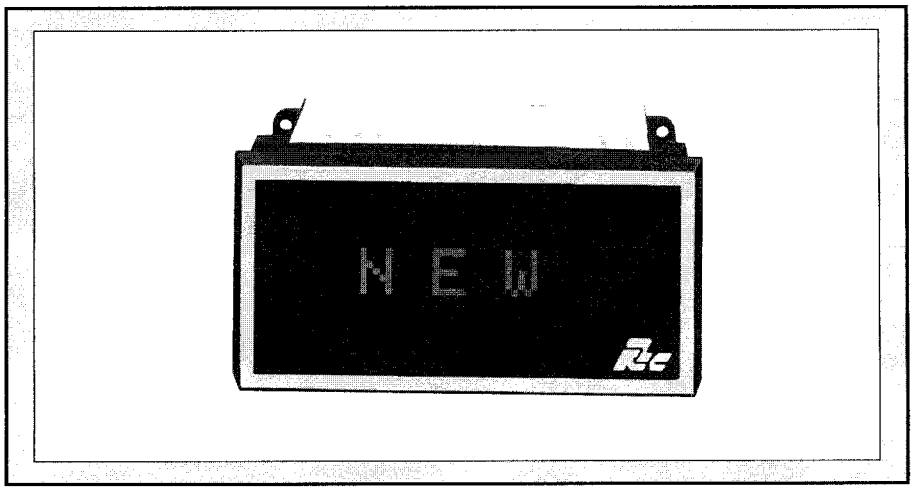
# THE APOLLO MESSAGE DISPLAY SLAVE SERIES




## *MODEL MDS INSTRUCTION MANUAL*

## *INTRODUCTION*

*The Message Display Slave unit (MDS) is a programmable display in our multi-purpose series of industrial control products that is field-programmable to solve multiple applications. This series of products is built around the concept that the end user has the capability to program different messages into the unit in order to adapt to various indication and process requirements.*

*The Slave Display which you have purchased has the same high quality workmanship and advanced technological capabilities that have made Red Lion Controls the leader in today's industrial market.*

*Red Lion Controls has a complete line of industrial indication and control equipment, and we look forward to being of service to you now and in the future.*

**UL Recognized Component,**
**File # E 171375**

**CAUTION**: Read complete instructions prior to installation and operaton of the unit.

**CAUTION**: Risk of electric shock.

**GENERAL DESCRIPTION** ..... 1
Safety Summary ..... 1
**CONNECTING THE MDS TO AN IBM® COMPATIBLE COMPUTER OR ASCII TERMINAL** ..... 2
System Requirements ..... 2
**MODES OF OPERATION** ..... 3
The RLC Mode ..... 3
The Serial Mode ..... 3
The Transmit Mode ..... 4
**SELECTING A MODE OF OPERATION** ..... 6
Units with a Parallel Port ..... 6
Units without a Parallel Port ..... 6
**CONFIGURATION FILE SET-UP** ..... 7
The Configuration Screen ..... 7
Unit Function Configuration ..... 8
Unit Address - RLC and Serial Modes only ..... 8
Error Handling - RLC and Serial Modes only ..... 8
Parallel Port Configuration ..... 8
Type - MDI only ..... 8
Debounce Time - Transmit Mode only ..... 8
Sample Time - Transmit Mode only ..... 9
ED Time Out - MDI only ..... 9
Serial Port Configuration ..... 9
Transmit Delay ..... 9
Serial Terminator - RLC and Serial Modes only ..... 9
ED Time Out - Transmit Mode only ..... 9
XON/XOFF - RLC and Serial Modes only ..... 9
Output/Busy Pin Configuration - RLC and Serial Modes only ..... 9
Status ..... 10
Logic Level ..... 10
Time Out ..... 10

**CONFIGURATION FILE SET-UP *(Cont'd)*** ..... 7
Display Configuration ..... 10
Intensity ..... 10
Blink Time ..... 10
Block Scroll Time ..... 10
Character Scroll Time ..... 11
Day/Month Names - MDI only ..... 11
**MESSAGE STRING FEATURES AND ASSEMBLY (RLC and Serial Modes Only)** ..... 12
RLC Mode Message String Format ..... 12
Serial Mode Message String Format ..... 12
Message String Construction ..... 13
Transmitting Extended Characters - 80h (128) to B8h (184) ..... 13
Displaying Text on the Bottom Line ..... 13
Block and Character Scrolling ..... 14
Blinking Text ..... 14
Including the Serial Terminator Character as Text ..... 14
Setting the Message Time Out Value ..... 14
Configuring the Output Pin ..... 15
Setting and Displaying the Current Time ..... 15
Configuring and Displaying Elapsed Timers ..... 16
MDS Control Code Sequences ..... 16
**ELAPSED TIMERS** ..... 17
**OUTPUT PIN** ..... 18
**HANDSHAKING** ..... 19
XON/XOFF ..... 19
Busy Mode ..... 19
**PARALLEL PORT (optional)** ..... 20
ITEM/ADDR Line ..... 20
STROBE Line ..... 20
DATA Lines (D0 - D7) ..... 22

**SERIAL COMMUNICATIONS** ..... 23
Communication Format ..... 23
Unit Address ..... 23
Serial Port Operation ..... 23
Serial Receive Buffer ..... 24
Serial Transmit Buffer ..... 24
Serial Terminator Character ..... 25
Serial Transmit Delay ..... 25
Serial Port Timing Diagrams ..... 25
**UNIT POWER-UP AND RESET** ..... 27
**COMMANDS** ..... 28
Assembling Commands ..... 28
RLC Mode ..... 28
Serial Mode ..... 28
Transmit Mode ..... 28
Command Syntax ..... 29
Configure Unit Functions - C01 ..... 31
Configure Unit Clock - C02 ..... 32
Configure Parallel Port (PRL Port) - C04 ..... 33
Configure Serial Port (SRL Port) - C05 ..... 33
Configure Output/Busy Pin - C06 ..... 34
Configure Elapsed Timers (ELP) - C07 - 'T' mnemonic ..... 35
Configure Unit Display - C08 ..... 36
Restore Default Unit Configuration - C17 ..... 37
Load Character and/or Day/Month File - C18 ..... 37
Halt/Reset Unit - C19 ..... 37
Cancel Message - C20 ..... 37
Request Message - C21 - 'M' mnemonic ..... 37
Transmit Text For Message - C23 ..... 37
Echo Character String - C24 ..... 38

**INSTALLATION AND CONNECTIONS** ..... 39
EMC Installaton Guidelines ..... 40
Wiring Considerations ..... 40
Vcontrast ..... 41
AC Power Wiring ..... 42
DC Power Wiring ..... 42
Serial Wiring ..... 42
Serial Terminal Descriptions ..... 43
**DIP SWITCH SETTINGS** ..... 44
Configuration and Self-Test Functions ..... 44
Unit Hardware Configuration ..... 45
Unit Configuration File Settings ..... 45
Parallel Port Configuration and Logic Levels ..... 46
Serial Port Loop-Back Test ..... 47
Memory Test ..... 47
Setting the RLC Mode ..... 47
Setting the Serial Mode ..... 48
Clearing the Unit ..... 48
**CHARACTER SET** ..... 49
Default Character Set ..... 49
**ERROR HANDLING** ..... 51
**TERMINAL EMULATOR FUNCTION** ..... 52
Function Keys ..... 52
Special Key Stroke Functions ..... 53
Terminal Emulator Record Function ..... 54
**SPECIFICATIONS AND DIMENSIONS** ..... 55
**APPLICATIONS** ..... 57
TRANSMIT MODE APPLICATION ..... 57
MDS RLC MODE APPLICATION ..... 57
MDS SERIAL MODE APPLICATION ..... 57
**TROUBLE SHOOTING GUIDE** ..... 58

**APPENDIX - USING THE SFMD SOFTWARE TO CREATE AND SIMULATE MDS MESSAGES** . . . . 59
Message Editing Screen . . . . 60
File/Unit Info . . . . 60
Message Destination - Display & MDS Unit . . . . 60
Message Time Out . . . . 60
MDS Unit Address . . . . 60
Message Scrolling . . . . 61
Message Priority - MDI only . . . . 61
Message Queuing - MDI only . . . . 61
Message Chaining - MDI only . . . . 61
Configuring the Output Pin . . . . 61
Periodic Messages - MDI only . . . . 62
Configuring Elapsed Timers . . . . 62
Collecting Embedded Data . . . . 62
Message Text Area . . . . 63
Message Simulator . . . . 66
Simulator Function Keys . . . . 66
Assembled Message String . . . . 67
Simulator Display . . . . 67
**ORDERING INFORMATION** . . . . 68

## GENERAL DESCRIPTION

The Message Display Slave *(MDS)* unit is a simple-to-use, yet versatile alphanumeric message center. The broad flexibility and functionality of the MDS make it adaptable to a wide variety of applications, including:

Display of Measured Values
Indication of Warning, Error, and Alarm Conditions
Indication of Manufacturing Process Steps
Display of Machine Start-Up and Operation Procedures
Display of RLC Message Display Intelligent (MDI) Unit Messages
Display of Information from other Serial RLC Units

The MDS is capable of displaying messages up to 127 characters in length. A message can include any of the 153 customizable characters, including the standard 96 character ASCII set. Individual lines of the message text can be programmed to scroll in a block or character fashion. Individual characters, blocks, lines of text and entire Messages can be programmed to blink, a well.

The MDS can be configured for one of three modes of operation:

RLC Mode
Serial Mode
Transmit Mode

In the RLC Mode, the MDS receives and displays Messages transmitted in an RLC Mode compatible format from one or more Message Display Intelligent (MDI) units. An MDI unit assembles and transmits the Message text to the MDS for remote display. Message text may include Current and Elapsed Timer values, as well as any data collected by the MDI.

In the Serial Mode, the MDS displays any ASCII text string terminated with the programmed Serial Terminator received over its Serial Port. Special control code sequences are available for accessing the blinking and scrolling features of the MDS. Other control codes are available for accessing the bottom line of dual line displays, and for displaying Current and/or Elapsed Time values. You can also assign a unique unit address to an MDS unit, for applications requiring individual addressing of multiple MDS units in the Serial Mode.

In the Transmit Mode, which requires the optional Parallel Port, the MDS interfaces directly with other RLC products with serial communications. The MDS accepts BCD unit address and data item numbers over the Parallel Port, and constantly polls the selected unit for the designated item over the Serial Port. The MDS display is updated with the information it receives from the selected RLC unit. Therefore, you can use a single MDS placed at a centralized location to retrieve and display information from multiple RLC units at remote locations.

Optional IBM® compatible software for configuring and programming the MDS is available from RLC, and the MDS's nonvolatile memory retains all programmed Configuration settings when power is removed from the unit. The Message Display User software (SFMD), with its easy-to-use menus, extensive prompts and on-line Help functions, assists in fitting the MDS to the application at hand. You can easily create and save Configuration Files with the SFMD software. Extensive file handling features are included, such as uploading, downloading, and printing of files.

Message Simulation and Terminal Emulation functions also come with the software. Although the MDS has no Message storage capability of its own, you can use the SFMD Message Editing function to construct the Message strings to transmit to the MDS, for either the RLC Mode or the Serial Mode. The Message Simulator is available to show you the actual Message string to transmit, as well as simulate the action of the MDS once the MDS receives the Message string. You can then test the MDS unit by entering and transmitting the assembled string to the MDS through the Terminal Emulator.

The metal front bezel of the MDS meets NEMA 4/IP65 requirements, allowing for wash down when properly installed. Modern surface-mount technology and extensive testing make the unit extremely reliable in industrial environments. Connections are made on removable terminal blocks, simplifying installation.

### SAFETY SUMMARY

All safety related regulations, local codes and instructions that appear in the manual or on equipment must be observed to ensure personal safety and to prevent damage to either the instrument or equipment connected to it. If equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

# CONNECTING THE MDS TO AN IBM® COMPATIBLE COMPUTER OR ASCII TERMINAL

To configure the MDS, connect the unit to an ASCII Terminal or a computer running a Terminal Emulator program. A typical configuration system is shown in Figure 1. RLC supplies the Message Display User Software *(SFMD)*, that runs on IBM® compatible PC's and assists you, through its user-friendly menus, numerous prompts, and on-line Help functions, in interfacing with and programming the MDS.

With the SFMD software, you can program multiple Configuration files for an MDS unit and save them to disk. You can then download any Configuration file, as needed, from the computer's serial port to one or more MDS units on the communications loop. The SFMD software also includes utilities for message simulation and terminal emulation. With the Message Simulator, the user can verify the proper operation of a message on the computer screen without having to transmit the message to the MDS. The Terminal Emulator can assist the user in verifying and testing the operation of the messages the MDS is to receive. It can also be used to monitor and trouble-shoot serial port communications.

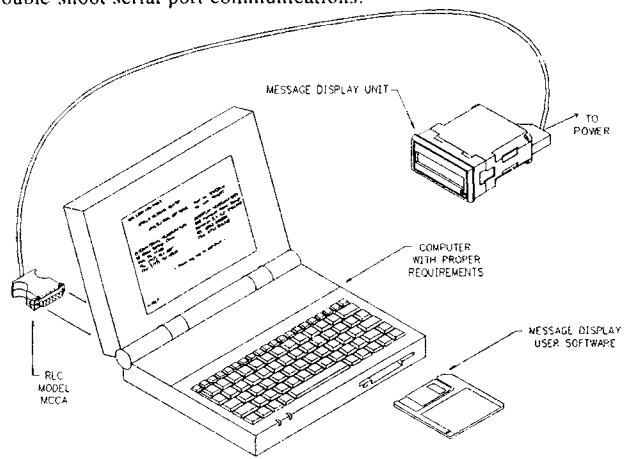



Figure 1 - Configuration System

## System Requirements

**IBM® compatible PC-AT (286 or greater) with:**

RS232 serial port — 640K RAM
DOS 3.0 or later — 1.2 or 1.44 Meg floppy drive
Monitor with MDA, CGA, EGA, or VGA graphics card
*Note: The SFMD software will not run on a PC-XT computer.*

**Cabling:**

RLC Model MCCA cable for initial set-up of a single unit.
RLC GCM232 Serial Converter Module for continuous interface with single or multiple units *(RS232/20 mA Current Loop)*, Serial communications cable, RLC Model APS supply or equivalent *(12 VDC power supply)*.

**Message Display User Software (SFMD)**

Available on 3.5" or 5.25" floppy disks.
*Note: Setup may only be performed via the serial port.*

Since the MDS transmits and receives over a 20 milliampere current loop, and the computer running the SFMD software communicates over an RS232 serial port, you must convert between these two communications formats. There are two methods for accomplishing this task.

1. When programming a single MDS unit, the preferred and easiest method for achieving this conversion is through the use of an RLC MCCA cable. The MCCA cable is prewired and ready to use. Plug the 25 pin D-type connector at one end of the cable into either COM1 or COM2 of the PC, and plug the terminal block at the other end of the cable into the bottom terminal receptacle of the MDI. Apply power to the MCCA cable, either AC or DC, as required. (See the MCCA Bulletin for details).
2. When programming multiple MDS's, or connecting to a network of serial devices, an RLC GCM232 Serial Converter Module and an RLC Model APS power supply are recommended. Connect the 20 milliampere loop of the MDS to the GCM232 module as shown in the "Installation and Connections" section of this manual, and plug the GCM232 module into COM1 or COM2 of the PC.

If the SFMD software is not installed on the PC, or you are unfamiliar with the operation of the software, refer to the SFMD Software Manual for more information.

# MODES OF OPERATION

## The RLC Mode

The RLC Mode of the MDS unit provides an easy interface for remotely displaying Messages programmed in, and processed by, one or more Message Display Intelligent (MDI) units. In this mode, the MDS expects to receive Message strings transmitted in an RLC Mode compatible format. An MDI unit assembles and transmits the Message string, that includes the appropriate information from any of the functions described above, along with the Message Text.

The Message Record in the MDI issuing the transmission specifies the address of the MDS unit receiving the transmission. The MDI precedes the Message string with this unit address. An MDS unit in the RLC Mode expects to see this unit address preceding all transmissions it is to process. Only an MDS unit with the matching address actually processes and displays the Message after receiving the transmission. All other MDS units on the loop whose addresses do not match the address contained in the transmitted string, ignore that transmission.

When the transmitting MDI encounters a Current Time field in the Message Text, it inserts the current value of its Real Time Clock (RTC), along with the specified format of the Current Time field, in the Message string. When the MDS processes the Message string, it sets its internal software clock to this transmitted value, thus "slaving" its own clock to the transmitting MDI's RTC.

The MDI performs a similar function when encountering Elapsed Timer fields in the Message Text. In this case, the MDI inserts the current configuration and value of the designated timer, along with the specified Elapsed Time format, into the transmitted string. When the MDS processes this string, it sets its Elapsed Timers to the configurations and values specified in the Message string, thus "slaving" its own Elapsed Timers to those of the transmitting MDI.

If the Message in the MDI configures the Output Pin, this configuration information is included in the transmitted Message string. When the MDS processes the string, it configures its Output Pin to the settings contained in the Message string, thus "slaving" its own Output Pin to that of the transmitting MDI.

The MDI includes the time out value of the Message, and the Blinking and Scrolling specifications for the Message text, in the Message string, and replaces all Index and Local Embedded Data Item fields with their respective values before transmitting the string. The MDI then terminates the assembled Message string with its own terminator, which must match that of the receiving MDS unit.

## The Serial Mode

The Serial Mode of the MDS unit provides an easy interface for displaying ASCII character strings. You transmit the desired character string, and terminate the string with the MDS's programmed Serial Terminator character. The MDS then displays the character string. No special control code characters or sequences are required to transmit simple ASCII text strings to the MDS while the unit is in the Serial Mode. In addition, the MDS automatically scrolls any line of text that contains more characters than the MDS can display.

A unit address function is available in the Serial Mode. Unlike the RLC Mode, however, if a Message string is NOT preceded by a unit address string, all Serial Mode units on the loop receive and process the transmission. You are required to provide an address for a Message string only if you wish to transmit the string to a particular Serial MDS.

All of the functions and methods described for the RLC Mode are available in the Serial Mode, as well.

### *The Transmit Mode*

In the Transmit Mode, which requires the optional Parallel Port, the MDS interfaces directly with other RLC products that have serial communications. The MDS accepts BCD unit addresses and data item numbers from a BCD thumbwheel or rotary switch connected to the Parallel Port, and repeatedly polls the addressed unit for the designated item, over the Serial Port. The MDS updates its display with the information it receives from the selected RLC unit. Therefore, you can use the MDS in the Transmit Mode to interrogate multiple RLC units in remote locations from one centralized location.

The MDS interrogates the other units on the serial loop by accessing the "Print Request" function, which is present in all serial RLC products. The MDS repeatedly issues the following character string:

**NxxP***

| | | |
|---|---|---|
| 'N' | - | RLC Unit Address identifier. |
| xx | - | Target Unit Address, 00 - 99. |
| 'P' | - | Print Request mnemonic code. |
| '*' | - | RLC Serial Terminator. |

The default value for the Target Unit Address is 0, and the default Target Data Item is 1. You can change the Target Unit Address by performing the following steps:

1. Set the BCD value on the Parallel Port DATA lines
2. Set the ITEM/ADDR line to the ADDR level
3. Apply a pulse to the STROBE line.

The Target Data Item the MDS displays is entered in a similar fashion, with the ITEM/ADDR line set to the ITEM level. The current Target Unit Address and Target Data Item number are saved in nonvolatile memory when power is removed.

You can retrieve different data items from a single Target unit by strobing in the desired data item number, without changing the unit address number. Similarly, you do not need to change the Target Data Item number when the Target Unit changes, allowing you to easily retrieve the same data item from multiple Target units.

With the Target Unit Address and Target Data Item selected, the MDS issues a "Print Request" Command for the Target Unit, and waits for a response. The MDS locates the Target Data Item in the response string by counting the Carriage Return (CR) characters contained in the response. When the Target Data Item is identified, the MDS extracts the character string for the Item, and updates the display with the new information.

Example: To display the second data item received from an RLC Gemini 5200 unit with address 3 and Print Option 4 selected, you would perform the following steps:

1. Place the BCD value "03" on the Parallel Port DATA lines.
2. Place the ITEM/ADDR control line in the ADDR state.
3. Activate the STROBE pin.
4. Place the BCD value "02" on the Parallel Port DATA lines.
5. Place the ITEM/ADDR control line in the ITEM state.
6. Activate the STROBE pin.

The MDS transmits:

**N03P***

and the Gemini 5200 with address 3 transmits the following response:

| | |
|---|---|
| "**3 CTA 123456**"<CR> | (Data Item 1) |
| "**3 CTB 123456**"<CR> | (Data Item 2) |
| "**3 CTC 123456**"<CR> | (Data Item 3) |

The MDS extracts the second data item, "**3 CTB 123456**" and displays the Target Data Item string, based on the type of transmission sent from the Target. Since the Target Unit is set to transmit its unit address and mnemonics along with the data, the MDS Block scrolls the unit address and mnemonic, and then the numerical data value, as shown:

" **3 CTB** "
" **123456** "

If the Target Unit had been set to transmit only the numerical data value, the MDS would display just the single block of data. If the MDS detects an asterisk '*' in the Target Data Item string, indicating a data overflow condition, it displays "**OVERFLOW** " along with the data, and the address and mnemonic, if sent. All data is centered on the display, and the Block Scroll configuration setting determines the rate at which the MDS displays the blocks.

If a response to the "Print Request" Command is not received within the Serial Port ED Time Out period, the MDS displays:

"**NO UNIT xx**" (xx - Target Unit)

If a response is received, but the selected Target Data Item is not present in the string, or is not received within the Serial Port ED Time Out period, the MDS displays:

"**NO ITEM xx**" (xx - Target Item)

In either case, the MDS continues to issue "Print Request" strings to the Target Unit, and look for the Target Data Item in the response, until either the Target Unit Address or the Target Data Item number is changed.

The MDS updates the display once every 250 milliseconds to once every five seconds depending on the serial baud rate, the Serial Port ED Time Out value, the type of Target Unit being interrogated, and the number of characters contained in the Target Unit's response.

You can configure the Serial Port Time Out value, and the Block Scroll setting in the RLC Mode of operation. See *Selecting A Mode Of Operation*, page 6, for details.

## SELECTING A MODE OF OPERATION

### Units with a Parallel Port

The settings present on the Parallel Port DATA lines on every unit power-up determine the mode of operation for an MDS unit with the Parallel Port. The MDS Transmit Mode is the only mode that accesses the Parallel Port during normal operation, and is the default mode for MDS units with the Parallel Port. To access either of the other two modes (RLC or Serial mode), DATA lines D5, D6, and D7 must be tied to one of the following combinations of logic levels:

| D7 | D6 | D5 | |
|---|---|---|---|
| 1 | 1 | 1 | RLC Mode |
| 1 | 0 | 1 | Serial Mode |

The DATA SINK/SRC and DATA LOGIC DIP switches must be set accordingly so that the MDS interprets the voltage levels at D5, D6, and D7 correctly. With the DATA SINK/SRC switch in the SINK position (DOWN) and the DATA LOGIC switch in the POSITIVE (UP) position, all data lines are pulled HI and interpreted as 1's. Therefore, if you wish to use an MDS w/Port in the RLC Mode, select these settings for the two DIP switches. To place the MDS in the Serial Mode, connect a jumper wire from the D6 line (Pin 7) to the Parallel Port COMMON line (Pin 11) on TBB. This jumper will pull D6 LO and the MDS will interpret it as a 0.

The preceding combinations are invalid in the BCD numbering sequence. Consequently, with the DATA SINK/SRC and DATA LOGIC DIP switches set correctly, and a BCD thumbwheel switch attached to the Parallel Port, the MDS will always power-up in the Transmit mode. While in the Transmit Mode, the MDS accepts Unit Addresses and Data Item numbers over the Parallel Port in BCD. Applying a valid pulse to the STROBE line (3 msec minimum) causes the MDS to read the ITEM/ADDR line and the DATA lines. The ITEM/ADDR line value indicates how the MDI should interpret the DATA lines, either as a Unit Address (ADDR), or as a Data Item (ITEM).

### Units without a Parallel Port

You can only choose between the RLC Mode and the Serial Mode for units without the Parallel Port, since the Transmit Mode of operation requires the optional Parallel Port. The desired mode is selected by the settings of the DIP switches on the side of the unit. To select the RLC Mode, set the BR0, BR1, BR2, and PARITY switches in the UP position, the EVEN/ODD switch in the DOWN position, and the 7/8 BIT switch in the UP position. To select the Serial Mode, set the BR0, BR1, and the BR2, switches in the up position, the PARITY switch in the down position, and the EVEN/ODD and 7/8 Bit switches in the up position.

With the appropriate DIP switches set, apply or cycle power to the unit. The selected mode now appears on the display, either RLC MODE or SRL MODE. To lock in the selected mode, move any of the Baud rate switches, BR2, BR1, or BR0, to the DOWN position. You can now set the switches as required for your application. If you need to change the mode of operation at a later time, repeat the above procedure for the desired mode.

You can verify the currently selected mode of operation by setting the BR2, BR1, and BR0 switches to the UP position, and the remaining switches to the DOWN position. Apply or cycle power to the unit and the MDS displays the hardware configuration of the unit, including the current mode.

(See *DIP Switch Settings*, page 43, for details)

## CONFIGURATION FILE SET-UP

The Unit Function, Parallel Port, Serial Port, Output Pin, and Display Configuration settings are all contained in an MDS's Configuration File. These Configuration File settings determine the manner in which an MDS interfaces with other units, as well as how it processes and displays the Message strings it receives.

There are two different methods available to you for configuring the MDS. Most users will want to use the SFMD software Configuration File Editing screen, shown in Figure 2, and described in detail in the remainder of this section.

Some applications, however, may require that the MDS be configured "on-line" from a controlling device such as an RLC Message Display Intelligent (MDI) unit, a computer, or a PLC. In this case, you can separately access each of the MDS's configuration settings through Commands C01 to C08 by transmitting the appropriate Command string to the unit over the serial port. See *Commands*, page 28, for more details.

### The Configuration Screen

You can program the Configuration settings on an IBM® compatible computer (PC) with the optional Message Display User Software (SFMD). The settings are programmed in a Configuration File (".CFG" file extension), which you can save on the PC's hard or floppy disk drives. The Configuration File is then downloaded into the nonvolatile memory of the MDS via the PC's serial port.

You must first place the MDS in either the RLC Mode or the Serial Mode in order to download a file to, or upload a file from the MDS, as well as issue Commands and/or Message strings to the unit. The Transmit Mode of operation DOES NOT respond to any of these transmissions. MDS Units without the optional Parallel Port cannot be placed in the Transmit Mode, and this limitation does not apply to those units. The Mode of operation for MDS Units with the Parallel Port is determined by the Parallel Port value on every unit power-up. You can select the RLC Mode of operation by setting the DATA SNK/SRC DIP switch on the back of the unit for SNK and the DATA LOGIC DIP switch on the side of the unit for '+' (UP). (See *MODES OF OPERATION*, page 3, for details).

You can program Configuration File settings in the Configuration Editing Screen. The upper left corner of the Configuration Screen displays the name of the current Configuration File. To the right of the file name are displayed the last date and time that the file was modified. Located at the bottom of the screen's border is an Active Prompt Line, which displays basic information about the currently highlighted field. If you require more information about a particular field, on-line Help for the highlighted field can be accessed by pressing **F1-HELP**. General Help is also available from the Main Menu:

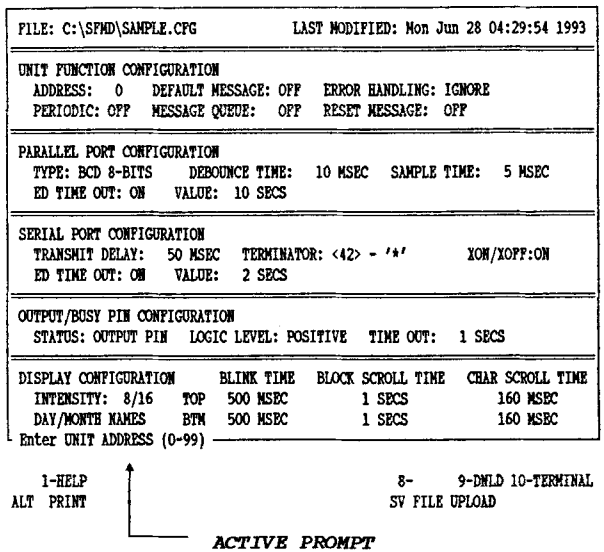



Figure 2 - Configuration Editing Screen

Although you program the Configuration File settings on the PC's screen, the MDS does not receive these settings until you download the new Configuration File containing those settings to the unit. Therefore, the MDS continues to access the current settings contained in its Configuration File until you download a new file.

This screen is used for configuring both MDS and MDI (Message Display Intelligent) units. The following functions shown on the screen do not apply to the MDS, and are ignored:

**Default Message**
**Periodic Message**
**Message Queue**
**Reset Message**

The Configuration File settings are described in groups, in the order in which they appear on the screen.

## Unit Function Configuration

The Unit Function settings enable and disable various major functions and features of the MDS. The definition and operation of each Unit Function is described below. You can also access all of the Unit Function settings described below with Command C01. See *Command C01 - Configuration Unit Functions*, page 31, for details on the proper usage of the C01 Command.

### Unit Address - RLC and Serial Modes only

The Unit Address can be set from 0 to 99. As with other RLC products, a unit with Address 0 does not require the "N00" address prefix when receiving serial communications. If multiple MDS's are on the communications loop that you will need to access individually, you should assign a unique address to each unit.

If you are using an MDI to transmit to an MDS which is in the RLC Mode, the message in the MDI must have the MDS UNIT Destination selected and a non-zero address specified for the MDS receiving the transmission.

### Error Handling - RLC and Serial Modes only

You can configure the MDS to generate Error Codes upon encountering certain Error conditions. You can choose how the MDS responds to an Error condition from the following Error Handling methods.

DISPLAY - The MDS displays an Error Message containing the Error Code. The Error Message blinks and has a time out value of 16 seconds.

TRANSMIT - The MDS transmits the Error Code over the serial port as an ASCII character string.

DISPLAY & TRANSMIT - The MDS displays and transmits the Error Code.

IGNORE - The MDS continues to detect Error conditions, but gives no indication of the Error.

*Note: With the Serial Port XON/XOFF function enabled, the MDS will always transmit the Error Code regardless of the Error Handling setting you have selected.*

(See *ERROR CODES*, page 50, for details on the processing of Error conditions).

## Parallel Port Configuration

The MDS's optional Parallel Port is a unidirectional input port. It consists of eight DATA lines *(D0 to D7)* and two CTRL lines *(STROBE and ITEM/ADDR)*. The DATA and CTRL lines are separately DIP switch selectable for sinking or sourcing current and positive or negative logic. (See the *PARALLEL PORT*, page 20, for details on the selection of these DIP switch settings).

You can also access all of the Parallel Port configuration settings described below with Command C04. See *C04 - Configure Parallel Port*, page 33, in the COMMAND section for details.

### Type - MDI only

The MDS only accepts 8 bit BCD information over the Parallel Port.

### Debounce Time - Transmit Mode only

The Debounce Time specifies the minimum time the MDS waits after the end of one valid STROBE pulse before detecting the beginning of the next STROBE pulse. This value ranges from 10 milliseconds to 2.5 seconds.

(See the *PARALLEL PORT*, page 20, for details on the usage and operation of the Debounce timer).

### *Sample Time - Transmit Mode only*

The Sample Time setting specifies the minimum time the DATA lines and the ITEM/ADDR line must be stable following the start of the STROBE pulse before the MDS accepts their values as valid. You can set the Sample Time value from 1 to 255 milliseconds.

(See the *PARALLEL PORT*, page 20, for details on the usage and operation of the Sample timer).

### *ED Time Out - MDI only.*

This function is not implemented in the MDS.

## *Serial Port Configuration*

The serial communication port is a full duplex, optically isolated two-way 20 milliamp current loop. The baud rate, parity, and data bit settings are DIP switch selectable. These settings should conform to those of the system in which the MDS is to be installed. (See *Serial Communications*, page 23, for details on the usage and operation of the Serial Port).

You can also access all of the Serial Port configuration settings described below with Command C05. See *Command C05 - Configure Serial Port*, page 33, for details on the proper usage of the C05 Command.

### *Transmit Delay*

The Transmit Delay ranges from 0 to 2.5 seconds, in 0.01 second intervals, and establishes the minimum time between the end of one MDS transmission and the beginning of the next transmission.

(See *Serial Communications, Serial Transmit Delay*, page 25, for details on the usage and operation of the Serial Port Transmit Delay function.)

### *Serial Terminator - RLC and Serial Modes only*

The Serial Terminator character signals the MDS that it should process the contents of its Serial Receive buffer. All message strings and commands must be terminated with the programmed Serial Terminator. The Terminator character can be programmed from 01h to 07h (1 to 7), 09h to 1Ah (9 to 26), 1Ch to 2Fh (28 to 47), and 3Ah to 40h (58 to 64). The factory default value of the Terminator character is '*', 2Ah (42).

(See *Serial Communications, Serial Terminator*, page 25, for details on the usage and operation of the Serial Port Termination.)

### *ED Time Out - Transmit Mode only*

The Embedded Data (ED) Time Out value ranges from 0 to 254 seconds and specifies how long the MDS should wait to receive a Target Data Item over the Serial Port in the Transmit mode. You can also turn the ED Time Out function OFF, causing the MDS to wait indefinitely for a data item.

(See *Modes Of Operation, Transmit Mode*, page 4, for details on the usage and operation of the ED Time Out function.)

### *XON/XOFF - RLC and Serial Modes only*

The XON/XOFF setting and the Output Pin function setting select the communications handshaking protocol the MDS uses when receiving and processing Command and Message Requests. The MDS transmits the ASCII XOFF character, 13h (19), to indicate the Unit Busy condition, and the ASCII XON character, 11h (17), to indicate the Unit Ready condition. The XON/XOFF Mode and the BUSY setting of the Output Pin Function cannot both be selected at the same time.

(See *Handshaking*, page 19, for details on the usage and operation of the XON/XOFF function.)

## *Output/Busy Pin Configuration - RLC and Serial Modes only*

The MDS contains one open-collector Output pin. You can select the function of this pin and its Active and Inactive logic levels, and you can specify a default Time Out value. *(See the OUTPUT PIN section for details on the usage and operation of the Output/Busy Pin.)*

You can also access all of the Output Pin configuration settings described below with Command C06. See *Command C06 - Configure Output/Busy Pin*, page 34, for the proper usage of the C06 command.

### *Status*

The Output Pin can be configured for one of the following states:

OUTPUT mode
BUSY mode
DISABLED

In the OUTPUT Mode, the pin functions as an independent Output pin that you can configure and control with Messages and Commands.

In the BUSY Mode, the MDS uses the pin for the READY/BUSY handshaking protocol. In the BUSY Mode, the XON/XOFF Mode of the Serial Port is automatically turned OFF since both handshaking functions cannot be in use at the same time. With the pin in the BUSY Mode, Messages cannot access the Output Pin.

When set for DISABLE, the Output Pin has no function.

(See the *Output Pin* and *Handshaking* sections for details on the Output and Busy modes of the Output/Busy Pin.)

### *Logic Level*

The Output Pin can be configured for positive or negative logic. In positive logic, with the open-collector pulled-up through a resistor, the pin outputs a logic "1" (does not conduct) when set to the ON state in the OUTPUT Mode, or while the unit is in the BUSY condition in the READY Mode. In negative logic, with the open-collector pulled-up through a resistor, the pin outputs a logic "0" (conducts) for the ON state and the BUSY condition.

The ON state and the BUSY condition are considered the "Active" levels of the pin. The OFF state and the READY condition are considered the "Inactive" levels of the pin.

(See *Output Pin*, page 18, for details on the various combinations of output levels for the Output Pin.)

### *Time Out*

The MDS contains a dedicated Output Time Out function which you can configure to automatically turn the Output pin OFF after a predetermined Time Out interval.

The default Output Time Out value can be set from 10 to 1260 milliseconds, 1 to 63 seconds, or 1 to 63 minutes. Messages and Commands can access this value, or can supply their own Time Out value, independent of the default value.

(See *Output Pin*, page 18 , for details on the usage and operation of the Output Time Out function.)

## *Display Configuration*

The MDS utilizes a Liquid Crystal Display (LCD) with direct LED back lighting. Display options include 1x10 and 2x20 character formats, positive and negative image display types, and Red (negative image only) and Yellow-Green back lighting. You can adjust the brightness of the display, and the various blink and scroll times.

The Display configuration settings described below can be accessed with Command C08. See *Command C08- Configure Unit Display*, page 36, for details on the proper usage of the C08 Command.

### *Intensity*

The Backlight Intensity of the display is software adjustable through 17 levels and ranges from 0 (OFF) to 16 (FULL INTENSITY) in 1/16 level increments. The Intensity of the backlight is current controlled, and each increment of the Intensity level accounts for about 10 milliamps of current drawn by the MDS.

### *Blink Time*

You can have the MDS blink any character or group of characters in a Message by surrounding the desired characters in the Message transmission string with the BLINK control code sequence. All blinking characters on a given display line blink in unison at the designated Blink time for that line. The Blink time is adjustable from 10 milliseconds to 1.27 seconds and can be set separately for the top and bottom lines of a 2x20 display.

(See *Message String Construction, Blinking Characters*, page 14, for details on the usage and operation of the Blink feature.)

### *Block Scroll Time*

Selecting Block Scrolling for a line of text causes the MDS to display each block of text (generally one or more consecutive words in a Message) one after the other on that line. The Block Scroll time sets the amount of time the MDS displays each Block of text for a given line. Once the last

block of text for a line has been displayed, the MDS repeats the line from the first block.

A line of text can be designated for Block Scrolling by inserting the BLOCK SCROLL control code sequence anywhere in the text for that line in a message string transmission. You then designate blocks of text by inserting the SEGMENT control code sequence at the desired locations in the text. The Block Scroll time can range from 10 to 1270 milliseconds or 1 to 127 seconds per Block and can be set separately for each line of a 2x20 display.

(See *Message String Construction, Block Scrolling*, page 14, for details on the usage and operation of the Block Scroll feature.)

### *Character Scroll Time*

Selecting Character Scrolling for a line causes the MDS to advance the text for that line across the display from right to left, one character position at a time. The Character Scroll time sets the amount of time a single shift in character position takes. Once the last character of a line has scrolled off the display to the left, the MDS repeats scrolling the line from the right. You can designate a line of text for Character Scrolling by inserting the Character Scroll control code sequence anywhere in the text for the line in a message string transmission.

In the Serial Mode, the MDS automatically Character Scrolls any line of text containing more characters than the display can hold. In this case, it is not necessary to include the Character Scroll control code sequence in the Message string.

The Character Scroll time can be adjusted from 10 milliseconds to 1.27 seconds per character position and you can select separate scroll times for each line of a 2x20 display. You can also designate separate blocks of text for Character Scrolling by inserting the SEGMENT control code sequence at the desired locations in the text. In this case, the MDS Character scrolls each block of text across the display separately, one block at a time.

(See *Message String Construction, Character Scrolling*, page 14, for details on the usage and operation of the Character Scroll feature.)

### *Day/Month Names - MDI only*

The MDS contains a table with the names of the days and months. Through the placement of the CURRENT TIME control code sequence and the proper format codes, you can have the MDS insert these names in the Message text.

(See *Message String Construction, Current Time And Date*, page 15, for details).

Unlike the MDI, the MDS does not contain battery-backed RAM, and cannot save this table when power is removed from the unit. You can, however, load the table "on-line", either with an MDI, or through another serial device programmed to do so. (See *Command C18*, page 37, for details on loading a custom Day/Month name table.)

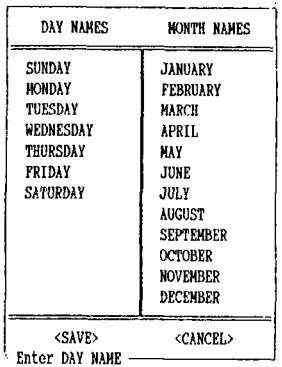



Figure 3 - Day/Month Names

# MESSAGE STRING FEATURES AND ASSEMBLY (RLC and Serial Modes Only)

The primary function of the MDS, in either the RLC Mode or the Serial Mode, is to receive serial transmission strings and to display Messages. This Message string can consist of message text, as well as specify a number of additional functions that the MDS executes when processing and displaying the Message string.

### *RLC Mode Message String Format*

In the RLC Mode, the unit expects to receive Message strings in the following format:

**NxxM:tt...t<term>**

| | | |
|---|---|---|
| 'N' | - | Unit Address identifier, indicates that a Unit Address follows. |
| xx | - | Unit address, 0 - 99. |
| 'M:' | - | Indicates that the Message text follows. |
| tt...t | - | Message text, up to 122 characters. |
| <term> | - | Programmed Serial Terminator character. |

The "Nxx" portion of the Message string is not required for an MDS unit with address 0:

M:tt...t<term>

The RLC Mode is primarily used for "slaving" one or more MDS units to an RLC Message Display Intelligent (MDI) unit. The Message string is stored in an MDI as part of a Message record. The MDI assembles and transmits the string on request. The SFMD software can be used to construct and display a Message string by creating the Message for the DISPLAY AND MDS Destination in the Message Editing screen, and then simulate the message with the Message Simulator function. The Simulator both shows the resulting Message string as the MDI would transmit it to an MDS, and simulates how the MDS would display the received string. (See the *Appendix, Using The SFMD Software To Create And Simulate Messages*, page 58, for details.)

### *Serial Mode Message String Format*

In the Serial Mode, the unit expects to receive serial transmission strings in the following format:

**tt...t<term>**

| | | |
|---|---|---|
| t...t | - | Message text, up to 127 characters |
| <term> | - | Programmed Serial Terminator character |

Notice that there is no Unit Address identifier specified for this mode. All units in the Serial Mode display all Message strings they receive, unless the string is preceded with the Unit Address control code:

**<CTRL-N>xxM:tt...t<Term>**

| | | |
|---|---|---|
| <CTRL-N> | - | Unit Address control code, 0Eh -( 14), indicates that a Unit Address follows. |
| xx | - | Unit address, 0 - 99. |
| 'M:' | - | Indicates that the Message text follows. |
| tt...t | - | Message text, up to 122 characters. |
| <term> | - | Programmed Serial Terminator character |

Notice that this string is identical to the RLC Mode format, with the 'N' Unit Address identifier replaced with the <CTRL-N> control code. In the SFMD Terminal Emulator, the <CTRL> character can be entered by holding down the <CTRL> key and hitting the 'N' key. If you are using a text editor, or other software to enter, construct, and/or store Message strings, enter this character by holding down the <ALT> key and typing "14" on the numeric keypad.

The Serial Mode is primarily used to receive ASCII text strings from a computer, a PLC, or other serial device. If you do not need to specify any of the special functions of the MDS, transmit the desired text strings verbatim. If you want to include one or more of the special functions in a Message string, you must edit the existing text to insert control code sequences which specify the various functions of the MDS.

The SFMD software can be used to construct and display these Message strings for you by creating the Message for the DISPLAY AND MDS Destination in the Message Editing screen, and then calling the Message Simulator function. The Simulator shows you the resulting Message string as you should transmit it to an MDS (with a few minor exceptions), and

simulates the MDS display as it would appear upon receiving the string. In the Serial Mode, the MDS will automatically character scroll a line containing one or more blocks of text having more characters than the Display's line length. (See the *Appendix, Using the SFMD Software to Create and Simulate MDS Messages,* page 58, for details).

## Message String Construction

The MDS begins processing a Message string upon receipt of the programmed Serial Terminator character. The processing of a Message string entails the MDS executing the various functions the Message string specifies. The MDS can process only one Message string at a time. The maximum processing time of any Message string is 10 milliseconds and while busy processing a Message, the MDS cannot receive additional information over the serial port. As long as the MDS is not already busy processing a Message string, it processes all strings as it receives them, replacing the Message currently on the display with each new Message it receives.

The MDS processes a Message string in the following sequence:

1. Configures the Output Pin, if specified.
2. Configures any specified Elapsed Timers.
3. Configures the Top and Bottom lines of the display for the designated scrolling technique, if any.
4. Places the Message on the display and starts the Message Time Out timer.

The various functions you can specify and include in a Message string are described below. These functions are available in both the RLC and the Serial Modes of operation. Specify a special function for a Message by inserting the appropriate control code sequence in the Message string, as described below. All control code sequences have the form:

**^x OR ▼ x**

^x or ▼ x - Message string control code, 5Eh (94) and 1Fh (31) are interchangeable.

x - Control code character, 20h (32 - ' ') to 2Ch (44 - ',').

*Note: For all following examples, the '^' control code is shown, but the '▼' code can be used as well. Also the examples are given for the Serial Mode. For the RLC Mode you must append "M:" to the beginning of each message string.*

### *Transmitting Extended Characters - 80h (128) to B8h (184)*

The extended characters (characters from 80h (128) to B8h (184)) all require 8 bits for transmission. If your serial port is configured for 8 data bits, include any extended character in a Message string by holding down the <ALT> key and typing in the decimal number of the character on the numeric keypad. For applications using 7 data bits on the serial port, you cannot transmit a character greater than 7Fh (127), since the character requires 8 bits for transmission. Therefore, you must convert all extended characters into a 7-bit compatible form. This form is as follows:

**^x**

'^' - Message string control code, 5Eh (94).

x - Extended character - 50h (80).

Find the decimal equivalent of the extended character you wish to transmit. Subtract 80 decimal from it, and enter the resulting character following the '^' control code. When the MDS receives this sequence, it converts the control sequence back into the original extended character.

**Example:** To transmit the 'ü' character, find its decimal equivalent, 129, and subtract 80 from it, resulting in 49. Either type in this number while holding down the <ALT> key, or press the key for its ASCII equivalent, which is the character '1'. The resulting control code sequence then becomes:

**^1**

## Displaying Text on the Bottom Line

You can instruct the MDS to place text on the bottom line of a 2x20 display by inserting the "^+" control code sequence at the end of the top line of text. All text following this control sequence is displayed on the Bottom line. You can specify a separate scrolling technique for each line of text by including the appropriate control code sequence anywhere in the text for a given line. See Block and Character Scrolling below.

**Example:** To display "Top Line" on the top line of the display, and "Bottom Line" on the bottom line of the display, transmit the following text string:

**Top Line^+Bottom Line<term>**

### *Block and Character Scrolling*

You can instruct the MDS to Block or Character scroll a line of text by including the appropriate control code sequence in the text for a line. To Block scroll a line, include the "^&" control code sequence anywhere in the line, and insert the "^)" End-of-Block sequence at the end of each Block of text. To Character scroll a line, include the "^'" control code sequence anywhere in the line. To separately Character scroll Blocks of text, insert the "^)" End-of-Block sequence at the end of each Block of text.

**Example:** To Block scroll the transmission string "This Message Block scrolls", transmit the following text string:

**^&This^)Message^)Block^)scrolls<term>**

Notice that there is no End-of-Block control code at the end of the Message string. The MDS assumes an End-of-Block whenever it encounters the end of a line of text.

**Example:** To Character scroll the transmission string "This Message Character scrolls", transmit the following text string:

**^'This Message Character scrolls<term>**

In the Serial Mode, the MDS automatically character scrolls the text for any line longer than the display can hold. This is not true in the RLC Mode. Any characters exceeding the length of the display are truncated, unless character scrolling is explicitly specified for that line.

### *Blinking Text*

You can instruct the MDS to Blink text by surrounding the desired text with the Blink control code sequence, "^<space>". The MDS Blinks all text located between two Blink control sequences at the Blink rate specified for the line on which the text appears.

**Example:** To cause the Message "This Line Blinks" to Blink on the display, transmit the following text string:

**^<space>This Line Blinks^<space><term>**

### *Including the Serial Terminator Character as Text*

You can include the Serial Terminator character in the Message text by inserting the "^(" control code sequence at the desired location in the Message string. When the MDS encounters this control sequence, it replaces the sequence with the programmed Serial Terminator character.

**Example:** To display the Message *** CAUTION *** on an MDS with the Serial Terminator set to '*', transmit the following text string:

**^(^(^( CAUTION ^(^(^(<term>**

### *Setting the Message Time Out Value*

Including an optional Message Time Out value in a Message string determines how long the MDS displays that Message. The MDS automatically cancels a Message when the Message's Time Out value expires. If you do not select the Time Out function for a Message, the MDS displays the Message indefinitely, until the Message is canceled, or a new Message string is received.

The Message Time Out value can range from 10-1260 milliseconds, 0-63 seconds, or 1-63 minutes. A Time Out value of 0 seconds indicates that the MDS should process the Message and, after placing the Message on the display, immediately cancel the Message. This feature is useful for configuring the Output Pin, and/or Elapsed Timers, with a Message string.

You can include a Time Out value for a Message with the following format:

**^$xxx**

^ - MDS control code character, 5Eh (94).
$ - MDS Message Time Out control code.
xxx - MDS Time Out value, 3-digits with leading zeroes included.

Construct 'xxx' as follows:

| | | |
|---|---|---|
| MILLISECONDS | = | 128 + (1 to 126), for 10 to 1260 Milliseconds, in 10 millisecond increments. |
| MINUTES | = | 64 + (1 to 63) for Minutes. |
| SECONDS | = | 0 + (0 to 63) for Seconds |

A value of 255 indicates that the Message should not Time Out. This is the default value, and you do not need to include the Time Out control code sequence if the Message does not Time Out.

**Example:** To have the Message "This Message times out in 10 Minutes" appear on the display for ten minutes, you would transmit the following string:

**^$074This Message times out in ten Minutes<term>** **(74 =64 + 10)**

### *Configuring the Output Pin*

The Output Pin can be configured in a Message string. When included, the MDS expects to receive the Output Pin configuration in the following format:

**^#xxx**

| | | |
|---|---|---|
| ^ | - | MDS control code character, 5Eh (94). |
| # | - | MDS Output Pin control code. |
| xxx | - | MDS configuration settings, 3-digits with leading zeroes included. |

Construct 'xxx' as follows:

Add 128 to UNLOCK the Output Pin, before changing settings.
Add 64 to turn the Pin ON.
Add 32 to turn the Pin OFF.
Add 16 to TOGGLE the Pin's state.
Add 8 to LOCK the Pin after changes have been made.
Add 4 to HALT the Time Out timer.
Add 2 to RUN the Time Out timer from the default value.
Add 1 to RUN the Time Out value from a value contained in the Message or RESUME the Time Out from the existing value.

**Example:** To have a Message UNLOCK the Output Pin, turn the Pin ON, RUN the Time Out from the default value, and LOCK the Pin after all the changes have been made, include the following string in the transmission:

**^#202** (202 = 128 + 64 + 8 + 2)

The Output Pin Time Out value can be set in a Message string. When included, the MDS expects to receive the Output Pin Time Out configuration in the following format:

**^%xxx**

| | | |
|---|---|---|
| ^ | - | MDS control code character, 5Eh (94). |
| % | - | MDS Output Time Out control code |
| xxx | - | MDS Time Out value, 3-digits with leading zeroes included. |

Construct 'xxx' as follows:

| | | |
|---|---|---|
| MILLISECONDS | = | 128 + (1 to 126), for 10 to 1260 Milliseconds, in 10 millisecond increments. |
| MINUTES | = | 64 + (1 to 63) for Minutes. |
| SECONDS | = | 0 + (1 to 63) for Seconds. |

**Example:** To have a Message set the Output Time Out value to 500 milliseconds, include the following string in the transmission:

^$178 (178 = 128 + 50)

### *Setting and Displaying the Current Time*

You can set and display the Current Time in a Message string. The MDS expects to receive the Current Time configuration in the following format:

**^!uussnnhhaaddmmyyff...f^!**

| | | |
|---|---|---|
| ^ | - | MDS control code character, 5Eh (94). |
| ! | - | MDS Current Time control code. |
| uussnnhhaaddmmyy | - | 16-digit value of the Current Time, with leading zeros, in the following format: |
| uu | - | Hundredths of seconds (00 - 99) |
| ss | - | Seconds (00 - 59) |
| nn | - | Minutes (00 - 59) |
| hh | - | Hours (00 - 23). |
| aa | - | Day of the Week (01 - 07, Sunday is 01). |
| dd | - | Day of the Month (01- 31). |
| mm | - | Month (01 - 12, January is 01). |
| yy | - | Year (00 - 99; 80 - 99 is 1980 - 1999, 00 - 79 is 2000 to 2079). |
| ff...f | - | Format string for Current Time (optional- See the Time Codes section for details). |

When the MDS receives this string, it sets the Current Time to the included 16-digit value. If the string includes a Current Time format, the MDS displays the value of the Current Time at the location, and in the format you specify. You can also use Command C02 to set the Current Time.

**Example:** To set the Current Time in the MDS to October 26, 1993, 10:15:00.00 AM, and display the time in the format 26Oct93 10:15 AM, transmit the following string:

**^!0000151003261093DDMAY2 HA:NN P^!<term>**

See *Message Text Area* under *Using The SFMD Software To Create And Simulate MDS Messages*, page 58, for details.

### *Configuring and Displaying Elapsed Timers*

You can configure and display one or more Elapsed Times in a Message string. The MDS expects to receive an Elapsed Timer configuration in the following format:

**^"xxxsdvuussnnhlhiff...f^"**

| | | |
|---|---|---|
| ^ | - | MDS control code character, 5Eh (94). |
| " | - | MDS Elapsed Time control code. |
| xxx | - | Three-digit Elapsed Timer number, leading zeros must be included. |
| s | - | Status flag, 'N' for ON, 'F' for OFF. |
| d | - | Direction flag, 'U' for UP, 'D' for DOWN. |
| v | - | Overflow flag, 'V' clears an Overflow, '<SPACE> ' does not. |
| uussnnhlhi | - | 10-digit value of the Elapsed Timer, with leading zeros in the following format: |
| uu | - | Hundredths of seconds (00 - 99). |
| ss | - | Seconds (00 - 59). |
| nn | - | Minutes (00 - 59). |
| hi | - | Tens and Ones Hours (00 - 99). |
| hl | - | Thousands and Hundreds Hours (00 - 99). |
| ff...f | - | Format string for the Elapsed Time (optional- See *Time Codes*, page 64 , for details). |

When the MDS receives this string, it sets the specified Elapsed Timer to the included 10-digit value. If the string includes an Elapsed Time format, the MDS displays the value of the specified Elapsed Timer at the location, and in the format, specified in the string. You can also use Command C07, 'T' mnemonic, to configure the Elapsed Timers.

**Example:** To set Elapsed Timer 10 to RUN DOWN from 1 Hour, and display the Timer's value in the form xx:xx:xx, transmit the following string:

**^"010ND 0000000100HL:NN:SS^"<term>**

See *Message Text Area* in the Appendix *USING THE SFMD SOFTWARE TO CREATE AND SIMULATE MDS MESSAGES*, page 58, for details.

### *MDS Control Code Sequences*

The following is a list of the control code sequences the MDS recognizes.

| | | |
|---|---|---|
| '^<SPACE> ' | - | Blink operator. |
| '^!' | - | Current Time operator, followed by the 16 digit value of the Current Time, an optional Current Time format string, and a closing '^!'. |
| '^"' | - | Elapsed Time operator, followed by the three digit Timer number, a three character Timer configuration sequence, the ten digit value of the Elapsed Timer, an optional Elapsed Time format string, and a closing '^"'. |
| '^#' | - | Output Pin Configuration operator, followed by a three-digit value. |
| '^$' | - | Message Time Out operator, followed by a three-digit value. |
| '^%' | - | Output Pin Time Out operator, followed by a three-digit value. |
| '^&' | - | Block scrolling operator, included in a line if that line BLOCK scrolls. |
| '^'' | - | Character scrolling operator, included in a line if that line CHARACTER scrolls. |
| '^(' | - | Terminator operator, allows you to use the Serial Terminator character as a Message text character. |
| '^)' | - | END-OF-BLOCK operator. |
| '^+' | - | END-OF-LINE operator. |
| '^0' to '^h' | - | Extended characters, 80h (128) to B8h (184). |

## ELAPSED TIMERS

The MDS contains sixteen separate, fully configurable Elapsed Timers. Any Message can be programmed to access, configure, and/or display these Elapsed Timers. You can also access and configure any Elapsed Timer with the C07 Command ('T' mnemonic). The MDS constantly updates all Elapsed Timer values according to the current configuration for each Timer. Each Elapsed Timer contains the following configurable attributes.

**Timer Disposition:** Not implemented in the MDS. On every unit Power-up or Reset, the MDS resets the Timer configuration to the factory default settings (that is, the MDS halts the Timer, and sets the Direction to DOWN, clears the VALUE to ZERO, and clears the Overflow/Underflow flag).

**Timer Status:** A Timer is either ON (running) or OFF (halted). Turning a Timer ON or OFF does not reset its value.

**Timer Direction:** Each Timer can run UP or DOWN. The direction of the Timer can be changed while it is ON (running), there is no need to halt or reset the Timer.

**Timer Value:** A Timer can accumulate up to 10,000 hours, to 0.01 second resolution. A Timer can be set to any value within the range from 0000:00:00.00 to 9999:59:59.99. You do not need to halt the Timer to set its value.

**Timer Overflow/Underflow:** An UP Timer rolls-over (overflows) from 9999:59:59.99 to 0000:00:00.00. A DOWN Timer rolls-over (underflows) from 0000:00:00.00 to 9999:59:59.99. Each Timer contains an Overflow/Underflow flag that indicates this condition. This flag can be reset as necessary.

**Trigger Function:** Not implemented in the MDS.

**Trigger Type:** Not implemented in the MDS.

**Trigger Value:** Not implemented in the MDS.

An Elapsed Timer value is displayed in a Message by inserting the ELAPSED TIMER control code sequence, followed by a three digit timer number, a ten-digit Timer value, the appropriate Time format codes, and a closing control code sequence in the Message Text. If a displayed Timer is running, the MDS displays the running value as it changes. See *Message String Construction*, page 13, for details on specifying an Elapsed Timer in a Message string.

## OUTPUT PIN

The MDS has one NPN open-collector Output Pin that can be assigned to one of two modes, the READY/BUSY handshaking mode, or the Message and Command controlled OUTPUT Mode. With the Output pin assigned to the READY/BUSY Mode, the MDS indicates its current READY/BUSY condition by outputting the appropriate logic levels on the pin. (See *Handshaking*, page 19, for details).

With the Output pin assigned to the OUTPUT Mode, you can transmit Messages strings and issue Commands to configure the pin. The Output pin in the OUTPUT Mode contains the following configurable attributes.

**Logic Level:** The Output pin can be configured for positive or negative logic. This setting applies to both the output mode and the READY/BUSY Mode of the pin. The following table lists the Output Pin conduction state for the various Logic Level and Mode settings:

| Logic Level | Output Mode | READY/BUSY Mode | Pin State |
|---|---|---|---|
| - | ON | BUSY | Conducts |
| - | OFF | READY | Does Not Conduct |
| + | ON | BUSY | Does Not Conduct |
| + | OFF | READY | Conducts |

**Pin Status:** The Output Pin can be turned ON or OFF. You can also TOGGLE its current status from ON to OFF and OFF to ON. The actual voltage level appearing at the Output Pin is determined by the Logic Level setting, above.

**Lock Status:** You can LOCK the pin from accepting any further configuration changes. With the pin LOCKed, the MDS ignores any attempt to configure the pin, whether by a Message or through a serial Command string. To configure a LOCKed Output Pin, first UNLOCK the pin. The LOCK function allows you to implement a two-level priority system for the Output Pin, and prevent unintentional reconfiguration of the pin.

**Time Out Status:** The Output Pin contains a Time Out function, that you can RUN and HALT. The Time Out function is disabled in the HALTed state. Setting the Time Out to RUN enables the Time Out function, but does not automatically turn the Output Pin ON, you must do this explicitly. At the expiration of the Output Time Out interval, the MDS automatically turns the Output Pin OFF. You can use this function to issue fixed length pulses over the Output Pin.

**Time Out Value:** Specify the value the MDS should use for the Output Time Out. The MDS stores a default Output Pin Time Out value in its Configuration File. RUN the Time Out from this value, or specify a separate value. You can also RESUME the Time Out function from its HALTed value. The Output Pin Time Out value can range from 10 to 1260 milliseconds, 1 to 63 seconds, or 1 to 63 minutes.

Configure the Output Pin with a Message by inserting the Output Pin control code sequence, followed by a three-digit configuration value in the message text. The Output Time Out value can be set by inserting the Output Time Out control code sequence, followed by a three-digit time out value in the message text. (See *Message String Features And Assembly*, page 12, for details on configuring the Output Pin in a Message string.)

## HANDSHAKING

The MDS can be configured for one of two Handshaking protocols. The Handshaking protocol determines how an MDS responds to various interactions with PLC's, computers, and other controlling devices. Through the chosen protocol, a controlling unit can monitor the status of one or more MDS's with which it communicates. The choice of Handshaking protocols is based on the type of interface established between the controlling unit and the MDS. The XON/XOFF protocol applies to serial interfaces, the READY/BUSY protocol applies to Parallel Port, or hardware interfaces.

Applications involving PLC's, computers, or extensive operator interaction with the MDS should specify one of the Handshaking protocols. Applications with little or no timing considerations may not need a Handshaking protocol. In addition, applications limited strictly to RLC products should NOT use a Handshaking protocol.

### XON/XOFF

With the MDS configured to follow the XON/XOFF protocol, the unit transmits the ASCII XOFF character (13h - 19) when it enters a BUSY condition, and the XON character (11h - 17) when it enters the READY condition. The MDS responds to all Message requests by transmitting an XOFF. When the MDS has completed processing the current Message, it transmits the XON character, signifying that it is READY to process the next Message request or Command string. While configured for the XON/XOFF protocol, the MDS always transmits error codes for any error conditions it encounters, regardless of the Error Handling configuration setting. You can set and verify the XON/XOFF Serial Port configuration setting with Command C05.

### Busy Mode

With the Output/Busy pin configured for the BUSY mode, the MDS follows the READY/BUSY protocol. Whenever the MDS enters the BUSY condition, it sets the Output Pin to the BUSY state. When the unit returns to the READY condition, the MDS sets the Output Pin to the READY state. The READY and BUSY states are defined by the Output Pin Logic Level configuration setting. For POSITIVE logic, the pin conducts whenever the MDS is READY. For NEGATIVE logic, the pin conducts whenever the MDS is BUSY.

The MDS responds to all Message and Command strings by taking the Output Pin to the BUSY state. Once the MDS has completed processing the received string, it returns the Output Pin to the READY state, signifying that the unit is now READY to receive the next Message or Command string.

| | READY STATE | BUSY STATE |
|---|---|---|
| POSITIVE LOGIC | Conducts | Does Not Conduct |
| NEGATIVE LOGIC | Does Not Conduct | Conducts |

You can set and verify the Output/Busy Pin configuration settings with Command C06.

## PARALLEL PORT (optional)

The MDS's Parallel Port is an input-only port and accepts information in the 8-bit BCD format. The port consists of two CTRL lines (STROBE and ITEM/ADDR) and eight DATA lines. The DATA lines and the CRTL lines are separately DIP switch selectable for current Sinking or Sourcing, and Negative or Positive logic.

The MDS determines its mode of operation on every unit power-up according to the settings on the Parallel Port. With a BCD thumbwheel or rotary switch, or any other device that asserts a two-digit (8-bit) BCD number, attached to the Parallel Port, the MDS always powers-up in the Transmit Mode. The MDS Transmit Mode is the only mode that accesses the Parallel Port during normal operation, and is the default mode for MDS units with the Parallel Port. (See *Selecting A Mode Of Operation*, page 6, for details).

While in the Transmit Mode, the MDS accepts a Target Unit Address and a Target Data Item number over the Parallel Port in BCD. Applying a valid pulse to the STROBE line (3 msec minimum) causes the MDS to read the ITEM/ADDR line and the DATA lines. The ITEM/ADDR line value indicates how the MDI should interpret the DATA lines, either as a Unit Address (ADDR), or as a Data Item (ITEM).

The Parallel Port functions in the following manner. Upon detecting a valid STROBE pulse (3 msec minimum length), the MDS begins Sampling the Parallel Port DATA lines and the ITEM/ADDR Control line. These lines must remain stable for the duration of one Sample time in order for the MDS to accept their values. Once the programmed Sample Time (1 to 255 msec) has been attained, the MDS processes the DATA lines according to the ITEM/ADDR line value. The MDS restarts the Sample Time if any of the lines change level (noise) before the Sample time expires.

A programmable Debounce Time (10 to 2550 msec) commences after the STROBE line returns to the inactive state. This Debounce Time effectively removes unwanted contact bounce and noise spikes on the STROBE line, thus eliminating spurious STROBE pulses. Any additional activity on the STROBE line during the Debounce time causes the MDS to restart the Debounce Timer. Apart from this, the MDS ignores the STROBE line activity during the Debounce Time. Once the Debounce Time expires, the MDS is again ready to accept information over the Parallel Port in the same sequence.

You can set and verify a unit's current Parallel Port configuration settings with the C04 Command. See *Command C04 - Configure Parallel Port*, page 33, for details on the C04 Command.

### ITEM/ADDR Line

The ITEM/ADDR line indicates whether the information on the DATA lines is a Unit Address or a Data Item number. The ADDR level indicates an Address, the ITEM level indicates an Item number.

The appropriate ITEM and ADDR levels are shown below for the two CRTL LOGIC DIP switch positions.

| CRTL LOGIC | ADDR LEVEL | ITEM LEVEL |
|---|---|---|
| NEG (DOWN) | LO | HI |
| POS (UP) | HI | LO |

### STROBE Line

The MDS begins sampling the Parallel Port DATA lines (DO-D7) and the ITEM/ADDR line after detecting a valid length pulse on the STROBE line. A valid STROBE pulse consists of any transition to the active state of the STROBE line for 3 milliseconds or longer while the Debounce timer is not running.

The MDS accepts the port line values at the end of the programmed Sample time, whether or not the STROBE line has returned to the inactive state. The Sample time tells the MDS how long the port must be stable before accepting the line values as valid, and is programmable from 1 to 255 milliseconds. The MDS automatically adjusts the Debounce time to always be longer than the Sample time.

The Debounce timer starts when the STROBE line returns to its inactive state and determines how long the MDS waits before again accepting STROBE pulses. The Debounce time eliminates spurious STROBE pulses due to contact bounce and noise spikes, and is programmable from 10 to 2550 milliseconds. The MDS automatically adjusts the Sample time to always be shorter than the Debounce time.

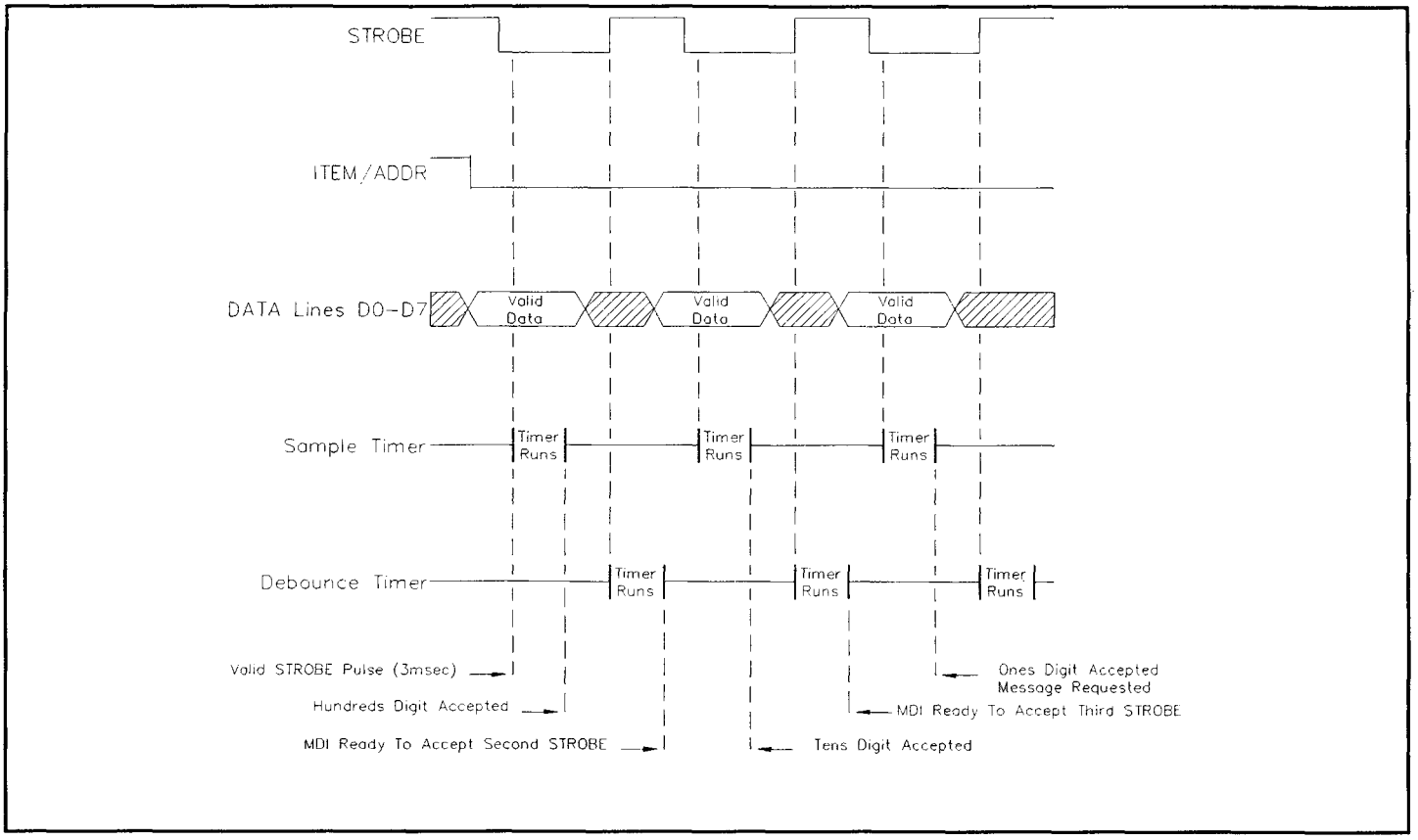



Figure 4 - Parallel Port Timing (Transmit Mode)

### *STROBE Line (Cont'd)*

The appropriate STROBE levels are shown below for the two CRTL LOGIC DIP switch positions.

| CRTL LOGIC | ACTIVE STROBE LEVEL |
|---|---|
| NEG (DOWN) | LO |
| POS (UP) | HI |

#### *DATA Lines (D0 - D7)*

The MDS always interprets the information on the DATA lines as a two digit BCD value in the range from 00 to 99. Upon receiving a valid STROBE pulse, the MDS converts the 2-digit BCD number to its decimal equivalent according to the following tables, and then issues a request for the selected Unit Address, or Data Item number.

The appropriate DATA PIN levels are shown for the two DATA LOGIC DIP switch positions.

| DATA LOGIC | DATA '1' LEVEL | DATA '0' LEVEL |
|---|---|---|
| NEG (DOWN) | LO | HI |
| POS(UP) | HI | LO |

| BCD TABLE | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| DATA LINES | | | | BCD | DATA LINES | | | | BCD |
| D7 | D6 | D5 | D4 | VALUE | D3 | D2 | D1 | D0 | VALUE |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 10 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 20 | 0 | 0 | 1 | 0 | 2 |
| 0 | 0 | 1 | 1 | 30 | 0 | 0 | 1 | 1 | 3 |
| 0 | 1 | 0 | 0 | 40 | 0 | 1 | 0 | 0 | 4 |
| 0 | 1 | 0 | 1 | 50 | 0 | 1 | 0 | 1 | 5 |
| 0 | 1 | 1 | 0 | 60 | 0 | 1 | 1 | 0 | 6 |
| 0 | 1 | 1 | 1 | 70 | 0 | 1 | 1 | 1 | 7 |
| 1 | 0 | 0 | 0 | 80 | 1 | 0 | 0 | 0 | 8 |
| 1 | 0 | 0 | 1 | 90 | 1 | 0 | 0 | 1 | 9 |
| 1 | 0 | 1 | 0 | Invalid | 1 | 0 | 1 | 0 | Invalid |
| 1 | 0 | 1 | 1 | Invalid | 1 | 0 | 1 | 1 | Invalid |
| 1 | 1 | 0 | 0 | Invalid | 1 | 1 | 0 | 0 | Invalid |
| 1 | 1 | 0 | 1 | Invalid | 1 | 1 | 0 | 1 | Invalid |
| 1 | 1 | 1 | 0 | Invalid | 1 | 1 | 1 | 0 | Invalid |
| 1 | 1 | 1 | 1 | Invalid | 1 | 1 | 1 | 1 | Invalid |

## SERIAL COMMUNICATIONS

The MDS's serial port consists of two 20 milliamp current loops which provide full duplex, isolated, two-way communications. ASCII terminals, programmable controllers, computers, and Red Lion Controls products with 20 milliamp current loop communications are just some of the devices that can be connected to the serial port. Multiple units can be connected to the serial loop, but the actual number of serial connections that the MDS can drive is limited by its serial hardware specifications.

There are two separate loops, a transmit loop (SO+ & SO-) and a receive loop (SI+ & SI-). The built in +20mA current source can drive either the transmit loop or the receive loop. The -20mA SRC (Common) is the current loop return connection. To minimize problems with ground loops, keep the -20mA SRC isolated from earth ground.

The following operations can be performed over the serial port:

Uploading and Downloading Files
Transmitting and Receiving Message Text
Receiving Commands
Collecting Data from other RLC Units (Transmit Mode only)
Transmitting Error Codes

You can set and verify the Serial Port configuration settings with the C05 Command. Through the appropriate DIP switch settings, the MDI can perform a serial loop-back test to verify the proper operation of its serial communications hardware.

### Communication Format

The MDS transmits characters by switching the current ON and OFF in its 20 milliamp transmit current loop, and it receives characters by monitoring the switching action on its 20 milliamp receive loop. The data format and baud rate must be identical between the transmitting and receiving units for the data to be interpreted correctly.

The baud rate, parity bit, and data bit parameters are DIP switch selectable.

The available baud rates are; 300, 600, 1200, 2400, 4800, 9600, and 19200. The available data formats are:

11 bits: 1 start bit, 8 data bits, Parity bit, 1 stop bit.
10 bits: 1 start bit, 8 data bits, 1 stop bit.
10 bits: 1 start bit, 7 data bits, Parity bit, 1 stop bit.
9 bits: 1 start bit, 7 data bits, 1 stop bit.

#### Unit Address

If only one MDS is present on a loop, an address number of "0" may be assigned to that unit. Assigning an address of zero to an MDS eliminates the need for including an address identifier when communicating with that unit. With multiple MDS's present on the loop, it is recommended that you assign each MDS a unique address, from 1-99, unless you want more than one unit to respond indentically to a given transmission. Although there are 100 valid addresses, the MDS's built in current source, if used, can only drive a loop with a maximum of 6 units on it. You can set the unit address of an MDS with the C01 Command.

If using an MDI to transmit to an MDS in the RLC Mode, the message in the MDI must have the MDS UNIT Destination selected and a non-zero address assigned to the MDS.

### Serial Port Operation

The MDS utilizes two buffers for its Serial Port, a 128 byte Receive Buffer and a 256 byte Transmit Buffer. The MDS can service both the Receive Buffer and the Transmit Buffer at the same time.

### *Serial Receive Buffer*

With a few exceptions, the MDS places all characters it receives directly into its receive buffer. The MDS loads characters into the buffer, without processing them, until it receives its programmed Serial Terminator character. The Terminator character informs the MDS that it should process the current contents of its Receive buffer.

The MDS can accept character strings up to 128 bytes in length, including the Terminator character. Should the MDS receive a 129th character without an intervening Terminator, it empties the Receive buffer and then places the 129th character in the buffer. The MDS DOES NOT issue an Error Code for this condition.

Several characters invoke special Receive buffer functions. The BACKSPACE character (08h - 8) causes the MDS to back up one character position in its Receive buffer, effectively deleting the last character received. Issuing repeated BACKSPACEs to an MDS eventually empties its Receive buffer. The MDS ignores the BACKSPACE character while its Receive buffer is empty.

The ESCAPE <ESC> character (1Bh - 27) causes the MDS to position its Receive buffer at the beginning. In effect, the <ESC> character deletes all characters sent to the MDS since the last Terminator character, clearing the Receive buffer.

The ASCII NULL character (00h - 0) acts as a Universal Serial Terminator character for all MDI and MDS units. Pressing <CTRL-END> while in the Terminal Emulator issues this character. See the Serial Terminator description later in this section for details.

The <CTRL-N> character (0Eh - 14) causes an MDS in the RLC Mode to insert its unit address prefix string into the Receive buffer at the current character position. Hitting <CTRL-HOME> while in the SFMD Terminal Emulator issues an <ESC>, that empties the unit's receive buffer first and then performs the same function as above. This function, combined with the <ESC> character and the Universal Terminator, allows you to access any MDI or MDS without knowing the unit address or the Serial Terminator of the unit.

Example: The character sequence "<CTRL-HOME>C1<CTRL-END>", issued from the SFMD Terminal Emulator to an MDS unit in the **RLC Mode**, causes the following actions:

<CRTL-HOME> - The SFMD Terminal Emulator transmits an <ESC> character, that clears the MDS buffer, followed by the <CTRL-N> character, causing the MDS to insert its unit address prefix in the Receive buffer. If the address is 34, the character string "N34" now appears in the Receive buffer.

C1 - The MDS loads these characters into its Receive buffer, which invoke Command C01 when the unit processes the Receive buffer.

<CTRL-END> - The Terminal Emulator transmits an ASCII null character, causing the MDS to terminate the character string in its Receive buffer and begin processing the string. The Receive buffer now contains: "N34C1". While processing the Receive buffer contents, the MDS identifies its own unit address, and consequently processes the remainder of the string.

The character string issues a request for Command C01, which causes the MDS to transmit its Unit Function configuration to the Terminal Emulator. The MDS's unit address, which was unknown to the operator, is included in this transmission.

*Note: Multiple MDI/MDS units on a loop will all receive and execute the preceding transmission, thus causing a collision between the multiple transmissions. This is not harmful to the units or the loop, but does result in a corrupted transmission.*

### *Serial Transmit Buffer*

All characters transmitted by the MDS originate from the Transmit Buffer. The MDS assembles a transmission string in its Transmit buffer before sending that string out on the serial port. Transmission strings are limited in length by the size of the Transmit buffer. The MDS can transmit character strings of 255 bytes, including the designated string terminator, which is usually the MDS's Serial Terminator character.

You can PAUSE, RESUME, and ABORT certain MDS transmissions by issuing the <ESC> and <CR> characters. While transmitting responses for any of the Commands, the MDS accepts the <ESC> character as a signal to PAUSE the transmission at the end of the current line. The MDS RESUMEs the transmission once it receives a <CR> character. If the MDS receives a second <ESC> character before the transmission is RESUMEd, the unit aborts the transmission by transmitting a closing <CR><LF> pair.

*Note: The MDS cannot process Message Time Outs, Commands, or any new Message strings, while a transmission is PAUSEd.*

### *Serial Terminator Character*

The MDS utilizes its Serial Terminator character both to recognize that it should process its Receive buffer, and to terminate most of its transmission strings in the Transmit Buffer. An MDS in the Transmit Mode always uses the '*' character to terminate its transmissions, and always looks for a <CR> when receiving a response from a unit.

With the MDS in either the RLC or the Serial Mode, you can program the Terminator in the range from 01h (1) to 40h (64), excluding the BACKSPACE character (08h - 8), the ESCAPE character (1Bh - 27), and the 10 numeric characters (30h - 48) to (39h - 57). As with all RLC products with serial communications, the default value for the Terminator is '*'.

A Universal Terminator character is also available for your use, the ASCII NULL character (00h - 0). All MDI and MDS units respond to this character as they do to their respective programmed Serial Terminator character. Therefore, if you forget, or cannot identify a particular unit's programmed Terminator, you can use this character in the interim. Many ASCII terminals and Terminal Emulation programs do not transmit the NULL character. However, you can issue this character from the RLC Terminal Emulator by pressing <CTRL-END>.

### *Serial Transmit Delay*

All Red Lion Controls products containing serial ports utilize Receive buffers. When a particular RLC unit receives its Terminator character, it must interrupt the current task to process the contents of the Receive buffer.

This interruption takes some finite amount of time, on the order of 10's or even 100's of milliseconds. The unit cannot receive any additional characters while it is processing the buffer, and, once finished, must return to the interrupted task at hand. Therefore, a method for establishing a minimum time between consecutive transmissions is necessary.

The MDS's Serial Transmit Delay setting controls the minimum delay time between the end of one transmission string and the beginning of the next string from the MDS. The Transmit Delay value can range from 0 seconds (NO delay) to 2.5 seconds, in 0.01 second increments. The MDS automatically runs a Transmit Delay at the end of every transmission string. It also inserts Transmit Delays after <CR><LF> pairs in all of its Command response transmissions.

In addition, provisions have been made to insert Transmit Delays anywhere in the Message Text that you deem necessary or essential. If your application requires that the MDS transmit its displayed Message, you can force the MDS to run a Transmit Delay at a particular location in the Message Text by inserting the <CTRL-T> character (14h - 20) at that location. When the MDS encounters the <CTRL-T> character while processing its Transmit buffer, it runs a Transmit Delay instead of transmitting the character. Once the Transmit Delay has expired, the MDS continues on to the next character in the buffer.

## *Serial Port Timing Diagrams*

When interfacing with the MDS over its Serial Port, certain considerations must be made regarding the timing of the various serial receptions and transmissions. Reliable communication over the Serial Port between the MDS and any units connected to it is dependent upon several factors; the serial baud rate, the serial terminator, and the serial transmit delay.

The serial baud rate setting directly affects the transmission and reception time for a character string. At 300 baud, the MDS takes approximately one second to transmit or receive a string of 30 characters. At 9600 baud, however, the same character string can be transmitted or received in 30 milliseconds. The following table lists the valid baud rates and the transmission time per character for each baud rate.

| | |
|---|---|
| 300 | 33.3 msecs/character |
| 600 | 16.7 msecs/character |
| 1200 | 8.3 msecs/character |
| 2400 | 4.2 msecs/character |
| 4800 | 2.1 msecs/character |
| 9600 | 1.0 msecs/character |
| 19200 | 0.5 msecs/character |

## *Serial Port Timing Diagrams (Cont'd)*

All serial communications with the MDS are buffered. The MDS places all characters it receives in its Serial Receive Buffer until the appropriate Serial Terminator character is received. Upon receipt of the Terminator, the MDS "reads" the contents of the Receive Buffer, and performs the action the character string in the buffer specifies. If the MDS does not "understand" the character string, or the string is not addressed to the MDS, the MDS clears its receive buffer, and no action is taken. If the string requests a valid function, such as a Message string or a Command, and the MDS subsequently finds an error in the data supplied for the function, the unit generates an Error Code, which is then processed according to the configured Error Handling method. (See SERIAL RECEIVE BUFFER, SERIAL TERMINATOR, and ERROR HANDLING sections for details).

A Serial Transmit Delay causes the MDS to delay transmitting the next character until the Delay time expires. Transmit Delays are incorporated in and between transmissions to accommodate the various serial timing and processing constraints of the units with which the MDS must communicate. (See the Serial Transmit Delay section for details).

The following diagrams illustrate the different timing constraints the MDS adheres to when receiving and processing Commands and Message strings. The READY/BUSY state of the MDS is shown for the various stages of processing. For the BUSY Mode of the Output Pin function, these states correspond to the appropriate Output Pin logic levels. For the XON/XOFF handshaking mode, the MDS's XON and XOFF transmissions are shown, as well. (See *Handshaking*, page 19, for details).

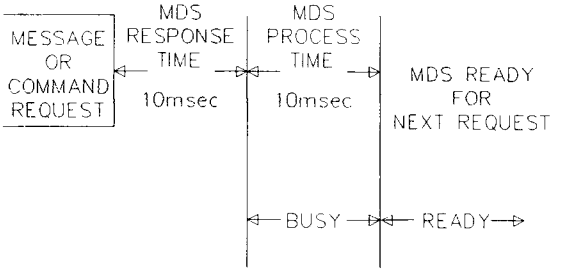



Figure 5 - READY/BUSY Handshake Protocol

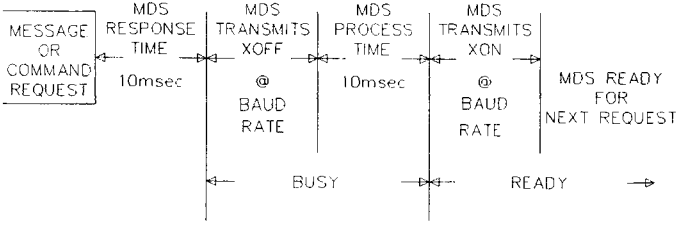



Figure 6 - XON/XOFF Handshake Protocol

## *UNIT POWER-UP AND RESET*

Upon the application of power (unit power-up), the MDS executes the following sequence of Reset functions in the RLC and Serial Modes:

1. Loads the default CHARACTER file from the MDS's ROM.
2. Loads the default DAY/MONTH table from the MDS's ROM.
3. Loads the Configuration file settings from the MDS's nonvolatile memory.
4. Sets the Output Pin to the current Configuration File settings. If the Output Pin is configured for the BUSY mode, the MDS sets the pin to the BUSY state. If the pin is in the OUTPUT mode, the MDS sets the pin to the OFF state. The actual voltage level of the pin is determined by the current Output Pin Logic Level configuration setting.
5. The MDS clears the configurations of all 16 Elapsed Timers. The MDS turns a CLEARed Timer OFF, sets the Timer's direction to DOWN and resets the value to 0000:00:00.00.
6. The MDS blanks its display and then enters the READY condition. If the XON/XOFF handshaking protocol is enabled, the MDS issues an XON character over the serial port. If the READY/BUSY protocol is in use, the MDS places the Output Pin in the READY state.

Upon the application of power (unit power-up), the MDS executes the following sequence of Reset functions in the Transmit Mode:

1. Reads the Target Unit Address and Target Data Item number from the nonvolatile memory.
2. Begins issuing Print Requests to the Target Unit.
3. Waits to receive the Target Data Item or times out according to the Serial Port ED Time Out value.
4. Displays the requested Target Data Item, if received.

# COMMANDS

The MDS contains an extensive Command structure, which is available in both the RLC and Serial Modes of operation. A full set of Commands is available to you for both configuring and interrogating the MDS. Commands can only be issued to the MDS by transmitting the desired Command string to the unit over the serial port.

You can use Commands to cause the MDS to transmit any of its current Configuration File settings in an easily readable mnemonic format. You can change any of these configuration settings *on-line* by issuing the appropriate Command. Commands also provide access to the Elapsed Timers and the Output Pin. Command String format and assembly is described in the following sections.

## Assembling Commands

### RLC Mode

The MDS can be assigned a specific Unit Address, which allows you to direct Message strings and Commands to specific units on the loop. In the RLC Mode, you must access an MDS unit having an address other than 0 with the following transmission format:

**NnnCxxdd..d<term>**

'N' - Signifies to the MDS that a unit address follows.

nn - The address number of the unit for which the Command string is intended, leading zeroes need not be included.

'C' - Signifies that a Command number follows.

xx - The number of the Command to be executed, leading zeroes need not be included.

dd..d - Represents any data the Command may require.

<term> - Programmed Serial Terminator.

In the RLC Mode, if an MDS has a unit address of 0, you do not need to include the Address identifier, "Nnn", in the Command string:

**Cxxdd..d<term>**

### Serial Mode

In the Serial Mode, if you wish to access a particular unit on the loop, you must substitute the <CTRL-N> character (0Eh - 14) for the Address identifier character, 'N':

**<CTRL-N>nnCxxdd..d**

If you want to issue a Command to all Serial units on the loop, you must eliminate the Address identifier string, "<CTRL-N>nn", and substitute the <CTRL-C>character for the Command identifier character, 'C':

**<CTRL-C>xxdd..d**

Any character string not preceded with either of the <CTRL-N> or <CTRL-C> characters is displayed in its entirety by every Serial Mode MDS unit receiving the string.

### Transmit Mode

You cannot issue Commands to an MDS in the Transmit Mode. If you need to configure an MDS unit that will be used in the Transmit Mode, you must first place it in either the RLC or the Serial Mode, issue the desired Commands to the unit, and then return the unit to the Transmit Mode.

***All future references to Commands assume that the RLC Mode of operation is selected, the Unit Address is set to 0, and the programmed Serial Terminator is issued at the end of the string.***

### *Command Syntax*

To help you locate and interpret command syntax, the following conventions are used in this document:

1. "UPPER-CASE" characters represent Command Codes and should be entered exactly as shown.
2. "lower-case" characters indicate that a variable field is to be filled in. The number of characters shown for the field indicates the maximum size of the field. Leading zeroes need not be included in numeric fields except where specifically indicated for a particular command.
3. "dd...d" indicates that a variable length field or variable length list should be entered.

Commands C01 through C08 accept multiple Command codes in a single data string.

Example: "C07T1P0UN"

"C07" - Command C07 configures the Elapsed Timer.
"T1" - TIMER Command Code. Specifies TIMER 1.
"P0" - PRESET Command Code. PRESETs the Timer to the value following the Command Code. "0" in this case.
"U" - UP Command Code. Sets the Timer direction to UP.
"N" - ON Command Code. Turns the Timer ON.

All Commands can be issued directly through their respective Command numbers, as shown above. A few selected Commands can also be issued with a Mnemonic code related to the specific function of the Command.

These selected Commands are listed below, along with their respective Mnemonic codes:

| MNEMONIC CODE | COMMAND NUMBER | DESCRIPTION |
|---|---|---|
| M | C21 | Display Message Text - The 'M' Mnemonic can be used in place of "C21", i.e. "C21M:Message" and "M:Message" both place the word "Message" on the display. |
| T | C07 | Elapsed Timer Command - The 'T' Mnemonic can be used in place of "C07", that is, "C07T1P0UN" and "T1P0UN" both set Timer 1 to 0, set the direction to UP, and turn Timer 1 ON. |

Command C07 allows you to access multiple Elapsed Timers in a single Command string with the '/' character. You can only request one transmission per command string and the transmission request must be the last entry in the string.

Example: "T1P0UN/T15F/T15"

"T1P0UN" Configures TIMER 1 with a PRESET of 0, sets its direction to UP, and turns the Timer ON
'/' Indicates another Command string follows.
"T15F" Turns TIMER 15 OFF.
'/' Indicates another Command string follows.
"T15" Causes the MDS to transmit Timer 15's current configuration.

*Note: The MDS executes a Command function as it processes the Command code and data for that function. If a Command is entered incorrectly, the MDS aborts processing for that Command at the first error it encounters and issues an Error Code based on the selected Error Handling method. All Command functions executed prior to the error remain in effect and all Command functions following the error are ignored.*

**MDS COMMAND LIST**

| COMMAND | FUNCTION | MNEMONIC |
|---|---|---|
| C01 | CONFIGURE UNIT FUNCTIONS | |
| C02 | CONFIGURE SOFTWARE CLOCK | |
| C03 | UNUSED | |
| C04 | CONFIGURE PARALLEL PORT | |
| C05 | CONFIGURE SERIAL PORT | |
| C06 | CONFIGURE OUTPUT/BUSY PIN | |
| C07 | CONFIGURE ELAPSED TIMERS | T |
| C08 | CONFIGURE UNIT DISPLAY | |
| C09 | UNUSED | |
| C10 | UNUSED | |
| C11 | UNUSED | |
| C12 | UNUSED | |
| C13 | UNUSED | |
| C14 | UNUSED | |
| C15 | UNUSED | |
| C16 | UNUSED | |
| C17 | RESTORE DEFAULT UNIT CONFIGURATION | |
| C18 | LOAD CHAR AND DAY/MONTH FILES | |
| C19 | HALT/RESET UNIT | |
| C20 | CLEAR DISPLAY | |
| C21 | DISPLAY MESSAGE TEXT | M |
| C22 | UNUSED | |
| C23 | TRANSMIT MESSAGE TEXT | |
| C24 | TRANSMIT CHARACTER STRING | |

### *Configure Unit Functions - C01*

Command C01 configures the MDS's Unit Function Configuration settings. The following is the list of valid Command string formats for the C01 Command.

*Note: The MDS does not have a Real Time Clock, consequently the clock is set to zero on unit power-up.*

| | |
|---|---|
| C01 | The MDS transmits the current Unit Function Configuration settings as shown.<br>`FUNCTIONS CFG`<br>`ADDR-aa CLK-n/f DFLT- ERR-t`<br>`PRDC- QUE- RST-`<br>Mnemonic — **Description**<br>ADDR — Unit Address, aa (0-99)<br>CLK — Clock Status, n/f (ON/OFF)<br>DFLT — Not implemented in MDS<br>ERR — Error Handling Type, t [Both (B), Display (D), Ignore (I), Transmit (T)]<br>PRDC — Not implemented in MDS<br>QUE — Not implemented in MDS<br>RST — Not implemented in MDS |
| C01Aaa | Sets the Unit ADDRESS to aa (0-99). |
| C01CF | Turns the Unit CLOCK function OFF, the internal CLOCK halts. |
| C01CN | Turns the Unit CLOCK function ON, the internal CLOCK runs. |
| C01EB | Sets the Unit ERROR Handling function to BOTH Display and Transmit Error messages. |
| C01ED | Sets the Unit ERROR Handling function to DISPLAY Error messages. |
| C01EI | Sets the Unit ERROR Handling function to IGNORE Error messages. |
| C01ET | Sets the Unit ERROR Handling function to TRANSMIT Error messages. |

*Note: Command codes can be strung together to configure multiple settings in a single command string.*

**Example:** C01A01CNEI

A01 - Sets the Unit Address to 1.
CN - Runs the internal software CLOCK.
EI - Sets the ERROR handling method to IGNORE.

### *Configure Unit Clock - C02*

Command C02 configures the MDS software Clock. The following is the list of valid command string formats for the C02 Command.

| | |
|---|---|
| C02 | The MDS transmits the current software CLOCK configuration as shown.<br>`CLOCK CFG - n/f`<br>`DATE: mm-dd-yy a`<br>`TIME: hh:nn:ss.uu`<br>n/f - ON/OFF (Clock is running or halted)<br>mm - MONTH (01-12)<br>dd - DATE (01-31)<br>yy - YEAR (00-99)<br>a - DAY (1-7, Sunday is 1)<br>hh - HOUR (0-23)<br>nn - MINUTE (0-59)<br>ss - SECOND (0-59)<br>uu - HUNDREDTHS (00-99)<br>*Note: The years 1980 to 1999 are indicated by 80 to 99, years 2000 to 2079 are indicated by 00 to 79.* |
| C02Dmm-dd-yy a | Sets the DATE to mm-dd-yy a.<br>mm - MONTH (01-12)<br>dd - DATE (01-31)<br>yy - YEAR (01-99)<br>a - DAY (1-7, Sunday is 1)<br>*Note: Leading zeroes must be entered.*<br>*Note: The years 1980 to 1999 are indicated by 80 to 99, years 2000 to 2079 are indicated by 00 to 79.* |
| C02Thh:nn:ss | Sets the TIME to hh:nn:ss.<br>hh - HOUR (00-23)<br>nn - MINUTE (00-59)<br>ss - SECOND (00-59)<br>*Note: Leading zeroes must be entered.*<br>*Note: The years 1980 to 1999 are indicated by 80 to 99, years 2000 to 2079 are indicated by 00 to 79.* |

| | |
|---|---|
| C02F | Turns the CLOCK function OFF. The CLOCK halts. |
| C02N | Turns the CLOCK function ON. The CLOCK runs. |
| C02+D | Adjusts the Date forward 1 DAY. |
| C02-D | Adjusts the Date backward 1 DAY. |
| C02+tttD | Adjusts the Date forward ttt (1-255) DAYs. |
| C02-tttD | Adjusts the Date backward ttt (1-255) DAYs. |
| C02+H | Adjusts the Time forward 1 HOUR. |
| C02-H | Adjusts the Time backward 1 HOUR. |
| C02+tttH | Adjusts the Time forward ttt (1-255) HOURs. |
| C02-tttH | Adjusts the Time backward ttt (1-255) HOURs. |
| C02+M | Adjusts the Time forward 1 MINUTE. |
| C02-M | Adjusts the Time backward 1 MINUTE. |
| C02+tttM | Adjusts the Time forward ttt (1-255) MINUTEs. |
| C02-tttM | Adjusts the Time backward ttt (1-255) MINUTEs. |
| C02+S | Adjusts the Time forward 1 SECOND. |
| C02-S | Adjusts the Time backward 1 SECOND. |
| C02+tttS | Adjusts the Time forward ttt (1-255) SECONDs. |
| C02-tttS | Adjusts the Time backward ttt (1-255) SECONDs. |

*Note: Command codes can be strung together to configure multiple settings in a single command string.*

**Example**: C02D09-20-92 1T09:05:00N

D09-20-92 1 - Sets the DATE to 9-20-92, SUNDAY.

T09:05:00 - Sets the TIME to 9:05:00 AM.

N - Turns the INTERNAL Clock ON.

### *Configure Parallel Port (PRL Port) - C04*

Command C04 configures the Parallel Port settings. The following is the list of valid command string formats for the C04 Command.

| | |
|---|---|
| C04 | The MDS transmits the current PRL PORT configuration as shown.<br>`PRL PORT CFG - BCD 8 Bits`<br>`DB-ddd0ms SM-sssms TO-`<br>DB - Debounce Time, ddd, in hundredths of seconds (1-255)<br>SM - Sample Time sss, in milliseconds (1-255)<br>TO - Not implemented in MDS |
| C04Dttt | Sets the PRL Port DEBOUNCE Time to ttt (1-255) in hundredths of seconds *(the SAMPLE time is automatically adjusted to be less than the DEBOUNCE time).* |
| C04Sttt | Sets the PRL Port SAMPLE Time to ttt (1-255) milliseconds *(the DEBOUNCE time is automatically adjusted to be greater than the SAMPLE time).* |

*Note: Command codes can be strung together to configure multiple settings in a single command string.*

**Example**: C04D5S3

D5 - Sets the DEBOUNCE Time to 50 milliseconds.
S3 - Sets the SAMPLE Time to 3 milliseconds.

### *Configure Serial Port (SRL Port) - C05*

Command C05 configures the Serial Port. The following is the list of valid command string formats for the C05 Command.

| | |
|---|---|
| C05 | The MDS transmits the current SRL PORT configuration as shown.<br>`SRL PORT - CFG - d,p,bbbbb`<br>`DL-ddd0ms ST-sss TO-ttts X-n/f`<br>d - Number of Data bits (7/8)<br>p - Parity type (Even (E), Odd (O), None (N))<br>bbbbb - baud rate (300 - 19200)<br>DL - Transmit Delay, ddd, in hundredths of seconds (0-255)<br>ST - Serial Terminator, sss, in decimal (1-7, 9-26, 28-47, 58-64, decimal)<br>TO - Embedded Data Time Out, ttt, in seconds (OFF, 0-254)<br>X - XON/XOFF Protocol Status (ON/OFF) |
| C05Dttt | Sets the SRL Port Transmit DELAY Time to ttt (0-255) in hundredths of seconds. |
| C05Ssss | Sets the SRL Port Receive STRING terminator to sss (1-7, 9-26, 28-48, 58-64). |
| C05TF | Turns the SRL Port DEFAULT Embedded Data TIME OUT function OFF - NO Time Out. |
| C05Tttt | Sets the SRL Port DEFAULT Embedded Data TIME OUT to ttt (0-254) seconds. |
| C05XF | Turns the SRL Port XON/XOFF function OFF (automatically selects the BUSY function for the OUTPUT/BUSY pin, if previously DISABLED). |
| C05XN | Turns the SRL Port XON/XOFF function ON (automatically DISABLEs the BUSY function for the OUTPUT/BUSY pin, if previously selected). |

*Note: Command codes can be strung together to configure multiple functions in a single command string.*

**Example**: C05D5S13T2

D5 - Sets the SRL Port Transmit DELAY Time to 50 milliseconds.
S13 - Sets the Receive STRING terminator to 13, <CR>.
T2 - Sets the DEFAULT Embedded Data TIME OUT value to 2 seconds.

### *Configure Output/Busy Pin - C06*

Command C06 configures the Output/Busy Pin. The following is the list of valid command string formats for C06.

| | |
|---|---|
| C06 | The MDS transmits the current OUTPUT/BUSY Pin configuration as shown.<br>`OUTPUT PIN CFG`<br>`abcde TO-ttt uuu`<br>`TO REM mm:ss.uu`<br>a - Enabled (E) or Disabled (D) with the Pin in the Output Mode, Busy (B) with the Pin in the Busy Mode<br>b - Positive logic (+) or Negative logic (-)<br>c - Output is ON (N) or OFF (F)<br>d - Output Time Out Value is RUNning (R) or HALTed (H)<br>e - Output Pin is LOCKed (L) or UNLOCKed (U)<br>TO - Output Default Time Out value, ttt, and units, uuu, (min/sec/10 ms)<br>TO REM - Time Out remaining in minutes (mm), seconds (ss), and hundredths of seconds (uu). Transmits "—TIMED OUT—" if the Time Out value has expired |
| C06B | Selects the BUSY Mode of the OUTPUT/BUSY Pin Function. Turns the SRL Port XON/XOFF Function OFF, if previously turned ON. |
| C06D | DISABLEs the OUTPUT/BUSY Pin Function. BUSY is selected if the SRL Port XON/XOFF Function is OFF. |
| C06E | ENABLES the OUTPUT Mode of the OUTPUT/BUSY Pin Function. |
| C06+ | Selects POSITIVE logic for the OUTPUT/BUSY Pin - Outputs a '1' (does not conduct) in the ON/BUSY state when pulled-up with a resistor. |
| C06- | Selects NEGATIVE logic for the OUTPUT/BUSY Pin - Outputs a '0' (conducts) in the ON/BUSY state when pulled-up with a resistor. |
| C06F | Turns the OUTPUT Pin OFF, if the OUTPUT function is ENABLED. |
| C06G | TOGGLES the state of the OUTPUT Pin, from ON to OFF, or OFF to ON, if the OUTPUT Function is ENABLED. |
| C06N | Turns the OUTPUT Pin ON, if the OUTPUT function is ENABLED. |
| C06R | RUNs the OUTPUT Time Out Timer - does NOT automatically turn the OUTPUT Pin ON. |
| C06H | HALTs the OUTPUT Time Out Timer - does NOT automatically turn the OUTPUT Pin OFF. |
| C06RD | RUNs the OUTPUT Time Out Timer and sets the Timer to the DEFAULT OUTPUT Time Out value. |
| C06RtttU | RUNs the OUTPUT Time Out Timer and sets the Timer to ttt (1-126) in HUNDREDTHs of a second. |
| C06RtttM | RUNs the OUTPUT Time Out Timer and sets the Timer to ttt (1-63) MINUTEs. |
| C06RtttS | RUNs the OUTPUT Time Out Timer and sets the Timer to ttt (1-63) SECONDs. |
| C06TtttH | Sets the OUTPUT TIME OUT default value to ttt (1-126) in HUNDREDTHs of a second. |
| C06TtttM | Sets the OUTPUT TIME OUT default value to ttt (1-63) MINUTEs |
| C06TtttS | Sets the OUTPUT TIME OUT default value to ttt (1-63) SECONDs. |
| C06L | LOCKs the OUTPUT/BUSY Pin configuration from further changes. |
| C06U | UNLOCKs the OUTPUT/BUSY Pin configuration. |

*Note: Command codes can be strung together to configure multiple functions in a single command string.*

**Example**: C06UE-R10MNL

U - UNLOCKs the OUTPUT/BUSY Pin configuration
E - ENABLEs the OUTPUT Pin function
– - Sets the OUTPUT Pin logic level to NEGATIVE
R10M - RUNs the OUTPUT Time Out Timer and sets the Timer to 10 MINUTEs
N - Turns the OUTPUT Pin ON
L - LOCKs the OUTPUT/BUSY Pin configuration.

### *Configure Elapsed Timers (ELP) - C07 - 'T' mnemonic*

Command C07 configures the ELAPSED (ELP) Timer function. The following is the list of valid command string formats for the C07 Command.

| | |
|---|---|
| C07 or T | The MDS transmits the current configurations of all 16 ELP Timers as shown.<br><br>`ELP TIMER tt CFG`<br>`abc M- T-`<br>`TMR hihl:mm:ss.uu`<br>`TRG`<br><br>tt - ELP Timer number (0-15)<br>a - ELP Timer is ON (N) or OFF (F)<br>b - ELP Timer Counts UP (U) or DOWN (D)<br>c - Indicates that the ELP Timer has OVERFLOWed or underflowed (V), blank otherwise.<br>M - Not implemented in the MDS<br>T - Not implemented in the MDS<br>TMR - hihl:mm:ss.uu - ELP Timer value in hundreds and thousands (hi) of hours, ones and tens (hl) of hours, minutes (mm), seconds (ss), and hundredths of seconds (uu).<br>TRG - Not implemented in MDS |

*Note: The ESCAPE key and the RETURN <RET> key PAUSE and RESUME the transmission of a list of ELP Timer Entries. Hitting <ESC> PAUSEs the transmission. Hitting <RET> RESUMEs the transmission. Hitting <ESC>while the list is PAUSEd cancels the remainder of the transmission.*

*Note: The MDS cannot process any message strings, message TIME OUTs, or commands while the transmission is PAUSEd.*

*Note: Any command string including the T command code can be issued without the command number "C07".*

| | |
|---|---|
| C07F<br>C07N<br>C07D<br>C07U<br>C07V<br>C07Ttt | Transmits a list of all OFF ELP Timers.<br>Transmits a list of all ON ELP Timers.<br>Transmits a list of all DOWN ELP Timers.<br>Transmits a list of all UP ELP Timers.<br>Transmits a list of all OVERFLOWed ELP Timers.<br>Transmits the current configuration for ELP TIMER tt (0-15). |
| C07TttF<br>C07TttG<br>C07TttN | Turns ELP TIMER tt (0-15) OFF - Timer halts.<br>TOGGLEs ELP TIMER tt's (0-15) ON/OFF status.<br>Turns ELP TIMER tt (0-15) ON - Timer runs. |
| C07TttD<br>C07TttU | Sets ELP TIMER tt (0-15) to count DOWN.<br>Sets ELP TIMER tt (0-15) to count UP. |
| C07TttV | Clears ELP TIMER tt's (0-15) OVERFLOW flag. |
| C07TttP0 | PRESETs ELP TIMER tt (0-15) to 0. |
| C07TttPhihlmmssuu | PRESETs ELP TIMER tt (0-15) to<br>hihl - HOURS (0000-9999)<br>mm - MINUTES (00-59)<br>ss - SECONDS (00-59)<br>uu - HUNDREDTHS (00-99)<br>*Note: Leading zeros are required. Any zeros following the last non-zero digit can be excluded. All excluded digits are automatically set to zero.* |

*Note: Command codes can be strung together to configure multiple functions in a single command string. Multiple ELP Timers can be configured with the '/' delimiter.*

**Example**: C07T0P0UN/T1N/T2F

| | | |
|---|---|---|
| T0 | - | Configures TIMER 0 |
| PO | - | PRESETs Timer 0 value to 0 |
| U | - | Sets Timer 0 Direction to UP |
| N | - | Turns ON Timer 0 |
| / | - | Indicates new Timer |
| TI | - | Configures TIMER 1 |
| N | - | Turns ON Timer 1 |
| / | - | Indicates new Timer |
| T2 | - | Configure TIMER 2 |
| F | - | Turns OFF Timer 2 |

### *Configure Unit Display - C08*

Command C08 configures the Unit Display parameters. The following is the list of valid command string formats for C08.

| | |
|---|---|
| C08 | The MDS transmits the current Unit Display configuration settings as shown.<br><br>`DISPLAY CFG INT-ii`<br>`BLINK BLOCK CHAR`<br>`TOP aaa0ms bbbuuu ccc0ms`<br>`BTM ddd0ms eeeuuu fff0ms`<br><br>INT - Backlight INTENSITY, ii (0-16)<br>aaa - Top Line BLINK time in hundredths of seconds (1-127)<br>ddd - Bottom Line BLINK time in hundredths of seconds (1-127)<br>bbb - Top line BLOCK SCROLL time (1-127) in uuu (0ms/sec)<br>eee - Bottom Line BLOCK SCROLL time (1-127)in uuu (0ms/sec)<br>ccc - Top Line CHARACTER SCROLL time in hundredths of seconds (1-127)<br>fff - Bottom Line CHARACTER SCROLL time in hundredths of seconds (1-127) |

| | |
|---|---|
| C08Iii | Sets the Display Backlight INTENSITY to ii (0-16) in 16ths. 0 is OFF, 16 is full ON. |
| C08TBttt | Sets the TOP BLINK time to ttt (1-127) in hundredths of seconds. |
| C08TBSttt | Sets the TOP BLOCK SCROLL time to ttt (1-127) in seconds. |
| C08TBSttt S | Alternate form of above. |
| C08TBSttt U | Sets the TOP BLOCK SCROLL time to ttt (127) in hundredths of seconds. |
| C08TCSttt | Sets the TOP CHAR SCROLL time to ttt (1-127) in hundredths of seconds. |
| C08BBttt | Sets the BTM BLINK time to ttt (1-127) in hundredths of a second. |
| C08BBStt | Sets the BTM BLOCK SCROLL time to ttt (1-127) in seconds. |
| C08BBSttt S | Alternate form of above. |
| C08BBSttt U | Set the BTM BLOCK SCROLL time to ttt (127) in hundredths of seconds. |
| C08BCSttt | Sets the BTM CHAR SCROLL time to ttt (1-127) in hundredths of seconds. |

*Note: Command codes can be strung together to configure multiple functions in a single command string.*

Example: C08I8TB10BB20TBS1BCS10

| | |
|---|---|
| I8 | - Sets the Display Backlight INTENSITY to 8/16ths |
| TB10 | - Sets the TOP BLINK Time to 100 milliseconds, ON for 100 msec, OFF for 100 msec |
| BB20 | - Sets the BTM BLINK Time to 200 milliseconds, ON for 200 msec, ON for 200 msec, OFF for 200 msec |
| TBS1 | - Sets the TOP BLOCK SCROLL Time to 1 second, Scrolls 1 BLOCK every second |
| BCS10 | - Sets the BTM CHAR SCROLL Time to 100 msec, Scrolls left 1 position every 100 msec |

### *Restore Default Unit Configuration - C17*

The C17 Command restores various settings within the MDS to their default settings. The following is a list of valid command string formats for the C17 Command.

| | |
|---|---|
| C17ALL | Restores the Configuration File, Character File, and the Day/Month File to their default settings. |
| C17CFG | Restores the Configuration File to its default settings. |
| C17CHR | Restores the Character File to its default settings. |
| C17DM | Restores the Day/Month Name File to its default settings. |

### *Load Character and/or Day/Month File - C18*

Command C18 receives the unit CHARACTER and/or DAY/MONTH Files over the serial port. The MDS unit must be in the RLC Slave Mode to receive the file. The following is a list of valid command string formats for the C18 Command.

| | |
|---|---|
| C18C | Informs the MDS that a CHARACTER file follows. The MDS expects to receive 768 bytes of data, representing the column pixel patterns that comprise the characters of a file. |
| C18D | Informs the MDS that a DAY/MONTH table follows. The MDS expects to receive 256 bytes of data, representing the DAY and MONTH names. |

*Note: The RLC Message Display Intelligent (MDI) unit can perform this function for you. See the MDI manual for details. If you need to perform either of these functions without the help of an MDI unit, contact the factory for the exact procedure.*

### *Halt/Reset Unit - C19*

Command C19 HALTs or RESETs the unit. The following is a list of valid command string formats for the C19 Command.

| | |
|---|---|
| C19HLT | HALTs the unit. Issuing the HALT Command causes an MDS to shut off its display and come to an orderly HALT. The only exit from the HALTed state is to cycle power to the unit. |
| C19RST | RESETs the unit. The unit undergoes a power-on RESET. Use this command if you need to have an MDS perform its power-up sequence, which includes clearing the Elapsed Timers and resetting the clock. |

### *Cancel Message - C20*

Command C20 cancels the Message on the Display. The following is a list of valid command string formats for the C20 Command.

| | |
|---|---|
| C20 | Cancels the DISPLAYed Message. |
| C20D | Alternate form of above. |

### *Request Message - C21 - 'M' mnemonic*

Command C21 issues a Message string. The following is a list of valid command string formats for the C21 Command.

| | |
|---|---|
| C21M:dd...d | Places a MESSAGE with text dd...d on the Display. |
| M:dd...d | Alternate form of above. |

### *Transmit Text For Message - C23*

Command C23 transmits the Message Text for the displayed Message. The following is a list of valid command string formats for the C23 Command.

| | |
|---|---|
| C23 | Transmits the DISPLAYed message text. |
| C23D | Alternate form of above. |

### *Echo Character String - C24*

Command C24 transmits (echoes) the received character string. This command is used to communicate with units downstream of the receiving unit. The following is the valid command string format for the C24 Command.

| | |
|---|---|
| C24:dd...d | Transmits the character string dd...d (echoes string). |

# INSTALLATION AND CONNECTIONS

## Installation Environment

The unit should be installed in a location that does not exceed the maximum operating temperature and provides good air circulation. Placing the unit near devices that generate excessive heat should be avoided.

Continuous exposure to direct sunlight may accelerate the aging process of the bezel. The bezel should be cleaned only with a soft cloth and neutral soap product. Do NOT use solvents.

## Panel Installation

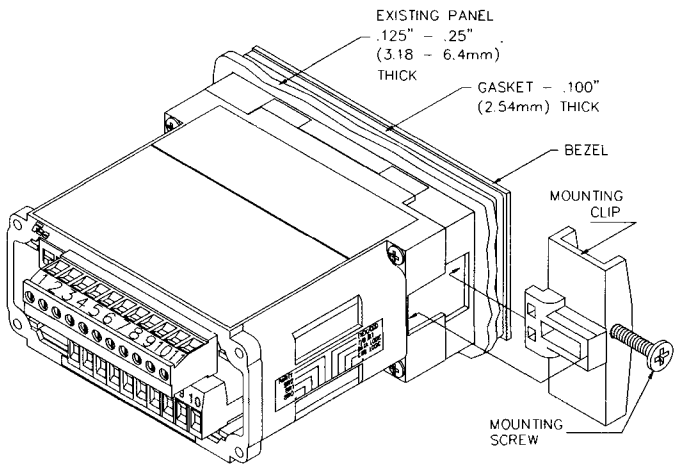


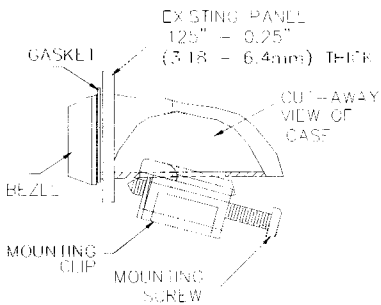

Figure 7 - Panel Installation

The Message Display Slave unit meets NEMA 4X/IP65 requirements for indoor use when properly installed. The units are intended to be mounted into an enclosed panel with a gasket to provide a water-tight seal. See Figure. Two mounting clips and screws are provided for easy installation. Consideration should be given to the thickness of the panel. A panel that is too thin may distort and not provide a water-tight seal. (Recommended minimum panel thickness is 1/8".)

After the panel cut-out has been completed and deburred, carefully slide the panel gasket over the unit from the rear, seating it against the lip at the front of the case.

Insert the unit into the panel. Install the mounting clips. Thread the screw into the clips until the pointed end just protrudes through the other side. Install each of the mounting clips by inserting the wide lip of the clips into the wide end of the hole located on either side of the case. Then snap the clip into the case.

Tighten the screws evenly to apply uniform compression, thus providing a water-tight seal.




*Caution: Only minimum pressure is required to seal the panel. Do NOT over tighten screws.*

## EMC INSTALLATION GUIDELINES

Although this unit is designed with a high degree of immunity to ElectroMagnetic Interference (EMI), proper installation and wiring methods must be followed to ensure compatibility in each application. The type of the electrical noise, source or coupling method into the unit may be different for various installations. The unit becomes more immune to EMI with fewer I/O connections. Cable length, routing and shield termination are very important and can mean the difference between a successful or a troublesome installation. Listed below are some EMC guidelines for successful installation in an industrial environment.

1. Use shielded (screened) cables for all Signal and Control inputs. The shield (screen) pigtail connection should be made as short as possible. The connection point for the shield depends somewhat upon the application. Listed below are the recommended methods of connecting the shield, in order of their effectiveness.
   a. Connect the shield only at the panel where the unit is mounted to earth ground (protective earth).
   b. Connect the shield to earth ground at both ends of the cable, usually when the noise source frequency is above 1 MHz.
   c. Connect the shield to common of the unit and leave the other end of the shield unconnected and insulated from earth ground.
2. Never run Signal or Control cables in the same conduit or raceway with AC power lines, conductors feeding motors, solenoids, SCR controls, and heaters, etc. The cables should be run in metal conduit that is properly grounded. This is especially useful in applications where cable runs are long and portable two-way radios are used in close proximity or if the installation is near a commercial radio transmitter.
3. Signal or Control cables within an enclosure should be routed as far away as possible from contactors, control relays, transformers, and other noisy components.
4. In extremely high EMI environments, the use of external EMI suppression devices, such as ferrite suppression cores, is effective. Install them on Signal and Control cables as close to the unit as possible. Loop the cable through the core several times or use multiple cores on each cable for additional protection. Install line filters on the power input cable to the unit to suppress power line interference. Install them near the power entry point of the enclosure.

The following EMI suppression devices (or equivalent) are recommended:

Ferrite Suppression Cores for signal and control cables:
- Fair-Rite # 0443167251 (RLC #FCOR0000)
- TDK # ZCAT3035-1330A
- Steward #28B2029-0A0

Line Filters for input power cables:
- Schaffner # FN610-1/07 (RLC #LFIL0000)
- Schaffner # FN670-1.8/07
- Corcom #1VR3

Note: *Reference manufacturer s instructions when installing a line filter.*

5. Long cable runs are more susceptible to EMI pickup than short cable runs. Therefore, keep cable runs as short as possible.

### Wiring Considerations

After the unit has been mechanically mounted, it is ready to be wired. All wiring connections are made on removable plug-in terminal blocks. All MDS units have a terminal block for the bottom board (TBA), and units with the optional Parallel Port have a terminal block for the top board (TBB). When wiring the unit, remove the terminal block and use the numbers on the label to identify the position number with the proper function. All conductors should meet voltage and current ratings for each terminal. Also cabling should conform to appropriate standards of good installation, local codes and regulations. It is recommended that power supplied to the unit (AC or DC) be protected by a fuse or circuit breaker.

Strip the wire, leaving approximately 1/4" bare wire exposed *(stranded wires should be tinned with solder).* Insert the wire into the terminal and tighten down the screw until the wire is clamped tightly. Each terminal can accept up to one 14-gauge, two 18-gauge or four 20-gauge wire*(s).*

After the terminal block is wired, install it into the proper location on the PC board. Wire each terminal block in this manner.

The MDS contains nonvolatile memory which retains its configuration settings. It is not necessary to provide battery backup to retain this programmed information. However, the MDS cannot retain customized CHARACTER or DAY/MONTH files, this information is lost on power-down and the default files are loaded on power-up. If customizing either of these files, either provide battery backup to the unit or reload the applicable files on every unit power-up.

### *Vcontrast*

The viewing angle of the display can be changed by adjusting the internal pot located on the side of the unit. The Vcontrast terminal located on TBA #9 allows for adjustment of the display viewing angle from a remote location.

To set-up the unit for remote adjustment, do the following:

1. Set the internal pot to the desired viewing angle.
2. Connect a 50 KΩ 1/4 Watt Pot between Vcontrast and common.

The external pot, which can be mounted on the front of the panel, can now be used to adjust the display's viewing angle.

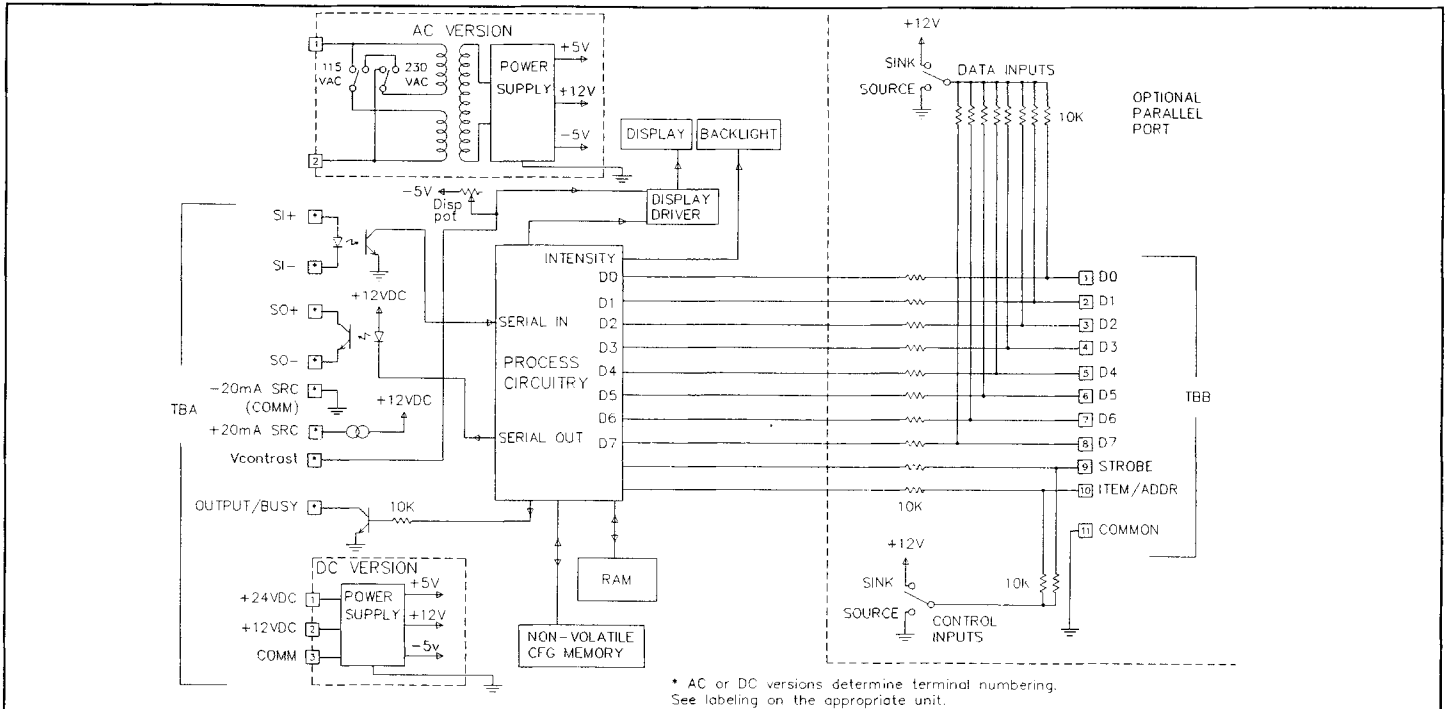



Figure 8 - Block Diagram

### AC Power Wiring

AC power is connected to TBA #1 and #2 located on the bottom terminal block. See Figure 8, Block Diagram. The voltage selector switch, located to the left side of TBA, is used to select the proper voltage. The switch is a slide movement type and can be set by using a small screwdriver. If the switch is showing "115", it is set for 115 VAC input. If "230" is showing, the switch is set for 230 VAC input.

*Caution: Before applying power to the unit, make sure the switch is set for the proper voltage setting.*

To reduce the chance of noise spikes entering the AC line and affecting the unit, the AC power should be relatively "clean" and within the specified ±10% variation limit. Connecting power from heavily loaded circuits or circuits which also power loads that cycle on and off, *(contactors, relays, motors, etc.)* should be avoided.

### DC Power Wiring

DC power is applied between TBA #1 and TBA #3 (common) for +24 VDC input or between TBA #2 and TBA #3 (common) for +12 VDC input. See Figure 8, Block Diagram. The DC power source must be capable of supplying the unit's rated current *(300 mA)*.

### Serial Wiring

It is recommended that shielded (screened) cable be used for serial communications. This unit meets the EMC specifications using Alpha #2404 cable or equivalent. There are higher grades of shielded cable, such as, four conductor twisted pair, that offer an even higher degree of noise immunity.

Connections for the 20milliamp current loop are made on TBA, the bottom terminal block. Power to the MDS and the current loop should be turned OFF before removing TBA and making any connections. Install wires into the proper location on the terminal block. Refer to the label on top of the unit for locations and descriptions of each terminal. When all necessary connections have been made, replace the terminal block. The Terminal block is keyed for proper installation.

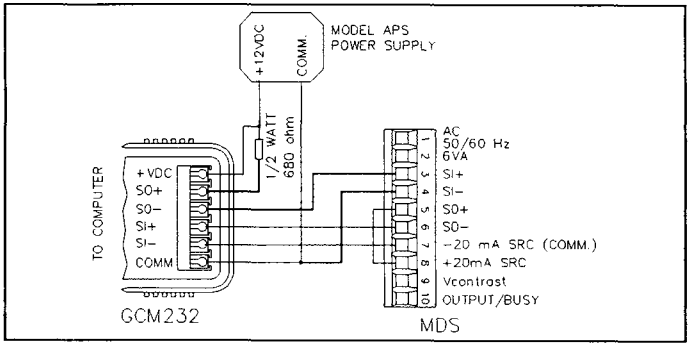



Figure 9 - AC Hook-up

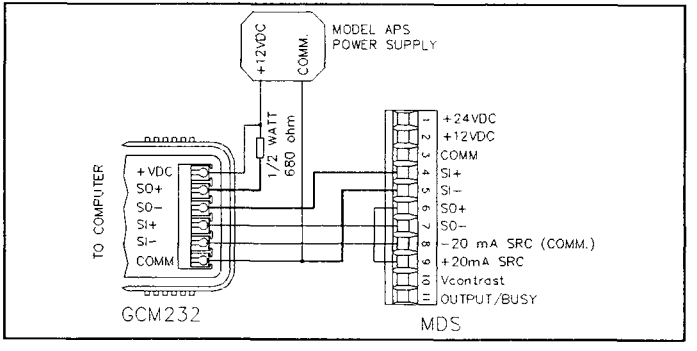



Figure 10 - DC Hook-up

### *Serial Terminal Descriptions*

**Receive Terminals**: SI+ (Serial In +)
SI- (Serial In -)

The MDS receives data, commands, and message strings on these terminals. Files from the SFMD software are downloaded to the MDS over these terminals, as well. Connect the transmit terminals of the device transmitting to the MDS, to these terminals.

**Transmit Terminals**: SO+ (Serial Out +)
SO- (Serial Out -)

The MDS issues all of its transmissions over these terminals. Files in the MDS are uploaded from these terminals to the SFMD software, as well. Connect the receive inputs of the device to which the MDS transmits, to these terminals.

**Current Loop Power**: The +20 mA SRC terminal provides a 20 milliamp source current for one of the loops. The -20 mA SRC (COMM.) terminal is the return path for the +20mA terminal.

See Figure 9, AC Hook-up, Figure 10, DC Hook-up, Figure 11, MCCA Cable AC Hook-Up, and Figure 12, MCCA Cable Hook-up DC for additional details.

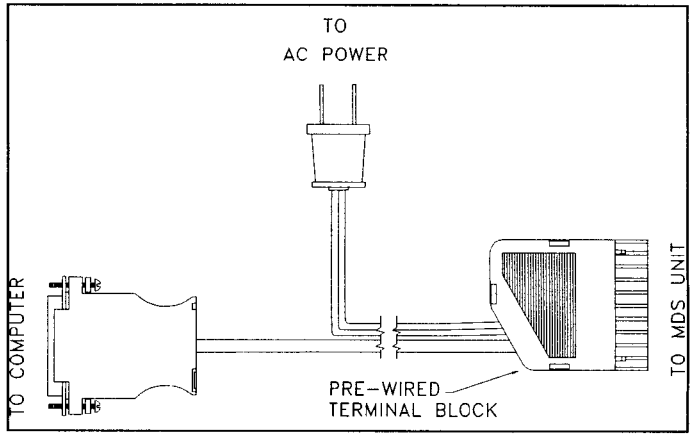



Figure 11 - MCCA Cable AC Hook-up

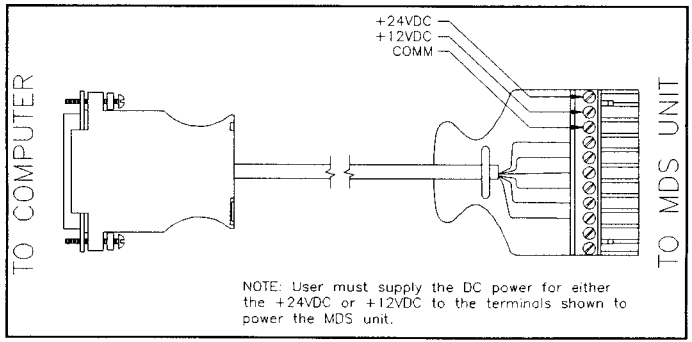



Figure 12 - MCCA Cable DC Hook-up

## DIP SWITCH SETTINGS

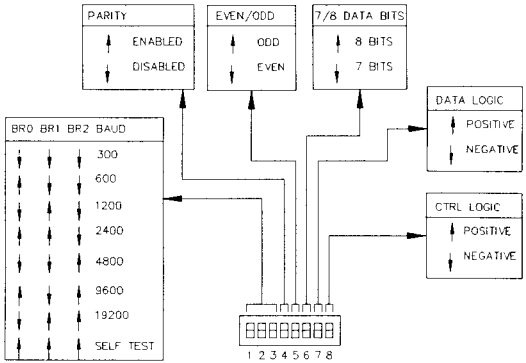



Figure 13 - DIP Switch Parameters

An eight position DIP switch is located on the side of the MDS. A two position DIP switch is located at the rear of units with the optional parallel port. The DIP switches are used to establish certain Serial and Parallel Port parameters. The parameter choices are shown in Figure 13, DIP Switch Parameters.

### Configuration and Self-Test Functions

The MDS contains several DIP switch selectable configuration and self-test functions that can be performed only on power-up. To invoke any of these functions, the three Baud Rate DIP switches (BR0, BR1, & BR2) on the side of the unit must be in the UP position when power is applied to the unit.

As long as the baud rate switches remain in the UP position, any of the DIP switch functions can be performed by changing the DIP switch settings to the next desired function. Configuration information and self-test results are shown on the MDS's Display. To exit the DIP switch functions, move any of the baud rate switches to its DOWN position. While executing the DIP switch functions, the MDS ignores all Message strings and Commands.

If the Baud Rate switches are placed in the Self-Test position while the unit is powered, the following Error Message is displayed in block scroll fashion at the current block scroll rate:

ERROR
INVALID
BAUD RATE
SETTING

The following functions can be performed by setting the DIP switches as shown (1 = DIP switch in the UP position, 0 = DIP switch in the DOWN position).

*Note: DATA LOGIC and CTRL LOGIC DIP switches are NOT used in the Self-Test Mode.*

**DIP SWITCH FUNCTIONS**

| BR0 | BR1 | BR2 | PARITY | EVEN/ ODD | 7/8 | DESCRIPTION |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | Displays the Unit Hardware Configuration. |
| 1 | 1 | 1 | 1 | 0 | 0 | Displays the Unit Configuration File Settings. |
| 1 | 1 | 1 | 0 | 1 | 0 | Displays the Parallel Port Configuration & current DATA & CTRL line levels. |
| 1 | 1 | 1 | 1 | 1 | 0 | Performs a Serial Self-Test. |
| 1 | 1 | 1 | 0 | 0 | 1 | Performs a Memory Self-Test. |
| 1 | 1 | 1 | 1 | 0 | 1 | Sets the RLC Mode of operation *(without Parallel Ports only)*. |
| 1 | 1 | 1 | 0 | 1 | 1 | Sets the Serial Mode of operation *(without Parallel Port only)*. |
| 1 | 1 | 1 | 1 | 1 | 1 | Clears nonvolatile Memory, restores all files to Factory Default settings. |

### *Unit Hardware Configuration*

The Unit Hardware Configuration can be displayed with the following DIP switch settings:

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 0 | 0 | 0 |

The MDS block scrolls the Hardware Configuration on the top line of the display.

| | |
|---|---|
| Hardware Version | VER - x.xx |
| Mode of Operation | MDS - "RLC", "SRL", or "TRX"<br>RLC - RLC Mode<br>SRL - Serial Mode<br>TRX - Transmit Mode |
| Display Size | "1 x 10" or "2 x 20" |
| Parallel Port | "PRL PORT" or "NO PORT" |

### *Unit Configuration File Settings*

The Unit Configuration File can be displayed with the following DIP switch settings:

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 1 | 0 | 0 |

The MDS block scrolls its current Configuration File Settings on the top line of the display. The first three blocks display a capital letter followed by eight flag settings (either 1 or 0). The remaining blocks display a lower case "parameter" letter followed by a three digit decimal value. A list of the parameters and their codes is shown below in Display order.

| CODE LETTER | DESCRIPTION OF SETTING | |
|---|---|---|
| C | Unit Configuration Byte | Factory default settings - 0110000 |
| | BIT 7 | 0 - Unused |
| | BIT 6 | 1 - Unused |
| | BIT 5 | 1 - Unused |
| | BIT 4 | 0 - XON/XOFF is OFF<br>1 - XON/XOFF is ON |
| | BIT 3 | 0 - SRL Mode<br>1 - RLC Mode |
| | BIT 2 | 0 - Unused |
| | BIT 1 | 0 - Unused |
| | BIT 0 | 0 - Unused |

| | | |
|---|---|---|
| H | Unit Hardware Byte | Factory default settings - xxxxxx10 Ported Units<br>11xxxx00 Portless Units |
| | BIT 7 | 0 - Parallel Port Control lines set for Negative Logic<br>1 - Parallel Port Control lines set for Positive Logic |
| | BIT 6 | 0 - Parallel Port Data lines set for Negative Logic<br>1 - Parallel Port Data lines set for Positive Logic |
| | BIT 5 | 0 - Serial Port set for 7 data bits<br>1 - Serial Port set for 8 data bits |
| | BIT 4 | 0 - Serial Port set for Even parity<br>1 - Serial Port set for Odd parity |
| | BIT 3 | 0 - Serial Port set for No parity<br>1 - Serial Port set for parity |
| | BIT 2 | 0 - Indicates a 1 x 10 Display<br>1 - Indicates a 2 x 20 Display |
| | BIT 1 | 0 - Unit has a Parallel Port<br>1 - Unit does not have a Parallel Port |
| | BIT 0 | 0 - Unused |

| CODE LETTER | DESCRIPTION OF SETTING | |
|---|---|---|
| O | Output Pin Configuration Byte<br>Factory default settings - 00000000 | |
| | BIT 7 | 0 - Output Pin in the Busy Mode (Can Only be in the Busy Mode if XON/XOFF is OFF)<br>1 - Output Pin in the Output Mode |
| | BIT 6 | 0 - Output Pin programmed for Negative Logic<br>1 - Output Pin programmed for Positive Logic |
| | BIT 5 | 0 - Output is OFF<br>1 - Output is ON |
| | BIT 4 | 0 - Output is Unlocked<br>1 - Output is Locked |
| | BIT 3 | 0 - Time Out is Halted (Output Mode Only)<br>1 - Time Out is Running (Output Mode Only) |
| | BIT 2 | 0 - Time Out value not in Hundreds<br>1 - Time Out value is in Hundreds (Time Out is in SECS if both BITs 1 and 2 are 0) |
| | BIT 1 | 0 - Time Out value not in Minutes<br>1 - Time Out value is in Minutes |
| | BIT 0 | 0 - Time Out has not Timed Out<br>1 - Time Out has Timed Out |

| CODE LTR | DESCRIPTION OF SETTING | RANGE | DEFAULT |
|---|---|---|---|
| c | Unit Address | 000-099 | 000 |
| d | Character Scroll Time Top Line | 001-127 | 016 |
| e | Character Scroll Time Bottom Line | 001-127 | 016 |
| f | Block Scroll Time Top Line | 001-127 | 001 |
| g | Block Scroll Time Bottom Line | 001-127 | 001 |
| h | Blink Time Top Line | 001-127 | 050 |
| i | Blink Time Bottom Line | 001-127 | 050 |
| j | Target Unit Address | 000-099 | 000 |
| k | Target Data Item | 001-099 | 001 |
| l | Serial Port Transmit Delay Time | 000-255 | 005 |
| m | Parallel Port Debounce Time | 001-255 | 005 |
| n | Display Backlight Intensity Level | 000-016 | 008 |
| o | Output Pin Time Out Value | 001-255 | 010 |
| p | Serial Port ED Time Out Value | 000-255 | 002 |
| q | Unused | | 001 |
| r | Error Handling Type | | 073 |
| *Note: Range includes ASCII B (066), D (068), I (073), and T (084)* | | | |
| s | Serial Port Terminator | 001-064 | 042 |
| *Note: Range excludes BS (008), ESC (027), and '0' to '9' 048-057)* | | | |
| t | Parallel Port Sample Time | 001-255 | 005 |

### *Parallel Port Configuration and Logic Levels*

The Parallel Port Configuration Test can be performed with the following DIP switch settings:

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 0 | 1 | 0 |

The MDS block scrolls the current Parallel Port DIP switch settings and the logic levels on the CTRL and DATA lines. The line levels shown are the actual levels at the Parallel Port pins, not the logic levels after translation for Positive or Negative logic. The line levels are read only when entering this test.

The results are Displayed as shown below:

PRL PORT

CTRL LVL (+/-) - CONTROL LOGIC DIP switch setting '+' (UP), '-' (DOWN)

DATA LVL (+/-) - DATA LOGIC DIP switch position '+' (UP), '-' (DOWN)

xxxxxxxxxx - H (HIGH) or L (LOW) indicating the current Logic level on each of the ten Parallel Port lines- Strobe, D7-D0, NEM/ADDR.

### *Serial Port Loop-Back Test*

The MDS must be wired in a serial loop-back configuration to properly perform the Serial Port Loop-Back test. See Figure 14 , Serial Loop-back.

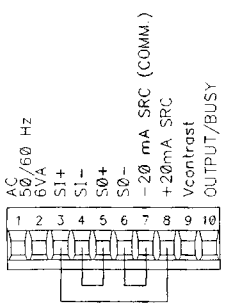



Figure 14, Serial Loop-back

The Serial Port Loop-Back test can be performed with the following DIP switch settings:

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 1 | 1 | 0 |

The MDS transmits 32 characters for each of the seven baud rates. Upon completion of the test, the MDS automatically transmits the test results, either 'P' (PASS) or 'F' (FAIL) at 1200 baud, 8 bits, and NO Parity. The MDS also places the test results on the display.

SRL - (PASS/FAIL)

### *Memory Test*

The MDS performs a limited Memory Test with the following DIP switch settings:

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 0 | 0 | 1 |

The MDS tests every memory location for Read and Write capability. The MDS does NOT change any of the Configuration settings. Upon completion of the test, the MDS automatically transmits the test results, either 'P' (PASS) or 'F' (FAIL) at 1200 baud, 8 bits, and NO Parity. The MDS also places the test results on the display.

MEM - (PASS/FAIL)

### *Setting the RLC Mode*

For MDS units without the Parallel Port, you select the RLC Mode of operation with the following DIP switch setting:

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 1 | 0 | 1 |

The MDS places "RLC MODE" on the display while in this DIP switch setting. For MDS units with the optional Parallel Port, see *Selecting A Mode Of Operation*, page 6.

### *Setting the Serial Mode*

For MDS units without a Parallel Port, you select the Serial Mode of operation with the following DIP switch setting:

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 1 | 1 | 0 |

The MDS places "SRL MODE" on the display while in this DIP switch setting. For MDS units with the optional Parallel Port, see *Selecting A Mode Of Operation*, page 6.

### *Clearing the Unit*

The following DIP switch setting, when applied at unit power-up, resets the configuration settings to their respective default values.

| PARITY | EVEN/ODD | 7/8 |
|---|---|---|
| 1 | 1 | 1 |

*Caution: This setting causes the MDS to clear its nonvolatile memory of all programmed information and restore the RLC Factory default settings.*

This DIP switch setting is only effective at power-up. If you accidentally switch into this setting at any other time, the MDS displays the "INVALID DIP SWITCH SETTING" Error Message.

## CHARACTER SET

The MDS contains 153 displayable characters, from 20h (32) to B8h (184). The 96 standard ASCII characters are included from 20h (32) to 7Fh (127). The extended set of characters, from 80h (128) to B8h (184) contains the European and special characters. A standard ASCII character can be included in a Message by transmitting its ASCII character code. For serial port configurations utilizing 8 data bits, an extended character can also be included in the Message text by transmitting its character code. However, since an extended character requires 8 data bits for transmission, if your serial configuration only uses 7 data bits, the extended character must be transmitted as a two character combination. The first character is always an MDS control code, either "^", 5Eh (94) or "▼", 1Fh (31). The second character is the difference of the extended character code and 50h (80). For example, to transmit the 'ü' character, 81 (129), you must first convert the 'ü' character (129 - 80 = 49, 31h, '1'), and then transmit the two character sequence "^1". See *Message String Features and Assembly*, page 12 , for details.

The SFMD software allows you to customize a character set to the particular needs of your system. Individual characters can be edited in the Character Editing screen. MDS units do not contain enough nonvolatile memory to retain a custom character set on power-down. If your application uses the MDS in the RLC Mode, and includes an RLC Message Display Intelligent (MDI) unit, you can load the custom character set into the MDI, and have the MDI automatically transmit the character set to one or more MDS units on the loop at power-up. See the Command C18 discussion in your MDI and MDS manuals for details.

If the application you are developing requires a customized character set and you are not utilizing an MDI in your application, contact the factory for details on loading the MDS with a character set.

The Command string "C17CHR" causes the MDS to restore its factory default character set. See *Command C17 - Restore Default Unit*, page 37, for details.

### Default Character Set

**ASCII TABLE OF EXTENDED CHARACTERS**

| HEXA DECIMAL | DECIMAL | MDS CHARACTER |
|---|---|---|
| 80 | 128 | Ç |
| 81 | 129 | ü |
| 82 | 130 | é |
| 83 | 131 | â |
| 84 | 132 | ä |
| 85 | 133 | à |
| 86 | 134 | å |
| 87 | 135 | ç |
| 88 | 136 | ê |
| 89 | 137 | ë |
| 8A | 138 | è |
| 8B | 139 | ï |
| 8C | 140 | î |
| 8D | 141 | ì |
| 8E | 142 | Ä |
| 8F | 143 | Å |
| 90 | 144 | É |
| 91 | 145 | æ |
| 92 | 146 | Æ |
| 93 | 147 | ô |
| 94 | 148 | ö |
| 95 | 149 | ò |
| 96 | 150 | û |
| 97 | 151 | ù |
| 98 | 152 | ÿ |
| 99 | 153 | Ö |
| 9A | 154 | Ü |
| 9B | 155 | ¢ |
| 9C | 156 | £ |

| HEXA DECIMAL | DECIMAL | MDS CHARACTER |
|---|---|---|
| 9D | 157 | ¥ |
| 9E | 158 | Á |
| 9F | 159 | Í |
| A0 | 160 | á |
| A1 | 161 | í |
| A2 | 162 | ó |
| A3 | 163 | ú |
| A4 | 164 | ñ |
| A5 | 165 | Ñ |
| A6 | 166 | Ó |
| A7 | 167 | Ú |
| A8 | 168 | ¿ |
| A9 | 169 | È |
| AA | 170 | À |
| AB | 171 | ø |
| AC | 172 | Ø |
| AD | 173 | ¡ |
| AE | 174 | Ù |
| AF | 175 | ã |
| B0 | 176 | õ |
| B1 | 177 | ' |
| B2 | 178 | ' |
| B3 | 179 | ≠ |
| B4 | 180 | Ω |
| B5 | 181 | ° |
| B6 | 182 | ± |
| B7 | 183 | ≥ |
| B8 | 184 | ≤ |

**ASCII TABLE OF STANDARD CHARACTERS**

| HEXA DECIMAL | DECIMAL | MDS CHARACTER |
|---|---|---|
| 20 | 32 | |
| 21 | 33 | ! |
| 22 | 34 | " |
| 23 | 35 | # |
| 24 | 36 | $ |
| 25 | 37 | % |
| 26 | 38 | & |
| 27 | 39 | ' |
| 28 | 40 | ( |
| 29 | 41 | ) |
| 2A | 42 | * |
| 2B | 43 | + |
| 2C | 44 | , |
| 2D | 45 | - |
| 2E | 46 | . |
| 2F | 47 | / |
| 30 | 48 | 0 |
| 31 | 49 | 1 |
| 32 | 50 | 2 |
| 33 | 51 | 3 |
| 34 | 52 | 4 |
| 35 | 53 | 5 |
| 36 | 54 | 6 |
| 37 | 55 | 7 |

| HEXA DECIMAL | DECIMAL | MDS CHARACTER |
|---|---|---|
| 38 | 56 | 8 |
| 39 | 57 | 9 |
| 3A | 58 | : |
| 3B | 59 | ; |
| 3C | 60 | < |
| 3D | 61 | = |
| 3E | 62 | > |
| 3F | 63 | ? |
| 40 | 64 | @ |
| 41 | 65 | A |
| 42 | 66 | B |
| 43 | 67 | C |
| 44 | 68 | D |
| 45 | 69 | E |
| 46 | 70 | F |
| 47 | 71 | G |
| 48 | 72 | H |
| 49 | 73 | I |
| 4A | 74 | J |
| 4B | 75 | K |
| 4C | 76 | L |
| 4D | 77 | M |
| 4E | 78 | N |
| 4F | 79 | O |

**ASCII TABLE OF STANDARD CHARACTERS**

| HEXA DECIMAL | DECIMAL | MDS CHARACTER |
|---|---|---|
| 50 | 80 | P |
| 51 | 81 | Q |
| 52 | 82 | R |
| 53 | 83 | S |
| 54 | 84 | T |
| 55 | 85 | U |
| 56 | 86 | V |
| 57 | 87 | W |
| 58 | 88 | X |
| 59 | 89 | Y |
| 5A | 90 | Z |
| 5B | 91 | [ |
| 5C | 92 | \ |
| 5D | 93 | ] |
| 5E | 94 | ^ |
| 5F | 95 | _ |
| 60 | 96 | ` |
| 61 | 97 | a |
| 62 | 98 | b |
| 63 | 99 | c |
| 64 | 100 | d |
| 65 | 101 | e |
| 66 | 102 | f |
| 67 | 103 | g |

| HEXA DECIMAL | DECIMAL | MDS CHARACTER |
|---|---|---|
| 68 | 104 | h |
| 69 | 105 | i |
| 6A | 106 | j |
| 6B | 107 | k |
| 6C | 108 | l |
| 6D | 109 | m |
| 6E | 110 | n |
| 6F | 111 | o |
| 70 | 112 | p |
| 71 | 113 | q |
| 72 | 114 | r |
| 73 | 115 | s |
| 74 | 116 | t |
| 75 | 117 | u |
| 76 | 118 | v |
| 77 | 119 | w |
| 78 | 120 | x |
| 79 | 121 | y |
| 7A | 122 | z |
| 7B | 123 | { |
| 7C | 124 | \| |
| 7D | 125 | } |
| 7E | 126 | ~ |
| 7F | 127 | ■ |

## *ERROR HANDLING*

The MDS contains an Error detection function which generates Error Codes for the various Error conditions it encounters. An Error Code is processed according to the type of Error Handling method selected.

**DISPLAY:** The MDS displays the Error Code as a blinking Message for 16 seconds. You can cancel any Error Message with the C20 Command. The Error Message appears on the display in the following format:

ERROR
exn

e - Error Code
x - Error Type
'c' - Command string error
'f' - Function error
n - Error Number (0-9)

**TRANSMIT:** The MDS transmits the Error Code over the serial port in the following format:
exn (See above for codes)

**DISPLAY & TRANSMIT**: The MDS both displays and transmits the Error Code as described above.

**IGNORE**: The MDS continues to detect Error conditions, but does not generate Error Codes for them.

You can set and verify the current Error Handling method with the C01 Command.

*Note: Error Codes are always transmitted if the Serial Port XON/XOFF Function is ON, regardless of the Error Handling type selected.*

ERROR CODES AND NUMBERS

**'c' - Command String Errors**

ec0 - Invalid command (Command > 24)
ec1 - Unused command (Commands 0, 3, and 9-16, 22)
ec2 - Unexpected or invalid command code
ec3 - Missing numeric command data
ec4 - Numeric command data out-of-range
ec5 - Numeric command data >255
ec6 - Non-numeric character in numeric field
ec7 - Unused
ec8 - Command buffer overflow

**'f' Function Errors**

ef0 - Output Pin is LOCKed, cannot make changes
ef1 - Unused
ef2 - Unused
ef3 - Invalid DIP switch setting

## TERMINAL EMULATOR FUNCTION

The SFMD software contains a Terminal Emulator which allows you to communicate with the MDS over the same serial connections that the SFMD software uses to upload from and download to the unit. Consequently, you can use the Terminal Emulator for issuing Commands and Message strings, interrogating the MDS, and most importantly, for testing, debugging, and trouble-shooting your particular application.

You can enter the Terminal Emulator from the Utilities pull-down menu, and from any of the three file editing screens. The current serial port settings are shown at the top of the Terminal Emulator screen, along with the currently specified MDS address for uploading and downloading purposes. You can alter these settings from within the Terminal Emulator.

### Function Keys

The active function keys are displayed at the bottom of the screen, along with their respective operations.

**F1-HELP** Requests on-line HELP for the Terminal Emulator function.

**F2-CFG TE** Activates the Terminal Emulator serial configuration fields. While configuring the serial settings, the Terminal Emulator ignores all serial transmissions. Press F2 to accept the new settings, or ESCAPE to keep the old settings. While you are configuring the serial settings, the CFG TE field BLINKs.

**F3-RECORD** Activates the RECORD function of the Terminal Emulator and requests a RECORD file name. Enter the file name, including the path, if different than the current path. Hit ENTER to accept the file name, or ESCAPE to keep the existing file. The RECORD field BLINKs while the Terminal Emulator is recording. If the specified file does not exist, it is created. If the file already exists, the Terminal Emulator begins to RECORD at the end of the current file, consequently, the existing data is kept. If the existing data is not desired, hit F4 to REWIND the file to the beginning.

**F4-REWIND** REWINDs the current RECORD file to the beginning. All existing data in the file is lost.

**F5-LOCK** LOCKs the keyboard from further entry. Press ALT-F5 to UNLOCK the keyboard. The LOCK field BLINKs while the keyboard is LOCKed.

**F6-PRINT** PRINTs the current RECORD file to the computer's parallel port.

**F10-EXIT** EXITs the Terminal Emulator. Closes any currently open RECORD file.

**Record File:** The name of the current RECORD file.

All characters received from the serial loop are displayed in regular type, all of the characters you enter, and any functions you perform, are displayed in bold type.

The Terminal Emulator displays the characters it receives exactly as transmitted, with the following exceptions:

1. Automatically inserts a Line Feed (LF, 0Ah - 10) for any unmatched Carriage Return (CR, 0Dh - 13) it receives.
2. Automatically inserts a CR for any unmatched LF it receives.
3. Displays <XON> for the ASCII XON code (11h - 17) and <XOFF> for the ASCII XOFF code (13h - 19).
4. Displays [CR] and [LF], respectively, for any CR and LF characters you enter.
5. Excluding CR, LF, XOFF, and XON, the hexadecimal value of any character received under 20h (32) or over 7Fh (127) is displayed in the form [xx], where "xx" is the two digit hexadecimal code for the character.

### *Special Key Stroke Functions*

Except for the previously mentioned function keys, and entering of a RECORD file name, the Terminal Emulator transmits all the characters you type. In addition, the following special key stroke functions are available for your use.

**CTRL-HOME** Transmits an ESCAPE (1Bh - 27) character, followed by the MDI/MDS universal Address code (0Eh - 14) and displays "[ADDR]". These characters cause an MDS unit in the RLC Mode to clear its receive buffer and then override its Unit Address and accept the transmission following the universal Address code. This function is useful if you have forgotten the unit address of an MDS. Please note that this function only works for the RLC Mode

**CTRL-END** Transmits the MDI/MDS universal Serial Terminator (00h - 0) and displays "[TERM]". Upon receiving this character, an MDS begins processing the contents of its serial buffer, just as if it had received its programmed Serial Terminator. This function is useful if you have forgotten the Serial Terminator for an MDS.

**Example**: You can determine the Unit Address (ADDR) and Serial Terminator (ST) of any MDS in the RLC Mode by entering the following Command strings:

```
Operator:   <CTRL-HOME>C1<CTRL-END>
Response:   FUNCTIONS CFG
            ADDR-99 CLK-ON DFLT- ERR-B
            PRDC-   QUE-   RST-

Operator:   <CTRL-HOME>C5<CTRL-END>
Response:   SRL PORT CFG - 7,O,9600
            DL-0050ms ST-042 TO-001s X-ON
```

**BACKSPACE** Transmits a BACKSPACE (08h - 8) character, causing an MDS to back-up one character position in its receive buffer. The character is removed from the Terminal Emulator's display, as well. The MDS ignores BACKSPACEs while its receive buffer is empty. However, the Terminal Emulator continues to delete characters from the screen.

**ESCAPE** Transmits an ESCAPE (1Bh - 27) character, causing an MDS to clear its receive buffer without processing the buffer's contents. The Terminal Emulator displays "[ESC]" for each stroke of the ESCAPE key.

If the MDS receives the ESCAPE character while it is in the process of transmitting a Command response, it pauses the transmit buffer at the end of the current line The MDS resumes the transmission upon receiving a CR. If it receives another ESCAPE character while the transmit buffer is paused, the MDS aborts the transmission.

**RETURN** Transmits a Carriage Return (CR, 0Dh - 13). If the Transmit Buffer of an MDS is currently paused, the MDS resumes transmission upon receiving the CR. Otherwise, the MDS treats a carriage return as any other character. The Terminal Emulator advances the display one line and displays "[CR]" for every stroke of the ENTER key.

**ALT-xxx** Transmits the character represented by xxx in decimal. Any character from 01h (1) to FFh (255) can be issued by holding down the ALT key while entering the decimal equivalent of the desired character on the numeric keypad. The character is transmitted when the ALT key is released.

### *Terminal Emulator Record Function*

The Terminal Emulator includes a RECORD function, which allows you to RECORD all of the actions of a Terminal Emulator session to the hard or floppy disk of the PC on which you are running the SFMD software. The RECORD function is useful for testing and debugging an MDS application. Complex MDS applications can be developed and verified through creative use of the Terminal Emulator and the RECORD function.

You enter the RECORD function by pressing **F3-RECORD**. The Terminal Emulator asks you for a file name, which can include a directory path, if desired. After you have typed in the RECORD file name, hit ENTER for the Terminal Emulator to open the file and begin RECORDing, or ESCAPE to cancel the function.

If the file you have specified does not exist, the Terminal Emulator creates it for you. If the file already exists, the Terminal Emulator begins RECORDing from the current end of the file, thus preserving any previous data. If you no longer require this data, hit **F4-REWIND** to position the file at the beginning.

While in RECORD, the Terminal Emulator RECORDs everything that appears on the screen, exactly as you see it, except for the regular and bold type styles. The RECORD file is stored in an ASCII text format, which you can review, edit, and print with almost any word processor. In addition, you can PRINT the contents of the current file from the Terminal Emulator by pressing the **F6-PRINT** key.

## SPECIFICATIONS AND DIMENSIONS

1. **POWER:**
   **AC Version:**
   115/230 VAC +/-10%, 50/60 Hz, 6 VA, switch selectable.
   **DC Version:**
   12/24 VDC +/-10%, 300 mA maximum, jumper selectable.
2. **DISPLAY:** *(available in various configurations).*
   **2x20:** 0.2" *(5mm)* high characters - without Parallel Port only.
   **1x10:** 0.35" *(9mm)* high characters - with Parallel Port only.
   **Transmissive STN LCD:** Negative Image with Red LED back lighting OR Positive or Negative Image with Yellow-Green LED back lighting. The brightness of the LED backlight is software adjustable through seventeen levels. The viewing angle of the display can be adjusted through an on-board pot. By connecting an external pot to the Vcontrast pin, the viewing angle can be adjusted from a remote location.
3. **PARALLEL COMMUNICATIONS:** *(Optional)*
   **BCD:** 8 bits.
   **PARALLEL PORT INPUTS:**
   **Data Inputs (D0 - D7) & Control Inputs (Strobe & Message/Data):** $V_{IH}$ = 8 $VDC_{MIN}$, $V_{IL}$ = 4V $DC_{MAX}$, $V_{MAX}$ = 30 VDC
   **Data SINK/SRC:** Sink or Source, switch selectable.
   **Control SINK/SRC:** Sink or source, switch selectable.
   **Data Logic Level:** Positive or negative, switch selectable.
   **Control Logic Level:** Positive or negative, switch selectable.
   **Current Sinking:** Internal 10KΩ pull-up, $I_{MAX}$=1.2 mA.
   **Current Sourcing:** Internal 10KΩ pull-down, $I_{MAX}$=3.1 mA @ 30VDC.
   **Debounce Time:** 0.01 to 2.5 seconds *(programmable).*
   **Strobe Time:** 3 to 255 msec *(programmable).*
4. **SERIAL COMMUNICATIONS:** 20 mA current loop, full-duplex.
   **Data Format:** Four types available, switch selectable.
   **11 bits:** 1 start bit, 8 data bits, 1 parity bit, 1 stop bit.
   **10 bits:** 1 start bit, 8 data bits, 1 stop bit.
   **10 bits:** 1 start bit, 7 data bits, 1 parity bit, 1 stop bit.
   **9 bits:** 1 start bit, 7 data bits, 1 stop bit.
   **Data Code:** ASCII
   **Unit Address:** Programmable from 0 to 99. *(The number of units in a single loop is limited by the hardware specifications.)*
   **Baud Rate:** 300 to 19200, switch selectable.
   **Parity:** Enabled or Disabled, switch selectable.
   **Even/Odd:** Parity Type, switch selectable.
   **7/8 BIT:** Data Bits, switch selectable.

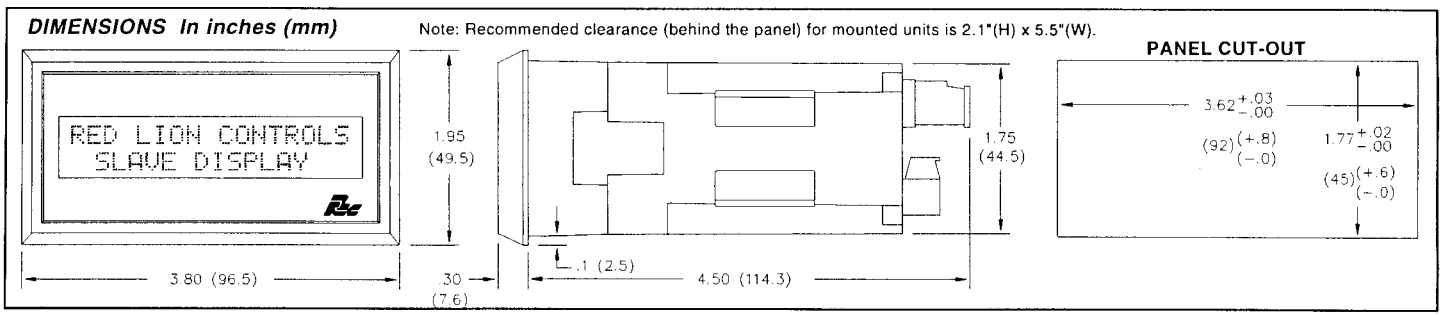



Figure 15 - MDS Dimension Drawing

**Serial Hardware:** Terminal TBA.

**+20mA SRC:** Provides 20 mA @ 12 VDC.

*Note: Can power up to 6 units in a loop.*

**-20mA SRC:** Loop return for +20 mA SRC.

**SO/Output Transistor Rating:** $V_{MAX}$ = 30 VDC. $V_{SAT}$ = 1 VDC @ 20 mA.

*Note: Transistor rating allows for up to 28 units in a loop.*

**SI/Input Diode Rating:** VF = 1.25 $VDC_{TYP}$: 1.5 $VDC_{MAX}$ @ 20 mA.

*Note: The compliance voltage rating of the source must be greater than the sum of the diode voltage drops around the loop. Typically a 30 VDC source (with adequate current capability) is capable of operating between 18 and 22 units in a loop.*

5. **OUTPUT/BUSY PIN:**

**Solid state:** NPN open-collector, current sinking, $V_{MAX}$ = 30 VDC, $I_{MAX}$ = 100 mA, $V_{SAT}$ = 1 $V_{MAX}$ @ 100 mA.

**Busy Mode:** Indicates the Ready/Busy status of the unit.

**Output Mode:** Output is activated from a Command or Message for a specified time out value.

**Time Out:** 10 msec to 63 mins or Latched.

6. **MEMORY:** nonvolatile memory retains all programmed Configuration settings.

7. **ENVIRONMENTAL CONDITIONS:**

**Operating Temperature**: 0 to 50°C

**Storage Temperature**: -20 to 60°C

**Operating and Storage Humidity**: 85% max. relative humidity (non-condensing) from 0°C to 50°C.

**Altitude**: Up to 2000 meters

8. **CERTIFICATIONS AND COMPLIANCES:**

UL Recognized Component, File #E171375

Recognized to U.S. And Canadian requirements under the Component Recognition Program of Underwriters Laboratories, Inc.

**Electromagnetic Compatibility**

**Immunity to EN 50082-2**

| | | |
|---|---|---|
| electrostatic discharge | EN 61000-4-2 | level 2; 4 Kv contact [1]<br>level 3; 8 Kv air |
| electromagnetic RF fields | EN 61000-4-3 | level 3; 10 V/m<br>80 MHz - 1 GHz |
| fast transients (burst) | EN 61000-4-4 | level 4; 2 Kv I/O [2]<br>level 3; 2 Kv power [3] |
| RF conducted interference | EN 61000-4-6 | level 3; 10 V/rms [3]<br>150 KHz - 80 MHz |
| power frequency magnetic fields | EN 61000-4-8 | level 4; 30 A/m |
| simulation of cordless telephones | ENV 50204 | level 3, 10 V/m<br>900 MHz – 5 MHz<br>200 Hz, 50 % duty cycle |

**Emissions to EN 50081-2**

| | | |
|---|---|---|
| RF interference | EN 55011 | enclosure class A<br>power mains class A |

*Notes:*

*1.Metal bezel of unit connected with ground lead from rear bezel screw to metal mounting panel.*

*2.For operation without loss of performance: I/O cables are routed in metal conduit connected to earth ground at both ends.*

*3.For operation without loss of performance: Install power line filter, RLC#LFIL0000 or equivalent.*

*Refer to the EMC Installation Guidelines section of the manual for additional information.*

9. **CONSTRUCTION:** Metal front bezel meets NEMA 4/IP65 specifications for indoor use in wash-down and/or dusty environments when properly installed.Installation Category II, Pollution Degree 2. Case is black high impact plastic *(panel gasket, mounting clips and screws included).*

10. **CONNECTIONS:** Removable terminal blocks.

11. **WEIGHT:** 1.1 lbs (0.5 Kg).

# APPLICATIONS

## TRANSMIT MODE APPLICATION

The customer has a process that is temperature sensitive and needs to monitor temperatures at various stages in the process from a single location. One or more RLC IMT (Intelligent Meter - Thermocouple) units have been installed at the necessary locations in the processing equipment. An MDS unit, configured for the Transmit Mode, is installed in the centralized process control panel. The MDS transmits to and receives data from the IMT's.

With an 8-bit (2 digit) BCD thumbwheel switch and push-button connected to the MDS's parallel port, the operator inputs the address of the desired unit by dialing in the address and pushing the button. The MDS interrogates the selected IMT and retrieves and displays the latest temperature reading.

## MDS RLC MODE APPLICATION

A manufacturing process is split between two buildings. Building #1 contains the test lab, and building #2 contains the actual process mixing equipment. When the lab has completed an analysis of the current mix, any formula changes must be transmitted to the mixing department. An intercom system was originally installed between the two buildings, but the mixing room is very noisy, and the formulae received over the intercom were frequently misinterpreted, or not heard at all. A more reliable method of communication needs to be established.

RLC proposes installing a Message Display Slave (MDS) unit configured for the RLC Mode of operation at each mixer. In this system each MDS is assigned a unique address and connected to an MDI unit located in the lab, that is attached to the lab PC via a serial port.

The lab technician calls up the appropriate message number on the MDI, which then asks the technician to supply the appropriate values for the revised mix from the PC's keyboard. The MDI inserts these values into the formula contained in the message and then transmits the completed message over the serial port to the MDS unit located at the appropriate mixer.

## MDS SERIAL MODE APPLICATION

Presently, the customer uses a software package to control the many processes throughout a manufacturing plant. The software performs data acquisition and logging, and makes decisions based on the collected information. Information on current plant conditions is transmitted to various printers located around the plant. Plant operators review the information on the printers and take whatever action is appropriate for the given condition(s). The system works well for the customer, and control of the plant is reliable and consistent.

However, the customer is concerned with the upkeep and maintenance of the printers, the cost of the paper and ribbons, and the downtime due to mechanical failures. RLC addressed these concerns by replacing the printers with Message Display Slave (MDS) units configured for the Serial (SRL) Mode of operation. An MDS unit in the SRL Mode displays any ASCII character string it receives that is terminated with the programmed Serial TERMINATOR character. The programmability of the TERMINATOR is essential when the MDS is interfaced with a third party software package, especially if the software cannot be programmed to transmit a specific termination character.

In addition, with unique addresses assigned to each MDS unit, the host software can then transmit information to specific unit(s). It is also possible to override the unit addresses and programmed TERMINATOR functions so that the same information can be transmitted to all MDS units in a system, regardless of their respective address and TERMINATOR settings.

With a wide variety of baud rate, parity, and data bit selections, the many display types, and several power supply options, the MDS unit is an ideal choice for replacing printers and other more cumbersome methods of providing plant-wide system information.

## TROUBLE SHOOTING GUIDE

For further technical assistance, contact technical support at the numbers listed on the back cover of the instruction manual.

| PROBLEMS | POSSIBLE CAUSE | REMEDIES |
|---|---|---|
| DARK DISPLAY<br>(No Backlighting Apparent) | 1. Power off<br>2. Improperly wired<br>3. Backlighting intensity level set to 0<br>4. Power in a Brown-out condition<br>5. Voltage selector switch in wrong position<br>6. Vcontrast pot *(internal/external)* adjusted improperly<br>7. No message text transmitted | 1. Verify power<br>2. Check wiring<br>3. Verify level setting<br>4. Verify power reading<br>5. Check switch setting<br>6. Adjust pot<br>7. Verify serial transmission |
| BLANK DISPLAY (Backlighting Apparent But No Discernable Characters) | 1. Vcontrast pot *(internal/external)* adjusted improperly<br>2. No message transmitted | 1. Adjust Pot<br>2. Verify serial transmission |
| MDS IS NOT RESPONDING TO SFMD SOFTWARE | 1. Serial set-up parameters do not match the MDS<br>2. GCM232 Module in wrong serial port of computer<br>3. GCM232 Module not powered<br>4.GCM232 Module improperly connected to MDS<br><br><br>5. Wrong address(es) being transmitted.<br><br>6. Incorrect Mode of Operation.<br>7. Modemm or mouse connected to Serial Port. | 1. Verify set-up parameters<br>2. Switch module to the port<br>3. Verify power<br>4. Check wiring<br>a. Pins 2 & 3 of GCM232 not connected properly to computer<br>5. Verify address by of MDS by viewing self-test indicator.<br>6. Set and verify the proper Mode of Operation.<br>7. Use a different serial port, or disconnect the modem or mouse. |
| MDS IS NOT RECEIVING DATA FROM SERIAL EXTERNAL UNIT(S) | 1. Improperly wired<br>2. Baud rates do not match MDS<br>3. Parity does not match MDS<br>4. Data bits transmitted do not match MDS<br>5. Proper terminator character not sent<br><br>6. External unit not powered | 1. Check wiring<br>2. Check DIP switch settings<br>3. Check DIP switch settings<br>4. Check DIP switch settings<br>5. Verify TRx mode<br>a. Retransmit with proper terminator<br>6. Verify power to unit |
| MDS IS NOT RESPONDING TO ITEM AND ADDRESS SELECTION OVER PARALLEL PORT | 1. Improperly wired<br>2. Data and control logic switches in wrong position<br>3. Data and control sink/source switches in wrong position<br>4. ITEM/ADDR line in wrong state when strobed | 1. Check wiring<br>2. Check switch settings<br>3. Check switch settings<br><br>4. Verify logic setting |
| MDS DISPLAYS EXTRA CHARACTERS, e.g.,"N00M:" "M001", etc | 1. MDS is in serial mode | 1. Switch MDS to RLC mode |

# APPENDIX - USING THE SFMD SOFTWARE TO CREATE AND SIMULATE MDS MESSAGES

The SFMD software contains extensive Message Editing and Simulation capabilities. These features are intended for creating and verifying Message Files that are eventually downloaded to an RLC Message Display Slave (MDS) unit. The MDS unit has **NO** Message File storage capability, and the SFMD software **DOES NOT** allow you to download a Message file to an MDS unit. However, you can use the software to assist you in creating, assembling, and testing complex Message strings intended for transmission to an MDS unit.

The Message Editing Screen of the SFMD software assists you in specifying the various special functions the MDS can execute. By programming the desired Message functions and text in the Message Editing Screen, and then simulating the Message, you can have the SFMD software assemble and display the required MDS Message string for you, as well as simulate the action and appearance of the Message, as if you had transmitted the Message string to an MDS unit.

## Message Editing Screen

Messages can be programmed and then simulated from the Message Editing Screen. The upper left corner of the screen indicates the current Message number that you are working on. Each Message has a specific Message number assigned to it, from 0 to 255. To the right of the Message number appears the size of the current Message Record, in bytes. The SFMD software updates the byte count whenever you SAVE the Message (press the F8- SVMSG key). A Message Record contains a minimum of 6 bytes and can contain a maximum of 255 bytes of information.

The current Message File path and name appear to the right of the Message Record size. Additional information about the current Message File and the MDS unit for which you are programming the Messages can be found in the File/Unit Info sub-menu. An active prompt line appears at the bottom of the screen's border, which displays basic information about the currently highlighted field. If you require more information about a particular field, on-line Help for the highlighted field can be accessed by pressing the F1-HELP key. General Help is also available from the Main Menu Screen. The Message Editing Screen appears as shown in Figure 16.

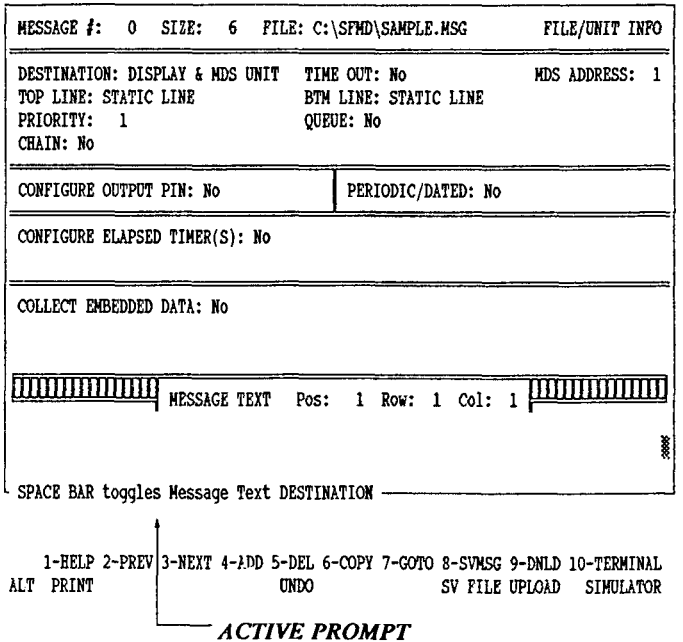



Figure 16 - Message Editing Screen

### File/Unit Info

The MESSAGE FILE AND MDI UNIT INFORMATION sub-menu, shown in Figure 17, displays general information about the current Message File you are editing and the MDI unit you last downloaded to or uploaded from. This menu provides the current SFMD Version number, the full Message FILE NAME and Message FILE PATH, the date and time of the last file modifications, and the number of programmed Messages in the file.

The Default and Reset Message settings only apply to an MDI unit. You can ignore these settings. The UNIT ADDRESS field should be set for the MDS unit with which the SFMD software must communicate. The corresponding unit information is then updated after each file download to or upload from that unit.

The MEMORY SIZE, MSG FILE SIZE field and the ED STORAGE fields only apply to the MDI and you can ignore these fields.

The DISPLAY TYPE field is used by the Message Simulator function to accurately simulate the MDS's display. Set this field for the type of display your unit contains.

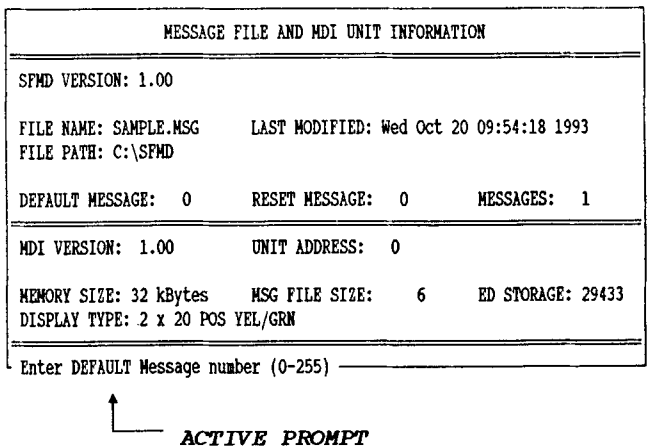



Figure 17 - Message File and Unit sub-menu

### Message Destination - Display & MDS Unit

Although there are several different Message destinations, the only one we are concerned with is the **DISPLAY & MDS UNIT** setting. Set the MESSAGE DESTINATION field to this setting. The **DISPLAY & MDS UNIT** Destination causes the Message Simulator to assemble the Message text into an MDS compatible transmission string. This type of transmission is intended for MDS units configured for the RLC Mode of operation. If you are using the MDS in the Serial Mode, a few minor adjustments to this string are necessary. *See the Message Simulator discussion later in this Appendix for details on these adjustments.*

### Message Time Out

You can specify a Time Out value for a Message. The Message Time Out value determines how long the MDS displays the Message. At the end of the Time Out, the MDS automatically cancels the Message and blanks its display. The Message Time Out value can range from 10-1260 milliseconds, 0-63 seconds, or 1-63 minutes. You can also turn the function OFF, in which case the Message does not Time Out.

### MDS Unit Address

The MDS UNIT ADDRESS field appears when the Message Destination is set for DISPLAY & MDS UNIT. Enter the address of the unit for which the Message string is intended. You cannot enter an address of 0. If a 0 address is a requirement of your application, or you are assembling Message strings for the serial mode, you will have to change or remove this information later.

### *Message Scrolling*

You can choose from among three different methods for displaying Message text for a given line of the display.

**STATIC** - The Message text characters do not scroll, but remain in place on the display.

**BLOCK SCROLLING** - The MDS displays BLOCKs of text, designated with the SEGMENT control code (♦s-<ALT-S>) in the Message Text area, one after another. Each BLOCK of text is displayed for the duration specified by the BLOCK Scroll Time for the particular line of the display on which it appears. Once the MDS displays the last BLOCK of text for a line it repeats the text for the line from the beginning.

For the RLC Mode, any BLOCK of text that exceeds the width of the display is truncated to the display width. All the remaining RLC mode characters for that BLOCK are ignored. For the SRL Mode, the MDS automatically character scrolls the line of text containing the BLOCK.

**CHARACTER SCROLLING** - The MDS advances the text one character position at a time, from right to left, across the display. The CHARACTER Scroll Time for a line determines the rate at which the text advances across the display. Once the last character of a line scrolls off the display to the left, the MDS repeats the text beginning from the right. You can designate BLOCKs of text for CHARACTER scrolling with the SEGMENT control code (♦s-<ALT-S>). In this case, the MDS character scrolls each BLOCK of text separately across the screen.

You do not need to specify the CHARACTER SCROLLING feature in Message Strings intended for MDS units in the SRL Mode. In the SRL Mode, the MDS automatically CHARACTER SCROLLS a line of text if any block of text in that line contains more characters than the width of the display.

You can specify separate scrolling techniques for the top and bottom lines of text for a Message. You can also configure the MDS's BLOCK and CHARACTER Scroll Times for separate top and bottom line values.

### *Message Priority - MDI only*

Not implemented in the MDS.

### *Message Queuing - MDI only*

Not implemented in the MDS.

### *Message Chaining - MDI only*

Not implemented in the MDS.

### *Configuring the Output Pin*

The MDS has one NPN open-collector Output Pin which you can assign to the MDS's handshaking BUSY Mode, or the Message and Command controlled OUTPUT Mode. With the Output Pin assigned to the OUTPUT Mode, you can program Messages to configure the pin. The MDS processes the new configuration settings for the Output Pin when the Message string is received. You program a Message to configure the Output Pin by setting CONFIGURE OUTPUT PIN to "Yes" and hitting ENTER or clicking the mouse with the cursor positioned on the CONFIGURE OUTPUT PIN field. You can program a Message string to perform any or all of the following Output Pin functions:

**UNLOCK ON ENTRY:** If the OUTPUT Pin is LOCKed, the Message must first UNLOCK the pin before it can access any other pin functions, otherwise, any attempt to modify the pin's configuration is ignored.

**LOCK ON EXIT:** The OUTPUT Pin can be LOCKed from accepting any changes. You can have the MDS LOCK the pin after it has performed the Output Pin functions the Message specifies.

**STATUS:** The Message can set the Pin to the ON state, the OFF state, or it can TOGGLE the current state of the Pin from ON to OFF or OFF to ON. The actual ON and OFF states of the Output Pin are determined by the pin's current Logic Level setting.

**TIME OUT STATUS:** The Output Pin contains a Time Out function. At the expiration of the Time Out value, the MDS automatically turns the Output Pin OFF. You can have the Message RUN or HALT this Time Out function and set the Time Out value. HALTing the Time Out function freezes the Time Out at its current value. RUNning the Time Out function does not automatically turn the Output Pin ON, you must do this explicitly. If the Message RUNs the Time Out, you must also select one of the following three Time Out sources (DEFAULT, MESSAGE, RESUME).

**SOURCE:** If you select DEFAULT, the MDS loads the Time Out value from the default Output Pin Time Out Configuration setting and then sets the Time Out status to RUN.

If you select MESSAGE, the MDS loads the Time Out value included in the Message record and then sets the Time Out status to RUN. The Message can specify the Output Time Out in the range from 10 to 1260 milliseconds, 1 to 63 seconds, or 1 to 63 minutes.

If you select RESUME, the MDS RUNs the Time Out from its current value. If the value has already Timed Out, the Time Out status immediately reverts to HALT.

If you do not desire to change a particular Output Pin function for a Message, select "N/C" or "NO CHANGE" for that function. *(See the OUTPUT PIN section for details).*

### *Periodic Messages - MDI only*

Not implemented in the MDS.

### *Configuring Elapsed Timers*

The MDS contains sixteen separate, fully configurable Elapsed Timers. Messages can access, configure, and display multiple Elapsed Timers. You can also access and configure any Elapsed Timer with the C07 Command. The MDS constantly updates all Elapsed Timer values according to the current configuration for each timer.

You can display an Elapsed Timer value in a Message by inserting the ELAPSED TIMER control code (♦e-<ALT-E>) followed by the two-digit timer number, the appropriate Time format codes, and a closing control code (♦e) in the Message Text. If a displayed Timer is running, the MDS displays the running value as it changes. See *Message Text Area* later in this section for details on specifying an Elapsed Timer value in the Message Text.

You can program a Message to configure an Elapsed Timer by setting CONFIGURE ELAPSED TIMERS to "Yes" and hitting ENTER or clicking the mouse with the cursor positioned on the desired Elapsed Timer number. A Message can access any of the following Elapsed Timer functions through the Elapsed Timer Configuration sub-menu. See Figure 18.

**ELAPSED TIMER:** This field indicates the Elapsed Timer you are configuring.

```
                ELAPSED TIMER CONFIGURATION
═══════════════════════════════════════════════════════════
      ELAPSED TIMER:  1          DISPOSITION: N/C

TIMER          DIRECTION: UP
STATUS: ON     VALUE: 0000:00:00.00    CLEAR OVERFLOW: No

TRIGGER        TYPE:  N/C              SET MESSAGE: No
STATUS: N/C    VALUE: NO CHANGE        MESSAGE: N/C
═══════════════════════════════════════════════════════════
            <SAVE>   <CANCEL>   <DELETE>
Enter Timer RESET/PRESET value, SPACE BAR for N/C ———————
```

Figure 18 - Elapsed Timer Configuration menu

**DISPOSITION:** Not implemented in the MDS.
**TIMER STATUS:** You can turn the Timer ON (run) or OFF (halt).
**DIRECTION:** A Timer can run UP or DOWN. You can change the direction of the Timer without halting or resetting the Timer.
**VALUE:** You can preset the Timer to any value within the range from 0000:00:00.00 to 9999:59:59.99.
**CLEAR OVERFLOW:** A Timer value rolls-over on Overflow or Underflow and the MDS sets a flag to indicate this action. You can have the Message reset this flag, if necessary.
**TRIGGER STATUS:** The Trigger function is not implemented in the MDS unit.
**TYPE:** Not implemented in the MDS.
**VALUE:** Not implemented in the MDS.
**SET MESSAGE:** Not implemented in the MDS.
**MESSAGE:** Not implemented in the MDS.

If you do not need to make a change to a particular Elapsed Timer feature, select "N/C" or "NO CHANGE" for that feature. (See *Elapsed Timers*, page 17, for details).

### *Collecting Embedded Data*

The Embedded Data function is not implemented in the MDS.

### *Message Text Area*

The Message Text area of the SFMD software functions as a limited text editor. The character position (Char Pos), Row (Row), and Column (Col) identifiers in the lower middle of the screen show the current position of the cursor in the text area. The following key-stroke functions are available in the Message Text area:

**HOME:** Moves the cursor to the beginning of the current row.
**END:** Moves the cursor to the end of the current row.
**INSERT:** Toggles between the INSERT and the OVERTYPE modes. The INSERT mode allows you to insert new characters anywhere in the existing Message Text. The OVERTYPE mode allows you to type new characters over the existing text.
**DELETE:** DELETEs the character at the current cursor position.

MESSAGE TEXT CONTROL CODES

| FUNCTION | CONTROL CODE | KEY STROKE | USAGE | DESCRIPTION | |
|---|---|---|---|---|---|
| BLINK | ♦b | ALT-B | ♦btt..♦b | tt..t | Specifies the text characters you want to blink. You must enclose the blinking field with the "♦b" control code, as shown. |
| CURRENT TIME | ♦b | ALT-C | ♦cff..f♦c | ff..f | Specifies the Current Time field format string. You must enclose the Current Time field with the "♦c" control code,as shown. The MDS replaces the format codes you include with the appropriate Current Time values. See the Current Time Format Codes table for details. |
| LOCAL ED ITEM | ♦d | ALT-D | ♦dnn | nn | This control code only applies to Messages assembled and transmitted by an MDI unit. Specifies the number of the Local ED Item (01 to 15) that you wish to insert in the text. You must include any leading zeroes for this field. An MDI replaces this control sequence with the current value of the specified Local item. |
| ELAPSED TIME | ♦e | ALT-E | ♦enff..f♦e | nn | Specifies the number of the Elapsed Timer (00 to 15) that you wish to access. You must include any leading zeros for this field. |
| | | | | ff..f | Specifies the Elapsed Tiime field format string with the "♦e" control code, as shown. The MDS replaces the format codes you include with the appropriate values from the specified Elapsed Timer. See the Elapsed Time Format Codes table for details. |
| INDEXED ED ITEM | ♦i | ALT-I | ♦inn | nn | This control code only applies to Messages assembled and transmitted by an MDI unit. It specifies the number of the Indexed ED Item (00 to 95) that you wish to insert in the text. You must include any leading zeros for this field. The MDI replaces this control sequence with the current value of the specified Indexed item. |
| END-OF-LINE | ♦l | ALT-L | ♦l | | Enter this control code at the end of the text you want the MDS to place on the top line of its display. Any subsequent text will appear on the bottom line of the display. This control code is only valid with a 2x20 display. |
| END-OF-SEGMENT | ♦s | ALT-S | ♦s | | Enter this control code at the end of each segment (BLOCK) of text. You do not need to include this control code at the end of a line, or the end of the text, unless you want the MDS to display a blank BLOCK of text at either of those locations. |
| TRANSMIT DELAY | ♦t | ALT-T | ♦t | | Enter this control code in the text to cause the MDS to run a Transmit Delay at that point in the text. This code is only useful if the MDS transmits the Message text. |

You can use the ARROW keys to move around the Message Text area.

A Message's text can include any of the characters in the MDS's character set, along with most of the lower ASCII control characters, such as the Carriage Return (CR) and Line Feed (LF) characters. The only lower ASCII characters you cannot include in the Message Text are 00h (0), 09h (9), and 14h (20), which are reserved for use by the MDS. You can enter any of the standard ASCII characters, 20h (32) to 7Eh (126), by hitting the appropriate key on the keyboard. All other characters must be entered through a combination of key strokes.

The characters from 01h (1) to 0Fh (15) are entered by holding down the CTRL key and pressing the 'A' (01h) through 'O' (0Fh) keys, respectively. The characters from 10h (16) through 1Fh (31), and the extended ASCII characters, from 7Eh (127) to B8h (184) are entered by holding down the ALT key and typing in the decimal number of the desired character on the numeric keypad.

You can have the MDS insert the Current Time and Date and Elapsed Timer values anywhere in the Message text through the use of other control code sequences. For Current and Elapsed Time values, you must also include the format of the inserted values with a format code sequence, which specifies the time values you want to include.

You enter a control code by pressing the appropriate letter key while holding down the ALT key. The SFMD software displays all control codes as two-letter sequences comprised of the diamond (♦) character followed by the lower-case letter of the key you pressed. A table of the available control codes, their usage and description follows later in this section.

You can insert any number of blinking fields in the Message text by surrounding the desired text characters with the BLINK control code. For instance, to instruct the MDS to blink the word "Blink" in a Message, you must enter the BLINK control code, "♦b", by holding down the ALT key and pressing the 'B' key. The "♦b" sequence appears at the cursor location in the Message Text area. Type in "Blink" followed by another ALT-B control code. The sequence appears on the screen as "♦bBlink♦b". The MDS blinks the characters sandwiched between the two BLINK control codes whenever it displays the Message.

When inserting a Current Time or Elapsed Time value in the Message text, you must also specify the format of the time field. The format codes used to specify Current and Elapsed Time values are comprised of two-letter upper-case character sequences. Each format code sequence describes a particular time value, such as the Current YEAR, the Current MONTH, the Elapsed MINUTES, etc.

You string these format codes together to create a time field of your choosing and indicate the beginning and ending of the field with the appropriate control code. For instance, you would insert the Current Hours, Minutes, and Seconds in a Message by including the sequence "♦cHH:NN:SS♦c". The "♦c" sequence is the control code for a Current Time field and the "HH", "NN", and "SS" sequences are the HOURS, MINUTES, and SECONDS format codes, respectively. When the MDS encounters this sequence, it replaces the entire string with the current values of the fields you specified. A table of the available format codes, their usage and description follows later in this section.

Inserting an Elapsed Time value is similar, with the exception that you must also specify the Elapsed Timer number you want to reference. For instance, to insert the Minutes, Seconds, and tenths of seconds for Elapsed Timer 15 in the Message text, you would include the sequence "♦e15NN:SS.U♦e" in the text. The "♦e" sequence is the control code for an Elapsed Time field and the "15" sequence specifies Elapsed Timer 15. The "NN", "SS", and "U" sequences are the MINUTES, SECONDS, and TENTHS of seconds format codes, respectively.

## *Message Text Area (Cont'd)*

The various Message Text control and format codes are listed in the following tables.

*Note: These codes only apply to Messages assembled by an MDI for transmission to an MDS unit. Strings transmitted to an MDS from any other peripheral device must use the MDS control code sequence. See the MESSAGE STRING FEATURES AND ASSEMBLY section for details on the MDS control code sequence.*

### CURRENT TIME FORMAT CODES

| CODE | DESCRIPTION |
|---|---|
| AA | Day Name, Abbreviated (i.e.. MON) |
| AF | Day Name, Full (i.e.. MONDAY) |
| DD | Date, Numeric (i.e.. 1-31) |
| HA | Hours, 12 Hour Clock (Civilian Time: 01-12) |
| HH | Hours, 24 Hour Clock (Military Time: 00-23) |
| HM | Same as above (HH) |
| MA | Month Name, Abbreviated (i.e.. JAN) |
| MF | Month Name, Full (i.e.. JANUARY) |
| MM | Month, Numeric (i.e.. 01-12, 1 = January) |
| NN | Minutes |
| P | AM/PM, can only be used with the "HA" code |
| SS | Seconds |
| U | Tenths of Seconds |
| UU | Tenths and Hundredths of Seconds |
| Y2 | Year, Two Digits (i.e.. 93) |
| Y4 | Year, Four Digits (i.e.. 1993) |

*Note: All Field Codes must be entered in CAPITAL LETTERS.*

### ELAPSED TIME FORMAT CODES

| CODE | DESCRIPTION |
|---|---|
| HH | Elapsed LOW Hours, ones & tens |
| HI | Elapsed HIGH Hours, hundreds & thousands |
| HL | Elapsed LOW Hours, ones & tens |
| NN | Minutes |
| SS | Seconds |
| U | Tenths of Seconds |
| UU | Tenths and Hundredths of Seconds |

*Note: All Field Codes must be entered in CAPITAL LETTERS.*

You can use any of the characters not included in either the Current Time or Elapsed Time Format Code tables as text in a Current Time or Elapsed Time Field. Consequently, any of the punctuation characters are available for use as field separators, such as the colon :, the hyphen - , the slash /, the period '.', etc.

**Example**: Given these Current Time and Elapsed Time values:
Current Time: Thursday, August 8, 1993 16:49:23.93
Elapsed Timer 00: 0001:23:45.67
Elapsed Timer 15: 9998:56:32.01

The MDS would replace the following Current and Elapsed Time format strings with the values shown:

| FORMAT STRING | DISPLAYED OR TRANSMITTED VALUE |
|---|---|
| "♦cDDMAY2 HA:NN:SS P♦c" | " 8AUG93 4:49:23 PM" |
| "♦cAF, MF DD, Y4♦c" | "THURSDAY, AUGUST 8, 1993" |
| "♦cHH:NN:SS.U♦c" | "16:49:23.9" |
| "♦e00HIHL:NN:SS.UU♦e" | "0001:23:45.67" |
| "♦e15HH:NN:SS.U♦e" | "98:56:32.0" |

It is possible to program more characters for a line than the MDS can display. In addition, if a line includes any Current or Elapsed Time fields, once the MDS processes these fields, the resulting line may be longer than the MDS can display. In cases such as this, the MDS truncates the excess text for a line in the RLC Mode, or automatically character scrolls the line in the Serial Mode. You can avoid this problem by configuring the Message to CHARACTER scroll the line or BLOCK scroll segments of text that do not exceed the display line length.

The MDS ignores any lower ASCII control characters (01h (1) to 1Eh (30)) it finds in the Message text. Therefore, if your application requires that these characters be present in the Message string, they will not interfere with the operation of the MDS, and the MDS will not display them.

### *Message Simulator*

The SFMD software contains an extensive Message Simulator function which allows you to observe the operation of the Message string you are currently editing without constantly transmitting the Message string to an MDS. Pressing ALT-F10 from the Message Editing screen invokes the Simulator, which appears in the lower half of the Message screen. If there is no currently open Configuration file, the Simulator asks you to open one.

The Simulator then executes the current Message based on the parameters you have specified for that Message. The Message Text appears on the Simulator's display exactly as it would appear on an MDS of the appropriate configuration. The Simulator uses the FILE/UNIT INFO settings from the current Message file to assemble its display. It can simulate both 1 x 10 and 2 x 20 displays of either positive or negative images with red or yellow-green back lighting.

The Simulator executes the following MDS Message features:

1) Configures and displays the Output Pin status, and runs the Output Time Out, as specified in the Message string.
2) Configures, runs, and displays the Elapsed Timers as specified in the Message string.
3) Assembles and displays the Message string, as it would be transmitted to an MDS, in the RLC Mode compatible form.
4) Displays the Message text using the default Character set, and the current Configuration file settings for the Blink time and the Block and Character scroll times.
5) Runs the Message Time Out, if desired.
6) Informs you of any Message execution errors, where appropriate.

Current and Elapsed Time fields are filled in with their appropriate values. Elapsed Timer configurations and values are retained so long as you do not exit the SFMD software. You can, therefore, perform simulations on blocks of Messages that configure and display Elapsed Timers.

### *Simulator Function Keys*

| | |
|---|---|
| F1-HELP | Displays HELP for the Simulator function. |
| F2-PAUSE | PAUSEs the Simulator. The Simulator display, and all time values, are frozen. |
| F3-CONTINUE | RESUMEs the Simulator. Also, CONTINUEs to the next Embedded Data Item, or to the next message in a chain. |
| F4-CANCEL MSG | CANCELs the current Message. Returns to the Message Editing screen. |
| F5-AUTO T/O | Toggles the automatic Message Time Out feature. If AUTO T/O is selected, the Simulator automatically cancels the Message when its Time Out expires. If AUTO T/O is OFF, the Message's Time Out does not run. Disable the AUTO T/O feature if you want to observe the operation of a Message without having it repeatedly time out on you. |
| F6-AUTO CHAIN | The CHAIN function is not implemented in the MDS. |
| F10-EXIT | EXITs the Message Simulator, returning control to the Message Editing screen. |

### *Assembled Message String*

The Simulator assembles and displays the Message string for the special functions and Message Text you have specified in the Message Editing screen. After the following minor modifications, you can copy and transmit this string to the MDS unit to accomplish the tasks you have selected for the Message. The displayed string may include the **"<XON>"**, **"<XOFF>"**, **"<TRX>"** and **"<CLR>"** strings the Simulator normally displays when simulating MDS transmissions. Delete all occurrences of these strings from the assembled transmission.

The Simulator prefixes an address string, "**Nxx**", for the Unit Address set in the Message Editing screen. This address specifies the MDS unit (in the RLC Mode) that is to process and display the transmission. The SFMD software does not allow you to set the Unit Address to 0. If you need to transmit the Message string to Address 0, delete the "**N**xx" Address Identifier string from the transmission, or replace the "xx" unit address with "00".

If your application uses the Serial Mode, and you are transmitting to a specifically addressed unit, you should transmit the **<CTRL-N>** (0Eh - 14) Address code in place of the '**N**' Address Identifier. If you are not addressing a particular unit, do not transmit the "**NxxM:**" string that precedes the Message text. If neither of these actions is taken, an MDS in the Serial Mode will accept the specified characters as part of the Message text.

While assembling the transmission string, the Simulator performs a number of control code conversions in order to construct the string in an MDS recognizable format. The Simulator converts all control codes and extended characters into two-character MDS control sequences. These two-character sequences all begin with the '▼' (1Fh - 31) character followed by an "operator" character. You can retain the '▼' character if your application can transmit it, or you can substitute the '^' (5Eh-94) character which is a standard ASCII character.

The Simulator terminates the transmission string with the Serial Terminator character programmed in the current Configuration file. The total transmission string length for an MDS unit, including the Serial Terminator character, is limited to the size of the Serial Receive buffer, 128 bytes. The Simulator function checks the length of the resulting Message string and informs you if the string is too long for the MDS to receive intact.

See the MESSAGE STRING FEATURES AND ASSEMBLY section for details on the various control code sequences the MDS recognizes.

### *Simulator Display*

After assembling the Message string transmission, the Simulator displays the Message Text in the Simulator Display Area, just as the text would appear on an MDS. The Message Text is displayed according to the SCROLL settings specified in the Message string, as well as the BLINK time and the BLOCK and CHARACTER scroll times contained in the current Configuration file.

If the Message displays any Current Time or Elapsed Time fields, the Simulator fills in the current values for these fields. If the Message includes a Time Out value, the Simulator blanks the display when the Time Out value expires.

### *Ordering Information For MDS*

| MODEL NO. | DESCRIPTION | PARALLEL PORT | PART NUMBERS | |
|---|---|---|---|---|
| | | | 12/24VDC | 115/230VAC |
| MDS | 1x10 Red, Negative Image | YES | MDS1R01D | MDS1R01A |
| | 1x10 Yel/Grn, Negative Image | YES | MDS1Y01D | MDS1Y01A |
| | 1x10 Yel/Grn, Positive Image | YES | MDS3Y01D | MDS3Y01A |
| | 2x20 Red, Negative Image | NO | MDS2R00D | MDS2R00A |
| | 2x20 Yel/Grn, Negative Image | NO | MDS2Y00D | MDS2Y00A |

### *Ordering Information For Message Display Accessories*

| MODEL NO. | DESCRIPTION | PART NUMBERS |
|---|---|---|
| SFMD | Apollo Message Display User Software (3 1/2";1.44M) | SFMD0 |
| MCCA | AC Communications Adapter | MCCA0000 |
| | DC Communications Adapter | MCCA1000 |

*Note: Only one copy of SFMD software is required for multiple units.*

## *LIMITED WARRANTY*

*The Company warrants the products it manufactures against defects in materials and workmanship for a period limited to one year from the date of shipment, provided the products have been stored, handled, installed, and used under proper conditions. The Company's liability under this limited warranty shall extend only to the repair or replacement of a defective product, at The Company's option. The Company disclaims all liability for any affirmation, promise or representation with respect to the products.*

*The customer agrees to hold Red Lion Controls harmless from, defend, and indemnify RLC against damages, claims, and expenses arising out of subsequent sales of RLC products or products containing components manufactured by RLC and based upon personal injuries, deaths, property damage, lost profits, and other matters which Buyer, its employees, or sub-contractors are or may be to any extent liable, including without limitation penalties imposed by the Consumer Product Safety Act (P.L. 92-573) and liability imposed upon any person pursuant to the Magnuson-Moss Warranty Act (P.L. 93-637), as now in effect or as amended hereafter.*

*No warranties expressed or implied are created with respect to The Company's products except those expressly contained herein. The Customer acknowledges the disclaimers and limitations contained herein and relies on no other warranties or affirmations.*

MDS / IM-C 2/00
DRAWING NO. LP0297

**INTERNATIONAL HEADQUARTERS**
20 Willow Springs Circle, York, Pa. 17402 (717) 767-6511 FAX: (717) 764-0839
Web site- http://www.redlion-controls.com E-mail- sales@redlion-controls.com

**RED LION CONTROLS (UK)**
Tapton Park, Chesterfield, Derbyshire S41 OTZ ENGLAND +44 1246 222122 FAX: +44 1246 221222