

BRX REMOTE I/O: BX-DMIO, BX-EBC100 & GS-EDRV100



CHAPTER 14

In This Chapter...

What it is.....	14-2
BX-DMIO & BX-EBC100 I/O Controllers	14-3
General Specifications	14-3
Power Supply Connector	14-5
AC Power Wiring (BX-DMIO-M & BX-EBC100-M)	14-6
Power Supply Specifications.....	14-6
Power Supply Connections	14-6
DC Power Wiring (BX-DMIO-M-D & BX-EBC100-M-D)	14-7
Power Supply Specifications	14-7
Power Supply Connections	14-7
Ethernet Communications	14-8
DIP Switch Specifications	14-8
Status Indicators.....	14-9
Supported Modules.....	14-10
Configuration	14-11
Wiring and IP address selection	14-11
System Configuration.....	14-11
Error Reporting:	14-15
Clearing Ethernet I/O Slave Errors and Warnings.....	14-17
Ethernet I/O Master Startup and Shutdown Sequences	14-18
Ethernet Master I/O Update Cycle	14-20
GS-EDRV100	14-21
GS-EDRV100 Setup	14-21
GSREGRD – GS EDrive Register Read	14-22
GSREGWR – GS EDrive Register Write.....	14-26

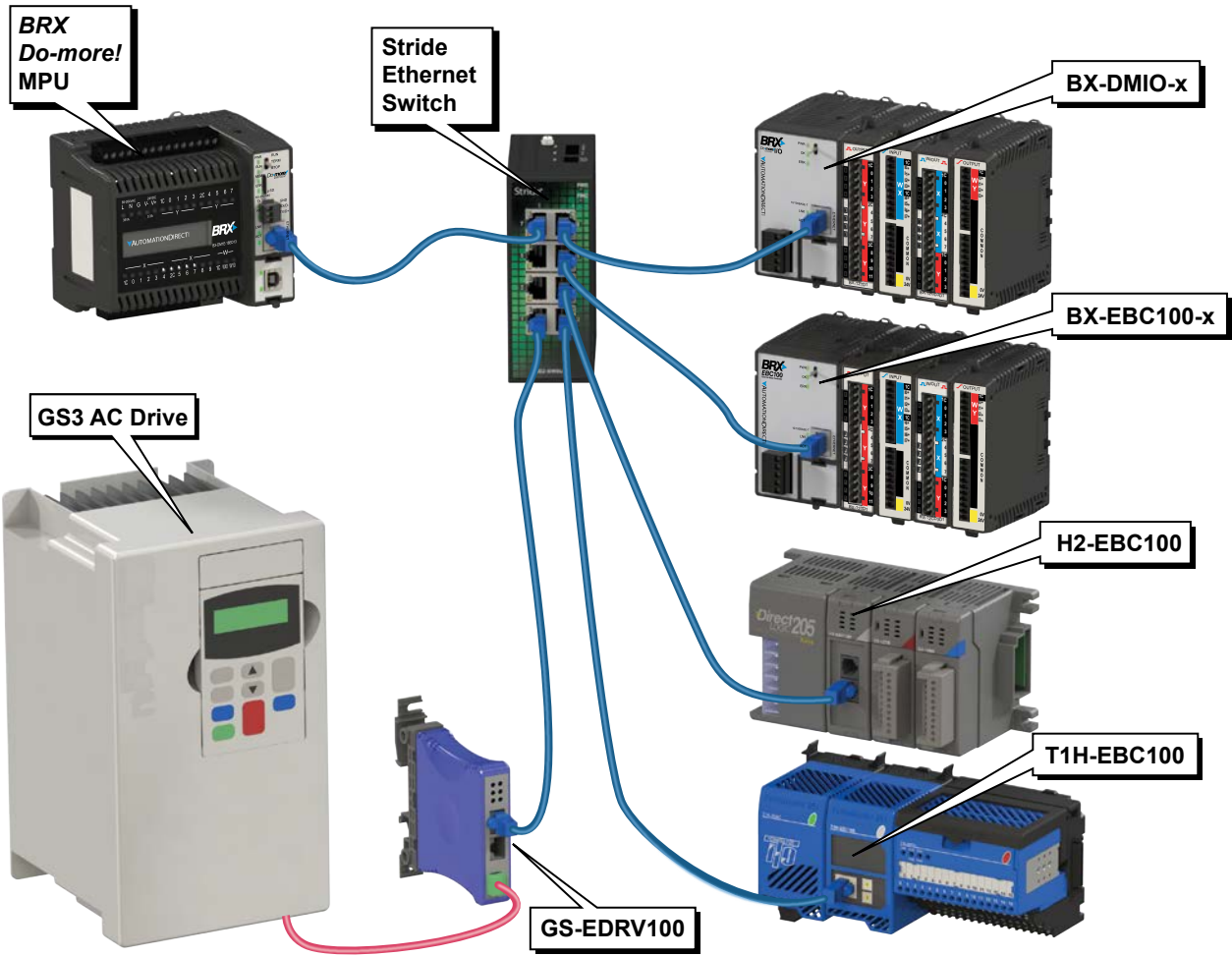
What it is

The BX-DMIO Modules are low-cost remote I/O controller modules that provide, through an Ethernet connection, a remote I/O drop to any PLC utilizing the Do-more! technology.

The BX-EBC100 Modules are remote I/O controller modules that provide, through an Ethernet connection, a remote I/O drop to any PLC utilizing the Do-more! technology or can be used with the DirectLOGIC PLC family using an Ethernet Remote Master (ERM100).

Comparison Chart		
	BX-DMIO	BX-EBC100
Lowest Cost	Yes	No
Supported Protocols	Do-more! Ethernet Remote I/O	Do-more! Ethernet Remote I/O Host Ethernet Remote I/O Modbus TCP
Can Be Used With	<div>BX-DM1E MPUs</div> <div>H2-DM1E CPUs</div> <div>T1H-DM1E CPUs</div> <div>Up to 16 slaves can be used by the Masters listed above.</div>	<div>BX-DM1E MPUs</div> <div>H2-DM1E CPUs</div> <div>T1H-DM1E CPUs</div> <div>H2-ERM100s</div> <div>H4-ERM100s</div> <div>Up to 16 slaves can be used by the Masters listed above.</div>

In this chapter we will be focusing on using DMIO and EBC100 along with GS-EDRV100 to provide a complete remote I/O solution for a Do-more! technology based PLC.



BX-DMIO & BX-EBC100 I/O Controllers

General Specifications

- RJ45 port for Ethernet communications
- No support for Pluggable Option Module (POM)
- Supports up to 8 additional Expansion Modules
- No built-in discrete or analog I/O
- AC and DC powered units available

BX-DMIO Controllers			
Part Number	External Power	Expansion Modules	Can be Used With
BX-DMIO-M	120–240 VAC	8	BX-DM1E MPUs H2-DM1E CPUs T1H-DM1E CPUs
BX-DMIO-M-D	12–24 VDC	8	



BX-DMIO-M
Do-more! I/O Controller

BX-EBC100 Controllers			
Part Number	External Power	Expansion Modules	Can be Used With
BX-EBC100-M	120–240 VAC	8	BX-DM1E MPUs H2-DM1E CPUs T1H-DM1E CPUs H2-ERM100s H4-ERM100s
BX-EBC100-M-D	12–24 VDC	8	



BX-EBC100-M
Ethernet Base Controller

General Specifications, continued

General Specifications				
Part Number	BX-DMIO-M	BX-DMIO-M-D	BX-EBC100-M	BX-EBC100-M-D
Operating Temperature	0° to 60°C (32° to 140°F)			
Storage Temperature	-20° to 85°C (-4° to 185°F)			
Humidity	5 to 95% (non-condensing)			
Environmental Air	No corrosive gases permitted			
Vibration	IEC60068-2-6 (Test Fc)			
Shock	IEC60068-2-27 (Test Ea)			
Enclosure Type	Open Equipment			
Agency Approvals	UL61010-2 - UL File # E185989 Canada and USA CE Compliant EN61131-2			
Noise Immunity	NEMA ICS3-304			
EU Directive	See the "EU Directive" topic in the Help File			
Heat Dissipation	10.8 W Max	8.6 W Max	10.8 W Max	8.6 W Max
Weight	198g (7oz)	170g (6oz)	198g (7oz)	170g (6oz)
Software Version	Do-more! Designer 2.3 or later			



NOTE: Eight (8) Expansion Modules can be connected to add I/O capability. Only Discrete and Analog Expansion Modules are supported.



NOTE: For specifications for H2-EBC100, T1H-EBC100 or GS-EDRV100 please refer to the appropriate manual on the AutomationDirect.com website under Manuals/Docs.

Power Supply Connector

The power supply terminal block connectors are included with the BX-EBC100 and BX-DMIO. The power supply connectors are a screw type, removable terminal block. Replacement connectors and specifications are listed in the tables that follow.

AC Power Supply Connector Specifications (-M models)			
Part Number	BX-RTB05 (Included w/Unit)	BX-RTB05-1	BX-RTB05-2
Connector Type	Screw Type-90 deg	Spring Clamp Type-180 deg	Screw Type-180 deg
Wire Exit	180 deg	180 deg	180 deg
Pitch	5.0 mm	5.0 mm	5.0 mm
Screw Size	M2.5	N/A	M2.5
Recommended Screw Torque	< 3.98 lb·in (0.45 N·m)	N/A	< 3.98 lb·in (0.45 N·m)
Screwdriver Blade Width	3.5 mm	3.5 mm	3.5 mm
Wire Gauge (Single wire)	28–12 AWG	28–14 AWG	28–14 AWG
Wire Gauge (Two wire)	28–16 AWG	28–16 AWG (Dual Wire Ferrule Required)	28–16 AWG (Dual Wire Ferrule Required)
Wire Strip Length	0.3 in (7.5 mm)	0.37 in (9.5 mm)	0.37 in (9.5 mm)
Equiv. Dinkle P/N	5ESDV-05P-BK	5ESDSR-05P-BK	5ESDF-05P-BK



BX-RTB05



BX-RTB05-1



BX-RTB05-2

DC Power Supply Connector Specifications (-M-D models)			
Part Number	BX-RTB03 (Included w/Unit)	BX-RTB03-1	BX-RTB03-2
Connector Type	Screw Type-90 deg	Spring Clamp Type-180 deg	Screw Type-180 deg
Wire Exit	180 deg	180 deg	180 deg
Pitch	5.0 mm	5.0 mm	5.0 mm
Screw Size	M2.5	N/A	M2.5
Recommended Screw Torque	< 3.98 lb·in (0.45 N·m)	N/A	< 3.98 lb·in (0.45 N·m)
Screwdriver Blade Width	3.5 mm	3.5 mm	3.5 mm
Wire Gauge (Single wire)	28–12 AWG	28–14 AWG	28–12 AWG
Wire Gauge (Two wires)	28–16 AWG	28–16 AWG (Dual Wire Ferrule Required)	28–16 AWG
Wire Strip Length	0.3 in (7.5 mm)	0.37 in (9.5 mm)	0.3 in (7.5 mm)
Equiv. Dinkle P/N	5ESDV-03P-BK	5ESDSR-03P-BK	5ESDF-03P



BX-RTB03



BX-RTB03-1



BX-RTB03-2

AC Power Wiring (BX-DMIO-M & BX-EBC100-M)

Power Supply Specifications

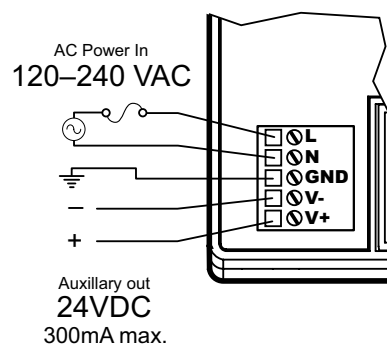
Power Supply Specifications	
Nominal Voltage Range	120–240 VAC
Input Voltage Range (Tolerance)	85–264 VAC
Rated Operating Frequency	47–63 Hz
Maximum Input Power	40VA
Cold Start Inrush Current	1.5 A, 2ms
Maximum Inrush Current (Hot Start)	1.5 A, 2ms
Internal Input Fuse Protection	Micro fuse 250V, 2A Non-replaceable
Acceptable External Power Drop Time	10ms
Under Input Voltage Lock-out	80VAC
Input Transient Protection	Input choke and line filter
Output Protection for Over Current, Over Voltage, and Over Temperature	Self resetting
Auxiliary 24VDC Output	24VDC @ 0.3 A max, <1V P-P Ripple, Integrated self-resetting short circuit protection, Isolated
Voltage Withstand (dielectric)	1500VAC Power Inputs to Ground applied for 1 minute 1500VAC Ground to 24VDC Output applied for 1 minute
Insulation Resistance	>10M Ω @ 500VDC

Power Supply Connections

Pin	Connection
1	L
2	N
3	GND
4	V–
5	V+



Removable Connector Included
ADC Part # BX-RTB05



WARNING: Do not exceed the 24VDC auxiliary power supply load limit of 300mA.



WARNING: The BRX System **MUST** have a proper earth ground. Do not operate the BRX MPU without proper earth grounding.

DC Power Wiring (BX-DMIO-M-D & BX-EBC100-M-D)

Power Supply Specifications

Power Supply Specifications	
Nominal Voltage Range*	12–24 VDC
Input Voltage Range (Tolerance)*	10–36 VDC
Maximum Input Voltage Ripple	< $\pm 10\%$
Maximum Input Power	30W
Cold Start Inrush Current	5A, 2ms
Maximum Inrush Current (Hot Start)	5A, 2ms
Internal Input Protection	Reverse polarity protection and undervoltage lockout via transistor circuit
Acceptable External Power Drop Time	10ms
Under Input Voltage Lock-out	<9VDC
Output Protection for Over Current, Over Voltage, and Over Temperature	Self resetting
Auxiliary 24VDC Output	None
Voltage Withstand (dielectric)	1500VAC power Inputs to ground applied for 1 minute
Insulation Resistance	>10M Ω @ 500VDC

* Class 2 or LPS Power Supply required.

Power Supply Connections

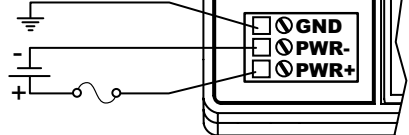
Pin	Connection
1	GND
2	PWR –
3	PWR +



Removable Connector Included
ADC Part # BX-RTB03



Class 2 or LPS
User Supplied Power
DC Power In
12–24 VDC



WARNING: The BRX System **MUST** have a proper earth ground. Do not operate the BRX MPU without proper earth grounding.

Ethernet Communications

The RJ-45 Ethernet port connector is located on the CPU faceplate. Rated at 10/100 Mbps, it accepts standard CAT5e cable and has built-in auto-crossover capability; no crossover cable is required.

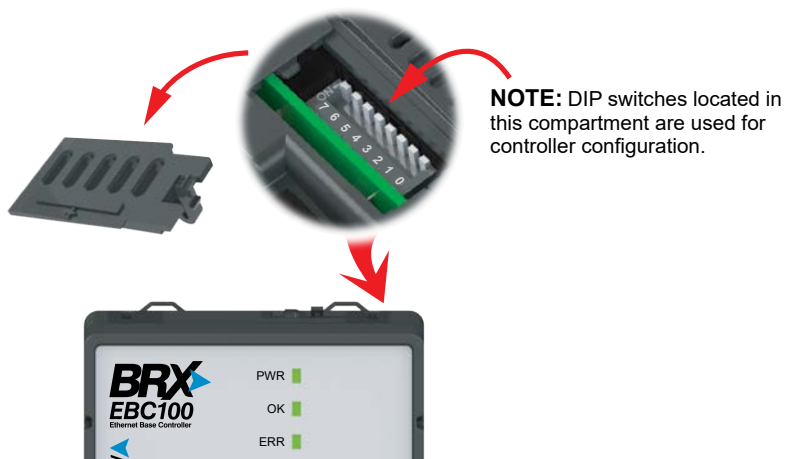
Ethernet Specifications		
	BX-DMIO-M (-D)	BX-EBC100-M (-D)
Port Type	Ethernet	
Description	Standard transformer-isolated Ethernet port with built-in surge protection.	
Transfer Rate	10Mbps and 100Mbps	
Port Status LED	LINK LED is solid GREEN when network LINK is established. ACT LED flashes GREEN when port is active.	
Supported Protocols (Server)	Do-more! Ethernet Remote I/O Embedded Web Server: HTTP (Unsecure)	Do-more! Ethernet Remote I/O HOST Ethernet Remote I/O Modbus TCP Embedded Web Server: HTTP (Unsecure)
Cable Recommendation	C5E-STxxx-xx from AutomationDirect.com	
Port Type	RJ45, Category 5, 10/100 BASE-T, Auto Crossover	
Ethernet Port Numbers:		
Do-more! Ethernet Expansion I/O (Server)	28784, UDP	28784, UDP
HOST Ethernet Remote I/O (Server)	-	28784, UDP
Modbus TCP (Server)	-	502, TCP
Embedded Web Server: HTTP (Unsecure)	-	80, TCP

14 DIP Switch Specifications

The BX-DMIO and BX-EBC100 Remote I/O controllers have a block of DIP switches, located on the top of the chassis, that are used to setup the Module ID, write protect the module and disable the Web configuration page. The DIP switch settings are read when the system is powered up. The default position for all DIP switches is OFF.



WARNING: The BX-DMIO and BX-EBC100 Remote I/O controllers must be powered down prior to changing DIP switch settings.



DIP Switch Specifications, continued

The following table describes the position and function associated with each DIP switch.

DIP Switch Settings		
DIP#	Switch Position	Function
7	ON	Reserved
6	ON	Reset the module to factory default
5	ON	Write protect the module. No writes to flash memory are allowed. Disable Web configuration page.
4	ON	Station number: DIP Switch 0–4: Supports addresses 1 to 31.
3	ON	
2	ON	All DIP Switch OFF: Station number from flash. Enter via Nedit 3 or Web configuration page. Supports addresses 1 to 247.
1	ON	
0	ON	



DIP Switch position is read when power is applied to the module. Power down the module before changing DIP switch settings.

DIP switches 0 to 4: Use to set a Module ID (0x00 up to 0x1F).

DIP Switch 5: When ON, it will write protect the module. It will not be possible to update the network settings, update the firmware or the booter version. It also disables access to the Web configuration page.

DIP Switch 6: When ON, the module is reset to factory defaults.

Status Indicators

Status Indicators		
Indicator	Status	Description
PWR	OFF	Base Power is OFF
	GREEN	Base Power is Good
OK	GREEN	Controller is Functioning Normally
	GREEN (Blinking)	Reset Mode (Booter without OS)
	YELLOW	Analog Module Mismatch
	OFF	Controller HW Error or Power is OFF
ERR	OFF	Controller is Functioning Normally
	RED	Controller HW Error or Communications Watchdog Timeout
	RED (Blinking)	Reset Mode (Booter without OS)
LNK	OFF	No Ethernet Link
	GREEN	Ethernet Link is Good
ACT	OFF	No Ethernet Activity
	GREEN (Blinking)	Ethernet Activity

Supported Modules

List of modules supported by BX-DMIO & BX-EBC100 and corresponding number of I/O points

Module	Number of Points		Module	Number of Points		Module	Number of Points							
	In	Out		In	Out		In	Out						
Discrete Input			Discrete Input/Output			Analog Input								
BX-08NF3	8		BX-08CD3R	4	4	BX-04ADM-1	4							
BX-08ND3	8		BX-12CD3D1	8	4	BX-04AD-1	4							
BX-08NB	8		BX-12CD3D2	8	4	BX-08AD-1	8							
BX-08NA	8		BX-16CD3D1	8	8	BX-16AD-1	16							
BX-08SIM	8		BX-16CD3D2	8	8	BX-04AD-2B	4							
BX-12ND3	12		BX-16CF3F2	8	8	BX-08AD-2B	8							
BX-12NB	12		Temperature			BX-16AD-2B	16							
BX-12NA	12					BX-04AD-3	4							
BX-16NF3	16					BX-08AD-3	8							
BX-16ND3	16					Analog Output								
BX-16NB	16													
BX-16NA	16		Temperature Input/Analog Output											
BX-32ND3	32													
Discrete Output														
BX-05TRS		5	Temperature Input/Analog Output			BX-04DA-1		4						
BX-05TRS-1		5	BX-4RTD4DA-1	4	4	BX-08DA-1		8						
BX-08TD1		8	BX-4THM4DA-1	4	4	BX-04DA-2B		4						
BX-08TD2		8	BX-4UT4DA-3	4	4	BX-08DA-2B		8						
BX-08TR		8	Temperature Input/Discrete Output			BX-04DA-3		4						
BX-08TRZ		8				BX-4UT4TD1	4	4	BX-08DA-3		8			
BX-08TA		8				BX-4UT4TD2	4	4	Analog Input/Output					
BX-12TD1		12				BX-4UT4TR	4	4						
BX-12TD2		12				List of modules NOT supported by BX-DMIO & BX-EBC100								
BX-12TR		12												
BX-12TA		12												
BX-16TD1		16												
BX-16TD2		16												
BX-16TF2		16												
BX-16TR		16												
BX-16TRZ		16												
BX-32TD1		32												
BX-32TD2		32												

List of modules NOT supported by BX-DMIO & BX-EBC100

BX-HSIO1	BX-SERIO	BX-SERIO-4
BX-HSIO2	BX-SERIO-2	BX-APAD
BX-HSIO4		

Configuration

Configuration of a Do-more! Remote I/O network is fairly straightforward.

1. Hook up power and Ethernet cables to the PLC and all remote I/O drops.
2. Use the NetEdit3 utility to give the PLC and all remote I/O drops an IP address.
3. Use Do-more! Designer software to add the remote I/O drops to the PLC program.
4. Write your program.

Wiring and IP address selection

See Appendix E for information on Ethernet wiring and how to use NetEdit3 for IP address selection. Using the NetEdit3 tool is the easiest way to set IP addresses for use with a Remote I/O network.



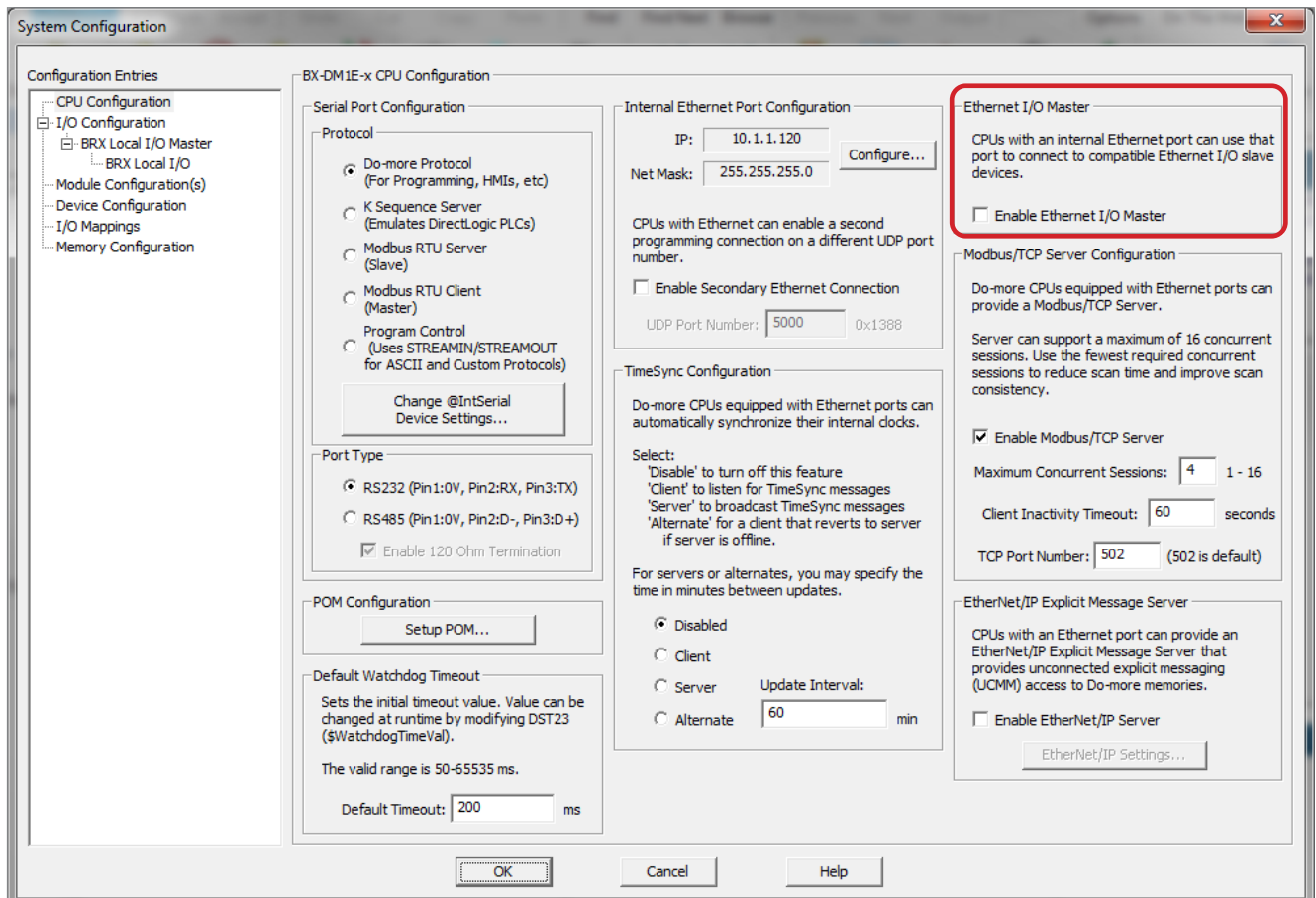
NOTE: We recommend that all Remote I/O networks be placed on isolated networks if at all possible.

System Configuration

CPU Configuration

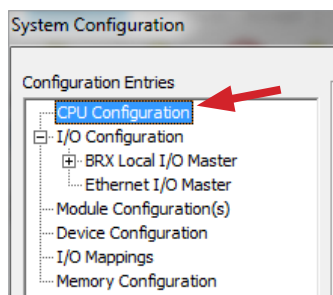
The first step to setting up a Do-more! Ethernet I/O Master is to enable it in the System Configuration.

You will find this under the CPU Configuration section of the System Configuration Dialog or by left clicking the Ethernet Port in the Dashboard and picking “Enable Ethernet I/O Master”.

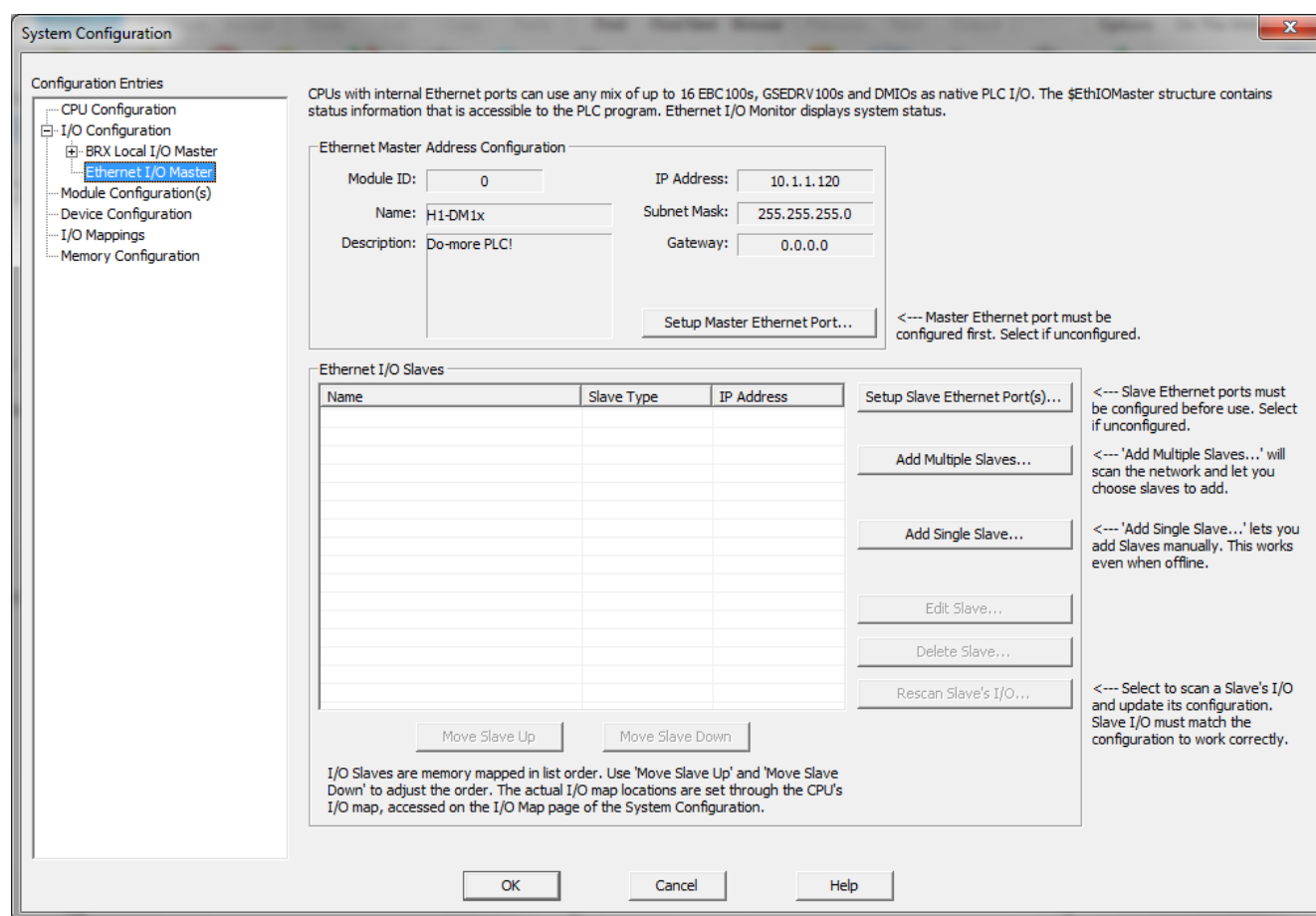


CPU Configuration, continued

Once the Ethernet I/O Master is enabled, you will notice that you now have a new entry under the I/O Configuration.



Clicking on the new Ethernet I/O Master entry will pop up this dialog.



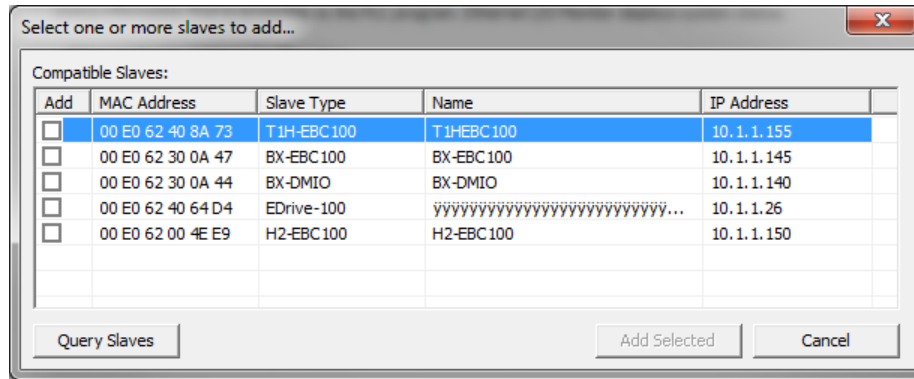
You are now ready to begin adding your slave devices to the Ethernet I/O Master.

If you have not setup IP addresses for your slave devices, now is the time to do so using the “Setup Slave Ethernet Port(s)” button.

If you have IP addresses assigned to your slave devices, select either “Add Multiple Slaves” or “Add a Single Slave”.

CPU Configuration, continued

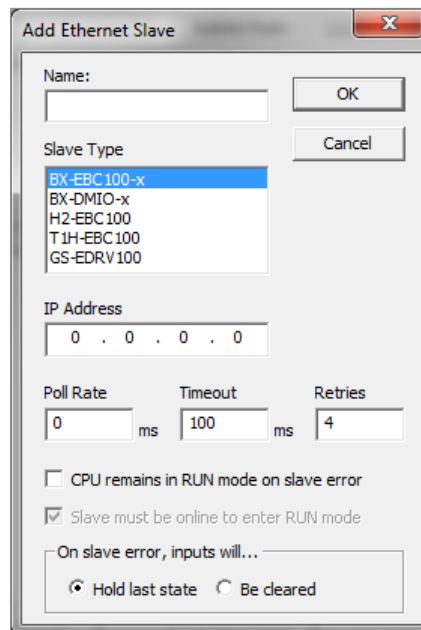
Add multiple Slaves will bring up the following dialog.



Assuming that your PC is on the same network as your slaves, you should see the slaves listed as above.

Pick the slaves that you wish to be attached to the Ethernet I/O Master by checking the box in the “Add” column and press “Add Selected”.

To add a single slave, you must know the IP address.



The first three items uniquely identify the Slave on the network, and describe the communication requirements.

Name specifies the name of the slave (any combination of alphanumeric and punctuation characters up to 255 characters), the **Slave Type** selects the type of device from the list (BX-DMIO-x, BX-EBC100-x, H2-EBC100, T1H-EBC100, or GS-EDRV100), and the **IP Address** specifies the TCP/IP address of the slave device.

CPU Configuration, continued

The next three items describe the communication requirements. All of the Slaves have an Ethernet Watchdog Timer that gets set when the Ethernet I/O Master initializes the Slave. The value of this timer is computed from the **Poll Rate**, **Timeout** and **Retries** settings for the Slave using the following formula:

$$\text{Ethernet Watchdog Timer} = \text{Poll Rate} + (\text{Timeout} * (\text{Retries} + 1))$$

If the Ethernet communication between the Ethernet I/O Master and this Slave is ever interrupted longer than the timer value the Slave will turn OFF all of the discrete outputs and set all of the analog output values to 0.

Poll Rate specifies the frequency (in milliseconds) at which the Ethernet I/O Master will read and write to the I/O modules in this slave device. This can be any value between 0 and 32767. The default value of 0 means as “often as possible”.

Timeout specifies the amount of time (in milliseconds) the Ethernet I/O Master will wait on a response from a read or write operation before it retries that operation. This can be any value between 25 and 32767.

Retries specifies the number of times the Ethernet I/O Master will retry a read or write operation before it cancels that operation and returns an error. This can be any value between 25 and 32767.

The last three options specify what actions the Ethernet I/O Master will take when it encounters an error from this I/O Slave:

CPU remains in RUN mode on slave error selects what action to take if the CPU is in RUN mode and this I/O Slave reports an error:

If **checked** the CPU will remain in RUN mode.

If **unchecked (default)** the CPU will drop into PROGRAM mode.

The **Slave must be online to enter RUN mode** specifies whether or not the CPU can go to RUN mode based on whether or not this I/O Slave is present on the network.

If **checked (default)** the CPU can only go to RUN mode if this I/O Slave is present on the network.

If **unchecked** the CPU can go to RUN mode even if this I/O Slave is NOT present on the network.

On slave error, inputs will ... selects what action to take with the current image register data for the discrete and analog input modules in this I/O Slave if an error is reported:

If **Hold last state (default)** is selected the image register data for all of the discrete inputs and all of the analog inputs will retain the last values that were received.

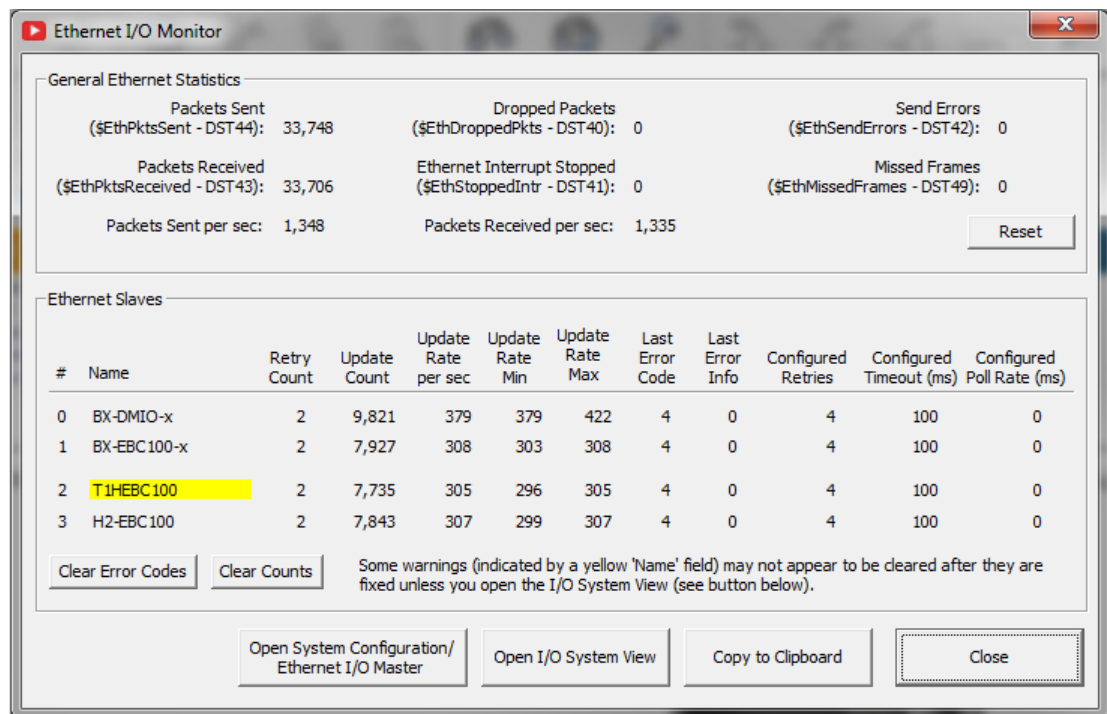
If **Be cleared** is selected the image register data for all discrete inputs will set to OFF, and the image register data for analog inputs will be set to 0.

The last steps are to save the configuration and download the program to the PLC.

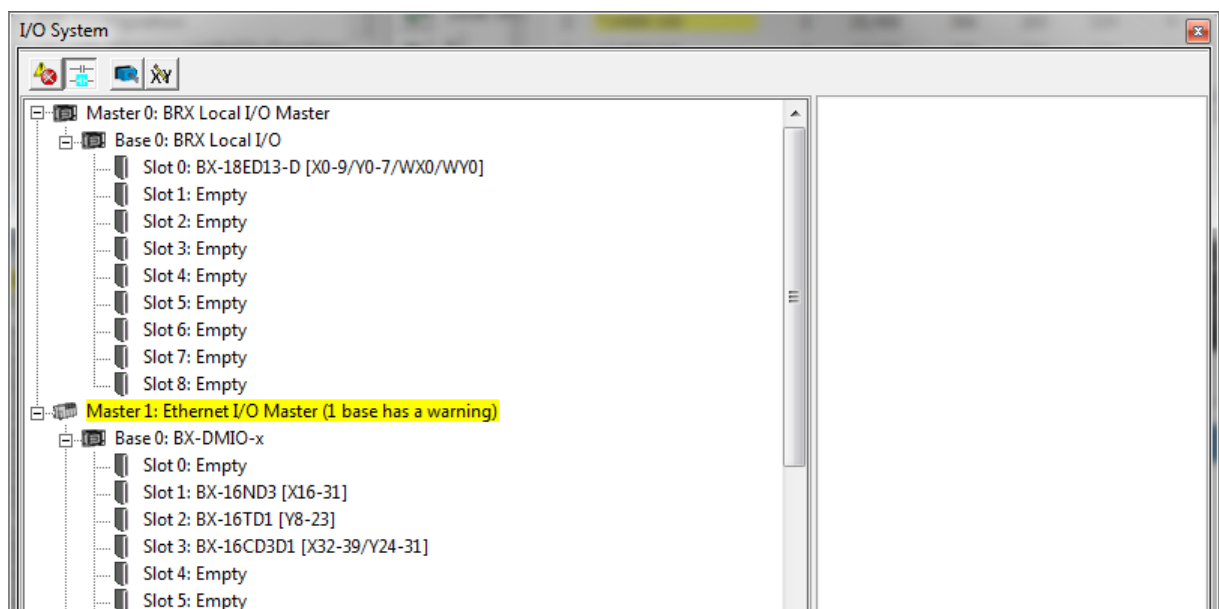
Error Reporting:

The **Ethernet I/O Master** can manage any mix of up to 16 Ethernet I/O Slaves. The \$EthIOMaster contains member fields that report the configuration status for each of the Slaves and report errors and warnings for each slot that contains an I/O module. The Ethernet I/O Master classifies Errors as any condition that will prevent the CPU from going to RUN mode; Warnings are conditions that allow the CPU to go RUN mode but an I/O module is reporting an issue that may require user intervention to resolve.

The Ethernet I/O Monitor is another utility that shows the current “health” of the Ethernet I/O Master and its configured Ethernet I/O Slaves.

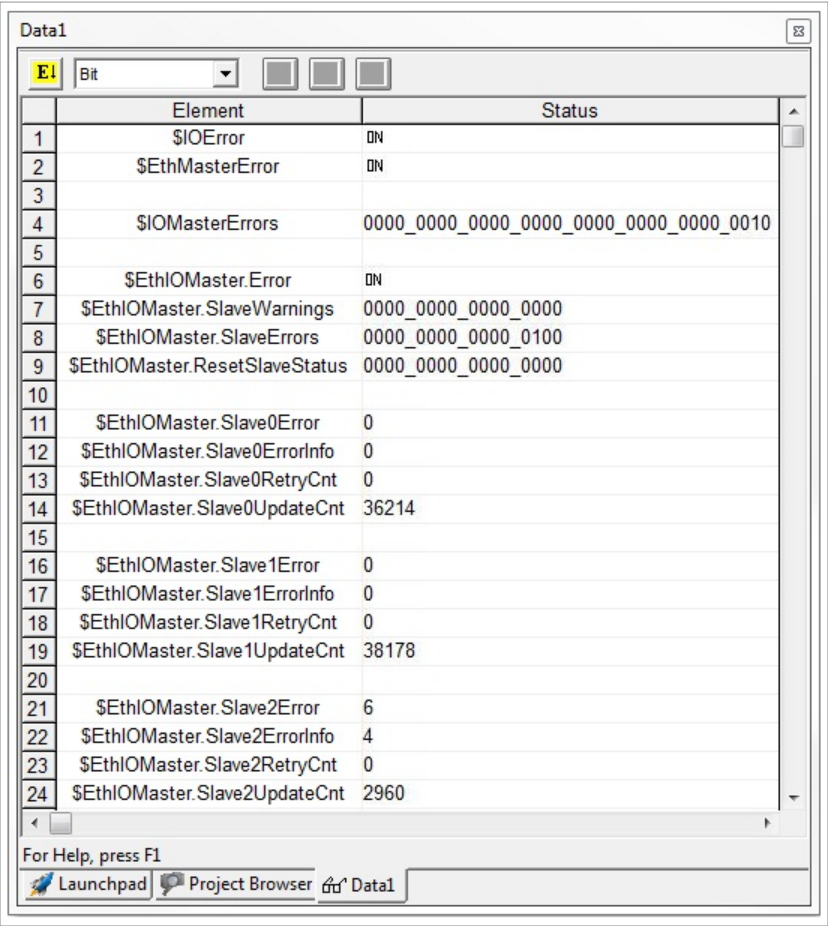


The I/O System View will display all of the current Error and Warning information for all of the I/O modules in all of the Ethernet I/O Slaves.



Error Reporting, continued

The Do-more! CPU has several built-in Bit and Word locations that can be utilized at runtime to detect Errors and Warnings in the Ethernet I/O subsystem. Refer to the following Data View to see the progression of error reporting that can be utilized in the ladder logic project:



	Element	Status
1	\$IOError	ON
2	\$EthMasterError	ON
3		
4	\$IOMasterErrors	0000_0000_0000_0000_0000_0000_0010
5		
6	\$EthIOMaster.Error	ON
7	\$EthIOMaster.SlaveWarnings	0000_0000_0000_0000
8	\$EthIOMaster.SlaveErrors	0000_0000_0000_0100
9	\$EthIOMaster.ResetSlaveStatus	0000_0000_0000_0000
10		
11	\$EthIOMaster.Slave0Error	0
12	\$EthIOMaster.Slave0ErrorInfo	0
13	\$EthIOMaster.Slave0RetryCnt	0
14	\$EthIOMaster.Slave0UpdateCnt	36214
15		
16	\$EthIOMaster.Slave1Error	0
17	\$EthIOMaster.Slave1ErrorInfo	0
18	\$EthIOMaster.Slave1RetryCnt	0
19	\$EthIOMaster.Slave1UpdateCnt	38178
20		
21	\$EthIOMaster.Slave2Error	6
22	\$EthIOMaster.Slave2ErrorInfo	4
23	\$EthIOMaster.Slave2RetryCnt	0
24	\$EthIOMaster.Slave2UpdateCnt	2960

In this example, you can see that there is an I/O Error (*\$IOError* is ON).

The I/O Master that is reporting the error is Ethernet I/O Master (bit 1 of *IOMasterErrors* is ON).

The Ethernet I/O Slave that is causing the Error (Slave #2) is indicated in the *\$EthIOMaster.SlaveErrors* structure member. With the .SlaveErrors structure member viewed in a Binary form, each Bit represents one of the 16 possible I/O Slaves, the lowest order Bit being Slave 0, the highest order Bit being Slave 15.

For each of the 16 potential I/O Slaves, there are two structure members that contain an error code that explains the problem and the slot number of the I/O module that is involved. The structure members are named *.Slave0Error* & *.Slave0ErrorInfo* through *.Slave15Error* & *.Slave15ErrorInfo*. In the example above, Slave #2 is reporting an error code 6 (I/O Mismatch) in Slot #4.

Error Reporting, continued

The following table lists the range of potential Error codes that can appear in the *.Slave0Error* through *.Slave15Error* fields:

Ethernet I/O Slave Error Codes		
Error Code	Error	Description
1	MODULE NOT RESPONDING	One or more of the I/O modules in the stored configuration is not detected in the base. Only applicable when the I/O Slave is a T1H-EBC100.
2	BASE CHANGED	A new I/O module is detected in the base that is not in the stored configuration. Only applicable when the I/O Slave is a T1H-EBC100.
3	SIGNATURE ERROR	The stored Link Monitor value does not match the Link Monitor value being returned by the I/O Slave—this can happen if the I/O Slave was power-cycled, or if the Slave's Link Monitor was reconfigured by a different master.
4	TIMEOUT	The I/O Slave did not respond within the allotted amount of time (Timeout x Retrys).
5	DEVICE MISMATCH	The I/O Slave that responded to a request was not the type of device (BX-DMIO, BX-EBC100, H2-EBC100, T1H-EBC100, or GS-EDRV100) that was expected. This can happen if the IP Address specified for the slave has been reassigned to a different type of slave.
6	I/O MISMATCH	I/O count of a module detected does not match the I/O count stored in the configuration. The corresponding <i>.SlaveXErrorInfo</i> member contains the slot number of the I/O module reporting the error.
7	MODULE CONFIG ERROR	An I/O module that requires a Module configuration did not have its configuration data correctly downloaded. The corresponding <i>.SlaveXErrorInfo</i> member contains the slot number of the I/O module reporting the error.

Clearing Ethernet I/O Slave Errors and Warnings

If any of the Ethernet I/O Slaves reports an I/O Warning, like a “broken transmitter” or “drive tripped” the remote slave will retain that warning until it is manually cleared. It does this to make sure that all of error and warning conditions that occur at any slave—specifically any intermittent ones—will be reported to the Ethernet I/O Master. So, after the problem that is causing the Error or Warning condition is resolved, there are two methods available to manually clear the retained status data from the Ethernet I/O system:

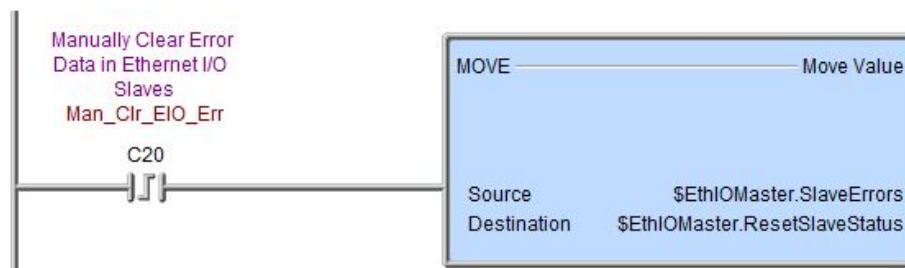
Using Do-more! Designer: opening the I/O System View will reset the Error and Warning information in each slave device when it polls each slave device for its Error and Warning data.

Runtime in Ladder Logic: in Do-more! version 2.0 and later, the *\$EthIOMaster* structure has a 16-Bit field named *.ResetSlaveStatus*. Each slave device is mapped to a bit in that 16-Bit register as follows: Bit 0 = Slave 0, Bit 1 = Slave 1, ... Bit 15 = Slave 15. Setting a Bit to a value of 1 will cause the Ethernet I/O Master to clear the retained Error and Warning data in the corresponding slave device and then set that Bit back to 0.

The preferred method for doing this at runtime is to perform a one-time copy of the contents of *\$EthIOMaster.SlaveErrors* to *\$EthIOMaster.ResetSlaveStatus*.

This will cause the Ethernet I/O Master to issue the reset command to each slave device that is currently showing an error. Each slave device that no longer has any error or warning data will leave its corresponding bit in *\$EthIOMaster.SlaveErrors* at 0. Any slave device that still has Error or Warning data will have its corresponding bit in *\$EthIOMaster.SlaveErrors* set back to 1.

Clearing Ethernet I/O Slave Errors and Warnings, continued



Ethernet I/O Master Startup and Shutdown Sequences

Once the Ethernet Master I/O configuration has been set the Do-more! CPU will perform the following sequences depending on the starting condition. Again, it is important to understand that the CPU places the same importance on the Ethernet Master I/O that it does on the Local I/O Master. The Ethernet I/O Master configuration must remain consistent for the CPU to operate normally.

Power ON Initialization Sequence is executed each time the CPU is powered ON.

- Query Network to make sure the currently configured Slaves are present on the network.
 - If all Slaves are present, each Slave is then queried to make sure that the I/O modules that are currently found in each Slave match the I/O modules that are in the stored configuration.
 - If all of the I/O modules match the current configuration, the Ethernet I/O Master will download the module configuration for all of the I/O modules that have one, for example, the CTRIO module.

The ERR LED (BX-DM1E, H2-DM1E) or STAT LED (T1H-DM1E) will be rapidly blinking RED while the module configurations are being downloaded.

- If all of the module configurations are downloaded successfully the initialization process is complete.

A unique value (called the Link Monitor) is written to each Ethernet I/O Slave. This unique Link Monitor value is included as part of each data packet between the Ethernet I/O Master and its Ethernet I/O Slaves. This allows the Ethernet I/O Master to verify that each response packet is from one of its own Ethernet I/O Slaves, and each Ethernet I/O Slave can know that each data packet it receives is from its own Ethernet I/O Master.

At this point the Ethernet I/O Master will begin normal updates for the I/O modules in all of its Ethernet I/O Slaves as described in the section below.

- If any of the Module configurations fail to download, the CPU will NOT go into RUN mode until the problem is resolved.

The ERR LED (BX-DM1E, H2-DM1E) or STAT LED (T1H-DM1E) will be solid RED.

The system Bit *\$IOError (ST152)* - "One or more I/O Master are reporting a problem with a module" will be ON. The error message can be seen on the System Status tab of the System Info utility.

The corresponding Bit for each Slave that is reporting a module error will be ON in the system structure member *\$EthIOMaster.SlaveError*.

Ethernet I/O Master Startup and Shutdown Sequences, continued

The appropriate structure member (*\$EthIOMaster.Slave0ErrorInfo* - *\$EthIOMaster.Slave7ErrorInfo*) will contain the number of the first slot that has an I/O module that failed to download its Module configuration.

- If one or more of the I/O modules is different than what is in the configuration (this could be a module that is missing or a module that is a different type), the CPU will NOT go into RUN mode until the problem is resolved.

The ERR LED (BX-DM1E, H2-DM1E) or STAT LED (T1H-DM1E) will be solid RED.

The system Bit *\$IOError (ST152)* - “One or more Ethernet I/O Masters are reporting a problem with a module” will be ON. The error message can be seen on the System Status tab of the System Info utility.

The corresponding Bit for each Ethernet I/O Slave that is reporting a module error will be ON in the system structure member *\$EthIOMaster.SlaveError*.

The appropriate structure member (*\$EthIOMaster.Slave0ErrorInfo* - *\$EthIOMaster.Slave7ErrorInfo*) will contain the number of the first slot that has an I/O module that is different than the configuration.

- If one or more of the Ethernet I/O Slave devices (that are configured as required to be present on the network) are NOT present on the network, the CPU will NOT go into RUN mode until the problems is resolved.

The ERR LED (BX-DM1E, H2-DM1E) or STAT LED (T1H-DM1E) will be solid RED.

The system Bit *\$IOError (ST152)* - “One or more Ethernet I/O Masters are reporting a problem with a module” will be ON.

The system Bit *\$EthMasterError (ST153)* - “The Ethernet I/O Master is reporting an error, most likely a slave is offline” will be ON. These error messages can be seen on the System Status tab of the System Info utility.

The corresponding Bit for each Ethernet I/O Slave that is NOT present will be ON in the system structure member *\$EthIOMaster.SlaveErrors*.

The Ethernet I/O Master will continue to query the network for missing Ethernet I/O Slaves at 10 second intervals. If one of the missing Ethernet I/O Slaves appears on the network, it will be initialized as described above.

After all of the missing Ethernet I/O Slaves are located the remainder of the initialization process will run, if that process completes successfully the ERR LED (BX-DM1E, H2-DM1E) or STAT LED (T1H-DM1E) will go OFF and the CPU can then be manually put into to RUN mode.

PROGRAM mode to RUN mode Sequence is executed each time the CPU transitions from PROGRAM mode to RUN mode. The process is very similar to the power-on sequence but there are a couple of notable exceptions that are described below. The resolution to any problems is the same as described in the power-on sequence.

- Query Network to make sure the currently configured Ethernet I/O Slaves are still present on the network.

Ethernet I/O Master Startup and Shutdown Sequences, continued

- If all Ethernet I/O Slaves are present, each one is then queried to make sure that the I/O modules that are currently found match the I/O modules that are in the stored configuration.
- If all of the I/O modules match the stored configuration the Ethernet I/O Master will initialize the image register locations with the Module configuration data for any that require it, for example: the Terminator I/O analog output modules, and the F2-8AD4DA-x modules. The Module configurations for all I/O modules are NOT downloaded again, only image register locations that need to be initialized with Module configuration data get processed.
 - After all of the initialization is complete the Ethernet I/O Master will begin normal updates for the I/O modules in its Ethernet I/O Slaves as described in the next subsection.
- If one or more of the I/O modules is different than what is in the configuration (this could be a module that is missing or a module that is a different type), the CPU will NOT go into RUN mode until the problems is resolved.
- If one or more of the configured Ethernet I/O Slaves are NOT present on the network, the CPU will NOT go into RUN mode until the problems is resolved.

RUN mode to PROGRAM mode is executed each time the CPU transitions from RUN mode to PROGRAM mode.

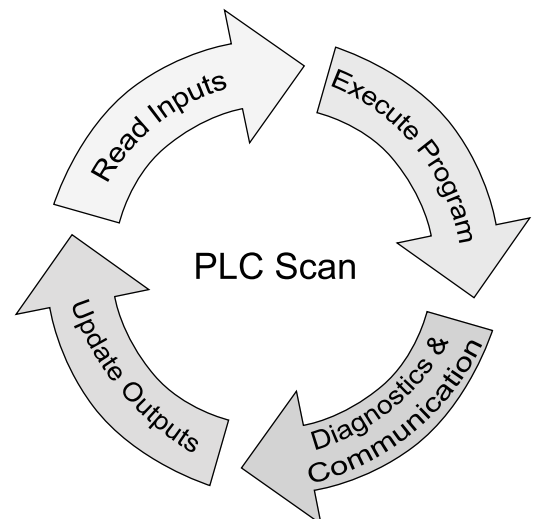
- All of the discrete outputs are turned OFF and a value of 0 is written to all of the channels of the analog output modules.

Ethernet Master I/O Update Cycle

Once it has been enabled and its Ethernet I/O Slaves configured, the CPU allows the Ethernet I/O Master to run during the Diagnostics and Communication slice of each scan. Each time the Ethernet I/O Master function is allowed to run it will update all of the configured Ethernet I/O Slaves.

When the CPU is in **PROGRAM mode**, only status data from the input modules in each Ethernet I/O Slave is read. That data is then used to update the Do-more's local copy in the image register. Any I/O modules that have structures instead of I/O, the structures will be read and the image register will be updated with that data. This allows both the input modules from Ethernet I/O Slaves and input modules in the local base to continue to update in PROGRAM mode.

When the CPU is in **RUN mode**, data for each output module is sent to the appropriate Ethernet I/O Slave, and for I/O modules that have structures instead of I/O, the I/O modules will be updated with data from their structures. The status data from the input modules in each Ethernet I/O Slave is read. Any I/O modules that have structures instead of I/O, the structures will be read and the image register will be updated with that data. The data from the input modules is then used to update the Do-more's local copy in the image register.

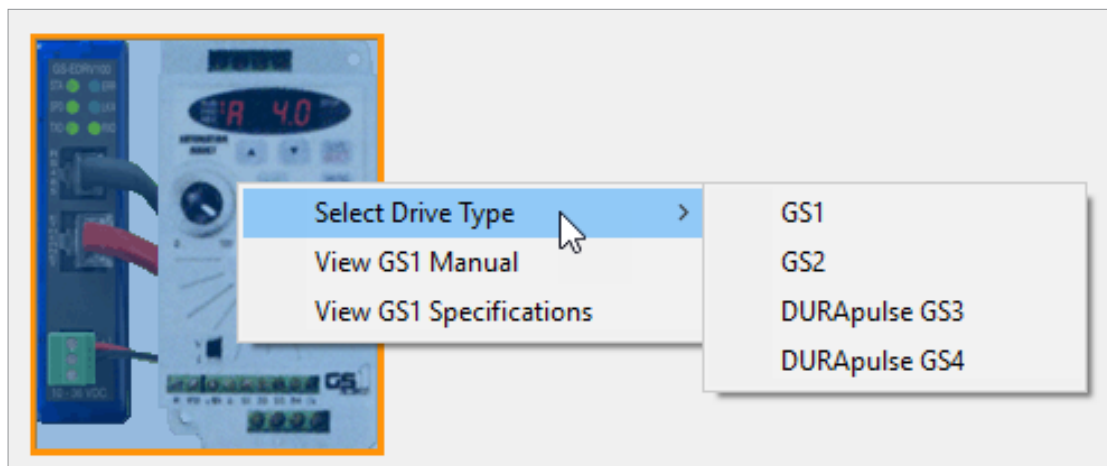


GS-EDRV100

The GS-EDRV100 is an Ethernet remote I/O slave device that allows an easy and convenient way to add an AutomationDirect GS1, GS2, GS3 or GS4 AC variable frequency drive to a Do-more! Ethernet Remote I/O or Host Ethernet Remote I/O network. This allows any Do-more! PLC (with a built-in Ethernet port), DL205 PLC or DL405 PLC (using the ERM100) to control any GS series drive as an Ethernet remote I/O drop.

GS-EDRV100 Setup

The process for adding a GS-EDRV100 is the same as adding any other remote I/O device. The only difference is that once you have the device added, you must go to the device in the I/O configuration section of the System Configuration and tell it which drive type you have (GS1, GS2, DURApulse GS3 or DURApulse GS4).



GS-EDRV100 Setup, continued

Once the drive is configured, you will have access to two instructions as well as a group of structure members. The structure members are as follows:

GSEDRV100 Structure Members			
Member	Read/Write ¹	Type ²	Meaning
\$GS1_000.RunCommand	W	UW	Set value to 0 to STOP, set value 1 to RUN (P9.27)
\$GS1_000.Direction	W	B	Set ON for FORWARD, turn this Bit OFF for REVERSE (P9.28)
\$GS1_000.RS485SpeedRef	W	B	Set the motor speed (P9.26)
\$GS1_000.OutputCurrent	R	UW	Shows the present Output Current
\$GS1_000.OutputFrequency	R	UW	Shows the present Output Frequency
\$GS1_000.StatusMonitor1	R	UW	Shows the present Status Monitor 1 (h2100) value
\$GS1_000.StatusMonitor2	R	UW	Shows the present Status Monitor 2 (h2101) value
\$GS1_000.ResetCommFault	W	B	Set ON to RESET FAULT, turn the Bit OFF for no action
\$GS1_000.ExtCommFault	R	B	ON if the AC Drive's Serial Comm Fault bit (P9.29) is ON
\$GS1_000.IntCommFault	R	B	ON if the GS-EDRV100's Serial cable to the GS Drive is disconnected or not functioning
\$GS1_000.BlockParm1	W	UW	Block Transfer Parameter 1 - User defined write value (P9.11)
\$GS1_000.BlockParm2	W	UW	Block Transfer Parameter 2 - User defined write value (P9.12)
\$GS1_000.BlockParm3	W	UW	Block Transfer Parameter 3 - User defined write value (P9.13)
\$GS1_000.BlockParm4	W	UW	Block Transfer Parameter 4 - User defined write value (P9.14)
\$GS1_000.BlockParm5	W	UW	Block Transfer Parameter 5 - User defined write value (P9.15)
\$GS1_000.BlockParm6	R	UW	Block Transfer Parameter 6 - User defined write value (P9.16)
\$GS1_000.BlockParm7	R	UW	Block Transfer Parameter 7 - User defined write value (P9.17)
\$GS1_000.BlockParm8	R	UW	Block Transfer Parameter 8 - User defined write value (P9.18)
\$GS1_000.BlockParm9	R	UW	Block Transfer Parameter 9 - User defined write value (P9.19)
\$GS1_000.BlockParm10	R	UW	Block Transfer Parameter 10 - User defined write value (P9.20)
\$GS1_000.BlockParm11	R	UW	Block Transfer Parameter 11 - User defined write value (P9.21)
\$GS1_000.BlockParm12	R	UW	Block Transfer Parameter 12 - User defined write value (P9.22)
\$GS1_000.BlockParm13	R	UW	Block Transfer Parameter 13 - User defined write value (P9.23)
\$GS1_000.BlockParm14	R	UW	Block Transfer Parameter 14 - User defined write value (P9.24)
\$GS1_000.BlockParm15	R	UW	Block Transfer Parameter 15 - User defined write value (P9.25)

1. R = Read Only, W = Read/Write

2. B = Boolean (Bit), UW = Unsigned 16-bit Integer

NOTE: GS4 drive users. The GS4 drives support up to Block Transfer #16. The Do-more! Designer heap-item structure supports up to Block Transfer #15. If the additional block transfer is needed, use the GSREGWR instruction, which allows the writing of all 16 Block Transfers at once (instead of using Do-more!'s heap-item structure). To read the value associated with Block Transfer #16, use the GSREGRD instruction.

GSREGRD – GS EDrive Register Read

The GS EDrive Register Read (GSREGRD) instruction is used to perform a series of read operations from parameter values in a GS Series AC Drive which is connected to a GS-EDRV100. The instruction can contain up to 50 individual READ operations which are performed in the order that they are listed in the instruction. If all of the listed READ operations were completed successfully the On Success selection will be executed. Processing of the listed READ operations will stop after the first failure and the On Error condition will be executed.

GSREGRD – GS EDrive Register Read, continued

This instruction can only be performed by the Ethernet I/O Master through a GS-EDRV100 which is connected to the GS Drive's serial port. When the Ethernet I/O Master is connected to a GS-EDRV100, it automatically reads a fixed set of parameters from the drive and places the data in the member fields of a GS Drive structure. This instruction is used to read values to parameter locations that are NOT already being read by that process. Parameter locations that are already being read from the GS Drive structure will still be displayed in the row editor with the name of the GS Drive structure, but they cannot be selected for use in this instruction.

Row#	Register	Destination
1	P21.00: Status Monitor 1	V0
2	P21.01: Status Monitor 2	V1
3	P21.07: Motor RPM	V2
4	P21.11: % Load	V3
5		

Parameters:



NOTE: Use the **F9** key to open the Default Element Selection Tool (the Element Picker or the Element Browser) or use the **Down-Arrow** key (Auto-Complete) on any parameter field to see a complete list of the memory locations that are valid for that parameter of the instruction.

Device - selects which of the configured GS-EDRV100 devices to use.

Create module - indicates that there are no GS-EDRV100 devices configured to perform this instruction. Selecting this option will open the Module Configuration section of the System Configuration utility where a GS-EDRV100 device can be created.

Structure - displays the name of the system-created structure for the selected GS-EDRV100.

The center section displays the table of individual write operations that will be performed each time this instruction is executed. The buttons below the table are used to manage the rows in the table:

Edit - opens the row editor to change the currently selected row.

GSREGRD – GS EDrive Register Read, continued

Insert - inserts an empty row before the currently selected row and opens the row editor .

Remove - removes the currently selected row.

Move Up / Move Down - moves the currently selected row up one row or down one row in the table respectively.

The row editor is used to add a new entry to the table, or modify an existing entry in the table. To open the row editor, double-click on the row, or highlight a row then click the **Edit** or **Insert** button.

Read GS1 Register

To select the GS Drive Parameter, fill in the P and ".dot" fields, or browse the complete list of parameters in the tree below.

GS1 Source Register P . Destination

Description

Range

Implied Decimal Places Default Value Modbus Address Hexadecimal Address

- P9.20: Block Transfer Parameter 10
- P9.26: Serial Comm Speed Reference
- P9.27: Serial Comm RUN Command
- P9.28: Serial Comm Direction Command
- P9.29: Serial Comm External Fault
- P9.30: Serial Comm Fault Reset
- P9.31: Serial Comm JOG Command
- P9.39: Firmware Version
- P9.41: GS Series Number
- P9.42: Manufacturer Model Information
- [-] P21.xx: Status Parameters
 - P21.00: Status Monitor 1**
 - P21.01: Status Monitor 2
 - P21.02: Frequency Command F
 - P21.03: Output Frequency H
 - P21.04: Output Current A
 - P21.05: DC Bus Voltage d
 - P21.06: Output Voltage U
 - P21.07: Motor RPM
 - P21.08: Scale Frequency (Low Word)
 - P21.09: Scale Frequency (High Word)
 - P21.11: % Load

Source Register - this is the Parameter number of the location in the GS Drive to read. The Source Register can be selected by entering the P XX.yy values in the two locations in the upper section, or by selecting one of the entries from the tree-control in the lower section.

GSREGRD – GS EDrive Register Read, continued

Destination - specifies the location to store the value read from the GS Drive.

The **Description**, **Range**, **Implied Decimal Places**, **Default Value**, **Modbus Address**, and **Hexadecimal Address** fields will display the appropriate information as the Source Register data is changed.

The **Browse to GSx Manual** and **Browse to GSx Specification** buttons will open the default web browser, which will then open a page on Automationdirect's web site that contains links to the User Manual or the Specification page for the type of GS Drive configured. These two options require a functional connection to the Internet and a PDF viewer.

The **Collapse All** button will collapse the Parameter Range lists so that only the Parameter Ranges are shown.

The **Expand All** button will fully expand all of the Parameter Ranges lists so that each selection in each Range is shown.

The **On Success** and **On Error** parameters specify what action to perform when this instruction completes. You do not have to use the same type of selection for both **On Success** and **On Error**.

If the **Set Bit** selection is used for either **On Success** or **On Error**, the specified BIT location will be SET OFF when the instruction is first enabled and will remain OFF until the instruction completes. Once complete, the appropriate Success or Error bit location will be set ON. The specified Bit location is enabled with a SET (Latch) operation meaning that it will remain ON even if the input logic for the instruction goes OFF.

If the **JMP to Stage** selection is used for either **On Success** or **On Error** the target Stage must be in the same Program code-block as this instruction. You cannot specify a target Stage that exists in a different Program code-block. When the operation finishes, the target Stage will be enabled the same way that a standalone Jump to Stage (JMP) instruction would do it. The JMP to Stage option will only be available if this instruction is placed in a Program code-block.

On Success - selects which of the following actions to perform if the operation is successful:

SET Bit - enable this selection then specify any writable bit location.

JMP to Stage - enable this selection then specify any Stage number from S0 to S127 in the current Program code-block.

On Error - selects which of the following actions to perform if the Close Device operation is unsuccessful:

SET Bit - enable this selection then specify any writable bit location.

JMP to Stage - enable this selection then specify any Stage number from S0 to S127 in the current Program code-block.

Automatically create the SG box for any NEW stage number - if either the On Success or On Error selections are set to JMP to Stage, this option will be enabled which will automatically create any target stage that does not already exist.

Below this rung - the new target stage will be created on a new rung following this instruction.

At end of code-block - the new target stage will be created as the last rung of this Program.

GSREGWR – GS EDrive Register Write

The GS EDrive Register Write (GSREGWR) instruction is used to perform a series of write operations to parameter values in a GS Series AC Drive which is connected to a GS-EDRV100. The instruction can contain up to 50 individual WRITE operations which are performed in the order that they are listed in the instruction. If all of the listed WRITE operations were completed successfully, the On Success selection will be executed. Processing of the listed WRITE operations will stop after the first failure and the On Error condition will be executed.

This instruction can only be performed by the Ethernet I/O Master through a GS-EDRV100 which is connected to the GS Drive's serial port. When the Ethernet I/O Master is connected to a GS-EDRV100, it automatically writes a fixed set of parameters from the member fields of a GS Drive structure. This instruction is used to write values to parameter locations that are NOT already being written by that process. Parameter locations that are already being written from the GS Drive structure will still be displayed in the row editor with the name of the GS Drive structure, but they cannot be selected for use in this instruction.

Parameters:



NOTE: Use the **F9** key to open the Default Element Selection Tool (the Element Picker or the Element Browser) or use the **Down-Arrow** key (Auto-Complete) on any parameter field to see a complete list of the memory locations that are valid for that parameter of the instruction.

Device - selects which of the configured GS-EDRV100 device to use.

GSREGWR – GS EDrive Register Write, continued

Create module - indicates that there are no GS-EDRV100 devices configured to perform this instruction, selecting this option will open the Module Configuration section of the System Configuration utility where a GS-EDRV100 device can be created.

Structure - displays the name of the system-created structure for the selected GS-EDRV100.

The center section displays the table of individual write operations that will be performed each time this instruction is executed. The buttons below the table are used to manage the rows in the table:

Edit - opens the row editor to change the currently selected row.

Insert - inserts an empty row before the currently selected row and opens the row editor.

Remove - removes the currently selected row.

Move Up / Move Down - moves the currently selected row up one row or down one row in the table respectively.

The row editor is used to add a new entry to the table, or modify an existing entry in the table. To open the row editor, double-click on the row, or highlight a row then click the Edit or Insert button.

Write GS1 Register

To select the GS Drive Parameter, fill in the P and "dot" fields, or browse the complete list of parameters in the tree below.

Source: GS1 Destination Register: P . ☐ Can set in RUN mode

Description:

Range:

Implied Decimal Places: Default Value: Modbus Address: Hexadecimal Address:

- [-] P0.xx: Motor Parameters
 - ... P0.00: Motor Nameplate Voltage
 - ... P0.01: Motor Nameplate Amps Factor
 - ... P0.02: Motor Base Frequency
 - ... P0.03: Motor Base RPM
 - ... P0.04: Motor Maximum RPM
- [-] P1.xx: Ramp Parameters
 - ... P1.00: Stop Methods
 - P1.01: Acceleration Time 1 (RUN mode)**
 - ... P1.02: Deceleration Time 1 (RUN mode)
 - ... P1.03: Accel S-curve
 - ... P1.04: Decel S-curve
 - ... P1.05: Acceleration Time 2 (RUN mode)
 - ... P1.06: Deceleration Time 2 (RUN mode)
 - ... P1.07: Select method to use for 2nd Accel/Decel
 - ... P1.08: Accel 1 to Accel 2 frequency transition
 - ... P1.09: Decel 2 to Decel 1 frequency transition
 - ... P1.10: Skip Frequency 1
 - ... P1.11: Skip Frequency 2
 - ... P1.12: Skip Frequency 3
 - ... P1.17: Skip Frequency Band
 - ... P1.19: DC Injection Voltage Level

GSREGWR – GS EDrive Register Write, continued

Source - specifies the location or constant value that contains the value to write to the GS Drive.

Destination Register - this is the Parameter number of the location in the GS Drive to which the Destination value will be written. The Destination Register can be selected by entering the P XX.yy values in the two location in the upper section, or by selecting one of the entries from the tree-control in the lower section.

The **Description**, **Range**, **Implied Decimal Places**, **Default Value**, **Modbus Address**, and **Hexadecimal Address** fields will display the appropriate information as the Source Register data is changed.

The **Browse to GSx Manual** and **Browse to GSx Specification** buttons will open the default web browser, which will then open a page on Automationdirect's web site that contains links to the User Manual or the Specification page for the type of GS Drive configured. These two options require a functional connection to the Internet and a PDF viewer.

The **Collapse All** button will collapse the Parameter Range lists so that only the Parameter Ranges are shown.

The **Expand All** button will fully expand all of the Parameter Ranges lists so that each selection in each Range is shown.

If all of the listed WRITE operations were completed successfully the On Success selection will be executed. Processing of the listed WRITE operations will stop after the first failure and the On Error condition will be executed.

The **On Success** and **On Error** parameters specify what action to perform when this instruction completes. You do not have to use the same type of selection for both **On Success** and **On Error**.

If the **Set Bit** selection is used for either **On Success** or **On Error**, the specified BIT location will be SET OFF when the instruction is first enabled and will remain OFF until the instruction completes. Once complete, the appropriate Success or Error bit location will be set ON. The specified Bit location is enabled with a SET (Latch) operation meaning that it will remain ON even if the input logic for the instruction goes OFF.

If the **JMP to Stage** selection is used for either **On Success** or **On Error** the target Stage must be in the same Program code-block as this instruction. You cannot specify a target Stage that exists in a different Program code-block. When the operation finishes, the target Stage will be enabled the same way that a standalone Jump to Stage (JMP) instruction would do it. The JMP to Stage option will only be available if this instruction is placed in a Program code-block.

On Success - selects which of the following actions to perform if the operation is successful:

SET Bit - enable this selection then specify any writable bit location.

JMP to Stage - enable this selection then specify any Stage number from S0 to S127 in the current Program code-block.

On Error - selects which of the following actions to perform if the Close Device operation is unsuccessful:

SET Bit - enable this selection then specify any writable bit location.

JMP to Stage - enable this selection then specify any Stage number from S0 to S127 in the current Program code-block.

Automatically create the SG box for any NEW stage number - if either the On Success or On Error selections are set to JMP to Stage, this option will be enabled which will automatically create any target stage that does not already exist.

Below this rung - the new target stage will be created on a new rung following this instruction.

At end of code-block - the new target stage will be created as the last rung of this Program.