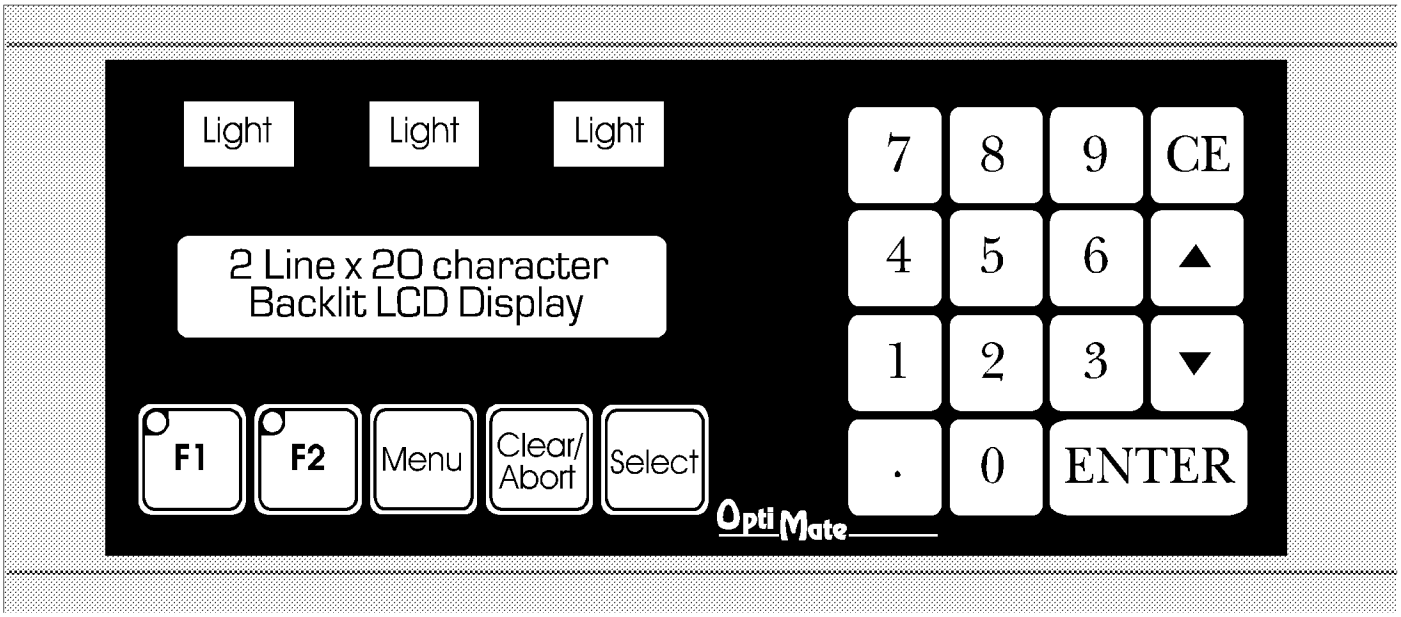


OM1510 Operator Terminal

The OM1510 Operator Terminal is a low cost/high performance man/machine interface with a broad range of operator input and display capabilities. The terminal includes a 2 line by 20 character LCD display, a full numeric keypad, two function keys, menu-tree selection keys and three LED light bar indicator lights.

OptiMate 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. Messages can also be configured into a layered menu tree. For computer applications, other messages can be sent directly over the serial communication port.

Indicator lights and 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 OM1510 Operator Terminal is part of Optimation's OptiMate® series. Each OptiMate module is designed to connect to a microprocessor or most PLC's with a single cable connection. OptiMate modules can be used individually, or together with any combination of other OptiMate modules.

When used with a microprocessor system, simple communications over either an RS232 or RS422 communications cable allow the microprocessor to directly control the lights and 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 OM1510 takes care of the rest.

Applications

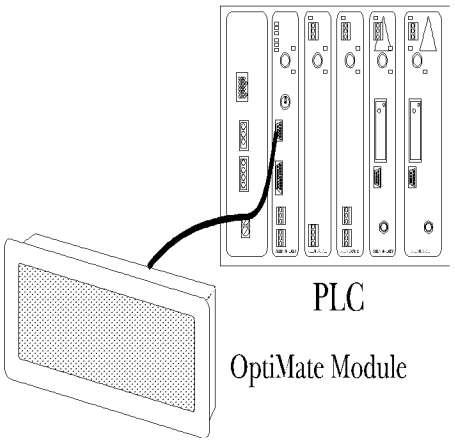
- Machine control
- Process control
- Security systems
- HVAC
- Plant monitoring/control
- PLC applications
- Microprocessor applications

Features

- 2 line x 20 character LCD
- Numeric keypad
- 2 User-defined function keys
- Menu-tree function selection
- 3 LED light bars
- PLC compatible
- RS232/RS422 communications
- Stand alone operation capable
- Multimodule operation capable

Contents

Configuration Options

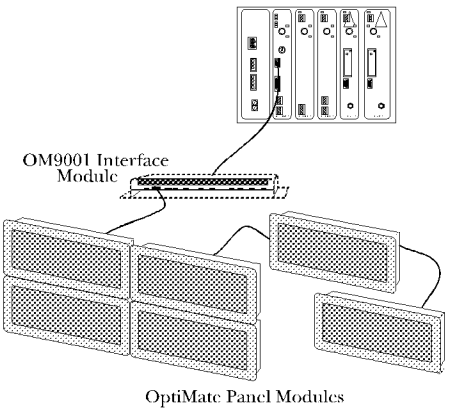


PLC Stand Alone

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

The OM1510 operator terminal uses a bank of PLC registers. Complete Operator interface is performed with 8 PLC registers for display message selection, data entry, function key interface and indicator light control. The OM1510 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 modules will perform these communications for most major PLC protocols. Configuration for particular PLC protocols and interconnect cabling is covered in the following pages.



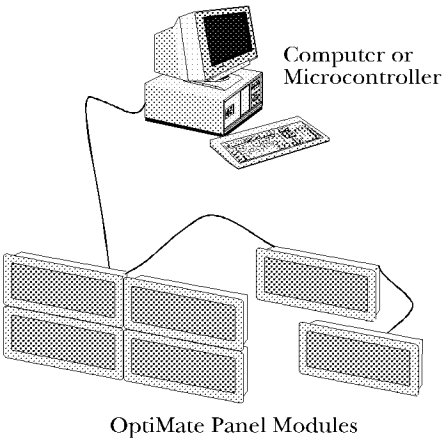
PLC Multi Modules

Larger systems involving operator panels can be successfully addressed using OptiMate modules. These applications utilize the OM9001 Communications Master to transfer data between the PLC and the individual OptiMate modules. OptiMate modules can be located together to form custom panels or they can be distributed anywhere within 4000 feet.

The OM9001 Communications Master provides a transparent interface between the PLC and a group of OptiMate modules. The communication interface between modules requires only four wires.

System configuration is simple via an interactive configuration program that runs on any IBM PC compatible computer.

This modular approach to custom applications provides a nearly limitless number of possibilities.



Microprocessor Based Systems

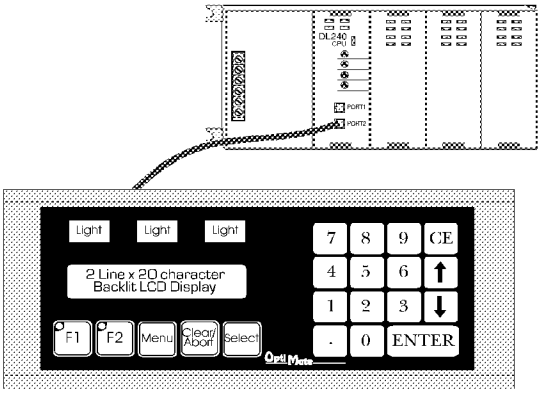
OptiMate modules can interface directly to most computers or microcontrollers. The modules communicate over either RS422 or RS232 serial communications. All that is required to interface OptiMate modules is a serial port. The OptiMate Hex protocol, detailed in this document, allows the user to directly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

Since each module 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 OM1510, these commands are messages for display and lamp control as well as function button status and data entry retrieval.

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

Use with a PLC



Memory Mapping

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

The term PLC register is used by Optimation for 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 OM1510 Terminal Module uses a bank of 8 contiguous PLC registers. The register set definition is shown in the table below.

OM1510 Terminal Module PLC Register Map

Holding Register	Register Function
X (first register of bank)	Top line message selection
X+1	Bottom Line message selection
X+2	Top line data/Function selection
X+3	Decimal point, Top line
X+4	Bottom line data
X+5	Decimal point, Bottom line
X+6	Status register
X+7	Control register

Register Definition

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

- Register X - 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 X+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 X+2 - Numeric Data associated with top line of display (described in more detail in following paragraphs).
- Register X+3 - For variable point data only. Numeric value of 0-5 for the number of digits after the decimal point for numeric data in top line of display.
- Register X+4 - Numeric Data associated with bottom line of display.
- Register X+5 - For variable point data only. Numeric value of 0-5 for the number of digits after the decimal point for numeric data in bottom line of display.
- Register X+6 - Status register

MSB															LSB
				↓	↑	ENT	MA	AB	FS	DA	SEL			F2	F1

Status Register

> F1-F5 are 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.

> ENT, Up arrow, Down arrow - are status of the ENTER and arrow buttons. Set to 1 when the button is active.

- Register X+7 - Control bits

MSB															LSB
						BD	MR	ME	DAK	L3F	L2F	L1F	L3	L2	L1

Control Register

> L1-L3 are on off control bits for each of the three indicator lamps. Set to 1 to turn the lamp on.

> L1F-L3F are flash control for each of the three indicator lamps. To flash a lamp, set the lamp on and lamp flash bits both to 1.

> 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 1500 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 OptiMate Configuration Editor, up to 160 predefined messages can be entered and stored in the OM1510. 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, let's 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 X and 22 in register X+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 incorporated 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 X+1 and '465' in register X+4. The bottom line of the display would then read "# of widgets sold: 465".

Displaying Data with a Decimal Point

The OM1510 terminal allows you to deal with two types of decimal point numerical displays. These two types are referred to as 1) pseudo-decimal, and 2) variable point decimal.

Pseudo-decimal 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 pseudo-decimal 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.

Variable point numbers include an integer value and a second integer that defines decimal point placement. These numbers are a little more complicated to handle within a PLC program, but may be necessary for certain applications. The OM1510 handles this type of number as an integer value in one register, and a second register containing a value, limited from 0 to 5, indicating the number of digits to the right of the decimal point.

Displaying Pseudo-Decimal Numbers

Pseudo-decimal 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 X and the value 735 in register X+2, the display would read "Temperature : 73.5" on the top line.

Displaying Variable Point Numbers

The OptiMate Configuration Editor allows a message to be marked to display a variable point number. When this is done, the OM1510 will use the value in register X+3 (top line) or X+5 (bottom line) for decimal point placement.

> Note : A decimal point should not be placed in the configured numeric string for a variable point number.

A typical variable point application might be the display of "Flow factor : ^^^^^" as message 12 selected for variable point display. If '12' is placed in register X+1, 4567 in register X+4 and 3 in register X+5, the bottom line will read "Flow factor : 4.567"

Numeric Data Entry

Numeric data can be entered through either the numeric keypad or the adjustment arrow keys. To do so the message must be marked for data entry in the configuration editor. Additionally, arrow adjustment is allowed only if selected in 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 decimal number types.

Data Entry from the Numeric Keypad

A message can be set up for data entry through the Configuration Editor. To do so, simply define the message with the numeric field. For keypad numeric entry, select data entry. If the number is to be variable point, select the variable point option.

When a data entry screen is required in the program, place the number of the preconfigured message in the message selection register. The digits marked by carets in the message definition will initially display as 0. As numbers are entered by the operator, they will appear in the rightmost caret position. Each new digit will be placed in the rightmost position and all other digits will be shifted over for each new digit entered. When the ENTER key is pressed, the number entered will be written to the proper data register in the PLC (X+2 or X+4). If the number is variable point, the decimal placement will be placed in the decimal point register (X+3 or X+5). The "data available" status bit (DA) will be set to inform the PLC program that data has been entered.

The OM1510 will automatically handle limitations on data entry. The following limiting actions will occur.

- The numeric entry will be limited to the maximum value that can be held in a 16 bit register (i.e. 65535, or 9999, if BCD)
- The OM1510 will not accept more digits than defined in the message screen.

Example of Keypad Numeric Data Entry

Suppose that you are designing an automatic banana peeler. With the configuration editor you define message 13 as "# of bananas : ^^^". With the configuration editor, you also select keypad data entry and integer.

When it is time to use this message in the display, your PLC program would put '13' in register X (for top line display). The operator would enter a number and press ENTER. When the operator presses ENTER, the value will be placed in PLC register X+2 and the data available status bit (DA) will be set. **It will remain set until a new message number is placed in register X, or the DAK bit is set.**

Data Entry/Adjustment with the Arrow Keys

There are certain times when 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 operation, it is commonly referred to as a "jog" operation.

A message can be set up for arrow adjustment through the Configuration Editor. To do so, define the message with a numeric field. Select arrow adjustment, select and enter minimum and maximum values.

> Note : Arrow adjustment is not available for variable point numbers.

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 (X or X+1) or the DAK bit is set.**

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 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 X (or X+1) and the current peel rate value in X+2 (or X+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 X+2 (or X+4). When the adjustment is complete, the operator would press ENTER. The OM1510 will then set DA. DA will remain set until a new message is selected via X (or X+1) or the DAK bit is set.

Menu Tree Operation

The OM1510 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 OM1510 allows up to four levels of menu. Each menu selection can be either a function or the next lower menu level. 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.

Sample Menu Tree

The example shown has three menu layers. The top level has four selections. Selection "Process Part Type 1" and "Process Part Type 2" each will place an associated function number in X+2 and set the FS flag when selected. "Manual Control" will bring up the next menu. Each item under "Manual Control" will place a function number to X+2 and set FS if selected. 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 until the PLC program clears the ME flag**. The display will return to function control of your program. When the function is complete, you may return to the same point in the menu by setting ME and MR. To require the user to start the menu selection process from the beginning, don't set MR.

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

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

Function Buttons

The OM1510 contains two 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 register holds the current state of both of the buttons. In a typical PLC application, these buttons would be mapped to control contacts for easy ladder logic interface.

Indicator Lights

Three general purpose indicator lights are located above the LCD display. These lights can be custom labeled and used for any purpose.

There are two control bits for each light in the control register. One bit controls whether the light is on or off. The other bit controls whether the light is flashing. The light must be turned on in order to flash.

In a typical PLC application, these bits would be mapped to control coils for easy ladder logic interface.

Configuration

Configuration of the OM1510 Terminal or system of OptiMate modules is performed via an IBM PC compatible computer. Optimation supplies configuration software that will allow you to select module configuration, system configuration and PLC protocol definition.

If the OM1510 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-module system, it must be configured for OptiMate Hex protocol.

> Note : When configuring, always remember to set the module address to address 31 (all switches on). Also, the ternination DIP switch(6) should be off for configuration.

Specific configuration of the OM1510 begins with defining the block of PLC register data 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 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 keypad data entry, select keypad data entry. Select variable point if the number is variable point (may not be used if a predefined decimal point is placed in the message).
- If the message has a field for numeric arrow (jog) adjustment, select arrow adjustment. Select and enter minimum and maximum values. Arrow adjustments are allowed only for integer and pseudo-decimal numbers.

Examples of Use with a PLC Direct PLC

Register Usage

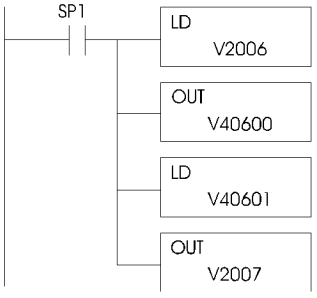
The OptiMate Configuration Editor allows you to configure a module to use a block of registers at a starting value that you define. For a PLC Direct 205 or 405 PLC the recommended memory to use is the general purpose data words starting at V2000 and V4000. For the 305 family, the recommended memory is the registers beginning at R400. Any block of registers within the data word range can be used.

The first six PLC registers in the block used by the OM1510 panel are used for numeric information. As such they are ideally suited for the general purpose data registers (V2000 and V4000 area for the 205/405 and R400 range for the 305). The last two registers use individual bits for control and status. These registers are better suited for the control relay register range of memory. The solution to this minor conflict is to define the base register address in general purpose data register memory and place a rung in your PLC program to copy the last two registers to/from control relay registers.

The following table lists the control relay register addresses for the various PLC Direct PLCs.

PLC Direct CPU	Control Relay Register Address range
DL230	V40600-V40617
DL240	V40600-V40617
DL330	R016-R037
DL330P	R016-R017 and R020-R027
DL340	R016-R037 and R100-R106
DL430	V40600-V40635
DL440	V40600-V40677

The examples on the following pages use an OM1510 connected to a PLC Direct 205/405 series PLC. The OM1510 is configured for a base address of V2000. **The following program rung should be placed in the program to copy the status register to V40600 and copy from V40601 to the control register**. With this rung placed into the PLC program, the status and control bits will be control relays. The register association is shown in the figure below.



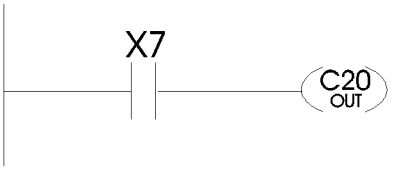
This will result in the following control relay association for the status and control registers.

Register	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	bit
	MSB															LSB	
V40600	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Status Register
V40601	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Control Register

Status Register			Control Register	
bit	relay		Bit	relay
F1	C0		L1	C20
F2	C1		L2	C21
SEL	C4		L3	C22
DA	C5		L1F	C23
FS	C6		L2F	C24
AB	C7		L3F	C25
MA	C10		DAK	C26
ENTER	C11		ME	C27
Up Arrow	C12		MR	C30
Down Arrow	C13		BD	C31

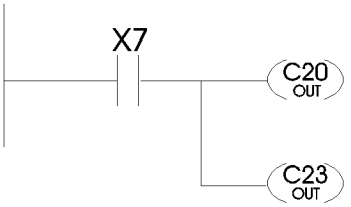
Lighting a Light

Lighting a light simply requires activating the control relay associated with the light. The following example will light the first light when input X7 is active. (Remember to place the register copy rung described previously in the program.)



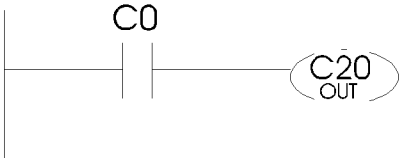
Flashing a Light

Flashing a light requires activating the control relay to turn the light on and the flash control relay. The following example will flash the first light when input X7 is active.



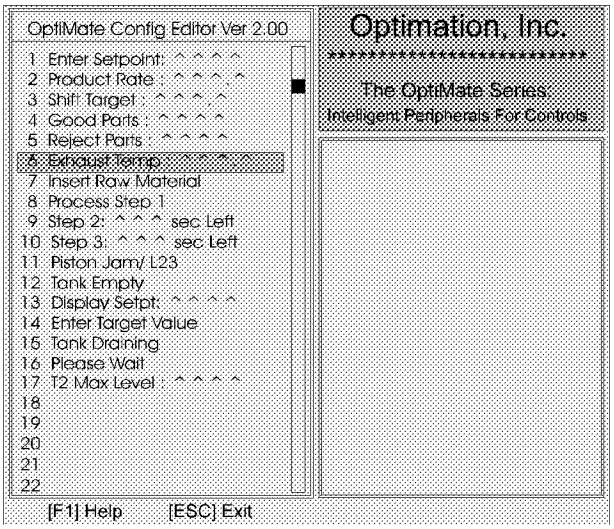
Using a Function Button

The five function buttons below the LCD display will appear as control relay coils in your program (assuming the register copy rung described previously is in your program). The following example lights light L1 when button F1 is active.

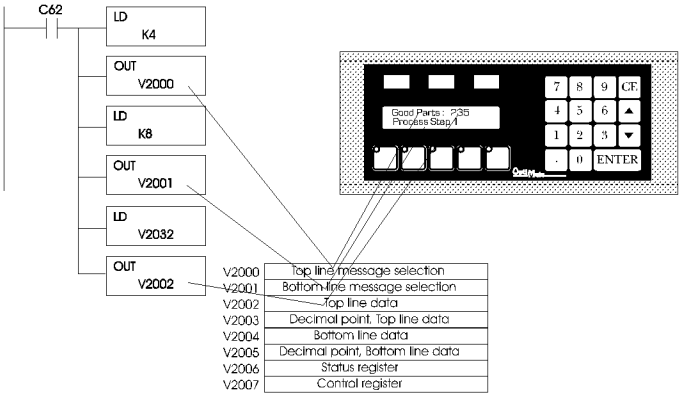


Displaying Messages on the LCD Display

Messages of various types can be configured via the OptiMate Configuration Editor and downloaded to the OM1510. The message definitions shown in the figure below will be used in all of the examples that follow.



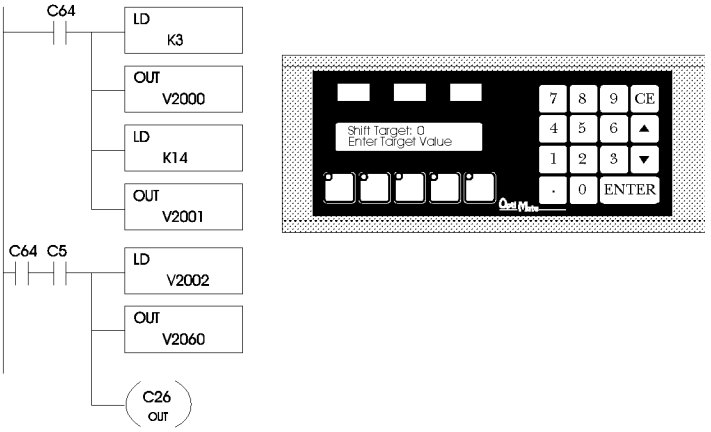
The following example shows a couple of messages being displayed to the LCD display. The top line uses data display message 4. The data for the data field is coming from V2032. The bottom line is text message 8.



Entering Data from the Keypad

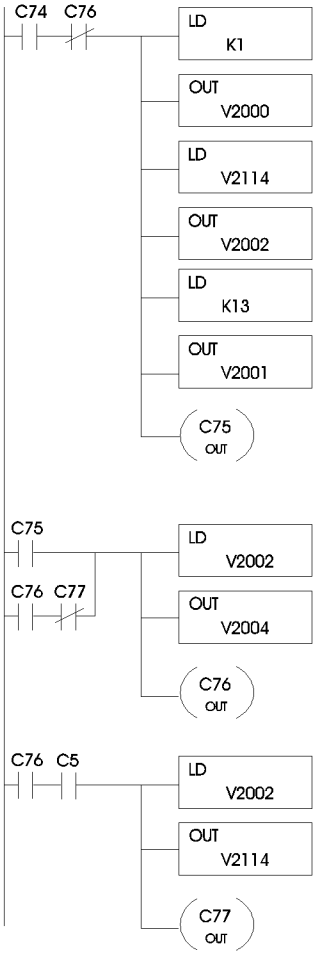
The example shown next illustrates keypad data entry. When control relay C64 is active, the top rung of the program will cause the messages shown to be projected on the display. The message shown on the top line is a keypad data entry message.

When the operator enters data on the keypad and presses "ENTER", the OM1510 sets the DA bit (energizes C5). The second program rung shows that when data is available, it will be transferred to V2060 and the data acknowledge bit set. When the OM1510 sees the data acknowledge bit set (C26), it will clear the data available



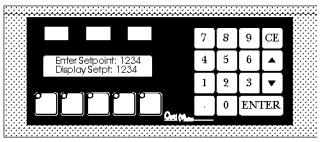
Arrow Adjustment of Setpoint Data

The figure below illustrates arrow adjustment of setpoint data (and a whole lot more). The process is enabled when C74 is energized. The first program rung places the "Enter Setpoint" and Display Setpt" messages in the top and bottom lines, intializes the setpoint value from the value in V2114 and enables the second rung.



The second rung continually copies the setpoint value to the display value so long as it is enabled. It latches itself until unlatched by the next rung, while disabling the first rung.

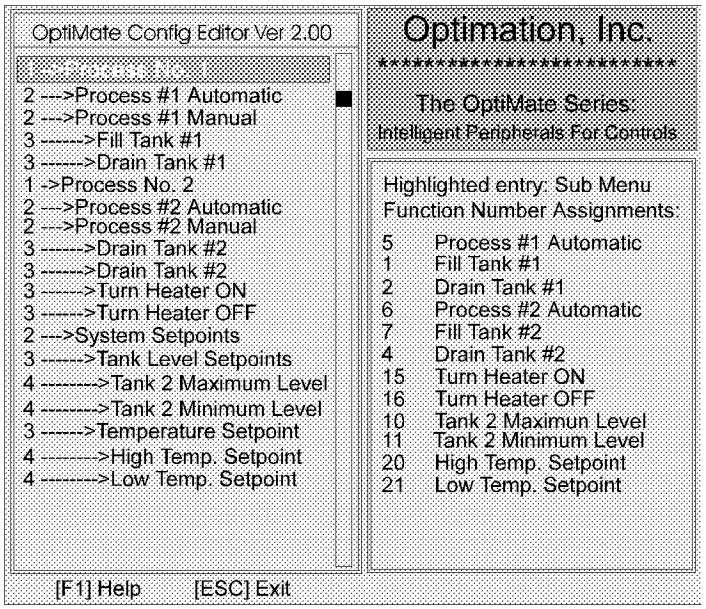
The third rung waits until the data available flag is set (C5), then copies the setpoint back to V2114. It also unlatches the second rung and, by activating C77, enables the next block of logic (whatever that might be) in the program.



Using a Menu Tree

The OM1510 allows you to predefine and use a layered menu tree for function selection. The operation of this menu tree is taken care of entirely by the OM1510 terminal. The only requirements 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 OptiMate Configuration Editor screen shown above displays the menu structure on the right hand side of the display and the function association on the left hand side. In other words, if the operator selects "Process #1 Automatic" from the menu tree, function number 5 would be placed in the function select register. If "Manual" is selected, no function number is selected; the terminal display will go to the next lower layer of the menu - "Fill Tank #1".

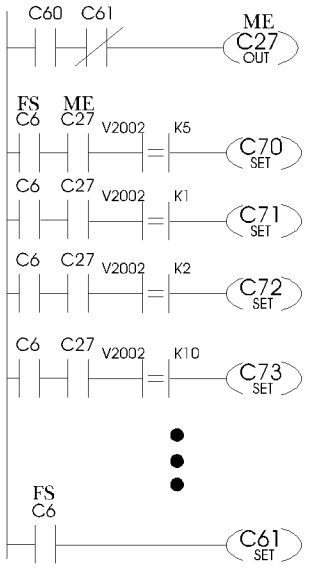
Any menu item that has lower level menu items below it will, when selected, branch into the next level. If the "MENU" button is pressed, the terminal will back the menu tree up to the next higher level (towards the trunk). Arrow keys will step the terminal through selection items on the same level on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

A Menu Tree Example

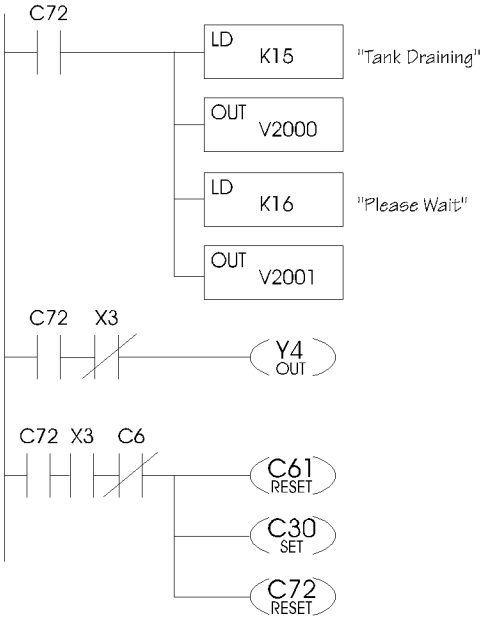
The program at the top of the next column illustrates menu tree function selection using the menu tree shown above. The first rung enables the menu tree when C60 is active.

> 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.

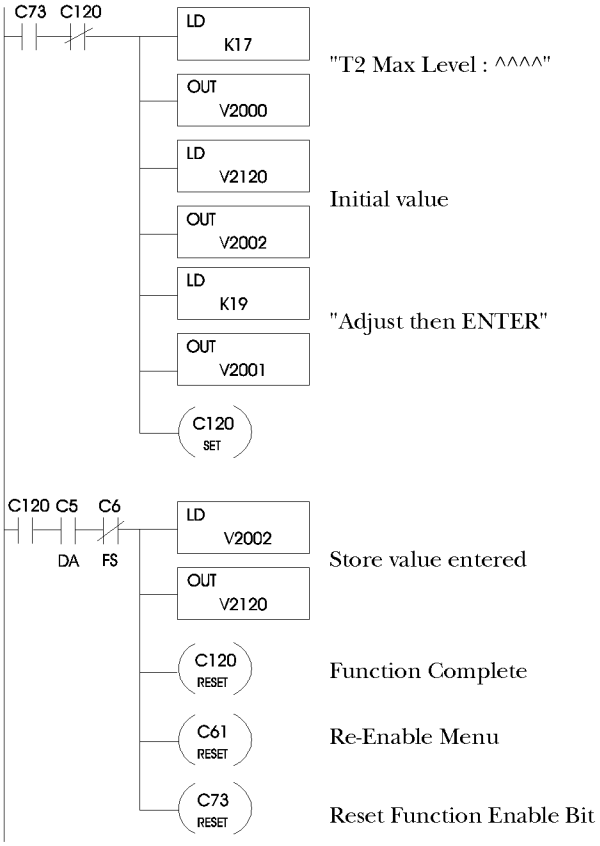
The second rung sets the appropriate function enable bit when a selection is selected from the OM1510. Function selection will activate the FS flag (which, based on our register copy shown earlier, will be C6). The value comparison to the value held in the function selection register will set the appropriate enable relay (C70, C71, etc.). The last rung logic shown will also disable the Menu Enable (ME) bit by setting C61. Note that the logic shown interlocks the function selects (setting C70, C71, etc.) with the function select and menu enable flags to ensure that only one function is enabled each time a function is selected.



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 above, this would result in control relay C72 being set. The first rung of the "Drain Tank #1" function, shown below, will put messages into the top and bottom lines of the display. The second rung energizes output Y4 to open the drain until level sensor input X3 senses that it is empty. The third rung re-enables the menu and sets it to return back to the "Drain Tank #1" selection when the tank is empty. The third rung also disables the function. By putting the function select relay (C6) in series, we force the program to wait until the function select relay has been cleared before re-enabling the menu.



The logic shown below is another typical example of how a setpoint function might be implemented. Suppose the selection for "Tank 2 Maximum Level" was made. Using the logic on the previous page, this will result in C73 being set. According to the first rung shown below, it selects the appropriate setpoint message for the top line and a prompt message for the bottom line. It also initializes the setpoint value for arrow adjustment. The first rung disables itself and enables the second rung. When the setpoint data is entered (after the FS flag is cleared), the second rung will copy the setpoint value back, clear C73 to disable the function, and clear C61 to re-enable the menu. If the menu return flag is also set, the OM1510 will return to the same point in the menu.



Examples of Use with an Allen Bradley PLC

Interfacing to A/B Memory

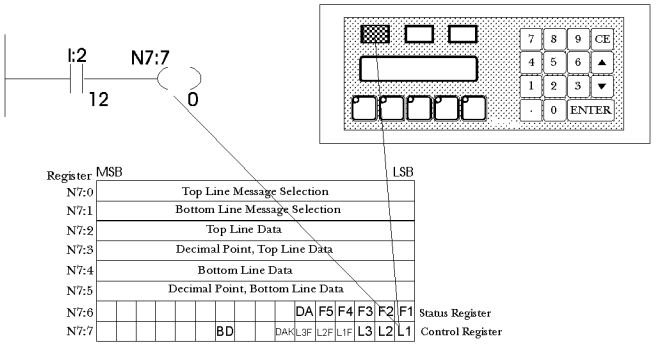
OptiMate modules interface to Allen Bradley SLC 5/03, SLC 5/04 and Micrologix PLCs via integer file type N. The 5/03 and 5/04 have file type N7 as standard. Other "N" type files can be created. The Micrologix has a fixed file type N7. Please refer to Allen Bradley documentaion for information on setting up and using "N" type files.

All of the examples that follow assume that the OM1510 module has been configured, through the OptiMate Configuration Editor, for a file number 7 and base register address 0. With this configuration, the status and control registers will be at N7:6 and N7:7 respectively. The following is a table relating status and control register bits to their N7 locations.

Status Register			Control Register	
bit	location		Bit	location
F1	N7:6/0		L1	N7:7/0
F2	N7:6/1		L2	N7:7/1
DA	N7:6/5		L3	N7:7/2
FS	N7:6/6		L1F	N7:7/3
AB	N7:6/7		L2F	N7:7/4
SEL	N7:6/8		L3F	N7:7/5
MA	N7:6/9		DAK	N7:7/6
ENTER	N7:6/10		ME	N7:7/7
Up Arrow	N7:6/11		MR	N7:7/8
Down arrow	N7:6/12		BD	N7:7/9

Lighting a Light

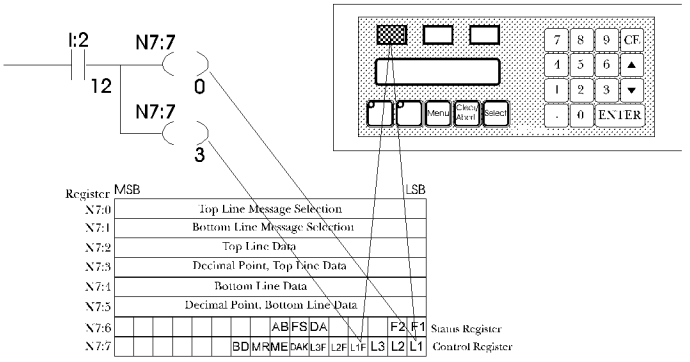
Lighting a light simply requires activating the register bit associated with the light. The easiest way to do this is to use the bit as a relay coil in the PLC program. The following example will light the first light when input I:2/12 is active.



Register Bit/Pushbutton Association

Flashing a Light

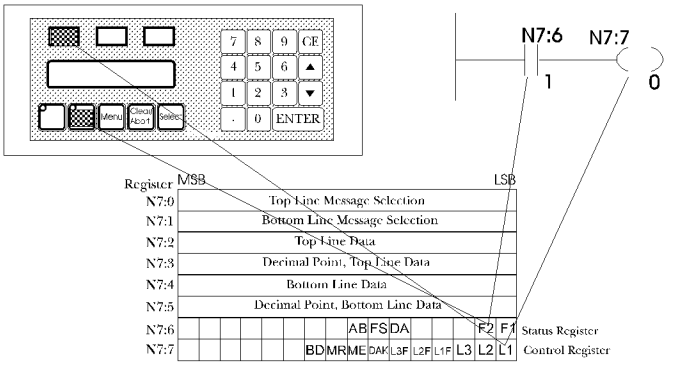
Flashing a light requires activating the register bit to turn the light on and set the flash control register bit. The following example will flash the first light when input I:2/12 is active.



Register Bit/Pushbutton Association

Using a Function Button

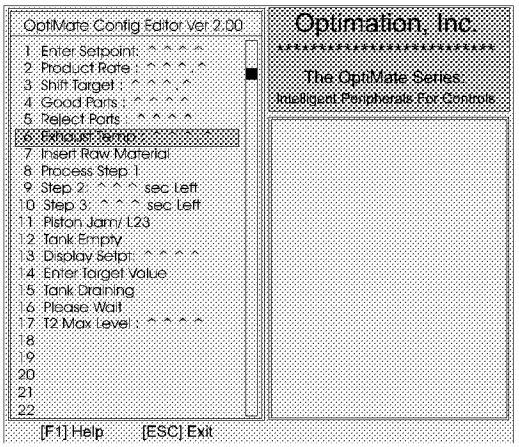
The five function buttons below the LCD display will appear as control relay coils in your program. The following example lights light L1 when button F2 is active.



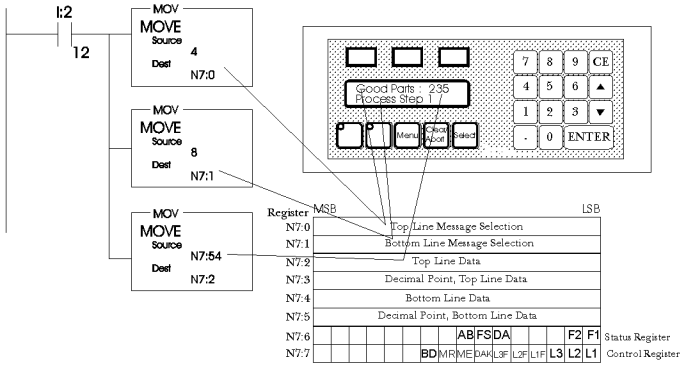
Register Bit/Pushbutton Association

Displaying Messages on the LCD Display

Messages of various types can be configured via the OptiMate Configuration Editor and downloaded to the OM1510. The message definitions shown in the figure below will be used in all of the examples that follow.



The following example shows a couple of messages being displayed to the LCD display. The top line uses data display message 4. The data for the data field is coming from N7:54. The bottom line is text message 8.

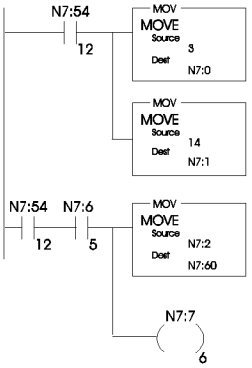


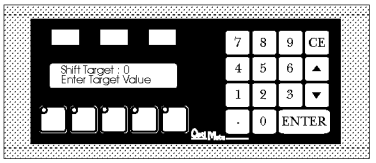
Register Bit/Pushbutton Association

Entering Data from the Keypad

The example shown next illustrates keypad data entry. When N7:54/12 is active, the top rung of the program will cause the messages shown to be projected on the display. The message shown on the top line is a keypad data entry message.

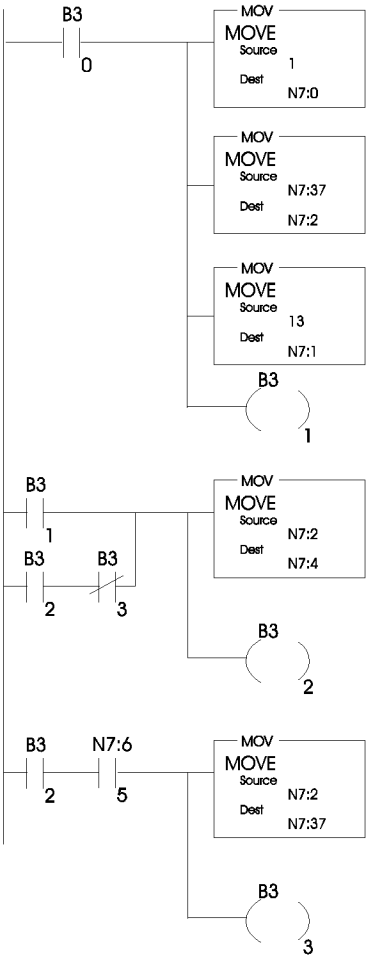
When the operator enters data on the keypad and presses "ENTER", the OM1510 sets the DA bit (energizes N7:6/5). The second program rung shows that when data is available, it will be transferred to N7:60 and the data acknowledge bit is set. When the OM1510 sees the data acknowledge bit set (N7:7/6), it will clear the data available.





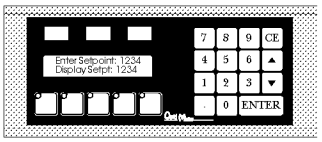
Arrow Adjustment of Setpoint Data

The figure below illustrates arrow adjustment of setpoint data (and a whole lot more). The process is enabled when B3/0 is energized. The first program rung places the "Enter Setpoint" and "Display Setpoint" messages in the top and bottom lines, intializes the setpoint value from the value in N7:37, and enables the second rung.



The second rung continually copies the setpoint value to the display value so long as it is enabled. It latches itself until unlatched by the next rung, while disabling the first rung.

The third rung waits until the data available flag is set (N7:6/5), then copies the

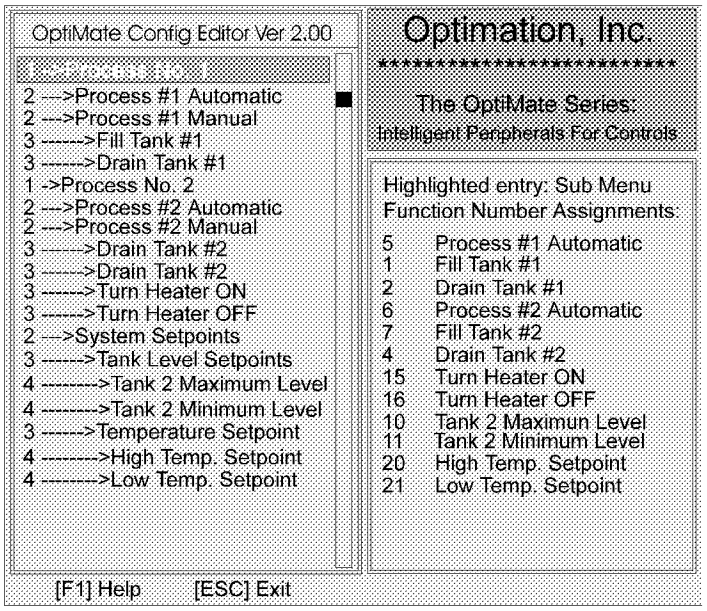


setpoint back to N7:37. It also unlatches the second rung and, by activating B3/3, enables the next block of logic (whatever that might be) in the program.

Using a Menu Tree

The OM1510 allows you to predefine and use a layered menu tree for function selection. The operation of this menu tree is taken care of entirely by the OM1510 terminal. The only requirements that the user has is to enable or disable the menu operation, and branch to the approriate function logic when a function selection is made.

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



The OptiMate Configuration Editor screen shown above displays the menu structure on the right hand side of the display and the function association on the left hand side. In other words, if the operator selects "Process #1 Automatic" from the menu tree, function number 5 would be placed in the function select register. If "Manual" is selected, no function number is selected; the terminal display will go to the next lower layer of the menu - "Fill Tank #1".

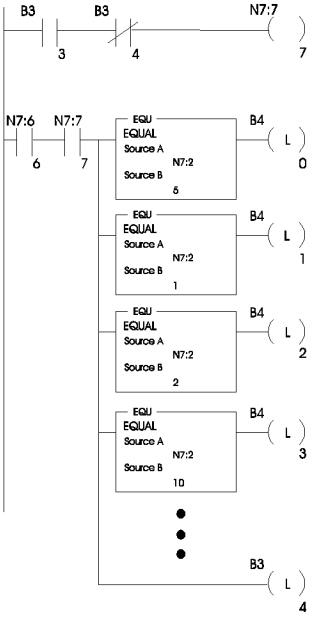
Any menu item that has lower level menu items below it will, when selected, branch into the next level. If the "MENU" button is pressed, the terminal will back the menu tree up to the next higher level (towards the trunk). Arrow keys will step the terminal through selection items on the same level on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

A Menu Tree Example

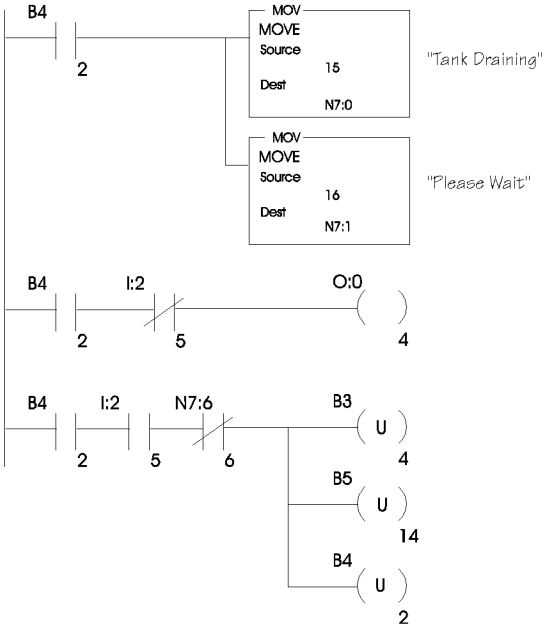
The program at the top of the next column illustrates menu tree function selection using the menu tree shown above. The first rung enables the menu tree when B3/3 is active.

> 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.

The second rung sets the appropriate function enable bit when a selection is selected from the OM1510. Function selection will activate the FS flag (N7:6/6). The value comparison to the value held in the function selection register will set the appropriate enable relay (B4/0, B4/1, etc.). The second rung shown will also disable the Menu Enable (ME) bit by setting B3/4.

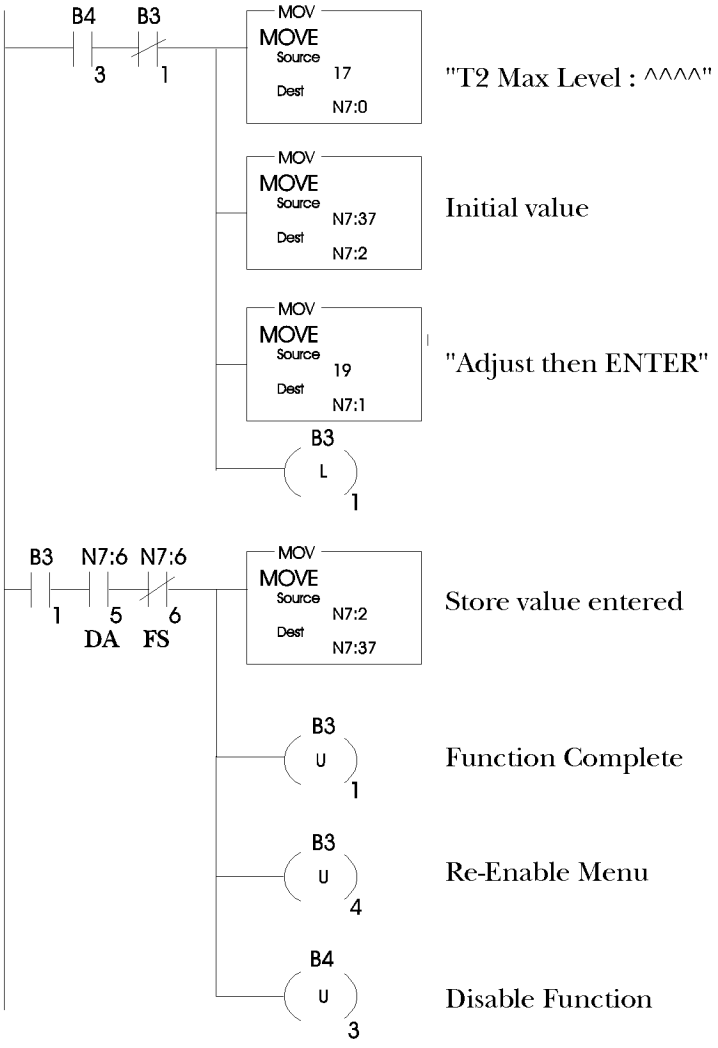


The program logic shown below illustrates how a typical function might be implemented. Suppose the selection was made by the operator to "Drain Tank #1". According to the logic shown above, this would result in control relay B4/2 being set. The first rung of the "Drain Tank #1" function, shown below, will put messages into the top and bottom lines of the display. The second rung energizes output O:0/4 to open the drain until level sensor input I:2/5 senses that it is empty. The third rung re-enables the menu and sets it to return back to the "Drain Tank #1" selection



when the tank is empty, and after the function select flag (N7:6/6) is cleared. The third rung also disables the function.

The logic shown below is another typical example of how a setpoint function might be implemented. Suppose the selection for "Tank 2 Maximum Level" was made. According to the logic on the previous page, this will result in B4:3 being set. The first rung shown below selects the appropriate setpoint message for the top line and a prompt message for the bottom line. It also initializes the setpoint value for arrow adjustment. The first rung disables itself and enables the second rung. When the setpoint data is entered (after the FS flag is cleared), the second rung will copy the setpoint value back, clear B3:1 to disable the function, and clear B3:4 to re-enable the menu. If the menu return flag is also set, the OM1510 will return to the same point in the menu.



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 module or read data from the module.

The OM1510 uses OptiMate Hex protocol for fast and easy communications. The OptiMate Hex protocol is defined in subsequent pages.

Module Address

A six-position DIP switch on the back of each OptiMate module provides a method for setting the address. By use of this DIP switch you can set the module address to any number between 0 and 31. (The sixth switch connects a terminating resistor and should be OFF in all cases except when the module is at the end of an RS422 cable).

Communications Protocols

To use an OptiMate module as a slave device in a microprocessor-based system, the module must be configured for OptiMate Hex protocol. The other options that must be set are 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, it must be downloaded to the module.

The OM1510 protocol for computer based operation is OptiMate Hex protocol.

Computer-Based Operation

All of the basic functionality described for PLC operation is also available to computer or microprocessor-based applications. In addition, computer-based systems can send messages directly for display on the LCD display.

The following is a synopsis of the OM1510 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 OM1510 can be instructed to display either predefined messages or a message string transmitted over the serial port. 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. This applies to both the predefined messages and messages sent directly from the computer host.

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. Keypad numeric data messages do not accept values.

Data Entry from the Numeric Keypad

A predefined message can be set up for numeric data entry through the Configuration Editor. A message can also be sent for data entry directly from the computer host. In either case, the numeric entry process on the OM1510 is the same.

When a number entry field is selected or transmitted for display, the digits marked with caret symbols will initially display as blanks. As numbers are entered by the operator, they will appear in the right-most caret position. Each new digit will be placed in the right-most position and all other digits will be shifted over for each new digit entry. When ENTER is pressed, the number entered will become available to be read by the host.

The OM1510 will automatically handle limitations on data entry. The following limiting actions will occur.

- The numeric entry will be limited to the maximum value that can be held in a 16 bit integer (i.e. 65535 or 9999, if BCD).
- The OM1510 will not accept more digits than defined in the message screen.

Example of Keypad Numeric Data Entry

Suppose that you are designing an automatic bread buttering machine. With the configuration editor you define message 17 as "# of slices : ^^^". With the configuration editor, you also select data entry and integer.

When it is time to use this message in the display, your program transmits a message selection for message 17. The operator would enter a number and press ENTER. When the operator presses ENTER, the data available status bit will be set. When the computer host reads the data, the bit will clear.

Data Entry/Adjustment with the Arrow Keys

There are certain times when 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 operation, it is commonly referred to as a "jog" operation.

A message can be set up for arrow adjustment either through the Configuration Editor or directly transmitted by the host.

Note : Arrow adjustment is not available for variable point numbers.

When the data screen is required in the program, either transmit a command for the pre-configured message and send its current value, or download both the message text and the 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 status bit will be set. When the computer host reads the data, the 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 carets in the message.

Function Buttons

The OM1510 contains two 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 both buttons along with other status.

Indicator Lights

Three general purpose indicator lights are located above the LCD display. These lights can be custom labeled and used for any purpose.

There are messages defined that allow the host computer to control the on/off state and flash condition of each light.

OptiMate OM1510 Hex Protocol

General Format

STX Module function ftn_data checksum
address

where Module address = 0 to 30
Function = 0xA0 ; General status/control
0xA1 ; Select predefined message display
0xA2 ; Send message text
0xA3 ; Send keyboard data entry message
0xA4 ; Send arrow key data adjustment message
0xA5 ; Read operator entered data
0xA6 ; Send data display message
0xA7 ; Send data for 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 no spaces between message fields

General status/control

STX Module 0xA0 control1 control2 checksum
address

where control1 = lamp control bits
bits 0, 1, 2 = lamp on off control for lamps 1 - 3 respectivley
bits 3, 4, 5 = lamp flash control for lamps 1 - 3 respectively
bit 6 = Menu enable
bit 7 = Menu return
control2 = misc control bits
bit 0 : Buzzer disable
bit 1 : Data acknowledge

Response

if message received and processed OK
STX term_stat display_stat ftn_select checksum

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

or
NAK **if any errors in message**

Select Predefined message

STX Module 0xA1 line mesg_no data checksum
address

where line = top (0x00) or bottom (0x01) line
mesg_no = number of the predefine (through configuration editor) message 0 - 160, hex iinteger, i.e. 33 = 0x21
data = 16 bit integer value plus DP placement (high byte, low byte, DP). 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**
or
NAK **if any errors in message**

Send message text

STX Module 0xA2 line text checksum
address

where line = top (0x00) or bottom (0x01) line
text = 20 character ASCII test message
example : STX 0x02 0xA2 0x00 Place Part in Collet 0x??(checksum)
Will display "Place Part in Collet" on the top line

Response

ACK **if message received and processed OK**
or
NAK **if any errors in message**

Send keyboard data entry message

STX Module 0xA3 line text checksum
address

where line = top (0x00) or bottom (0x01) line for variable point data entry
top (0x02) or bottom (0x03) line for integer or fixed point data entry
text = 20 character ASCII test message including caret characters for numeric data entry placeholders

example : STX 0x02 0xA3 0x01 Number of Parts: ^^^ 0x??(checksum)
Will display "Number of parts: " on the bottom line and allow a three digit number to display in the last 3 positions on the line

Response

ACK **if message received and processed OK**
or
NAK **if any errors in message**

Send arrow key data adjustment message

STX Module 0xA4 line text minimum maximum curr_val checksum
address

where line = top (0x00) or bottom (0x01) line
text = 20 character ASCII test message including caret characters for numeric data entry placeholders
minimum = minmum value allowed (16 bit integer, high byte, low byte)
maximum = maximum value allowed (16 bit integer, high byte, low byte)
curr_val = current value (16 bit integer, high byte, low byte)
example : STX 0x02 0xA4 0x01 Number of Parts: ^^^ 0x00 0x02 0x03 0x84 0x00 0x7B0x??(checksum)
Will display "Number of parts: 123" on the bottom line and allow a three digit number (between 1 and 900) to display in the last 3 positions on the line

Response

ACK if message received and processed OK
or
NAK if any errors in message

Read Operator Entered Data

STX Module 0xA5 checksum
address

Response

STX status data checksum if message received and processed OK

where status = 0x00 if ENTER key has not been pressed
= 0x01 if ENTER key has been pressed
data = operator entered data (16 bit integer + decimal placement, high byte, low byte, DP)
Notes : 1) This message can be used to read value of an arrow adjusted value while it is being adjusted.

or
NAK if any errors in message

Send data display message

STX Module 0xA6 line text curr_val checksum
address

where line = top (0x00) or bottom (0x01) line
text = 20 character ASCII test message including caret characters for numeric data entry placeholders
curr_val = current value (16 bit integer + decimal placement, high byte, low byte, DP)
example : STX 0x02 0xA6 0x01 -Shift Total- : ^^^^ 0x03 0x81 0x00 0x??(checksum)
Will display "-Shift Total- : 897" on thebottom line

Response

ACK if message received and processed OK
or
NAK if any errors in message

Send data for data display message

STX Module 0xA7 line curr_val checksum
address

where line = top (0x00) or bottom (0x01) line
curr_val = current value (16 bit integer, high byte, low byte)
example : STX 0x02 0xA6 0x00 0x03 0xE8 0x01 0x92
Will display "100.0" in the numeric display position of the message.
Note : If the message has a fixed decimal place, the transmitted decimal position will be ignored. In other words, if the same command had been sent to a line containing the predefined message "Tank 3 level : ^^.^^", the display would project "Tank 3 level : 10.00".

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

Broadcast message (sent to all modules)
Synchronize lamp flashing (between all system lamp modules)

STX Broadcast function checksum
address

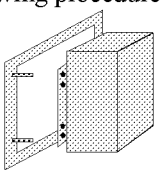
where broadcast address = 99
function = 0

Set Up and Interconnect

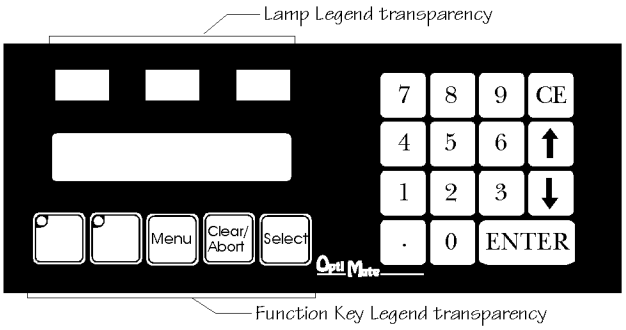
Legending the Lamps and Function Keys

Legending the OM1510 module is a relatively simple process that basically involves sliding legend transparencies into pockets in the panel overlay. Use the following procedure.

- Remove the bezel from the module. The bezel snaps to the module 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. A pattern is provided on the next to last sheet of this document.
  - > 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 or lettering machine to letter onto paper, then photocopy



- Cut along outline. Slide into overlay pocket. Lamp legends slide in from the top. Pushbutton legends slide in from the bottom.
- Re-attach bezel. Push bezel onto box until it snaps together.

Connection to the System

OptiMate modules are designed for communications connection to system devices. The module can be connected to a computer, PLC or communication master over the serial port (RS232 or RS422).

Connection to a Computer or PLC

Connection of an OptiMate module 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.

OptiMate Module RS232

Host Computer/PLC		OptiMate Module DB-15 Male
TX	———	3 RS232 RX
RX	———	2 RS232 TX
Sig Gnd	———	5 Sig Gnd

OptiMate Module RS422

Host Computer/PLC		OptiMate Module DB-15 Male
TX+	———	9 RS422 RX+
TX-	———	10 RS422 RX-
RX+	———	11 RS422 TX+
RX-	———	12 RS422 TX-

Refer to manufacturer's documentation for PLC or computer serial link connector pinouts.

Optimation sells interface cables for connection to several different PLCs as well as to IBM PCAT compatible ports.

Serial Connection to Communications Master

Connection to an Optimation Communications Master over a serial link is via RS422. The Communication master port connections are reversed from the module ports to enable direct pin to pin connection. For distances under 50 feet (in a low electrical noise environment), a ribbon cable connection works quite well. For longer distances or in noisy environments, a two pair shielded RS422 cable is recommended.

Configuration

Configuration Selections

OptiMate modules can be configured for the specific application by using the OptiMate Configuration Editor. The Configuration Editor runs on any IBM PC compatible computer. It allows the user to select the exact functionality to meet application requirements.

For the OM1510 module, the following are important configuration parameters.

Computer-Based Systems

Decision	Selection
Single/Multi Module	Choose Single module even if the system will contain several modules. The Multi module selection applies only to systems using a communications master. In computer-based systems, each module is configured independently. After configuration, multiple modules can be connected together to form a system.
Configuration starting point	First-time configuration, start with defaults for module. 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 your application
Menu	Use the template found in this manual to structure a working menu. Enter the menu as defined. Keep in mind the tree structure and the limit of 4 available menu layers.

Configuration must be downloaded from IBM PC compatible to each module. This is done over the serial link. Module address must be set to 31 prior to application of power for module to accept configuration data. The termination DIP switch should always be OFF during configuration. Communication cable is available from Optimation.
After configuring, module address should be changed to the address you will use for your application (something other than 31).

Single Module PLC-Based Systems

Decision	Selection
Single/Multi Module	Choose single module configuration
Configuration starting point	First-time configuration start with defaults for module. 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 your application
Menu	Use the template found in this manual to structure a working menu. Enter the menu as defined. Keep in mind the tree structure and the limit of 4 available menu layers.

Configuration must be downloaded from IBM PC compatible to the module. This is done over the serial link. Module address must be set to 31 prior to application of power for module to accept configuration data. The termination DIP switch should always be OFF during configuration. Communication cable is available from Optimation.
After configuring, module address should be changed to any address other than 31 to enable the module to communicate with the PLC.

Multi-Module PLC Applications (Uses Communications Master)

Decision	Selection
Single/Multi Module	Choose Multi module
PLC Type	This applies to the Communications master. Choose appropriate type
Protocol	This applies to the 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.
Module Protocol	Choose Optimate Hex
Buttons	Select momentary or alternate action as required for your application.
Messages	Define messages as required for your application.
Menu	Use the template found in this manual to structure a working menu. Enter the menu as defined. Keep in mind the tree structure and the limit of 4 available menu layers.

Configuration must be downloaded from IBM PC compatible to each module. This is done over the serial link. Module address must be set to 31 prior to application of power for module to accept configuration data. The termination DIP switch should always be OFF during configuration. Communication cable is available from Optimation.
After configuring, module address should be changed to the address you will use for your application (something other than 31).

Note: DIP switch 6 (Termination) applies only in RS422 communications. It should be on only on the module at the end of the cable. The switch should be OFF for all other cases, including RS232.

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 this page and using the template on the lower right to define all of your messages.

The next step is to use the configuration editor to enter the messages as defined. Remember to use carets wherever variable data is to be used. The configuration editor will guide you through other definable parameters, including data type, message type, etc.

Message #	Text (20Characters Max.)
1	Enter Setpoint: ^^^^
2	Product Rate: ^^^.^
3	Shift Target: ^^^.^

Example Message Definitions

Message #	Text (20Characters Max.)

Message Defintion Template

Defining a Menu

The figure on the right illustrates the process of defining a menu. The details of this process are as follows.

1. Define the message.
2. If the message is for function selection, define the message number.
3. Keep the "tree" structure. Proceed in level from general to more specific. To make sub- menus, increment the level number by one and place all items that pertain to the sub-menu at the same level.
4. Menu items that select lower level menus, rather than a function,do not have function numbers.
5. There is a limit of four layers.

Level #	Text (20Characters Max.)	Function# or Sub-Menu
1>—2>—3>—4>	Process No. 1	Sub_Menu
1>—2>—3>—4>	Process # 1 Automatic	Function # 5
1>—2>—3>—4>	Fill Tank # 1	Function # 1
1>—2>—3>—4>		
1>—2>—3>—4>		

Example Menu Definitions

Level #	Text (20Characters Max.)	Function# or Sub-Menu
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		
1>—2>—3>—4>		

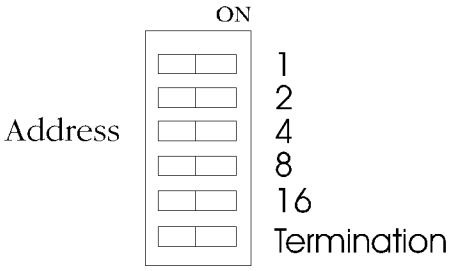
Menu Defintion Template

Addressing

Setting the module address is a matter of turning the module over and pressing the appropriate dip switches. There are 6 DIP switches; 5 of which have a numeric value listed next to the switch. To select an address, push (with a pencil or small screwdriver) the appropriate combination of switches down to the right.

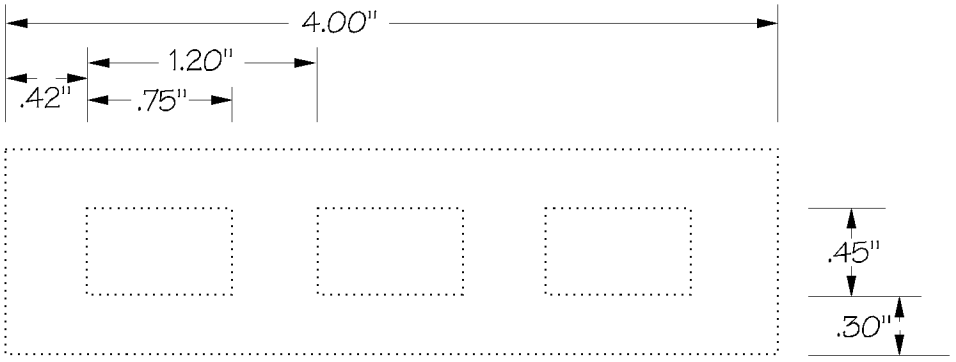
For example to select address 14, the 2, 4 and 8 switches should be pressed down to the right and the 1 and 16 switches down to the left.

Remember that for configuration, address 31 (all numeric switches 1 -5 in the "on" position to the right) must be selected first, then apply power to the module.

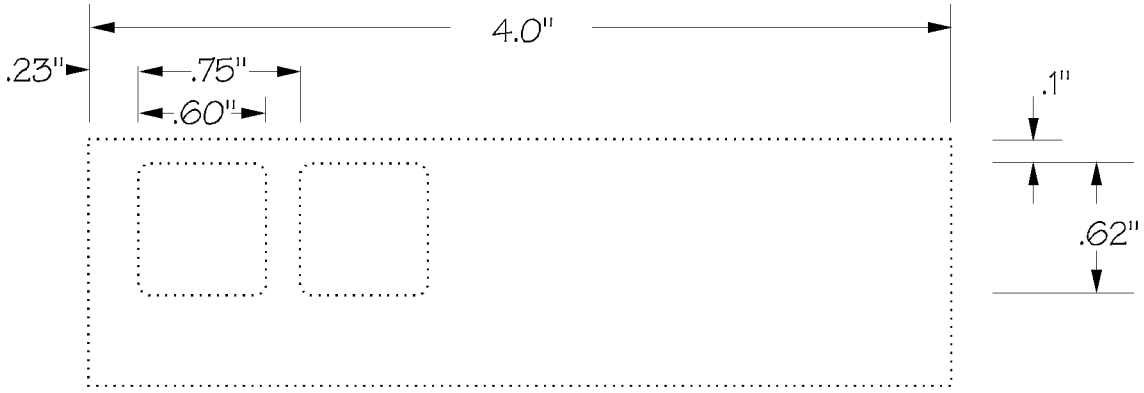


Power

OptiMate modules can operate on any voltage between 8 and 30 VDC. Power must be connected to the terminal plug located on the back of the module.



Lamp Label Strip Pattern

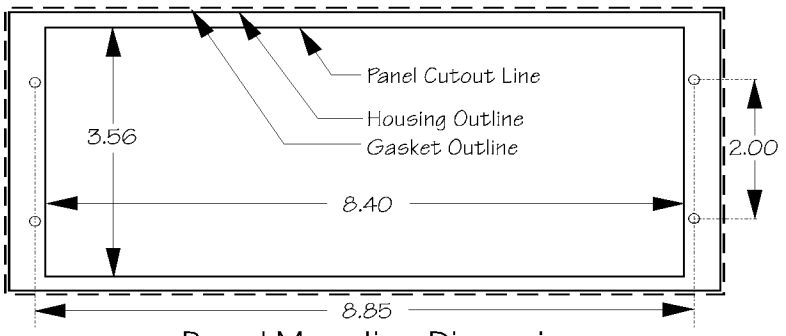


Pushbutton Label Strip Pattern

Specifications

Physical

- Recessed Mount Housing 9.5"L x 4.0"H x 1.75"D
- Cutout size for above 3.56"Hx8.46"L
- Panel Fasteners : Four, 6x32 threaded studs, shown above (on ends, symmetrical about center line)
- Weight : 19 ounces
- Colors : Dark gray housing with black panel. Numeric keys; blue with white letters. Other keypad keys; gray with white letters. Function keys; White with user supplied label.
- Lamp Colors : Red, Green, Yellow
- 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 (all lamps on) : 8 - 30VDC @ 5.7Watts
  420 mA @ 12VDC 240 mA @24VDC
- Power connector : Pluggable terminal block, 2 position

Communications

- RS232 and RS422
- 4800 to 19200 baud
- Compatible with most major PLC protocols
- OptiMate Hex Protocol for computer-based systems
- 15 pin female 'D' shell connector

Environmental

- Enclosure - NEMA 4 (when properly installed) recessed mount
- 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)
- Keypad data entry (Integer, fixed point or variable point)
- Arrow adjustment data message (Integer or fixed point)

Menu Tree

- Hierarchical (up to four levels)
- User-defined through configuration editor
- Stored in module's non-volatile memory
- User-defined association with function number
- Program control of enable/disable
- Each menu item takes one of the 160 available message

Menu Tree Operation

- Activation upon Menu Enable from host computer and Operator selection of "MENU"
- Operator control of scrolling and branch selection
- "MENU" selection will back up tree 1 level on each selection (to 1st level)
- Function selection provides function number and function select flag
- Timeout after 90 seconds of inactivity (goes back to displays selected through PLC or comm link)

Numeric Types & Values

- Integer
- Fixed Point (Pseudo-Decimal)
- Variable Point
- BCD (Values between 0 & 9999; with appropriate decimal placement)
- Binary (Values between 0 & 65535 with appropriate decimal placement)