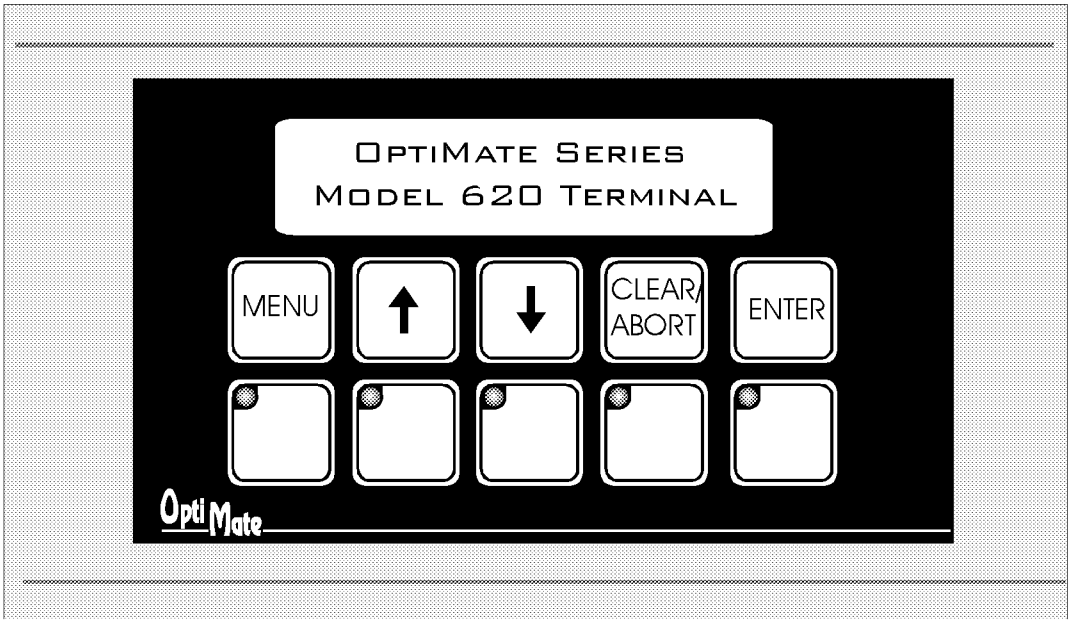


OM620J Operator Terminal

The OM620J Operator Terminal is a low cost/high performance man/machine interface with a broad range of operator input and display capabilities. The panel includes a 2 line by 20 character LCD display, arrow adjustment data entry, five function keys and menu tree capability.

OM-WINEDIT configuration software allows you to predefine up to 160 messages. These messages can be later selected for display by your PLC or computer program to display status, variable data and allow numeric data input.

Function keys can be custom labeled by the user with plastic inserts. The inserts can be custom legended with text and/or graphics, and slipped into a protective pocket behind the faceplate.

The OM620J Operator Terminal is part of Optimation's **OptiMate**® series. Each OptiMate panel is designed to connect to a microprocessor or most PLCs with a single cable connection. OptiMate panels can be used individually, or together with any combination of other OptiMate panels.

When used with a microprocessor system, simple communications over either an RS232 or RS422 communication cable allows the microprocessor to directly control the LCD display as well as read numeric data entry and function key status.

When used with a PLC, operation is transparent to the user. Terminal functions tie directly into your PLC ladder logic program. The OM620J takes care of the rest.

Applications

- Machine control
- Process control
- Security systems
- HVAC
- Plant monitoring/control
- PLC applications
- Microprocessor applications

Features

- 2 line x 20 character LCD
- Numeric arrow adjustment
- 5 User-defined function keys
- Menu tree capability
- PLC compatible
- RS232/RS422 communications
- Stand alone operation capable
- Multimodule operation capable

Contents

Configuration Options
- Stand Alone
- PLC Multi-Panel
- Multidrop

General Use with a PLC
Use with an Entertron PLC
Microprocessor-Based Systems
- OptiMate Hex Protocol

SetUp and Interconnect
- Legending the Function Keys
- Connection to the System

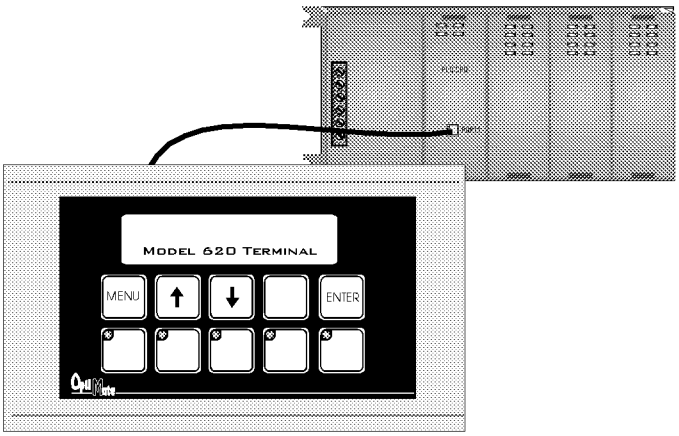
Configuration
- Configuration Selection
- Configuration using an Entertron PLC
- Message Definition Template
- Addressing
- Label Templates

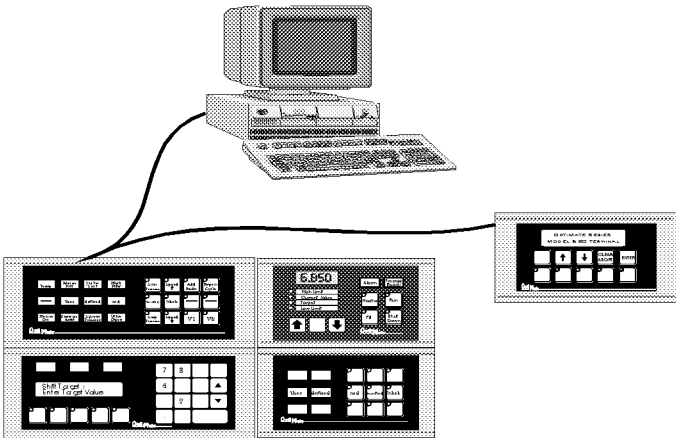
Specifications

OptiMate is a registered trademark of Optimation, Inc.

copyright ©2001, Optimation, Inc.

Configuration Options





PLC Stand Alone

OptiMate panels plug directly into most PLCs. A simple cable connection allows you to interface and control the OptiMate panel via PLC data registers and ladder logic.

The OM620J Operator Terminal uses a bank of PLC registers. Complete Operator interface is performed with 9 PLC registers for display message selection, data entry and function key interface. The OM620J continuously accesses these PLC registers and performs operations under ladder logic control on a real time basis.

PLCs are slave devices on their standard communications ports. This means that a panel attached to the standard port must control the transfer of information by reading and writing the PLC registers. OptiMate panels will communicate most major PLC protocols. Configuration for particular PLC protocols and interconnect cabling is covered in the following pages.

Microprocessor Based Systems

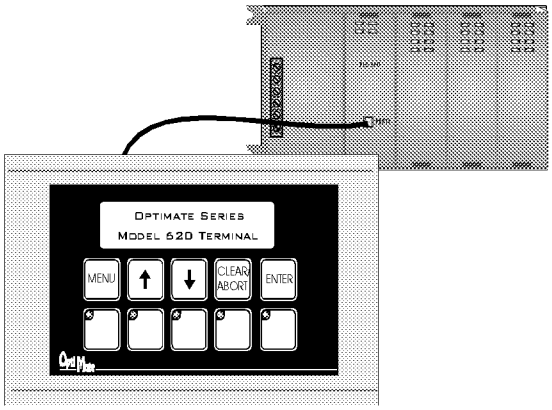
OptiMate panels can interface directly to most computers or microcontrollers. The modules communicate over either RS422 or RS232 serial communications. All that is required to interface OptiMate panels is a serial port and the ability to transmit and receive Hex numbers. The OptiMate Hex protocol, detailed in this document, allows the user to directly control panel operation and retrieve operator inputs.

Since each panel has its own unique address, up to 31 modules can be interfaced on one communications cable.

In a microprocessor-based system, the host microprocessor is the system master. The OptiMate modules are slave devices that respond to commands from the host. In the case of the OM620J, these commands are messages for display and data initialization as well as function button status and data entry retrieval.

Communications over RS422 allows placement of modules anywhere within a 4000 foot cable distance. Panels can be grouped together to form a larger panel. Panels can be grouped in several clusters all on the same communications cable.

General Use with a PLC



Memory Mapping

Memory mapping is a technique that "maps" the memory of an OptiMate panel into the registers of the programmable controller. By knowing where the data of a specific OptiMate panel is mapped, this data can be moved, changed or monitored using ladder logic.

The term PLC register is used for describing the area of memory within the programmable controller that can be used for data storage. PLC registers are sometimes known as data registers or internal registers.

MSB															LSB
16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

PLC Register

The OM620J Operator Terminal uses a bank of 9 contiguous PLC registers. The register set definition is shown in the table below.

OM620J Operator Terminal PLC Register Map

PLC Register	Register Function
M+0 (first register of bank)	Top line message selection
M+1	Bottom Line message selection
M+2	Top line data
M+3	Top line data 2 (for long BCD and floating-point)
M+4	Bottom line data
M+5	Bottom line data 2 (for long BCD and floating-point)
M+6	Function selection
M+7	Status register
M+8	Control register

Register Definition

The following describes the function of the registers shown in the table.

- Register M - When a number from 1 - 160 is placed in this register, the predefined message associated with that number will be displayed on the top line of the LCD display.
- Register M+1 - When a number from 1-160 is placed in this register, the predefined message associated with that number will be displayed on the bottom line of the LCD display.
- Register M+2 - Numeric data associated with top line display (described in more detail in following paragraphs)
- Register M+3 - For long BCD and floating-point data only. Numeric value is the four most significant BCD digits of data for the top line.
- Register M+4 - Numeric data associated with bottom line of display.
- Register M+5 - For long BCD and floating-point data only. Value is the four most significant BCD digits of data for the bottom line.
- Register M+6 - The function number selected through the menu tree.
- Register M+7 - Status register

MSB															LSB
					↓	↑	ENT	MA	AB	FS	DA	F5	F4	F3	F2	F1

Status Register

> F1-F5 - status of the five function keys. Set to 1 when the button is active.

> DA - Data available. Associated with data entry. Set to 1 when new data has been entered.

> FS - Function Selected. Indicates that a function has been selected through use of the menu tree.

> ENT - ENTER button is active. (Not necessarily Data Entry - see DA bit).

> AB - Abort key active.

> MA - Menu operation is currently active.

> Up Arrow - Up arrow key is active.

> Down Arrow - Down arrow key is active.

- Register X+8 - Control bits

MSB															LSB
											BD	DAK	MR	ME	

Control Register

> ME - Menu Enable. Must be set for menu operation to be enabled. (Normally, your control program would clear this bit when a function is in process).

> MR - Menu Return. Used at the end of a function. If set, will cause the menu to return to the same point in the menu tree as when the function started. If not set, function complete will not return directly into menu.

> DAK - Data acknowledge. This bit is used for repeat data entry into the same message. Setting to 1 acknowledges data entered from the keyboard as accepted by the PLC program. The 620 Terminal will clear the data on the screen message, clear the DA bit and allow new data entry. This bit must be cleared after DA (status register) is cleared.

> BD - Buzzer disable. When this bit is set to 1, the buzzer that beeps every time a button is pressed will be disabled (i.e. not beep).

Operational Overview

Displaying Messages on the LCD Display

Through the OM-WINEDIT configuration software, up to 160 predefined messages can be entered and stored in the OM620J. These messages are 20 characters long and can include a field for the display and/or entry of numeric data.

Any predefined message can be displayed on either the top or bottom line. The messages entered via the configuration editor are numbered 1 through 160. To display a particular predefined message on the display, simply place that message's number in the message selection register.

For example, lets assume that we have defined message #16 as "Mary had a little .." and message #22 as "white fleeced lamb". If we wanted to put these two lines on the top and bottom lines of the display respectively, we would simply need to put the number 16 in register M and 22 in register M+1.

If any number other than 1 to 160 is placed in a message selection register, the associated line will not change.

Placing Numeric Data in the Display

Certain predefined messages may incorporate a numeric data field. One numeric field per line is allowed. This field may be either a display data field or a data entry field. Messages that contain data are entered through the configuration editor with a caret symbol "^" as a place holder for each numeric digit.

An example of the use of numeric data is the message "#widgets sold: ^^^^". Assume that this is message # 36 entered through the configuration editor. Also assume that a total of 465 widgets have been sold today. To display the current number of widgets sold on the bottom line of the display, you would place `36` in register M+1 and `465` in register M+4. The bottom line of the display would then read "# widgets sold: 465".

Displaying Data with a Decimal Point

The OM620J panel allows you to display fixed point numbers. Fixed point numbers are numeric values that have a known decimal point placement and are simply handled as integer values within the PLC program. The only time you use an actual decimal point is for display to the operator. An example of a fixed point number is a program that uses temperature as a control variable. Within the program, all temperatures are scaled in tenths of a degree. The values are integer. A temperature of 73.5 degrees would be 735 in a data register. For the convenience of the operator, you would want the display to include the decimal.

Fixed point numbers are handled by simply placing a decimal point or period in the message field during configuration. In other words, the message "Temperature : ^^^.^" would be entered during configuration (say message 47). If 47 were placed in register M and the value 735 in register M+2, the display would read "Temperature : 73.5" on the top line.

Displaying BCD and Binary Numbers

Normally, numeric values to be displayed are values contained in one 16 bit register. One 16 bit-register will handle values between 0 and 65535 (in binary format), or 0 to 9999 (in BCD format). For this type of number, register M+2 is used for numeric value for the top line and register M+4 is used for the bottom line.

Displaying "Double" Numbers

The OM620J will handle larger numeric numbers. If you select the option "BCD double" when the display message is being defined, your display will handle numbers between 0 and 99,999,999. The OM620J will use data in the register pair M+2 and M+3 for the top line. Likewise, M+4 and M+5 are used for the bottom line. The data must be in BCD format.

When placing a "BCD double" number in the display registers, the first register numerically in the sequence of two registers (M+2 or M+4) will contain the 4 least significant digits of the number. The second register in the sequence (M+3 or M+5) contains the data for the 4 most significant digits of the "BCD double" number.

If the data displayed on the top line of the panel is 92345678, the top line data registers will contain the following: (shown in BCD/Hex format)

BCD Double Data	PLC Register
M+2	5678
M+3	9234

Displaying Floating-Point Numbers

The OM620J has the capability to display Floating-Point (or Real) numbers if you select the option "Float" when the display message is being defined in the OM-WINEDIT software.

Floating-point numbers can only be used with CPUs that support the IEEE 32-bit floating-point number format. The floating-point numbers are stored in the IEEE 32-bit floating-point format within the PLC. They always occupy 2 16-bit register locations regardless of the size of the number. Refer to the PLC manufacturer's programming documentation for more information on the IEEE 32-bit floating-point number format.

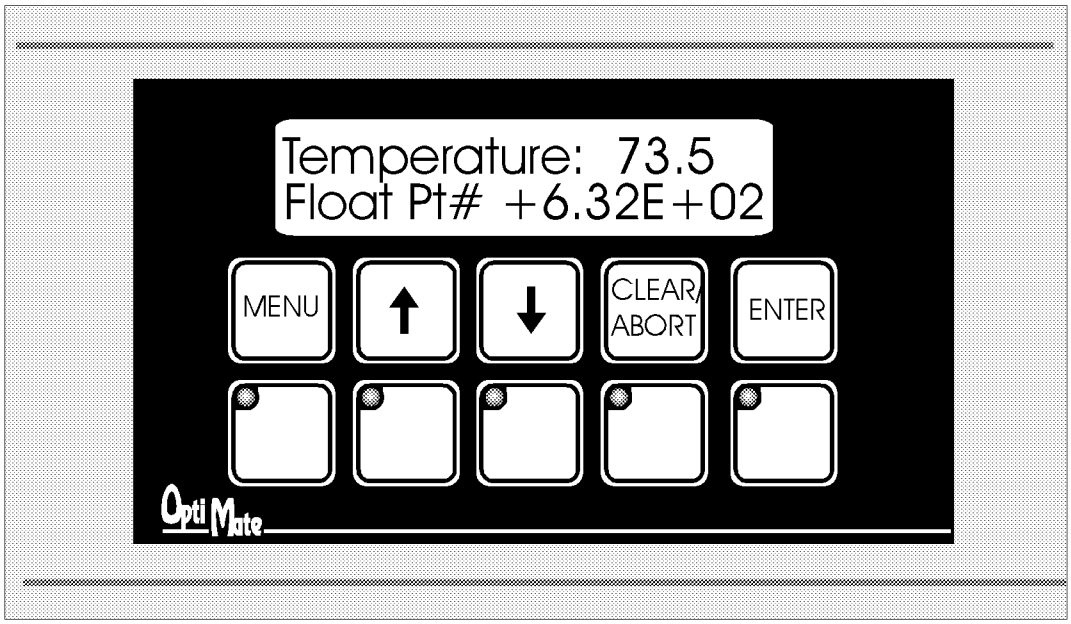
An IEEE 32-bit floating-point number has a range of -3.402823E+38 to +3.402823E+38. The OM620J will be able to display any number within that range. The panel always uses the format ±X.XXE±XX to display the numbers.

The panel does not have the ability to display all the significant digits of a floating-point number, it only displays the first 3 significant digits. The OM620J does not "round" the numbers up or down, instead it truncates the remaining digits. The two examples in the table below show the data contained in the PLC registers and the value displayed on the panel in its format. Notice how the data is truncated, not rounded.

PLC Registers	OM620J Display
12301.789	+1.23E+04
123.96783	+1.23E+02

The configuration of a floating-point number message is similar to any other message. First you select the message number, then you type in the text using 9 caret symbols "^" as a place holder for each of the 9 floating-point number symbols. Next, select the "Float" option for the data format.

Suppose you wanted to configure message #58 to display a floating-point number. In the OM-WINEDIT software you would select OM620J as module type. Then to configure message #58 simply select it with the mouse and type in a message in the following manner: "Float Pt ^^^^^^^^^". Also, select floating-point as the message format. To display a number, simply move it into either the top or bottom line data registers and load the appropriate message number into the corresponding top or bottom line



message selection register. If the number 632.15 is to be displayed in message #58, it will be displayed as the following: "Float Pt # +6.32E+02".

Numeric Data Entry

Numeric data can be entered through the adjustment arrow keys. To do so, the message must be marked for data entry via the configuration editor.

> Note : Only one data entry message may be active at a time. If data entry is selected simultaneously for both lines, unexpected operation may occur.

All of the numeric features described for data display apply to numeric data entry. This includes the definition of the caret symbol "^" place holder within the message as well as "double" number types.

Data Entry/Adjustment with the Arrow Keys

A message can be set up for arrow adjustment through the OM-WINEDIT software. To do so, define the message with a numeric field. Select arrow adjustment. If there are limits, select and enter minimum and maximum values.

When the data screen is required in the program, place the number of the preconfigured message in the selection register and its current value in the associated data register. The digits marked by carets will initially display the current value.

As the operator presses the up or down arrow key, the numeric value will increment or decrement respectively. As it is adjusted, the value will be continuously updated in the PLC data register. **When adjustment is complete, the operator will press the ENTER button. When this happens the data available (DA) status bit will be set. The DA bit will remain set until a new message number is placed in the message selection register (M or M+1) or the DAK bit is set.**

Adjustment of data will be confined to the limits defined through the configuration editor. Data also will be limited by the number of digits defined by carets in the message.

Example of Arrow Adjustment of Numeric Data

Suppose that your automatic banana peeler has a peel rate that can be adjusted between 1 and 50 bananas per second. With the configuration editor you define message 15 as "Set Peel Rate:^^.^". You would also select arrow adjustment and range limits of 10 to 500 (in tenths).

When this message is used, your PLC program would put 15 in register M (or M+1) and the current peel rate value in M+2 (or M+4). If the current peel rate was 5.7 bananas per second, the display would read "Set Peel Rate: 5.7". Pressing the arrow keys would adjust the value up or down while continuously writing the value to data register M+2 (or M+4). When the adjustment is complete, the operator would press ENTER. The OM620J will then set DA. DA will remain set until a new message is selected via M (or M+1) or the DAK bit is set.

Function Buttons

The OM620J contains five user definable pushbuttons. These pushbuttons can be custom labeled and used for any purpose.

The pushbuttons can be individually configured as either alternate-action or momentary pushbuttons. Alternate-action buttons alternate state each time they are pressed. Momentary buttons are active only while they are being pressed.

The status register (M+7) holds the current state of each of the five pushbuttons. In a typical PLC application, these pushbuttons would be mapped to control contacts for easy ladder logic interface.

Menu Tree Operation

The OM620J terminal is designed to allow you to create a menu tree for function selection. The menu tree allows for interactive selection of a required function operation from a "menu" or list of options.

The OM620J allows up to four levels of menu. Each menu selection can be either a function or the path to the next lower menu level (sub-menu). With this type of "tree" arrangement, you can construct an application menu that goes from general to specific. You can also place frequently used menu selections on the top layer and infrequent selections on lower layers.

The graphic below is an example of a typical menu tree.

- Process Part type 1
- Process Part type 2
- Manual Control
  - Turn Pump On
  - Turn Pump Off
  - Turn Heater On
  - Turn Heater Off
  - Turn Mixer On
  - Turn Mixer Off
- Setpoint Entry
  - Tank Level Setpoints
    - Set Minimum Level
    - Set Maximum Level
    - Set Low Alarm Level
    - Set High Alarm Level
  - Temperature Settings
    - Temperature Setpoint
    - Set Low Alarm Temp
    - Set High Alarm Temp

Sample Menu Tree

The example shown has three menu layers. The top level has four selections. When "Process Part type 1" is selected, the associated function number will be placed in X+6 and the FS flag set. "Manual Control" is a sub-menu which will bring up the next menu. When selected, each item under "Manual Control" is a function which will place a function number in X+6 and set the FS bit. The "Setpoint Entry" selection from the top level menu will bring up a second level menu of two items. Each second level item will in turn bring up another level. The items at the bottom level will perform actual function selections.

When a function is selected through the menu tree, the PLC program should start the function process. **Once a function is selected, the menu selection is locked onto the display until the PLC program clears the ME flag.** When the PLC program clears the ME flag, control of the display will return to your program. When the function is complete, you may return to the same point in the menu by setting the ME and MR flags. To require the user to start the menu selection process from the beginning ("Process Part type 1" in this case), don't set the MR flag.

The menu enable is totally under PLC program control. If for any reason while in the menu you need to take control of the display from the menu to the program, just clear the ME bit.

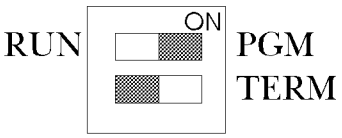
Each menu tree message takes up one of the 160 available total messages.

Configuration

Configuration of the OM620J terminal or system of OptiMate modules is performed via an IBM PC compatible computer with the Windows operating system. Optimation supplies OM-WINEDIT software that will allow you to select module configuration, system configuration and PLC protocol definition.

If the OM620J is to be operated stand alone with a PLC, the configuration selection must be made to select the proper PLC protocol information. If it is part of a multi-panel system, each panel must be configured under the multi-panel selection in OM-WINEDIT.

Note : When configuring, always remember to set the RUN/PGM DIP switch to PGM (towards the "ON"). Always wait for the module to reset (approximately 2 seconds) before downloading the configuration to the module.



Specific configuration of the OM620J begins with defining the block of PLC registers to be used. Next, each of the function buttons must be configured for either momentary or alternate-action operation. Then each of the messages used by the PLC program must be defined.

Message definition is very straightforward and easily accomplished. All that is necessary is the following sequence.

- Select the message number to enter.
- Type the message. Up to 20 characters are allowed. Any unused characters will be filled with blanks. One numeric field per message may be defined with caret `^` characters. One decimal point or colon may be placed within the field.
- If the message has a field for numeric arrow adjustment, select arrow adjustment. Select and enter minimum and maximum values if they apply.
- If applicable, select the data type for the message.

Examples of Use with an Entertron PLC

Register Usage

The OptiMate OM-WINEDIT software allows you to configure a module to use a block of 9 contiguous registers at a starting value that you define. For an Entertron PLC the recommended memory to use is the general purpose data memory starting at TC1. Any block of registers within the TC memory range can be used as long as it does not conflict with any devices that may be using registers in the TC memory area.

The first seven PLC registers in the block used by the OM620J panel are used for numeric information. As such they are ideally suited for the general purpose data memory registers TC1 - TCXX. (See table below for base register address ranges.) The last two registers use individual bits for control and status. These registers are better suited for the internal relays. The solution to this minor conflict is to define the base register address in the TC memory area and place a rung in your PLC program to copy the last two registers to/from internal relays. See the example at the bottom of the page.

The following table lists the base register addresses as well as the internal relay addresses for the various Entertron PLCs.

Entertron PLC	Valid Base Register Address Range
Standard OS	TC1 - TC16
Enhanced OS	TC1 - TC242
Modbus OS	TC1 - TC232
Elite 2000	TC1 - TC1016
Entertron PLC	**Internal Relay assignment**
Standard OS	RLY1 - RLY149
Enhanced OS	RLY1 - RLY149
Elite 2000	RLY1 - RLY512

With the memory mapping rung shown at the bottom of the page placed in the PLC program, the status and control bits will correspond to internal relays. The register association is shown in the figure below.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0 bit	
Register	MSB															LSB	
TC8					12	11	10	9	8	7	6	5	4	3	2	1	Status Register
TC9													16	15	14	13	Control Register

This will result in the following internal relay association for the status and control registers.

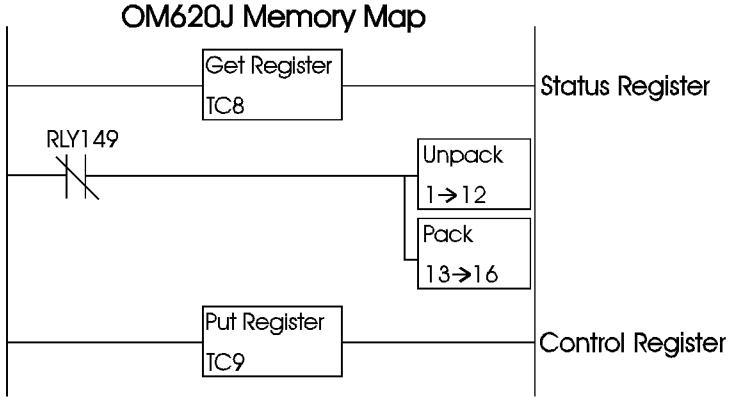
Status Register (M+7)			Control Register (M+8)	
Bit	Relay		Bit	Relay
F1	RLY1		ME	RLY13
F2	RLY2		MR	RLY14
F3	RLY3		DAK	RLY15
F4	RLY4		BD	RLY16
F5	RLY5			
DA	RLY6			
FS	RLY7			
AB	RLY8			
MA	RLY9			
ENTER	RLY10			
Up arrow	RLY11			
Down arrow	RLY12			

Memory Mapping Example

The program rung on the right shows a typical way of mapping the control and status registers to/from the internal relays of an Entertron PLC. In this example, the OM620J is configured for a base register address of TC1. **The program rung should be placed in the program to copy the status register TC8 (M+7) to internal relays beginning with RLY1 and copy from the internal relays beginning with RLY13 to the control register TC9 (M+8).** The middle rung should always be active. This ensures that the memory mapping happens every scan.

The location of the status and control registers depends on the base register address (the location of register M). In this case, the base register address is TC1. Any valid internal relays can be used in the memory map. If you do not use the relays shown in the example, adjust each lamp, button, etc., to correspond to the relays that you define.

OM620J Memory Map

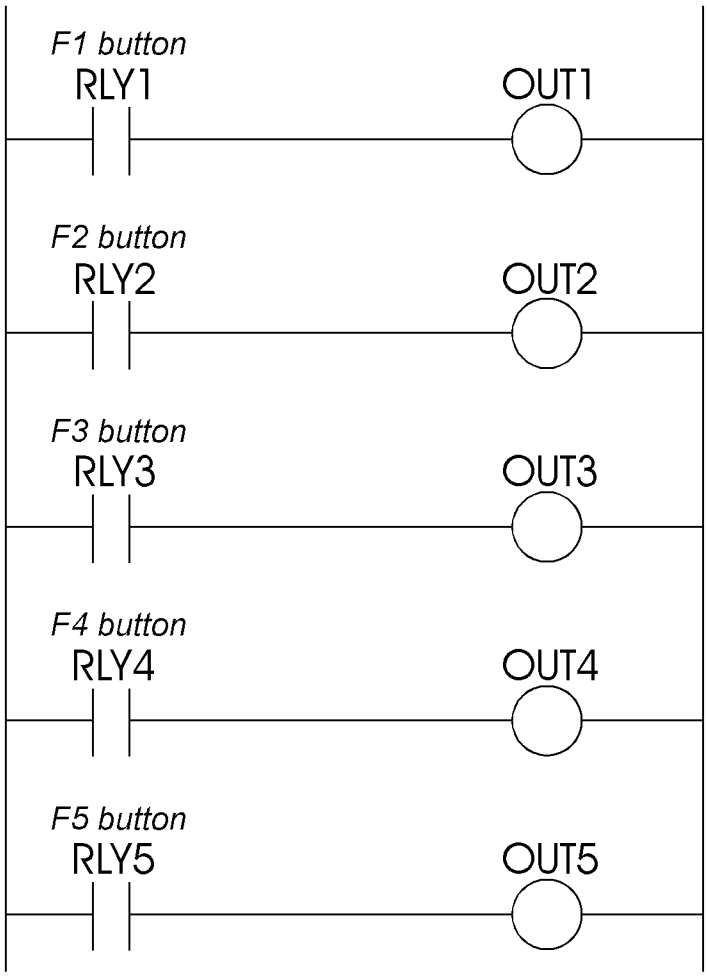


Programming Examples

The examples on the following pages use an OM620J connected to an Entertron PLC. The OM620J is configured for a base address of TC1. The following examples correspond to the memory mapping program rung shown on the previous page. That rung should be placed into the top of the PLC program.

Using a Function Button

The five function buttons below the LCD display will appear as internal relay coils in your program (assuming the memory mapping rung described previously is in your program). The example on the right turns ON output OUT1 when button F1 is active. Also, output OUT2 is turned ON when button F2 is active and output OUT3 is turned ON when button F3 is active. Output OUT4 is turned ON when button F4 is active and output OUT5 is turned ON when button F5 is active.



OM620J Configuration

Edit Help

Panel Address: 0

PLC Base Register Address: TC1

Pushbuttons

	Alternate	Momentary
#1:	◉	○
#2:	◉	○
#3:	◉	○
#4:	◉	○
#5:	◉	○

Panel:

Close

Labels

Write to Panel

Configure Messages:

Msg	Text	Action	Decimal	Format	Range
1:	"Enter Setpoint: ^^^^":	Arrow	Fixed	BCD	0 - 9999
2:	"Product Rate: ^^^.^ ":	Display	Fixed	BCD	
3:	"Display Setpt: ^^^^ ":	Display	Fixed	BCD	
4:	"Good Parts: ^^^^ ":	Display	Fixed	BCD	
5:	"Reject Parts: ^^^^ ":	Display	Fixed	BCD	
6:	"Count Val: ^^^^^^^^ ":	Display	Fixed	BCD Double	
7:	"AvgPart/Hr_^^^^^^^^^^":	Display	Fixed	Floating Point	
8:	"Process Step 1 ":				

Delete Message

Clear List...

Configure Menu Tree:

<< Promote Items >> Demote Items

Item	Fctn	Level	Text
1:		1	"Process No. 1 "
2:	5	2	"Process #1 Automatic"
3:		2	"Process #1 Manual "
4:	1	3	"Fill Tank #1 "
5:	2	3	"Drain Tank #1 "
6:		1	"Process No. 2 "
7:	6	2	"Process #2 Automatic"
8:		2	"Process #2 Manual "

Delete Menu Items

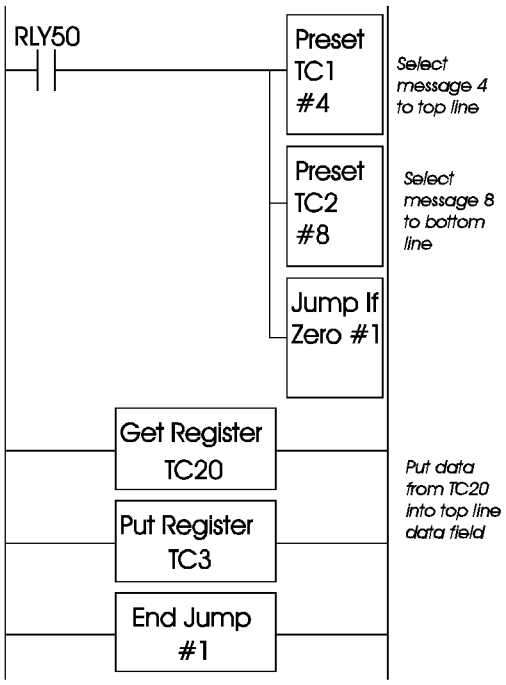
Check Menu Tree

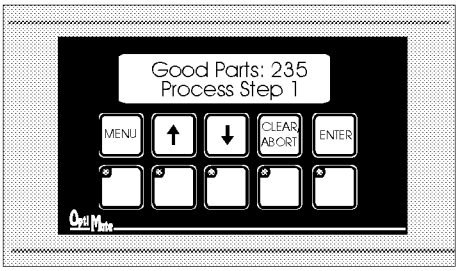
Displaying Messages on the LCD Display

Messages of various types can be configured via OM-WINEDIT and downloaded to the OM620J. The message definitions shown in the figure above will be used in all of the examples that follow.

The example on the right shows a couple of messages being displayed on the LCD display. This example will work for BCD or Binary data. The top line uses data display message 4. The data for the data field is coming from TC20. The bottom line is text message 8.

Entertron PLCs do not have an easy way of manipulating BCD Double or IEEE 32-bit floating-point numbers. Examples on the next page show how to display these types of numbers *if* the data is in the proper format within the PLC.

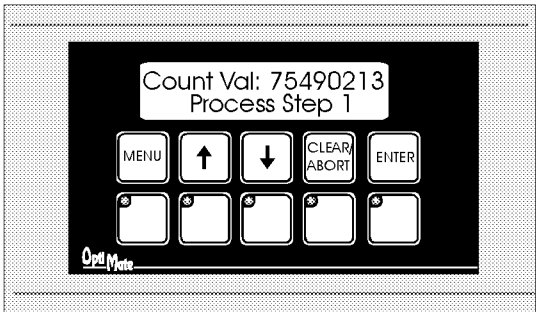


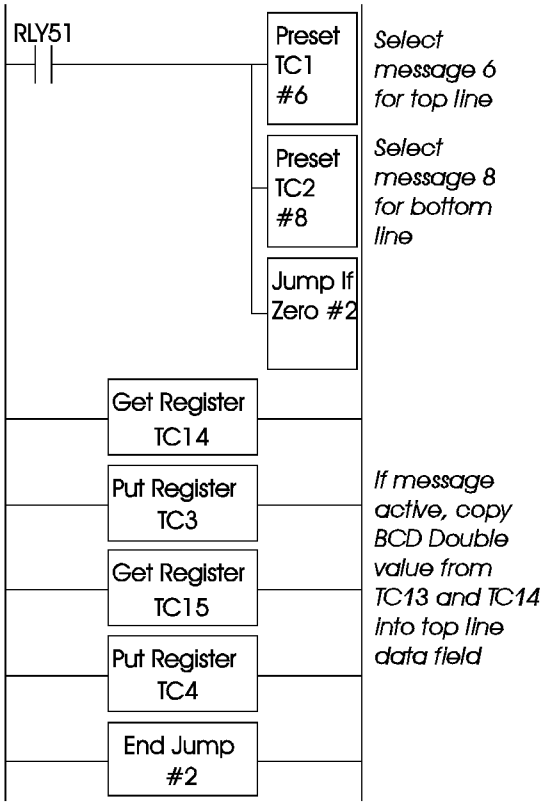


TC1	Top line message selection
TC2	Bottom line message selection
TC3	Top line data
TC4	Top line data 2 (for long BCD)
TC5	Bottom line data
TC6	Bottom line data 2 (for long BCD)
TC7	Function Selection
TC8	Status register
TC9	Control register

Displaying BCD Double Numbers

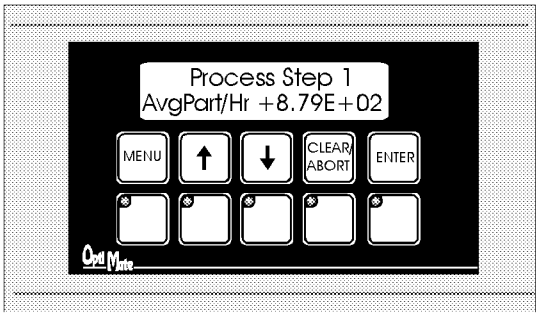
The example shown on the right is similar to the last example. The primary difference is that it uses a BCD double number in the top line display. The top line uses data display message 6, which has been configured as a BCD double display. The data for the data field is coming from TC14 and TC15. TC14 contains the least significant word while TC15 contains the most significant word of the BCD double value. The bottom line is text message 8.

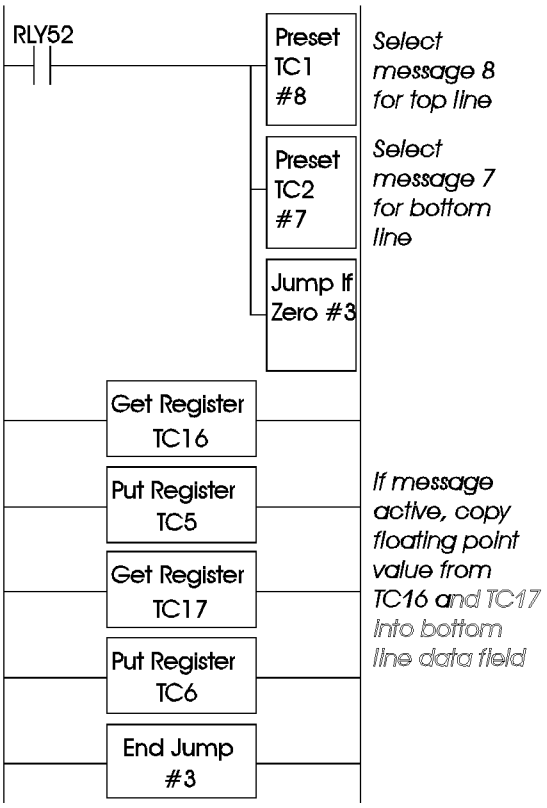




Displaying Floating-Point Numbers

The example shown on the right is similar to the previous example. The primary difference is that it displays a floating-point number. The bottom line uses data display message 7, which has been configured as a floating-point display message. Since the data is a floating-point number, it uses two 16-bit registers. The two registers have to be looked at together, not individually, for you to be able to make any sense of the data. In this example, the data is loaded from TC16 and TC17 to the bottom line display registers TC5 and TC6. The top line is text message 8.





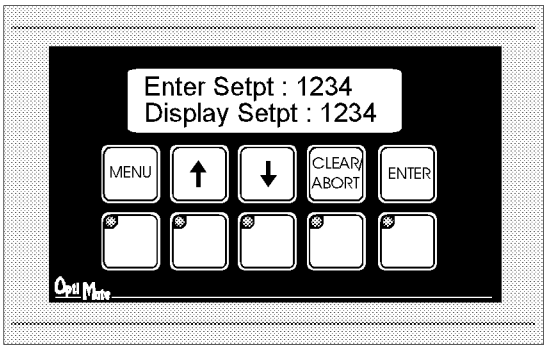
Arrow Adjustment of Setpoint Data

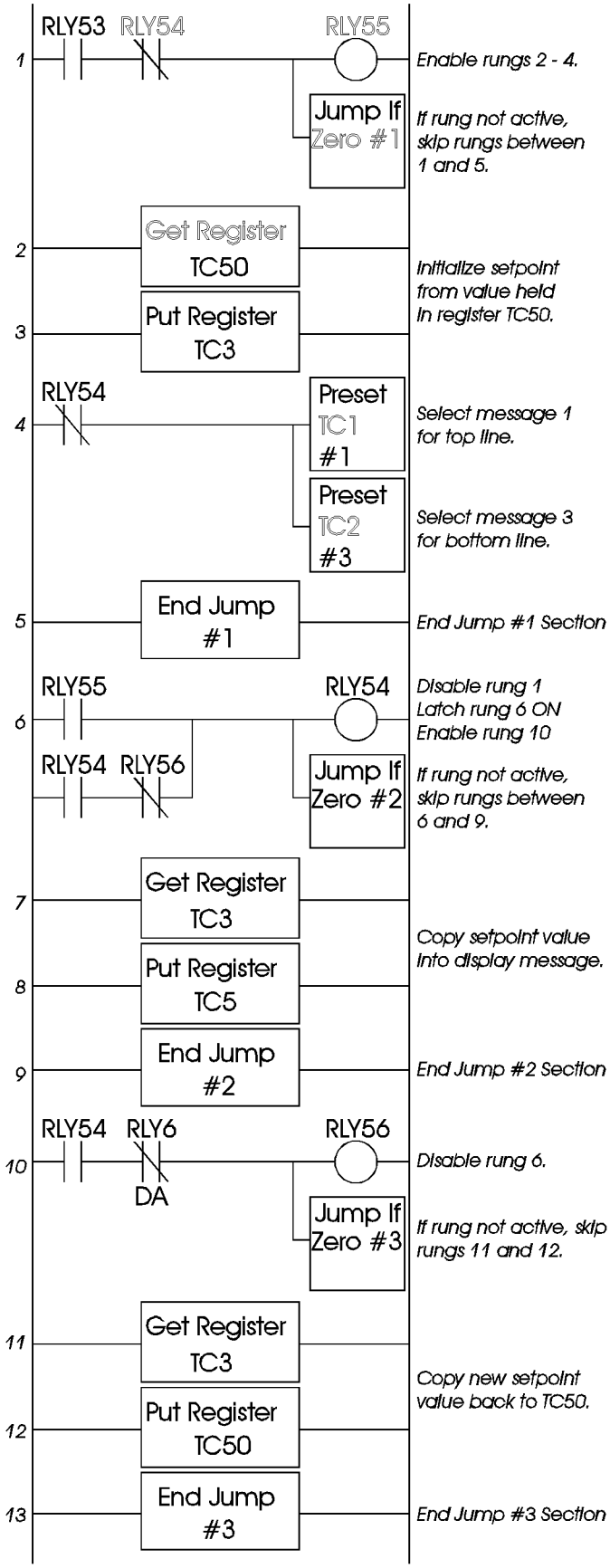
The figure to the right illustrates arrow adjustment of setpoint data. The example loads an arrow adjust message on the top line and a data display message on the bottom line. It reads the initial value for the arrow adjust message from a register and places it in the top line data register. Next, it continually copies the value of the arrow adjust message to the bottom line message. When the ENTER key is pressed, it copies the arrow adjust data back to the original register.

In the example, the process is enabled when RLY53 is energized. When enabled, rung 1 enables rungs 2, 3, 4 and 6. Rungs 2 and 3 initialize the setpoint value by copying the initial value from TC50 to the top line data register, TC3. Rung 4 places the "Enter Setpoint" and "Display Setpt" messages on the top and bottom lines of the LCD.

Rung 6 disables the first rung which disables rungs 1 - 4. It also latches itself on until unlatched by rung 10 and enables rungs 7 and 8. When enabled, the rungs 7 and 8 continually copy the setpoint value on the top line to the display message on the bottom line.

Rung 10 waits until the data-available flag (RLY6) is set, then copies the new setpoint value back to TC50. It also unlatches rung 6 which disables rungs 7 and 8. Rung 10 also turns itself off by disabling rung 6. Disabling rung 10 disables rungs 11 and 12.



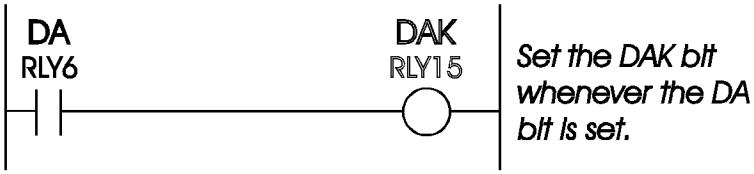


Acknowledging Data Entry

When the ENTER key is pressed during an arrow adjust message, the panel will set the data available (DA) bit in the status register. The arrow adjust message data will lock up (i.e. can't be changed by the user) until the DA bit is cleared.

The DA bit can be cleared by two means. The first is to change the arrow adjust message to a message with a different number. The other method used to clear the DA bit is to set the data acknowledge (DAK) bit.

If you want to use the DAK bit to clear the DA bit, the program example on the right shows how to implement that procedure. **This rung should be placed at the end of the program to ensure that it will be checked every scan.** The rung will remain active as long as the DA bit is active. When the module sees that the DAK bit is set, it will write to the status register and clear the DA bit. This will normally happen in less than a second. When the DA bit clears, the panel will unlock the arrow adjust message and the user can change it if necessary.



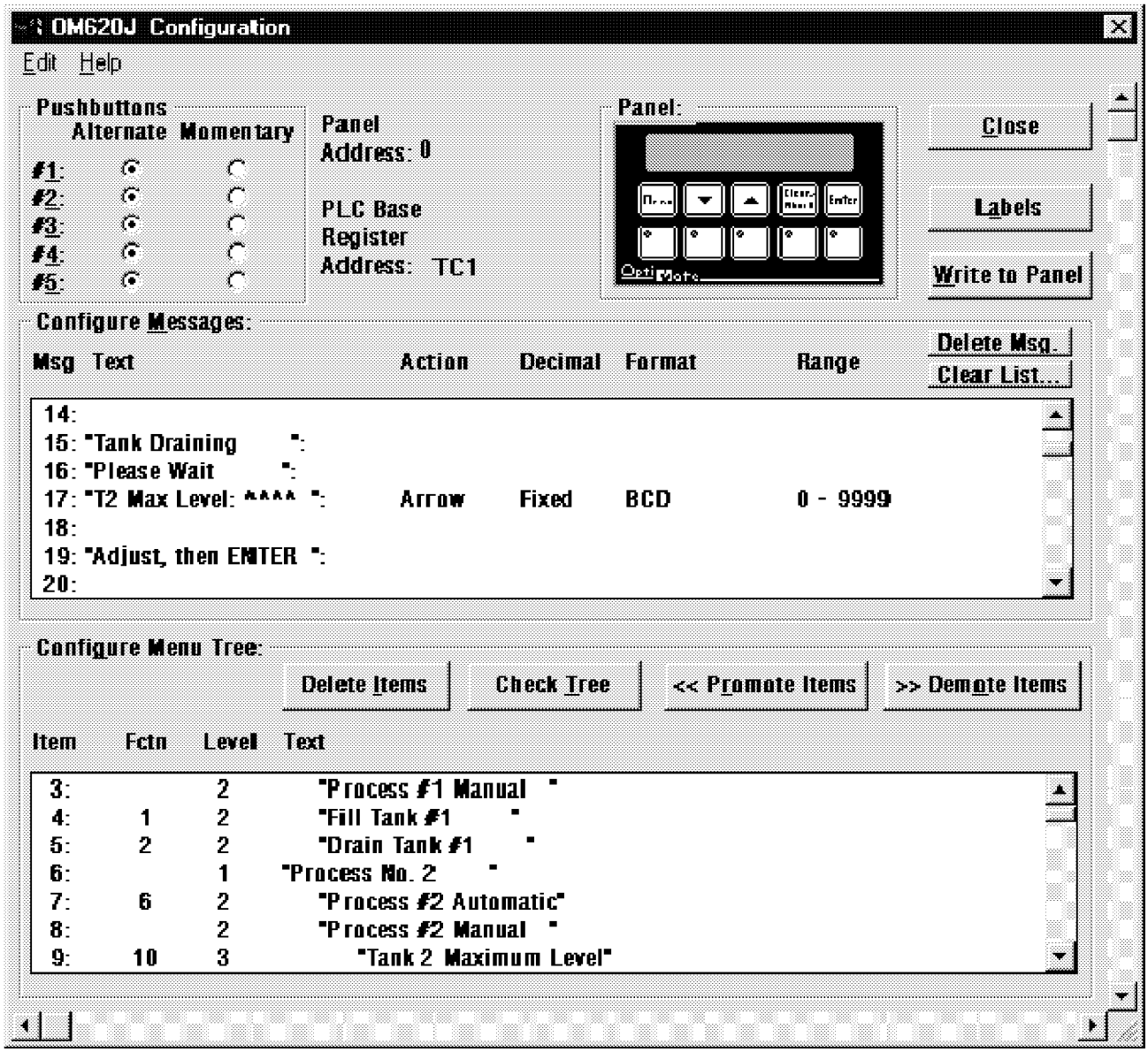
Using a Menu Tree

The OM620J allows you to pre-define and use a layered menu tree for function selection. The operation of this menu tree is taken care of entirely by the OM620J panel. The only requirement that the user has is to enable or disable the menu operation, and branch to the appropriate function logic when a function selection is made.

We will use the menu-tree definition shown below for our example.

The OM-WINEDIT Configuration Editor screen shown below displays the menu structure, including function associations, on the lower section of the display. For example, if the operator selects "Process #1 Automatic" from the menu tree, function number 5 would be placed in the function select register (M+6). If "Process No.2" is selected, no function number is selected; the terminal display will go to the next lower layer of the menu - "Process #2 Automatic".

Any menu item that has lower level menu items below it will, when selected, branch to the next level. If the "MENU" button is pressed, the terminal will back the menu tree up to the next higher level (towards the trunk). Arrow keys will, when pressed, step the panel through selection items on the same level of the same branch. The lowest level items on any branch will be function selections.



A Menu Tree Example

The program shown on the right illustrates menu tree function selection using the menu tree shown on the previous page. The first rung enables the menu tree by setting the ME bit when RLY100 is active.

> Note : Enabling the menu tree does not automatically put the terminal into the menu. Once enabled, the terminal will bring up the menu tree when the operator presses "MENU". Until then, the display is under PLC program control.

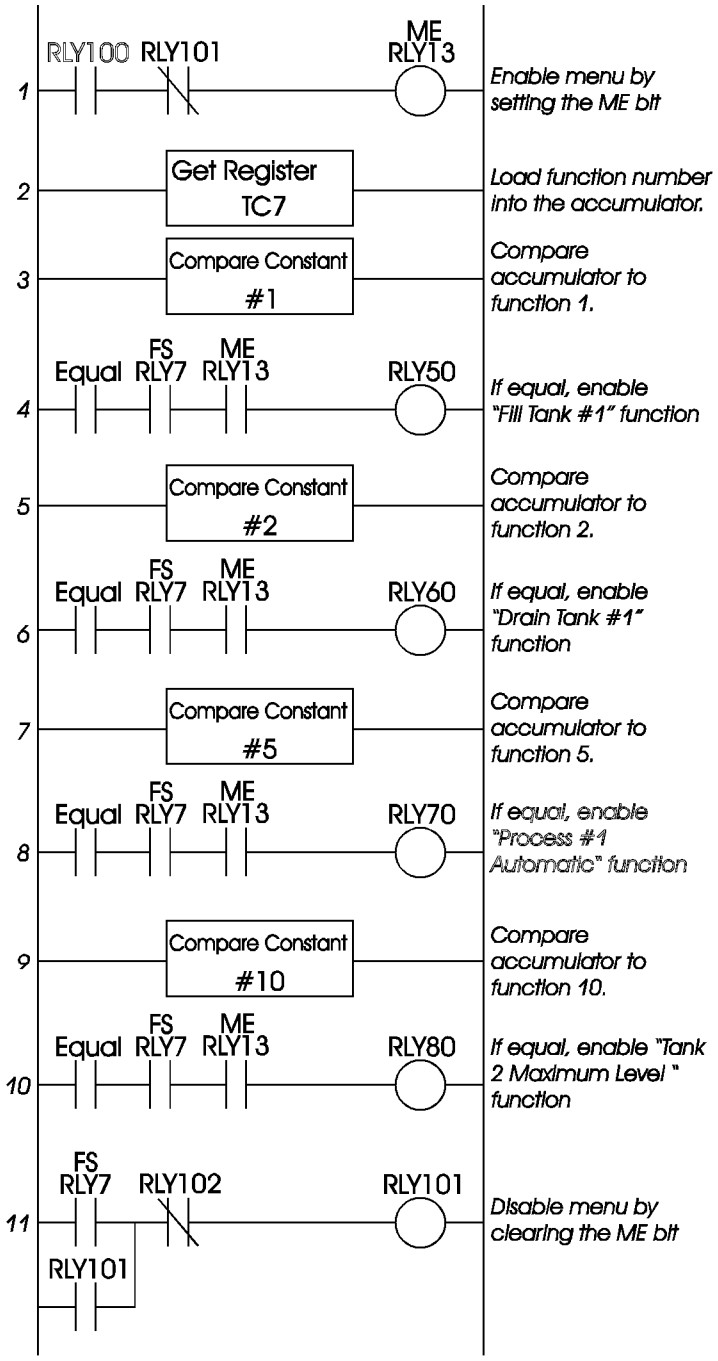
Decoding a Function Number

Decoding the function number consists of comparing the value in the function selection register (M+6) with the function numbers defined in the menu tree. This is shown in rungs 2 - 10. Rung 2 loads the function number selected in the OM620J into the accumulator. Rungs 3, 5, 7, and 9 compares the function number in the accumulator with a constant value that represents the various function numbers. When a match is found, the appropriate enable relay (RLY50, RLY60, etc.) (you choose it) should be energized.

The last rung shown will disable the Menu Enable (ME) bit by latching RLY101 on. When the ME bit turns off, the panel will exit the menu tree. Control of the OM620J's LCD will return to your program and the action your function takes will depend on your program. RLY102 is used to unlatch the rung. RLY102 should be activated at the end of each function to disable RLY101, therefore activating the ME bit (RLY13).

Notice that the logic shown interlocks the function enable bits (setting RLY50, RLY60, etc.) with the function select and menu enable flags (RLY7, RLY13) to ensure that only one function is executed each time a menu function is selected.

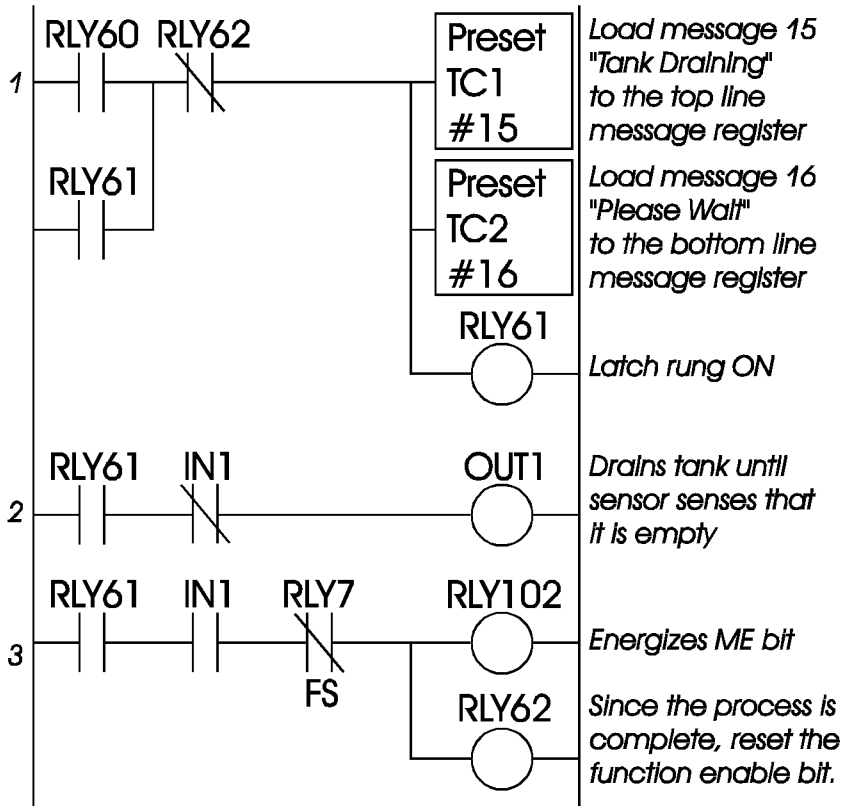
See the next two pages for implementation of functions.



Implementing a Menu Function that utilizes the Message Display Capability

The program logic shown below illustrates how a typical function might be implemented. Suppose the selection was made by the operator to "Drain Tank #1". **According to the logic shown on the previous page**, the selection of "Drain Tank #1" will place the function number 2 in register TC7 which in turn would result in internal relay RLY60 being energized. The first rung of the "Drain Tank #1" function, shown below, will put messages 15 ("Tank Draining") and 16 ("Please Wait") into the top and bottom lines of the display. RLY61 is energized to latch rung 1 on and enable rungs 2 and 3. The second rung energizes output OUT1 to open the drain until level sensor input IN1 senses that it is empty. When the tank is empty, the third rung re-enables the menu by energizing RLY102. The third rung also disables the function by energizing RLY62. By putting the function select relay (RLY7) in series, we force the program to wait until the function select has been cleared before re-enabling the menu.

Note: Multiple outputs coils cannot be used in a program. RLY102 will be needed multiple times if you have more than one function to implement. Therefore, you can use a different relay for each function and set each to activate RLY102 by using a logical OR type rung.

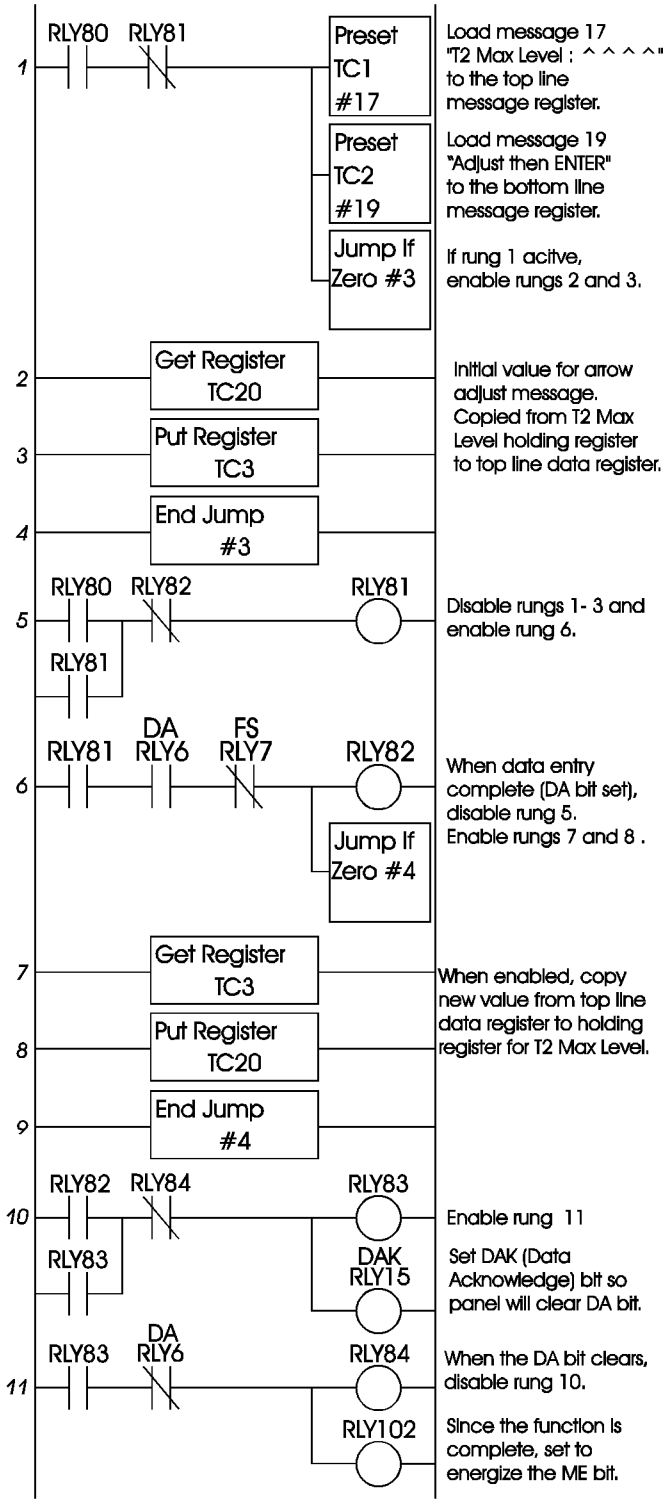


Implementing a Menu Function that uses an Arrow Adjustment Message

The logic shown at right is a typical example of how a setpoint function might be implemented. Suppose the menu tree selection for "Tank 2 Maximum Level" was made. From the logic shown in ***Decoding a Function Number***, the selection of "Tank 2 Maximum Level" will place function number 10 in register TC7, which in turn will result in relay RLY80 being set. Accordingly, the first rung shown at right selects setpoint message 17 (T2 Max Level: ^^^^") for the top line and message 19 ("Adjust then ENTER") for the bottom line. The first rung also enables rungs 2 and 3 which initialize the setpoint value for arrow adjustment by copying the value from its holding register (TC20) to the top line data register (TC3). Rung 5 disables rungs 1 - 3 and enables rung 6.

Rung 6 waits for the setpoint data to be entered (after the FS flag is cleared), then enables rungs 7, 8 and 10. Rungs 7 copies the new value from the top line data register and rung 8 places that value in its holding register (TC20).

Rung 10 latches itself on, sets the DAK bit and enables rung 11. Rung 11 waits until DA bit is cleared by the OM620J, then clears the DAK bit, re-enables the menu and disables rung 10. The DAK bit is cleared by disabling rung 10 (turning RLY84 on) and the ME bit is set by enabling RLY102 (as shown in the ***Decoding a Function Number*** example). If the menu return (MR) flag had been set, the OM620J would have returned to the menu tree item "Tank 2 Maximum Level".



Use in a Microprocessor-Based System

OptiMate modules can interface a microprocessor-based controller over a serial link. This link can be either RS232 (for point to point) or RS422 (for multidrop or point to point). In either case the microprocessor acts as the master. It can write data to the panel or read data from the panel.

The OM620J uses the OptiMate Hex protocol for fast and easy communications. The OptiMate Hex protocol is defined in subsequent pages.

Module Address

In a microprocessor system, each panel must have its own unique address. You define this address (between 0 and 30) using the OM-WINEDIT configuration editor. The panel will respond to the host only if it is properly addressed.

Communications Protocol

To use an OptiMate panel as a slave device in a microprocessor-based system, the panel must be configured for the OptiMate Hex protocol. The other options that must be set are module address, baud rate, parity and number of stop bits (note; if parity is set to even or odd, only one stop bit is allowed). Once selected, the configuration must be downloaded to the panel.

Computer-Based Operation

The OM620J protocol for computer based operations is the OptiMate Hex protocol.

All of the basic functionality described for PLC operation is also available to computer or microprocessor-based applications.

The following is a synopsis of the OM620J operation as it relates to computer based applications. In certain cases, more detail is provided under the same topic for PLC operation. The details of messages involved are covered in the protocol documentation which follows.

Displaying Messages on the LCD Display

Under computer-based operation, the OM620J can be instructed to display predefined messages. Selection of a predefined message simply requires transmitting a command which selects the particular message.

Placing Numeric Data in the Display

Any message containing caret '^' place holders will allow either the display of numeric data or numeric data entry.

Numeric data for the numeric data field can be transmitted by the host computer. If it is a display data field, it may be continuously updated. If it is a data adjustment message (arrow key adjustment), only the first value transmission will be accepted.

Data Entry/Adjustment with the Arrow Keys

Data entry in the OM620J can be performed via the arrow adjustment keys. This type of data entry is most applicable to cases where you want to have a number adjusted slightly, rather than entered from scratch. This happens many times with setpoints. It is also common when adjusting speeds and rates. When this is done in an operation, it is commonly referred to as a "jog" operation.

A message can be set up for arrow adjustment through the OM-WINEDIT Editor. When the data screen is required in the program, transmit a command for the preconfigured message and its current value. The digits marked by carets will initially display the current value.

As the operator presses the up or down arrow key, the numeric value will increment or decrement respectively. As it is adjusted, the value is continuously available to be read by the host computer. When adjustment is complete, the operator will press the ENTER button. When this happens the data available (DA) status bit will be set. When the computer host sends a DAK or changes the message, the DA status bit will clear.

Adjustment of data will be limited to within the limits defined through the configuration editor. Data also will be limited to the number of digits defined by the carets in the message.

Function Buttons

The OM620J contains five user definable function buttons. These buttons can be custom labeled and used for any purpose.

The buttons can be individually configured as either alternate action or momentary pushbuttons. Alternate action buttons alternate state each time they are pressed. Momentary buttons are active only while they are being pressed.

The status request message will return the current state of each of the five buttons along with other status.

> Note: All OptiMate Hex protocol messages are transmitted in hexadecimal number format. The message responses will be transmitted back to the host in the same number format.

OM620J OptiMate Hex Protocol

General Format

STX Module address function ftn_data checksum

Where STX = 0x02
Module address = 0 to 30
function = 0xA0 ; General status/control
0xA1 ; Select predefined message display
0xA5 ; Read operator entered data
0xA7 ; Send data for data display message
ftn_data = data specific to the function
checksum = 8 bit sum of all characters after address until checksum

Note : Spaces are shown for readability only. There are no spaces between message fields.
0xXX denotes a hexadecimal number.

General status/control

STX Module address 0xA0 control checksum

where control
bit 0 ; Menu enable
bit 1 ; Menu return
bit 2 ; Data acknowledge
bit 3 ; Buzzer Disable

response

STX term_stat disp_stat ftn_select checksum ***if message received and processed OK***

where term_stat = terminal status
bits 0-4 = Button status for function buttons 1-5 respectively (1 = active, 0 = inactive)
bit 5 : operator entered data available
bit 6 : function selected
bit 7 : abort key pressed
disp_stat = display status (1=TRUE)
bit 0 : Top line display data
bit 1 : Top line arrow adjust
bit 2 : Bottom line display data
bit 3 : Bottom line arrow adjust
bit 4 : Enter button
bit 5 : Up arrow
bit 6 : Down arrow
bit 7 : Menu active

or

NAK if any errors

Where NAK = 0x15

Select Predefined Message

STX Module Address 0xA1 line mesg_no data checksum

where line = top (0x00) or bottom (0x01) line
mesg_no = number of the predefined (through the configuration editor) message (1 - 160, hex integer, i.e. 33 = 0x21)
data = 4 bytes. The format depends on the message type selected
> For an integer type data message, the first two bytes are not used (send as 0's). The Third byte is the high 8 bits of the 16 bit integer data. The fourth byte is the low 8 bits.
> For BCD data type message, the data is sent MSByte through LSByte. For a long BCD Data type message, all 4 bytes (8 digits) are used. For a regular BCD message, only the last two bytes are used.
- Used as display data for numeric data display message.
- Used as initial value for arrow adjustment type message (integer portion only).
- Ignored for all other message types.

response

ACK ***if message received and processed OK***

Where ACK = 0x06

or

NAK ***if any errors in message***

Read Operator Entered Data

STX Module 0xA5 checksum
Address

response

STX status data checksum ***if message receive and processed OK***

where status = 0x00 if ENTER key has not been pressed
= 0x01 if ENTER key has been pressed

data = 4 bytes. The format depends on the message type selected
> For an integer type data message, the first two bytes are not used (send as 0's). The Third byte is the high 8 bits of the 16 bit integer data. The fourth byte is the low 8 bits.
> For BCD data type message, the data is sent MSByte through LSByte. For a long BCD Data type message, all 4 bytes (8 digits) are used. For a regular BCD message, only the last two bytes are used.

Note : This message can be used to read an arrow adjust message value while it is being adjusted.

or

NAK ***if any errors in message***

Send data for data display message

STX Module 0xA7 line data checksum
Address

where line = top (0x00) or bottom (0x01) line

data = 4 bytes. The format depends on the message type selected
> For an integer type data message, the first two bytes are not used (send as 0's). The Third byte is the high 8 bits of the 16 bit integer data. The fourth byte is the low 8 bits.
> For BCD data type message, the data is sent MSByte through LSByte. For a long BCD Data type message, all 4 bytes (8 digits) are used. For a regular BCD message, only the last two bytes are used.

response

ACK ***if message received and processed OK***

or

NAK ***if any errors in message***

Display status request

STX Module 0xA9 checksum
Address

response

STX top_msg bot_msg checksum ***if message received and processed OK***

where top_msg = last predefined message selected for top line
bot_msg = last predefined message selected for bottom line

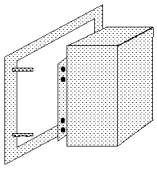
or

NAK ***if any errors in message***

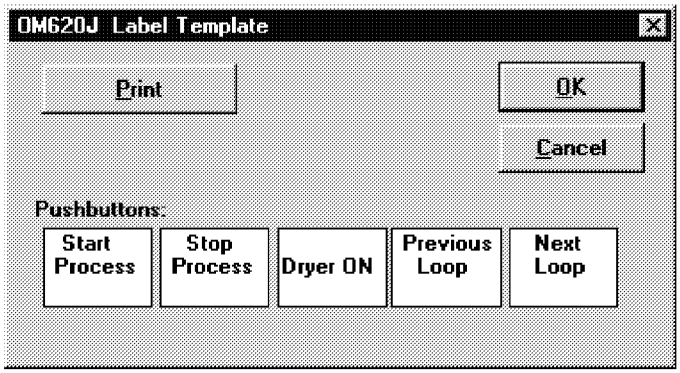
Set Up and Interconnect

Legending the Function Keys

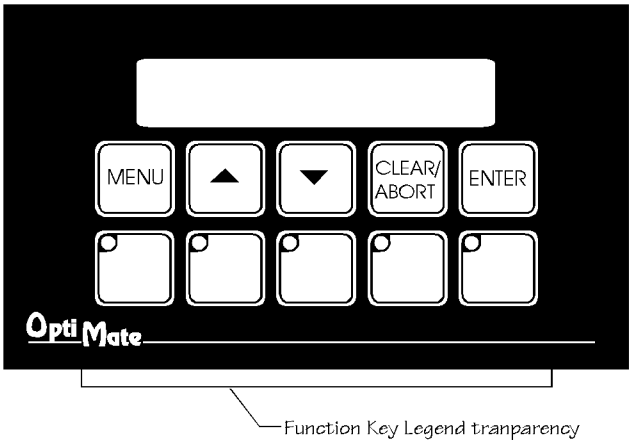
Legending the OM620J panel is a relatively simple process that basically involves sliding a label transparency into a pocket in the panel overlay. Use the following procedure.



- Remove the bezel from the panel. The bezel snaps to the panel box along the top and bottom edges. Pull the bezel out and over the snaps to remove.
- Create legend transparencies. There are a number of available options for doing so. Patterns are provided on the next to last sheet of this document.
  > Use the built-in label-making capability of the OM-WINEDIT software to create labels. Either print on the transparency directly or print on paper and photocopy onto the transparency. The figure below is a screen from OM-WINEDIT which illustrates the process.



Other options include the following:
  > Use a computer graphics program and a laser printer to create the transparency directly. Alternately print on paper and photocopy to a transparency.
  > Use press-on letters onto a transparency sheet.
  > Use a typewriter, lettering machine or press-on letters to letter onto paper, then photocopy.
- Cut along outline. Slide into overlay pocket. Pushbutton legends slide in from the bottom.
- Reattach bezel. Push bezel onto box until it snaps together.



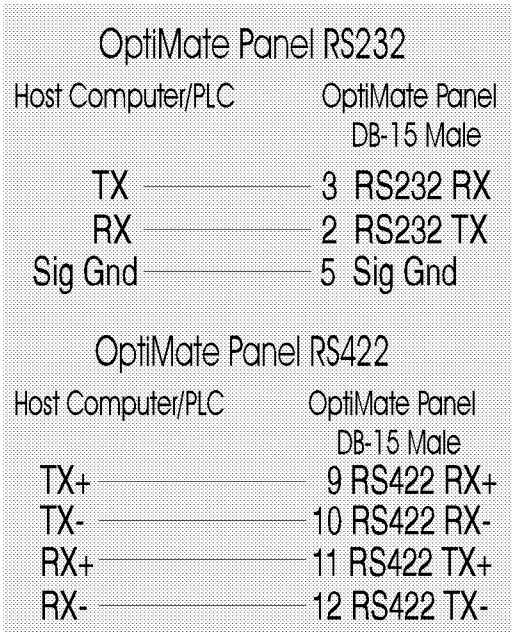
Connection to the System

OptiMate panels are designed for communication connection to system devices. The panel can be connected to a computer or PLC over the serial port (RS232 or RS422).

Connection to a Computer or PLC

Connection of an OptiMate panel to a computer or PLC can be accomplished over either an RS232 or RS422 link. RS232 is limited to one OptiMate module to one computer serial port. RS422 allows up to 31 modules to be connected to one computer port. Since PLCs are slave devices, the RS422 link for a PLC is limited to one OptiMate module.

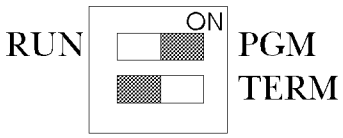
Refer to manufacturer's documentation for PLC or computer serial link connector pinouts. See the figure below for RS232 and RS422 pinouts for OptiMate modules.



Interface cables for connection to several different PLCs as well as to IBM PCAT compatible ports are available.

Termination

The termination DIP switch on the back of the panel switches on a terminating resistor. **This terminating resistor does not apply to an RS232 connection, therefore, it should be in the OFF position for RS232.** In an RS422 connected system, the termination DIP switch should be ***ON*** *in the last, and only the last, panel on the cable.*



Power

The OM620J panel will operate on any DC voltage between 8 and 30VDC. Power must be connected to the terminal plug located on the back of the module. Pin 1 of the plug is the 8-30VDC (+) terminal and pin 2 is the 0VDC (-) terminal. Steady state current is listed on the specification page.

There is a brief (0.5 to 2 millisecond) power on surge to 1.5 amps. This is typical of nearly any type of electronic equipment and is due to the initial charging of power capacitors. This surge is normally easily handled by commercial power supplies.

Configuration

Configuration Selections

OptiMate panels can be configured for the specific application by using the OM-WINEDIT Configuration Editor. The OM-WINEDIT Editor runs on any IBM PC compatible computer with Windows, Windows 95, 98 or NT. It allows the user to select the exact functionality to meet application requirements.

For the OM620J panel, the following are important configuration parameters.

Computer-Based Systems

Decision	Selection
Single/Multi Module	Choose single panel even if the system will contain several panels. The Multi panel selection applies only to systems using an OM9001. In computer based systems, each panel is configured independently. After configuration, multiple panels can be connected together to form a system.
Configuration starting point	First time configuration, start with defaults for panel. Subsequent configurations can utilize disk files you create.
PLC Type	Select OptiMate Hex
Protocol	Select appropriate baud rate, 8 data bits, #stop bits & parity. Note that if even or odd parity selected, only 1 stop bit is available.
Buttons	Select momentary or alternate action as required for your application
Messages	Define messages as required for application

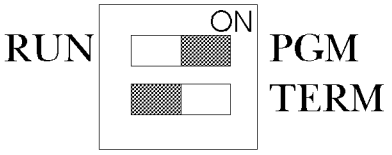
Single Module PLC-Based Systems

Decision	Selection
Single/ Multi Module	Choose single panel configuration
Configuration starting point	First time configuration start with defaults for the panel. Subsequent configurations can utilize disk files you create
PLC Type	Select appropriate PLC type
Protocol	Select appropriate baud rate, # data bits, # stop bits & parity. Note that if 8 data bits and even or odd parity selected, only 1 stop bit is available
Buttons	Select momentary or alternate action as required for your application
Messages	Define messages as required for application

Multi Module PLC Applications (Uses OM9001 Communications Master ... Not available with Entertron PLCs)

Decision	Selection
Single/Multi Module	Choose Multi module
PLC Type	This applies to the Communications master. Choose appropriate type
Protocol	This applies to the OM9001 Communications master. Choose appropriate baud rate, # bits, # stop bits & parity. Note that if 8 data bits and even or odd parity are selected, only 1 stop bit is available.
Address	Each panel must have a unique address
Panel Protocol	The OM-WINEDIT software will automatically select the OptiMate Hex protocol for communications between the OM9001 and the panel. (This is all transparent to the user)
Buttons	Select momentary or alternate action as required for your application
Messages	Define messages as required for your application.

Configuration must be downloaded from an IBM PC compatible computer to each panel. This is done over the serial link. Panel must be selected for "PGM" (DIP switch in back of the module) for module to accept configuration data. After the download to the panel is complete, wait a few seconds before switching the DIP switch from "PGM" to "RUN." The DIP switch must be in the "RUN" position for the module to operate with the selected host.



Configuration using an Entertron PLC

The OM620J requires that certain parameters be set in order for it to communicate with an Entertron PLC. The parameters are PLC type, protocol, baud rate, parity, stop bits and base register address.

If you are using the ASCII protocol, the following parameters should always be set:

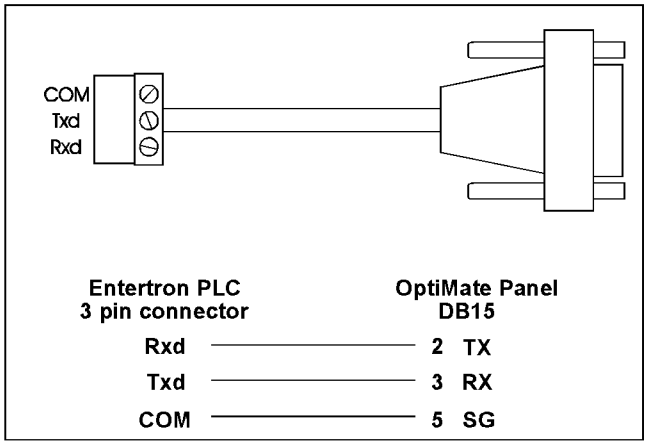
OM620J Communication Parameters Entertron ASCII protocol	
Baud rate	9600
Parity	none
Stop Bits	1
Data Bits	8

If you are using Modbus protocol, the following parameters should always be set:

OM620J Communication Parameters Entertron Modbus protocol	
Baud rate	9600
Parity	none
Stop Bits	2
Data Bits	8

For information on setting the base register address, see the section **Examples of Use with an Entertron PLC**.

The figure below shows the pinouts to connect an OM620J to an Entertron PLC CPU.



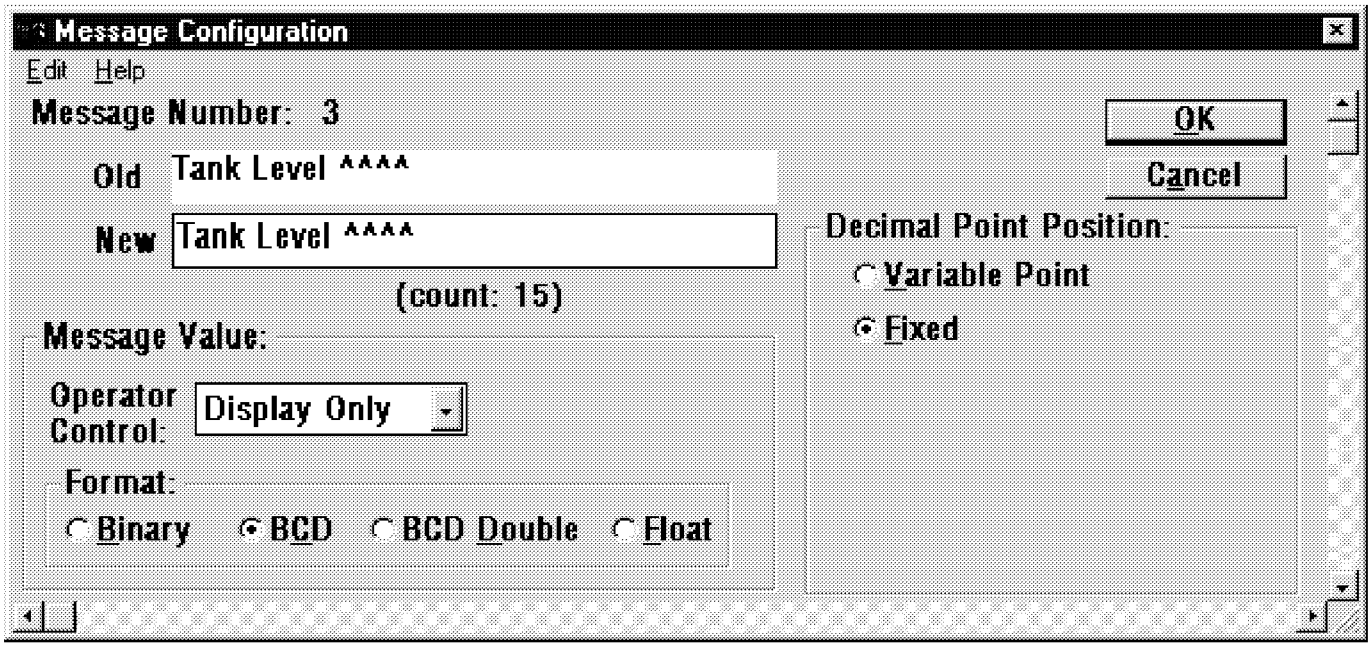
Creating Messages

The figure on right the illustrates the process of creating messages for your program to use. The first step involves simply using a message template to define, on paper, each of the messages. We suggest copying the template page and using it to define all of your messages.

The next step is to use the OM-WINEDIT editor to enter the messages as defined. Remember to use carets wherever variable data is to be used. The OM-WINEDIT editor will guide you through other definable parameter, including data type, message type, etc., as shown below.

Message #	Text (20Characters Max.)																			
1	E	n	t	e	r		S	e	t	p	o	i	n	t	:		^	^	^	^
2	P	r	o	d	u	c	t		R	a	t	e	:		^	^	^	.	^	
3	S	h	i	f	t		T	a	r	g	e	t	:		^	^	^	.	^	

Example Message Definitions

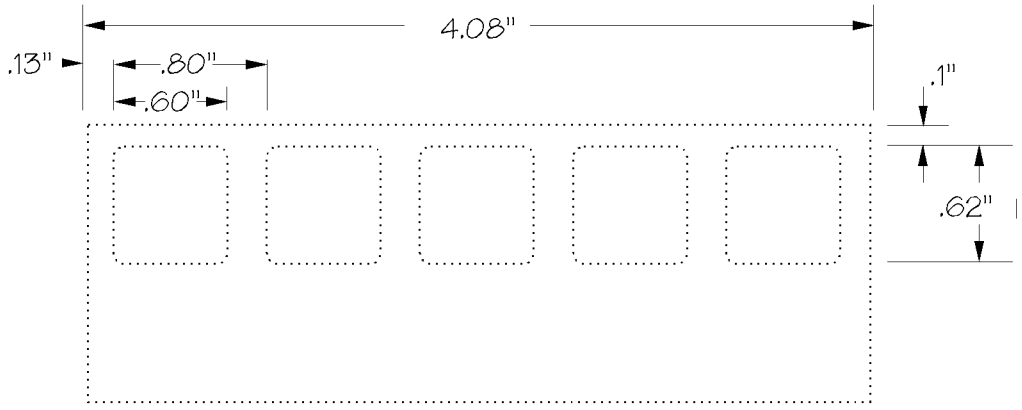


Label and Message Definition Templates

Message #	Text (20Characters Max.)

Message Definition Template

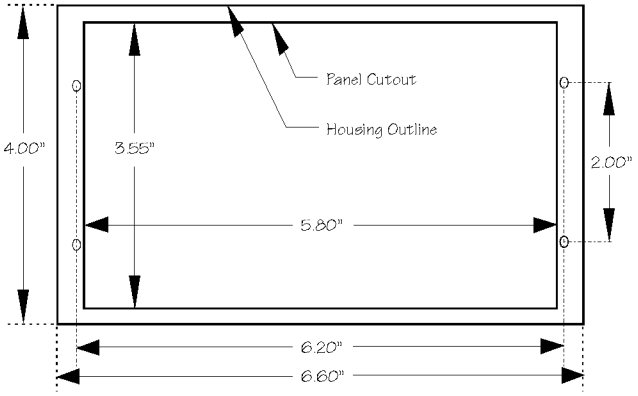
Pushbutton Label Strip Pattern



Specifications

Physical

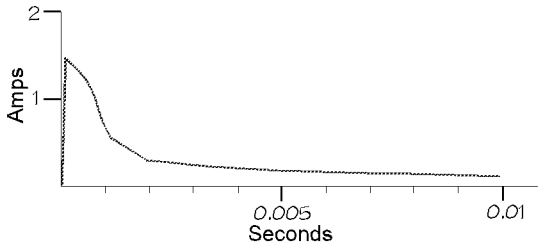
- Recessed Mount Housing 6.6”Lx4.0”H x 1.25”D
- Cutout size for above 3.55”Hx5.80”L
- Panel Fasteners : Four, 6x32 threaded studs, shown above (on ends, symmetrical about center line)
- Weight : 10 ounces
- Colors : Dark gray housing with dark gray panel. Keypad keys; white with dark gray letters. White with user supplied label.
- LCD Display : 2 line x 20 character STN with LED backlight character size : 5.5mm high x 3.2mm wide



Panel Mounting Dimensions

Electrical

- Power : 8 - 30VDC @ 1.8Watts
  150mA @ 12VDC 75mA @ 24VDC
- Power on surge (see figure below)
  1.5A for 2 milliseconds maximum



Power on Surge

- Power connector : Pluggable terminal block, 2 position

Communications

- RS232 and RS422
- 4800 to 19200 baud
- Compatible with Entertron PLCs using ASCII or Modbus
- OptiMate Hex protocol for computer based systems
- 15 pin female ‘D’ shell connector

Communications Failure Operation

Should the module (when not selected for configuration) ever fail to communicate successfully for a period of 12 seconds, the LEDs inset in the corner of the 5 buttons will all simultaneously flash at a rapid rate.

Environmental

- Enclosure - NEMA 4 (when properly installed)
- Temperature - 0 to 50 C
- Humidity - 95% non-condensing

Message Types (160 user defined messages available)

- General Text message
- Data display message (one data value per line)
- Arrow adjustment data message (integer or fixed point)

Numeric Types & Values

- Integer
- Fixed Point
- BCD (Values between 0 & 9999; with appropriate decimal placement)
- BCD Double (values between 0 & 99999999 with appropriate decimal placement)
- Binary (Values between 0 & 65535 with appropriate decimal placement)
- Floating-Point (Values between -3.402823E+38 to +3.402823E+38 in the format of ±X.XXE±XX)