attributes to their appropriate settings before you upload.

APPENDIX G - TROUBLESHOOTING

If you do your programming online, once you get everything programmed correctly and the unit is fully operational, then do a download to disk and all of the correct settings will be saved along with the messages.

MESSAGE 0-15 GET CALLED UP WITH NO PROBLEMS, BUT IF ANY # GREATER THAN 15 IS CALLED UP, A DIFFERENT MESSAGE IS DISPLAYED:

You have your *parallel port* set up opposite of what you are anticipating i.e. Binary instead of BCD or BCD instead of Binary. This will also affect the Remote I/O interface units which do not have *parallel port*. Only the data bits word is interpreted as a BCD or Binary number. Data sets are read as Binary only.

THE PMD IS IN THE DISPLAY MODE BUT NOTHING IS BEING DISPLAYED. Check:

1. Are you calling up an invisible message that clears the display?

2. Are you calling up a message that blanks the display?

3. Has the Display Saver Timeout feature been activated and blanked out the display?

Direct PLC interface units:

1. Have you done an unconditional move statement of the value 9 into the Message Control Word?

2. Do you have the *PLC port* set up as non-inverted? All units with remote I/O interfacing must be set up as non-inverted.

The PMD 400 Series units contain a unique troubleshooting tool which is called **PLC INTERFACE** located under the **TEST FUNCTIONS** option on the PMD. This allows you to view the four control bits and 16 data bits that control what message is to be displayed in the form of a pattern of 1s and 0s representing the Binary or BCD value that you sent to the display.

This option is very useful in verifying that the wiring is correct, the power supply is operational, and that you have the correct *PLC port* setting. The control bits will appear as four digits in the upper left-hand corner and the data bits will be at the bottom of the display. In the upper right-hand side, you will be informed as to how the *PLC port* is configured: source or sink and inverted or non-inverted.

GLOSSARY

AC	Abbreviation for alternating current
ASCII	Abbreviation for American Standard Code for Information Interchange; It represents a code for defining alphanumeric characters.
Adapter	An intermediate device that permits attachment of special accessories or provides special means for mounting; an electrical device that allows one electrical standard to be used with a different one
Alphanumeric	Pertaining to a character set that contains both letters and numerals, and usually other characters such as periods and dollar signs
Amps	Abbreviation for ampere; a unit of electrical current or rate of flow of electrons; One volt across one ohm of resistance causes a current flow of one ampere.
BCD	Abbreviation for Binary Coded Decimal; a system of number representation in which each digit of a decimal number is represented by a four-digit binary number
Baud rate	The speed at which serial data is transmitted and received; rated at bits per second
Bidirectional	Descriptive of a device that can transmit and receive data
Binary	A numbering system in which each digit can take on values from 0 to 1; the base 2 numbering system
Bit	Abbreviation for Binary digit; the smallest unit of information in the Binary numbering system; represented by the digits 0 and 1
Buffer	Electrical: an isolating circuit used in an electronic device to avoid reaction of a driven circuit on the corresponding driver circuit; Software: a memory location that stores transmitted information for later use.
Byte	A single group of bits processed together (in parallel); It can consist of a variable number of bits.
Byte diagram	Illustration that defines how bytes are utilized in message memory
Checksum	A calculation of all data bits being transmitted; is used by the receiving device for checking data integrity
Chain message	Links up to 115 individual messages together and displays at a user-defined rate

GLOSSARY

Circular message queue	A list of messages displayed in the order that they are triggered according to their message display time. The list wraps around from the last message back to the first message and continues to cycle through the queue until the user makes a change.
Computer interface	An electrical communication gateway between a computer and a peripheral piece of equipment
Control characters	ASCII code sequences used to make the display perform special tasks
Controller terminals	These wiring terminals interface to the programmable logic controller (PLC) or other device that operates the unit; terminals used by the controller are the *Parallel Port, the Message Control Terminals,* and *VDC In/Out terminals*
Daisy chain	A serial wiring method where the cable goes into one unit and out to the next unit
Data hold	An enable/disable control line for the *Parallel Port*
Data log	An area of memory set aside for a time- and date-stamped historical record of events in the unit
Data select	Two control inputs on the *Parallel Port* used to select the data set number
DC	Abbreviation for direct current
Debounce time	A time period during which the integrity of a signal's value is verified
Decrement	Gradual decrease in quantity.
Default	The initial state of attributes
Delimiter	A character that distinguishes a string of characters and therefore cannot be a member of the string e.g. Blink delimiter
DIN	Abbreviation for Deutsche Industrie Normenausschuss, the German association that determines the standards for electrical and other equipment in that country; It maintains a European standard which gives very precise cut-out mounting dimensions.
Dip switch	A mechanical switch housed in a dual inline pack used for selecting unit attributes

GLOSSARY

Display mode	The operating mode of the unit; When in this mode, the unit is controlled by a programmable controller or other intelligent device. The unit can also be interfaced to a printer, alarm device, slave message units, or computer while in this mode.
Dump	Sending all or part of the unit memory contents through a serial port
EEPROM	Abbreviation for Electronically Erasable Programmable Read Only Memory; This type of read-only memory can be altered with commands of the programming terminal and does not lose its contents when power is turned off.
EPROM	Abbreviation for Electronically Programmable Read Only Memory; a read only memory in which stored data can be erased by ultraviolet light or other means and reprogrammed bit by bit with appropriate voltage pulses
FIF0	Abbreviation for First-In, First-Out; a list in memory where the first item received is the first item retrieved
Group Number	A four-bit Binary number used as part of the unit identifier
HEX	See hexadecimal
Handshaking	The method used to coordinate sending and receiving data by means of signals which indicate when it is okay to transmit or receive data and when it is not
Hexadecimal	Base sixteen numbering system with digits 0 through F where F is equal to fifteen
IEEE	Abbreviation for Institute of Electrical and Electronics Engineers
Increment	Gradual increase in quantity.
Input devices	Devices such as limit switches, pressure switches, push buttons etc. that supply data to a controller
Interfacing	Interconnecting two pieces of equipment so they can communicate; The interconnection can be physical, electrical or logical.
Invisible messages	Messages sent to printer or slave but not to the master's own display
Isolated bidirectional adapter	Provides two-way interface between devices using RS-232C and RS-422A/RS-485A communication standards

GLOSSARY

LED	Abbreviation for Light-Emitting Diode; a pn junction that emits light when biased in the forward direction
Message attributes	Specific codes and commands which dictate the characteristics of a message e.g. blinking, scrolling etc.
NEMA standards	Abbreviation for National Electrical Manufacturers Association which provides standards for electrical equipment approved by the majority of its members
NEMA 4x	See NEMA standards. 4x enclosures provide a degree of protection against corrosion, windblown dust and rain, splashing water and hose-directed water, and remain undamaged by the formation of ice on the enclosure. They meet hose-down, external icing and corrosion-resistance design tests. UTICOR uses #316 grade stainless steel.
NEMA 12	See NEMA standards; 12 enclosures provide a degree of protection against dust, falling dirt and dripping noncorrosive liquids. They meet drip, dust and rust-resistance design tests.
Noise immunity	The ability of an electronic device to reject interfering noise signals
Null character	An ASCII character that is neither a displayed character nor a command but can be defined by the user; In UTICOR displays, the null character is used as a place holder in the current message when overlaying it on an existing message.
Offline programming	Message programs are developed with UTICOR software and stored to disk. The message file can then be uploaded to the 400 unit. The computer does not need to be connected to the unit for this programming process.
Online programming	The master unit is programmed directly from the computer. The unit must be connected to the programming computer.
PMD	Abbreviation for Programmable Message Display, copyrighted by UTICOR Technology, Inc.
Parallel port	16-input port through which message numbers and data information from a controller or other device is entered into the unit when it is in the display mode
Parameter	A constant or element, the value of which characterizes the behavior of one or more variables associated with a given system

GLOSSARY

Parity	A method of testing the accuracy of Binary numbers used in recorded, transmitted or received data; For even parity, the sum of the ones is an even number; for odd parity, the sum of the ones is an odd number.
Parity bit	An additional bit added to a Binary word to make the sum of the number of "1s" always even or odd
Peripheral equipment	In a data processing system, any unit of equipment, distinct from the central processing unit, which may provide the system with outside communication
Printer port	The place of access on a 400 unit that allows for communication between it and the specified printer
Protocol	A formal set of rules governing the communication between two devices; Some products can use either UTICOR protocol or ASCII protocol.
RAM memory	Abbreviation for Random Access Memory; RAM is a type of memory which requires that power be applied in order to retain the data stored in it.
ROM	Abbreviation for Read Only Memory; A ROM is a solid-state storage memory whose contents cannot be altered by the controller
RS-232	IEEE standard for serial communication with a 50 ft. distance limitation
RS-422	IEEE standard for serial communication with a 4000 ft. distance limitation
Real-time clock	A clock that indicates the passage of actual time
Scrolling message	A message that contains up to 235 characters and moves from the bottom of the display to the top, or from the right of the display to the left
Sink	When using the *Parallel Port* with a sinking module, the positive side of the electrical power supply is used for reference and the negative voltage level is equal to a logic one.
Slave port	The place of access on a master unit for serial connection to a slave display
Slo Blo	A type of fuse that will withstand short current surges
Source	When using the *Parallel Port*, this term refers to using the negative side of the voltage supply for reference and the positive side for logic one levels.
Stop bit	The last one or more bits sent in a synchronous data transmission to indicate the logical end of a series of bits

GLOSSARY

Unit address	A set of two numbers which uniquely identify a unit in a communication network
Unit number	A twelve-bit Binary number used to identify a member of a group
VAC	Abbreviation for Voltage Altering Current
VDC	Abbreviation for Voltage Direct Current
Variable data	Real-time numerical quantities displayed in stored messages
Vacuum fluorescent	An evacuated display tube in which the anodes are coated with a phosphor that glows when electrons from the cathode strike it; usually organized in a dot matrix to form characters on a display screen

Index

Index

Index

Index

Index

Index

Index

Index