

OI100 SERIES

OPERATOR INTERFACE



USER MANUAL



P.O.Box 24
STANFIELD 3613
SOUTH AFRICA

Tel: +27 (031) 7028033
Fax: +27 (031) 7028041
Email: proconel@proconel.com
Web: www.proconel.com

TABLE OF CONTENTS

1.	AN OVERVIEW OF THE OI100 OPERATOR INTERFACE	3
1.1	DESCRIPTION	3
2.	HARDWARE	4
2.1	SPECIFICATIONS	4
2.2	GROUNDING/SHIELDING.....	4
2.3	WIRING TERMINALS	5
2.3.1	Programming Port (OI100 and OI101).....	5
2.3.2	Communications Port (OI100 and OI101).....	5
2.3.3	4 Pin Connector (OI101)	5
2.4	SETTING THE INTERNAL JUMPERS.....	6
2.5	OI100 CPU.....	7
2.5.1	Program Memory.....	7
2.5.2	Data Memory.....	7
2.5.3	Data Memory Map.....	8
2.5.4	Digital Input Map.....	9
2.5.5	Digital Output Map.....	9
2.5.6	Timer Map.....	10
2.5.7	Counter Map.....	10
2.5.8	Control Relay Map.....	10
2.5.9	System Relay Map.....	11
3.	RS232/RS485 Modbus Communications Port	12
3.1	Modbus Master.....	12
3.2	Modbus Slave.....	12
4.	Programming the OI100	13
4.1	Programming the Display parameters.....	13
4.1.1	Step 1: Display page number	13
4.1.2	Step 2: Text Line 1	13
4.1.3	Step 3: Text Line 2	13
4.1.4	Step 4: Type.....	14
4.1.5	Step 5: Memory Address	14
4.1.6	Step 6: Slave ID.....	14
4.1.7	Step 7: Slave Address	14
4.1.8	Step 8: Format	14
4.1.9	Step 9: Decimal Point	14
4.1.10	Step 10: Submit	14
4.2	Programming the Communications parameters.....	15
4.2.1	Step 1: Modbus Master.....	15
4.2.2	Step 2: Poll Block number.....	15
4.2.3	Step 3: Remote ID	15
4.2.4	Step 4: Function.....	15
4.2.5	Step 5: Local Address	16
4.2.6	Step 6: Range.....	16
4.2.7	Step 7: Remote Address.....	16
4.2.8	Step 8: Submit	16
4.3	Programming the Ladder Logic to use the buttons.....	16
4.4	Editing the data on the display.....	17
4.4.1	Step 1: Entering Edit Mode.....	17
4.4.2	Step 2: Changing the current digit value.....	17
4.4.3	Step 3: Advancing to the next digit.....	17
4.4.4	Step 4: Exit Edit Mode.....	17
5.	Modbus Memory Map (MODULE TYPE = 44)	18
6.	Ladder Logic Function Blocks	20

1. AN OVERVIEW OF THE OI100 OPERATOR INTERFACE

1.1 DESCRIPTION

The OI100 OPERATOR INTERFACE has been developed as a versatile LCD display for displaying machine parameters and messages. The fact that the controller is programmable enables the user to program their own unique text messages and not be restricted by a pre-programmed unit.

The OI100 has a built in PLC which is programmed in ladder logic. PROCON's ProSoft windows-based PC software is used to generate the ladder diagram, compile the program, and then download the program to the OI100 via the programming port on the side of the unit.

The programming port supports the Modbus RTU protocol and all of the internal registers and I/O status can be accessed through this port.

The OI100 also has a second communications port which can be configured internally for RS232 or RS485 communications. This port can be configured as a Modbus master or Modbus slave and can read parameters to be displayed from a OI100 Series PLC or one of the many MOD-MUX I/O modules.

There are two formats of display available:

The OI100 Operator Interface is a hand held unit and has been developed as a tool for technicians to adjust parameters at the time of commissioning.

The OI101 is a panel mount display in the industry standard DIN 90mmX90mm format. This unit is designed as a machine operator interface to provide process information to the operator as well as the facility to adjust set-points, alarms and other process variables.

2. HARDWARE

2.1 SPECIFICATIONS

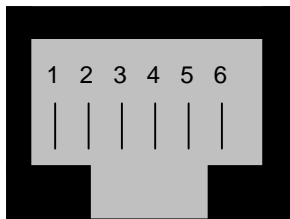
POWER REQUIREMENT:	OI100 – Powered from PL4XX PLC. OI101 – Powered from PL4XX PLC or by 12-24VDC.
DISPLAY:	2 Line X 16 Character LCD Display.
COMMUNICATION PORTS:	Prog. Port – RS232 Programming Port. Comms Port – RS232/RS485 Modbus Master/Slave Port.
BAUD RATE:	Prog. Port - 9600, 19200 (Default). Comms Port - 2400, 4800, 9600, 19200, 38400.
COMMS PARAMETERS:	Prog. Port - Parity- None. Stop Bits- 1. Comms Port - Parity- None, Even, Odd. Stop Bits- 1 or 2.
CONNECTORS:	OI100 – 2 x 6 PIN RJ12. OI101 – 2 x 6 PIN RJ12 and 1 x 4 PIN Connector/Plug with screw terminals.
DIMENSIONS:	OI100 - 63mm (HIGH) X 112mm (WIDE) X 30mm (DEEP) OI101 - 96mm (HIGH) X 96mm (WIDE) X 80mm (DEEP)
OPERATING TEMPERATURE:	-0°C to +60°C
STORAGE TEMPERATURE:	-0°C to +65°C
HUMIDITY:	up to 95% non condensing

2.2 GROUNDING/SHIELDING

In most cases, the OI100 will be installed in an enclosure along with other devices, which generate electromagnetic radiation. Examples of these devices are relays and contactors, transformers, motor controllers etc. This electromagnetic radiation can induce electrical noise into both power and signal lines, as well as direct radiation into the module causing negative effects on the system. Appropriate grounding, shielding and other protective steps should be taken at the installation stage to prevent these effects. These protective steps include control cabinet grounding, module grounding, cable shield grounding, protective elements for electromagnetic switching devices, correct wiring as well as consideration of cable types and their cross sections.

2.3 WIRING TERMINALS

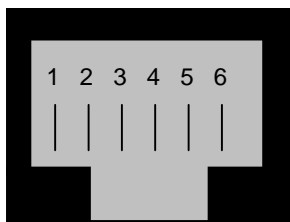
2.3.1 Programming Port (OI100 and OI101)



6-pin Female Connector

Programming Port Connections		
1	No Connection	
2	No Connection	
3	TXD	Transmit Data (RS232)
4	RXD	Receive Data (RS232)
5	0V	Power (-) GND
6	0V	Power (-) GND

2.3.2 Communications Port (OI100 and OI101)

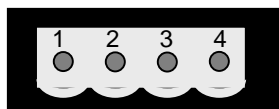


6-pin Female Connector

Communications Port Connections		
1	+5V	Power (+)
2	+5V	Power (+)
3	TXD	Transmit(RS232) / + (RS485)
4	RXD	Receive (RS232) / - (RS485)
5	0V	Power (-) GND
6	0V	Power (-) GND

When the OI100 is connected to a PL4XX series PLC using the supplied cross-over cable, the OI100 obtains its power from the PLC and no external power source is required.

2.3.3 4 Pin Connector (OI101)



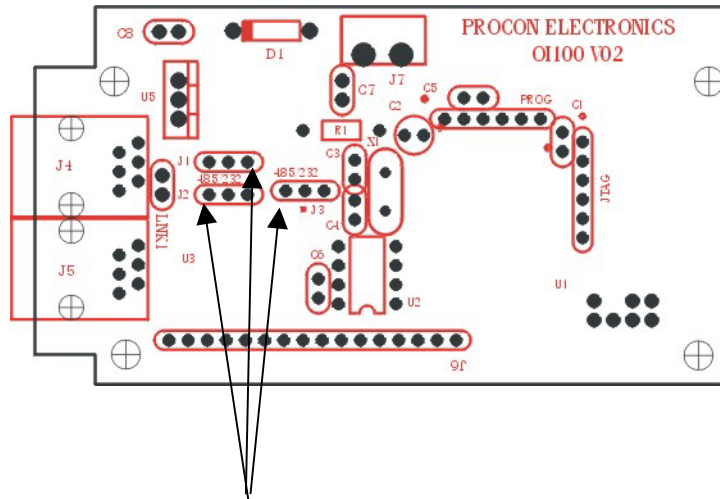
OI101 4 pin Connector		
1	0V	Power (-) GND
2	12/24V	Power (+)
3	+ (RS485) or TXD (RS232)	
4	- (RS485) or RXD (RS232)	

The 4 Pin Connector is used when the OI101 is connected to a device other than a PL4XX PLC.

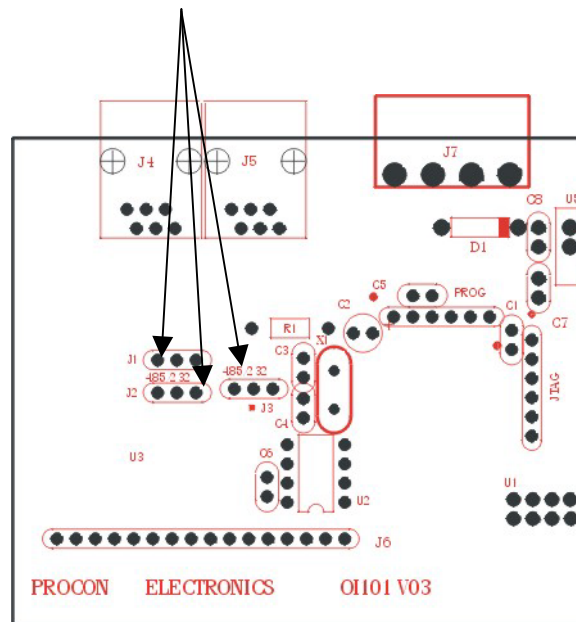
Note: Do not connect power to the 4 Pin Connector if the OI101 is connected to a PL4XX PLC using the cross-over cable.

2.4 SETTING THE INTERNAL JUMPERS.

The internal jumpers are used to select between RS232 and RS485 on the Communications port. The default factory setting is RS232. There are 3 jumpers which must all be on either RS232 or RS485.



Position of jumpers on the circuit boards.



2.5 OI100 CPU.

The CPU (central Processing Unit) performs all of the tasks that are required to make the PLC function and run your ladder program. Some of the tasks include:

1. Reading the status of the push buttons.
2. Executing the program.
3. Updating the display.
4. Doing diagnostics.
5. Servicing the communications ports.
6. Running the timers.

2.5.1 Program Memory.

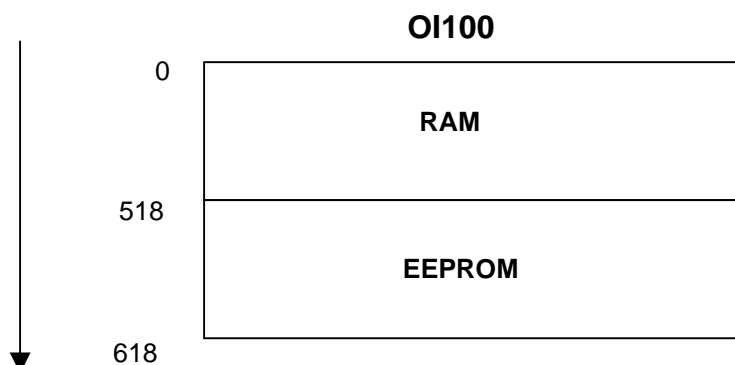
The programming port is used to program the PLC. The program which is sent from the PC using the ProSoft ladder editor, is stored in FLASH memory. This memory does not get lost when the power fails and so will remain permanently in the PLC until it is reprogrammed.

2.5.2 Data Memory.

All the variables used in the program are stored in Data memory. Both the Digital and Analog values are stored in this memory along with the timers, counters, and user memory.

The memory is divided up into 2 sections.

1. RAM – Random Access Memory. This memory is the most widely used memory and is where most of the data is stored. All timers, counters, I/O statuses and system information use this memory. If the power fails then all the information in this memory is lost and is re-initialized to zero when the PLC starts again.
2. EEPROM – This memory is used to store parameters such as set-points and configuration data as it retains its memory when the power is turned off. The one point to remember is that this memory can only be written to 10 000 times before it wears out so you must not write to this memory all the time as you can with RAM.



2.5.3 Data Memory Map.

All of the variables used in the PLC are stored in data memory. In order for your program to get access to these variables you need to know the memory address. The memory address starts at zero and the size depends on the PLC being used. Each memory location consists of 16 bits. Thus one memory location can be used to store the status of 16 digital I/O points or an analog value from 0 to 65535. Some of the ladder functions use two consecutive memory locations to store larger values. Refer to the ProSoft user manual to find out about the ladder functions.

OI100 MEMORY MAP			
Memory Type	Digital Reference	Memory Address	Quantity
Module Type = 44	-	M0	1
Digital Inputs	I1 to I2	M1	2
Digital Outputs	O1 to O2	M2	2
Timer Status	T1 to T64	M3 – M6	64
Counter Status	C1 to C64	M7 – M10	64
Control Relays	R1 to R64	M11 – M14	64
System Relays	S1 to S32	M15 – M16	32
Timer Memory	-	M19 – M82	64
Counter Memory	-	M83 – M146	64
User RAM Memory	-	M147 – M518	371
User EEPROM	-	M519 – M618	100

2.5.4 Digital Input Map.

MSB			DIGITAL INPUTS												LSB		ADDRESS	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	I2	I1	M1

Bit Number	Digital Input Number	Description
0	I1	Push Button 1
1	I2	Push Button 2
2	-	-
3	-	-
4	-	-
5	-	-
6	-	-
7	-	-
8	-	-
9	-	-
10	-	-
11	-	-
12	-	-
13	-	-
14	-	-
15	-	-

2.5.5 Digital Output Map.

MSB			DIGITAL OUTPUTS												LSB		ADDRESS
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	O2	O1	M2

Bit Number	Digital Input Number	Description
0	O1	Scroll Up
1	O2	Scroll Down
2	-	-
3	-	-
4	-	-
5	-	-
6	-	-
7	-	-
8	-	-
9	-	-
10	-	-
11	-	-
12	-	-
13	-	-
14	-	-
15	-	-

2.5.6 Timer Map.

TIMER STATUS															ADDRESS	
MSB							LSB									
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
T16	T15	T14	T13	T12	T11	T10	T9	T8	T7	T6	T5	T4	T3	T2	T1	M3
T32	T31	T30	T29	T28	T27	T26	T25	T24	T23	T22	T21	T20	T19	T18	T17	M4
T48	T47	T46	T45	T44	T43	T42	T41	T40	T39	T38	T37	T36	T35	T34	T33	M5
T64	T63	T62	T61	T60	T59	T58	T57	T56	T55	T54	T53	T52	T51	T50	T49	M6

2.5.7 Counter Map.

COUNTER STATUS																ADDRESS
MSB								LSB								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
C16	C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	M7
C32	C31	C30	C29	C28	C27	C26	C25	C24	C23	C22	C21	C20	C19	C18	C17	M8
C48	C47	C46	C45	C44	C43	C42	C41	C40	C39	C38	C37	C36	C35	C34	C33	M9
C64	C63	C62	C61	C60	C59	C58	C57	C56	C55	C54	C53	C52	C51	C50	C49	M10

2.5.8 Control Relay Map.

CONTROL RELAYS																ADDRESS
MSB								LSB								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
R16	R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	M11
R32	R31	R30	R29	R28	R27	R26	R25	R24	R23	R22	R21	R20	R19	R18	R17	M12
R48	R47	R46	R45	R44	R43	R42	R41	R40	R39	R38	R37	R36	R35	R34	R33	M13
R64	R63	R62	R61	R60	R59	R58	R57	R56	R55	R54	R53	R52	R51	R50	R49	M14

2.5.9 System Relay Map.

SYSTEM RELAYS															ADDRESS			
MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	LSB	0	
S16	S15	S14	S13	S12	S11	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1			M15
S32	S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17			M16

Bit Number	Digital Input Number	Description
0	S1	ON
1	S2	1st Scan
2	S3	0.1 Second Clock Period
3	S4	1 Second Clock Period
4	S5	1 Minute Clock Period
5	S6	CMP < MEM/K
6	S7	CMP = MEM/K
7	S8	CMP > MEM/K
8	S9	PLC Running
9	S10	PLC Re-Program Request
10	S11	PLC Re-Program Acknowledge
11	-	-
12	-	-
13	-	-
14	-	-
15	-	-

3. RS232/RS485 Modbus Communications Port

3.1 Modbus Master.

The communications port can be configured to function as a Modbus master device. To enable this mode you must make sure that the Modbus Master tick box is selected in Procon's ProSoft PLC programming software.

In this mode, you can configure the OI100 to read a range of registers from a remote Modbus slave or you can write a range of registers to a remote slave. You can configure up to 20 of these communications blocks.

The setup parameters are as follows:

- **Remote ID.** This is the network ID of the Modbus slave device.
- **Function.** You must enter a value of 3 to read a range of registers and a value of 16 to write to a range of registers. Function 3 reads registers in the slave and stores them in memory in the OI100. Function 16 reads memory in the OI100 and writes them to registers in the slave device.
- **Local Address.** This is the memory location in the OI100 where the data will be read from or written to. For example, if you want to access memory M1 then you must put a 1 into the local address field. (Do not put the Modbus address 30002).
- **Range.** This is the number of consecutive memory locations that will be transmitted.
- **Remote Address.** This is the register location in the slave device where data will be written to or read from. If you want to access a modbus register for example 40010 in the remote slave device, then you must put a value of 9 into this field.

3.2 Modbus Slave.

The communications port can be configured to function as a Modbus slave device.

When configured as a modbus slave, the OI100 will respond to network requests from a modbus master on the network. This could be another OI100, PLC or PC.

The modbus functions supported are as follows:

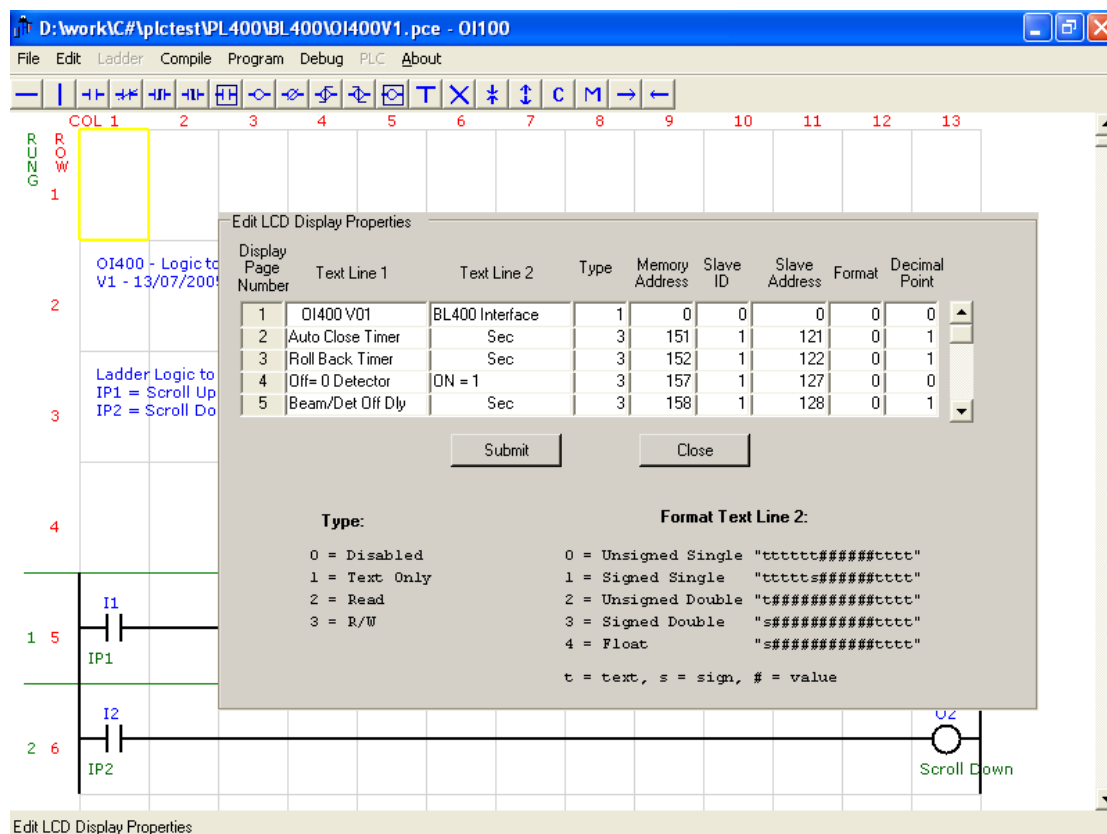
Modbus Slave Commands				
Modbus Function	Description	Memory start	Memory end	Max. Range
1or2	Reads a range of bits from any part of RAM	M0	M518	1600
3or4	Reads a range of registers from RAM, EEPROM and BBRAM.	M0	M638	100
5	Reads a single Bit from any part of RAM	M0	M518	1
6	Reads a single register from RAM, EEPROM and BBRAM.	M0	M638	1
15	Writes a range of bits to RAM.	M2	M518	1600

4. Programming the OI100

The OI100 is programmed using ProSoft Ladder Logic Software.

4.1 Programming the Display parameters.

The display is setup using the special table in ProSoft. The table is found in the Edit->Display Properties dialog box. An example program is shown below.



The columns of the table are described below. Each row represents a different screen on the display.

4.1.1 Step 1: Display page number

This is the page number that is to be edited. A maximum of 20 pages can be edited and displayed. Select a row for editing. The pages on the display are displayed in sequential order.

4.1.2 Step 2: Text Line 1

This is the text that will be displayed on the top line of the display. This line is for text only. A maximum of 16 characters can be displayed on a single line.

4.1.3 Step 3: Text Line 2

This is the text that will be displayed on the bottom line of the display. The data is also written to this line. Look at the format of the data to ensure that the text does not clash with the data.

4.1.4 Step 4: Type

The type configures the display with the following options:

- 0 – Disabled. The page is not displayed.
- 1 – Text Only. The page only displays text and does not display data.
- 2 – Read Only. Line 2 consists of text and data. The buttons cannot be used to edit the data.
- 3 – R/W. The data on line 2 is displayed and can be edited with the buttons. Data which has been configured with the Floating Point format cannot be edited.

4.1.5 Step 5: Memory Address

This is the memory address in the OI100 that contains the data to be displayed on line 2.

4.1.6 Step 6: Slave ID

This is the network ID where the updated data is sent to after being edited. If the ID is 0 then the data is updated in the OI100 memory. If the ID is greater than 0 then the data is updated in the memory of a remote slave device.

4.1.7 Step 7: Slave Address

This is the memory address where the updated data is sent to after being edited.

4.1.8 Step 8: Format

This is the format of the data that is displayed on line 2. Refer to the list of formats on the screen in ProSoft.

4.1.9 Step 9: Decimal Point

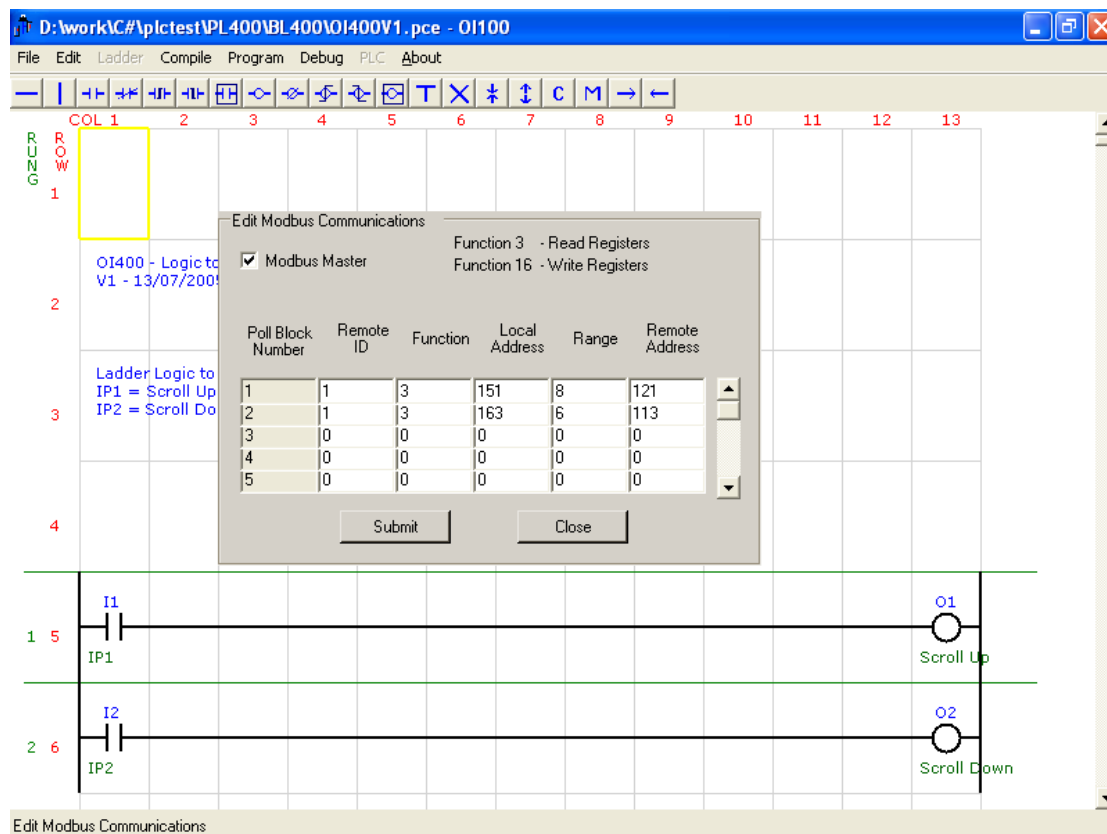
This enables to data to be displayed with a decimal point. The position of the decimal point is determined by the value given. A value of 0 disables the decimal point.

4.1.10 Step 10: Submit

Submit the data by clicking on the Submit button. Go back to step 1 to enter another page or click on the Close button to exit. Remember that these parameters will only be saved the next time the project is saved.

4.2 Programming the Communications parameters.

If the data that is to be displayed is on a remote slave device, then this data must first be fetched from memory in the remote slave and then saved in memory in the OI100. The communications data is setup by entering the Edit->Communications menu in ProSoft.



The columns of the table are described below. Each row represents a new communications request on the communications port.

4.2.1 Step 1: Modbus Master

The first step is to click on the Modbus Master tick box. This will configure the communications port as master for retrieving data from the remote slaves.

4.2.2 Step 2: Poll Block number

This is the communications poll number that is to be edited. A maximum of 20 blocks can be used. Select a row for editing.

4.2.3 Step 3: Remote ID

This is the network ID of the slave device where the data will be read from.

4.2.4 Step 4: Function

This is the value that determines if data is to be read from the remote slave or written to the remote slave. In our example the data is read so Function = 3.

4.2.5 Step 5: Local Address

This is the starting memory address in the OI100 where the data will be saved.

4.2.6 Step 6: Range

The range is the number of memory locations that will be read. The maximum value is 20.

4.2.7 Step 7: Remote Address

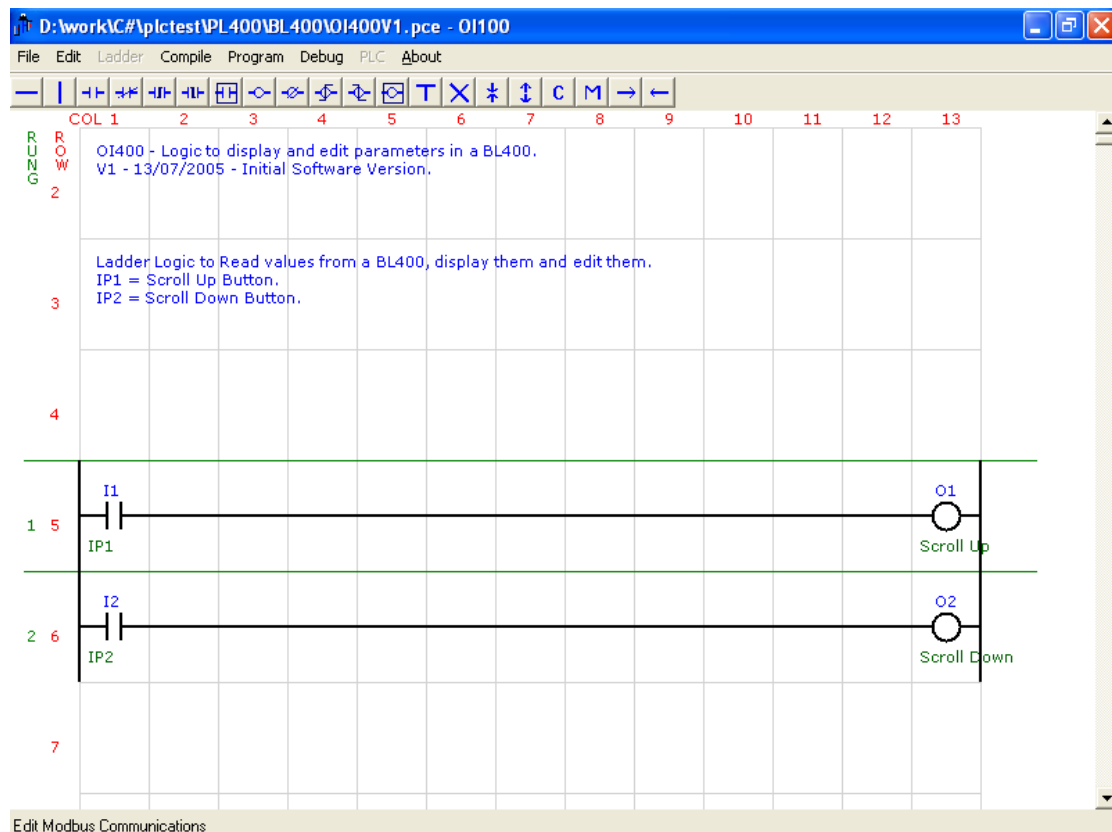
This is the remote address that contains the data.

4.2.8 Step 8: Submit

Submit the data by clicking on the Submit button. Go back to step 1 to enter another block or click on the Close button to exit. Remember that these parameters will only be saved the next time the project is saved.

4.3 Programming the Ladder Logic to use the buttons.

The buttons on the front of the OI100 can be used to scroll up and down through the display pages. This is done in the ladder program as shown below by connecting the inputs to the scroll outputs.



It is also possible to connect a timer to the scroll outputs so that the display scrolls automatically through the pages.

The current page number being displayed is stored in memory **M147**. It is possible to get the ladder program to change the page being displayed by writing a new value to this memory location.

4.4 Editing the data on the display.

The buttons on the front of the OI100 are used to edit the data that is currently being displayed. This feature will only be enabled if the TYPE is R/W (3) and is not a floating point format.

4.4.1 Step 1: Entering Edit Mode.

To enter the edit mode, both buttons must be pressed at the same time. The least significant digit will start to flash to shown that the display is in edit mode.

4.4.2 Step 2: Changing the current digit value.

Press the Down button to change the value of the current digit which is flashing. For each button press, the digit will increment up to 9 and then back to 0 again. When the correct value has been selected then advance to the next digit.

4.4.3 Step 3: Advancing to the next digit.

Press the Up button to advance to the next digit. Repeat steps 2 and 3 until the correct value has been entered.

4.4.4 Step 4: Exit Edit Mode.

To exit the edit mode, both buttons must be pressed at the same time. The new value will now be sent to the memory location as setup in the Display Properties dialog box. Either to the memory of the OI100 or to the memory of the remote slave device.

5. Modbus Memory Map (MODULE TYPE = 44)

The data in the OI100 is stored in registers. These registers are accessed over the network using the MODBUS RTU communication protocol.

There are 4 types of variables which can be accessed from the module. Each module has one or more of these data variables.

<u>Type</u>	<u>Start Address</u>	<u>Variable</u>
1	00001	Digital Outputs
2	10001	Digital Inputs
3	30001	Input registers (Analog)
4	40001	Output registers (Analog)

Modbus Address	Mem Addr	Register Name	Low Limit	High Limit	Access	Comments
10017	1.1	Digital Input 1	0	1	R	Status of Digital Inputs 1.
"	"	"	"	"	"	"
10032	1.16	Digital Input 16	0	1	R	Status of Digital Inputs 16.
00033	2.1	Digital Output 1	0	1	R/W	Status of Digital Outputs 1.
"	"	"	"	"	"	"
00048	2.16	Digital Output 16	0	1	R/W	Status of Digital Outputs 16.
00049	3.1	Timer 1	0	1	R/W	Status of Timer 1.
"	"	"	"	"	"	"
00112	6.16	Timer 64	0	1	R/W	Status of Timer 64.
00113	7.1	Counter 1	0	1	R/W	Status of Counter 1.
"	"	"	"	"	"	"
00176	10.16	Counter 64	0	1	R/W	Status of Counter 64.
00177	11.1	Control Relay 1	0	1	R/W	Status of Control relay 1.
"	"	"	"	"	"	"
00240	14.16	Control Relay 64	0	1	R/W	Status of Control relay 64.
00241	15.1	System Relay 1	0	1	R/W	Status of System relay 1.
"	"	"	"	"	"	"
00272	16.16	System Relay 32	0	1	R	Status of System relay 32.
30001	0	S/W Version / Module Type	N/A	N/A	R	High Byte = Software Version Low Byte = 44
30002	1	Digital Inputs	N/A	N/A	R	Digital Inputs in 16 bits.
40003	2	Digital Outputs	N/A	N/A	R/W	Digital Outputs in 16 bits.
40004	3	Timer Status	N/A	N/A	R/W	Timer Status 16 – 1
40005	4	Timer Status	N/A	N/A	R/W	Timer Status 32 – 17
40006	5	Timer Status	N/A	N/A	R/W	Timer Status 48 – 33
40007	6	Timer Status	N/A	N/A	R/W	Timer Status 64 – 49
40008	7	Counter Status	N/A	N/A	R/W	Counter Status 16 – 1
40009	8	Counter Status	N/A	N/A	R/W	Counter Status 32 – 17

40010	9	Counter Status	N/A	N/A	R/W	Counter Status 48 – 33
40011	10	Counter Status	N/A	N/A	R/W	Counter Status 64 – 49
40012	11	Control Relay	N/A	N/A	R/W	Control Relay 16 – 1
40013	12	Control Relay	N/A	N/A	R/W	Control Relay 32 – 17
40014	13	Control Relay	N/A	N/A	R/W	Control Relay 48 – 33
40015	14	Control Relay	N/A	N/A	R/W	Control Relay 64 - 49
40016	15	System Relay	N/A	N/A	R/W	System Relay 16 – 1
40017	16	System Relay	N/A	N/A	R/W	System Relay 32 – 17
-	17	-	N/A	N/A	-	Do not use – System only
-	18	-	N/A	N/A	-	Do not use – System only
40020	19	Timer 1 Value	0	65535	R/W	Timer range 0 to 65535.
“	“	“	“	“	“	“
40083	82	Timer 64 Value	0	65535	R/W	Timer range 0 to 65535.
40084	83	Counter 1 Value	0	65535	R/W	Counter range 0 to 65535.
“	“	“	“	“	“	“
40147	146	Counter 64 Value	0	65535	R/W	Counter range 0 to 65535.
40148	147	Display Page Number	0	20	R/W	0 to 20.
40149	148	User Memory	0	65535	R/W	0 to 65535.
“	“	“	“	“	“	“
40519	518	User Memory	0	65535	R/W	0 to 65535.
40520	519	Baud rate Programming Port	9600	19200	R/W	Default = 19200
40521	520	ID Prog. Port	0	255	R/W	Default = 1
40522	521	User EEPROM	0	65535	R/W	User EEPROM
“	“	“	“	“	“	“
40619	618	User EEPROM	0	65535	R/W	User EEPROM

6. Ladder Logic Function Blocks

The function blocks supported by the OI100 are listed below:

OI100 Function Blocks	
Function	Function Block Description
Timer 0.1Sec	Single input timer with 0.1 Second time base. The timer will run as long as the input is on. The timer will be reset to zero when the input is off.
Timer 0.01Sec	Single input timer with 0.01 Second time base. The timer will run as long as the input is on. The timer will be reset to zero when the input is off.
TimerA 0.1Sec	Accumulating timer with 0.1 Second time base. The timer will run as long as the input is on and stops when the input is removed. The timer will continue when the input is on again. The timer will be reset to zero when the reset input is on.
TimerA 0.01Sec	Accumulating timer with 0.01 Second time base. The timer will run as long as the input is on and stops when the input is removed. The timer will continue when the input is on again. The timer will be reset to zero when the reset input is on.
Counter	Up counter with reset input. The counter will count up when the count input goes from off to on. The counter will be reset to zero when the reset input is on. The counter output will go on when the count value is greater or equal to the preset value. The counter memory is addressed as the counter number + an offset
Counter Up/Dn	Up/Down counter with reset input. The counter will count up when the Up count input goes from off to on. The counter will count down when the Down count input goes from off to on. The counter will be reset to zero when the reset input is on. The counter output will go on when the count value is greater or equal to the preset value. The counter memory is addressed as the counter number + an offset of 16, so for example the value for counter 1 is in memory 17
NOP	This is a no operation function.
END	Placing this output function in the ladder program will indicate the end of the program. Any ladder after this function will not be run.
LD	Load the accumulator from memory(M) or with a constant(K).
LDD	The Load Double loads the accumulator with a 32 bit value from memory(M) or with a constant(K). The memory used is the two consecutive 16 bit memory locations, M & M+1.
LDF	The Load Float loads the accumulator with a float value from memory(M) or with a constant(F). The memory used is the two consecutive 16 bit memory locations, M & M+1.
OUT	Outputs the accumulator to memory(M).
OUTD	Outputs the 32 bit accumulator to two consecutive memory locations, M & M+1.
OUTF	Outputs the float accumulator to two consecutive memory locations, M & M+1.
AND	AND the accumulator with memory(M) or with a constant(K).
ANDD	AND the 32 bit accumulator with memory(M) or with a constant(K). The memory used is the two consecutive 16 bit memory locations, M & M+1.
OR	OR the accumulator with memory(M) or with a constant(K).
ORD	OR the 32 bit accumulator with memory(M) or with a constant(K). The memory used is the two consecutive 16 bit memory locations, M & M+1.
XOR	Exclusive OR the accumulator with memory(M) or with a constant(K).

OI100 Function Blocks	
Function	Function Block Description
XORD	Exclusive OR the 32 bit accumulator with memory(M) or with a constant(K). The memory used is the two consecutive 16 bit memory locations, M & M+1.
CMP	Compare the accumulator lower 16 bits with memory(M) or with a constant(K). If the value in the accumulator is less than the value in memory/constant then system bit S6 is turned on. If the value in the accumulator is equal to the value in memory/constant then system bit S7 is turned on. If the value in the accumulator is greater than the value in memory/constant then system bit S8 is turned on.
CMPD	Compare the 32 bit accumulator with memory(M) or with a constant(K). If the value in the accumulator is less than the value in memory/constant then system bit S6 is turned on. If the value in the accumulator is equal to the value in memory/constant then system bit S7 is turned on. If the value in the accumulator is greater than the value in memory/constant then system bit S8 is turned on.
CMPF	Compare the 32 bit accumulator with memory(M) or with a constant(F). If the value in the accumulator is less than the value in memory/constant then system bit S6 is turned on. If the value in the accumulator is equal to the value in memory/constant then system bit S7 is turned on. If the value in the accumulator is greater than the value in memory/constant then system bit S8 is turned on.
ADD	Add the memory(M) or constant(K) to the accumulator. The result is stored in the accumulator.
ADDD	Add the memory(M) or constant(K) to the 32 bit accumulator. The result is stored in the accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
ADDF	Add the memory(M) or constant(F) to the float accumulator. The result is stored in the float accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
SUB	Sub the memory(M) or constant(K) from the accumulator. The result is stored in the accumulator
SUBD	Sub the memory(M) or constant(K) from the 32 bit accumulator. The result is stored in the accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
SUBF	Sub the memory(M) or constant(F) from the float accumulator. The result is stored in the float accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
MUL	Multiply the accumulator with the memory(M) or constant(K). The result is stored in the accumulator
MULD	Multiply the 32 bit accumulator with the memory(M) or constant(K). The result is stored in the accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
MULF	Multiply the float accumulator with the memory(M) or constant(F). The result is stored in the float accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
DIV	Divide the accumulator by the memory(M) or constant(K). The result is stored in the accumulator.
DIVD	Divide the 32 bit accumulator by the memory(M) or constant(K). The result is stored in the accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
DIVF	Divide the float accumulator by the memory(M) or constant(F). The result is stored in the float accumulator. The memory used is the two consecutive 16 bit memory locations, M & M+1.
INC	Increment the memory(M). The result is stored in the memory(M)

OI100 Function Blocks	
Function	Function Block Description
INCD	Increment two consecutive memory(M) locations. The result is stored in the memory M & M+1.
DEC	Decrement the memory(M). The result is stored in the memory (M).
DECD	Decrement two consecutive memory(M) locations. The result is stored in the memory M & M+1.
INV	Invert the bits in the accumulator
MOV	Moves a variable in a memory location to a new location. The accumulator must already contain the address of the memory location to be moved.
SHL	The bits in the accumulator are shifted left by the memory(M) or constant(K). The lower bits are filled with zeros.
SHR	The bits in the accumulator are shifted right by the memory(M) or constant(K). The upper bits are filled with zeros.
CALL	This function is used to call a subroutine. The constant(k) is the label of the subroutine.
SUBR	This function is the start of a subroutine. The constant(k) is the label of the subroutine which is called by the call function.
RET	This function must be placed at the last line of a subroutine. The function can also be used in the subroutine for a conditional return.
RAND	A random number from 0 to 100 is placed in the accumulator
ACOSF	Arc Cosine of float accumulator
ASINF	Arc Sine of float accumulator
ATANF	Arc Tangent of float accumulator
COSF	Cosine of float accumulator
SINF	Sine of float accumulator
TANF	Tangent of float accumulator
SQRTF	Square Root of float accumulator
BTOF	The value in the 32 bit accumulator is converted to a float value and stored in the float accumulator.
FTOB	The value in the float accumulator is converted to a binary number and stored in the 32 bit accumulator.
RADF	The Radian of the float accumulator.
DEGF	The degrees of the float accumulator.
LOGF	The log of the float accumulator.
EXPF	The exponential of the float accumulator
PWRF	The power of the float accumulator.
COMM	Communications function. Enter a parameter number to select the data to be saved. 0 = Port Number (default = 1) 1 = Protocol (default = 0) 2 = Slave network ID 3 = PLC Memory Address 4 = Range 5 = Slave Address 6 = Timeout 7 = Function

