BANNER ENGINEERING

SC22-3E

Industrial Ethernet User's Guide

1/8/2016

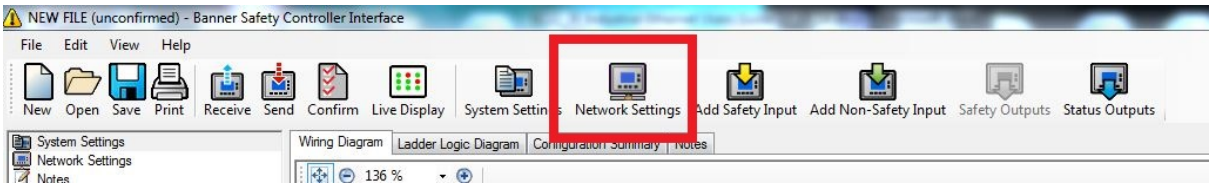
An aid for use in establishing Ethernet communications between the safety controller and a PLC or HMI.

Table of Contents

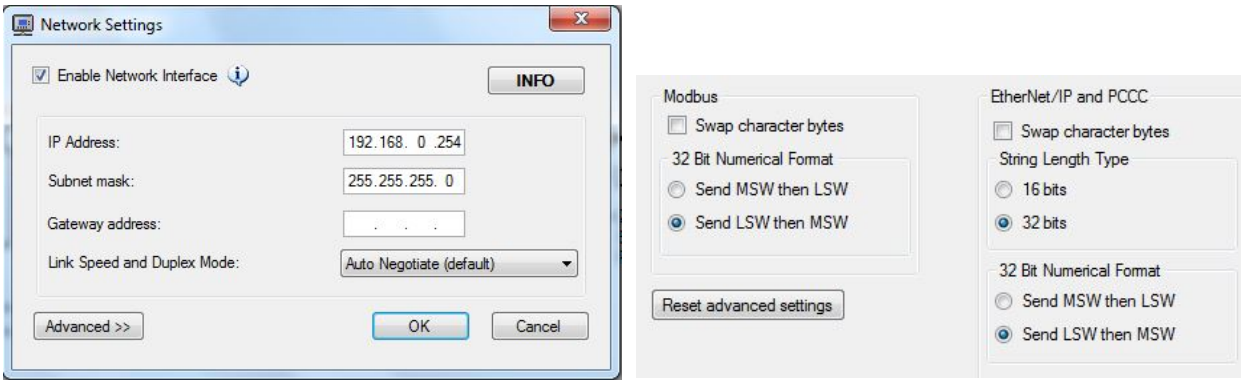
SC22-3E

Chapter 1: Configuring the Controller

By default, the SC22-3E Ethernet port is disabled. To enable the port, connect the safety controller to your PC via USB cable (SC-USB1, p/n 77724). Open the Banner Safety Controller PC Software and click on the Network Settings icon along the top of the screen (shown here in red box).

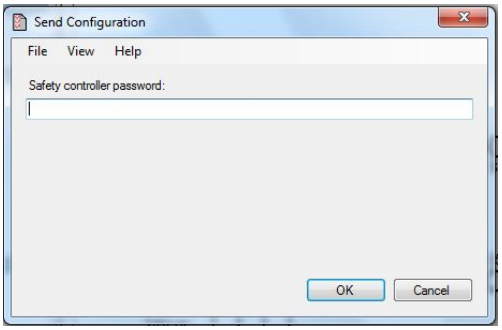


Place a check mark in the "Enable Network Interface" box then configure the IP Address and Subnet Mask as needed for your network and click on the OK button. Users can configure the Advanced network settings, if desired, by clicking on the "Advanced" button.



These are the default values for the SC22 Ethernet port and "Advanced" Industrial Ethernet options.

Click the Send icon along the top of the screen (shown here in red box), then enter the password and click OK.

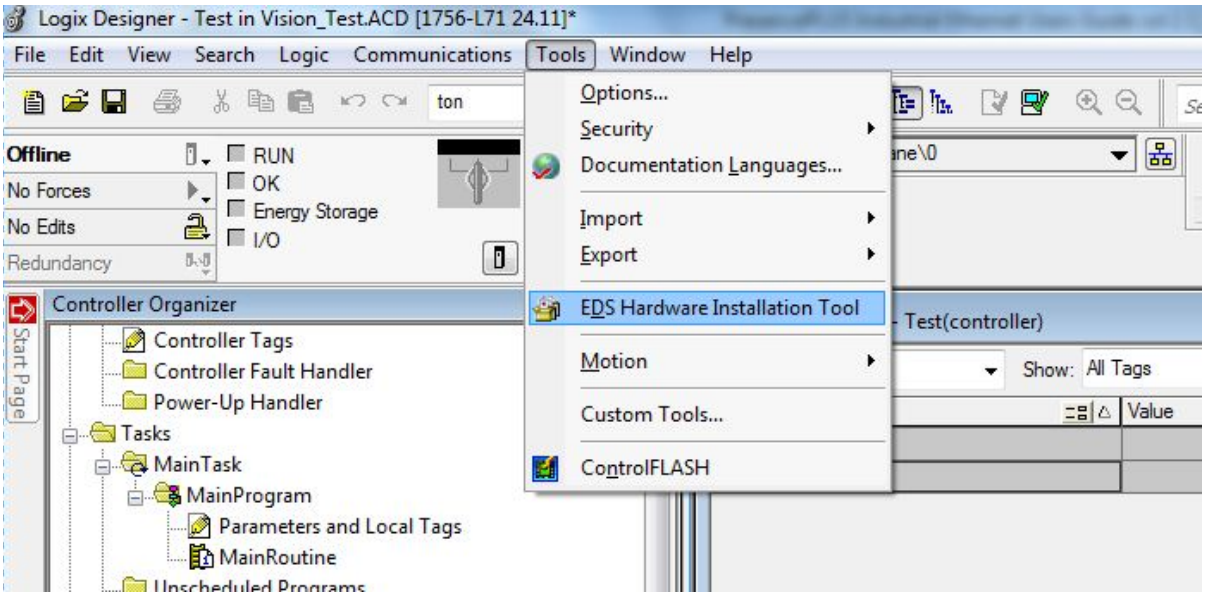


Chapter 2: EtherNet/IP

2.1 EtherNet/IP Configuration Using EDS File

Here is an example of using the SC22 EDS file to create a connection on a ControlLogix PLC.

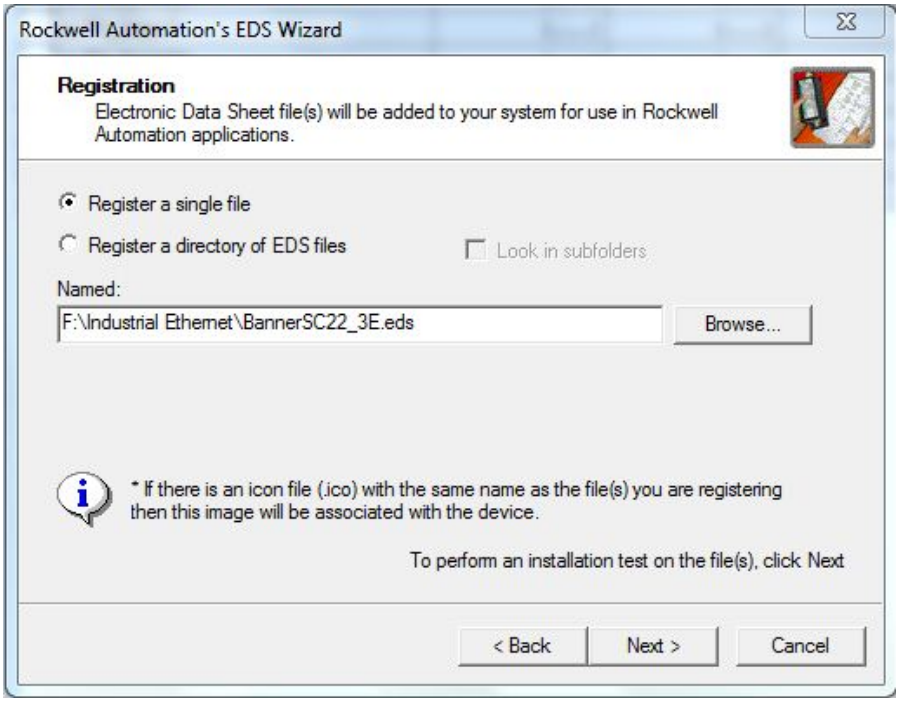
1. First we use the EDS Hardware Installation Tool to register the EDS file.



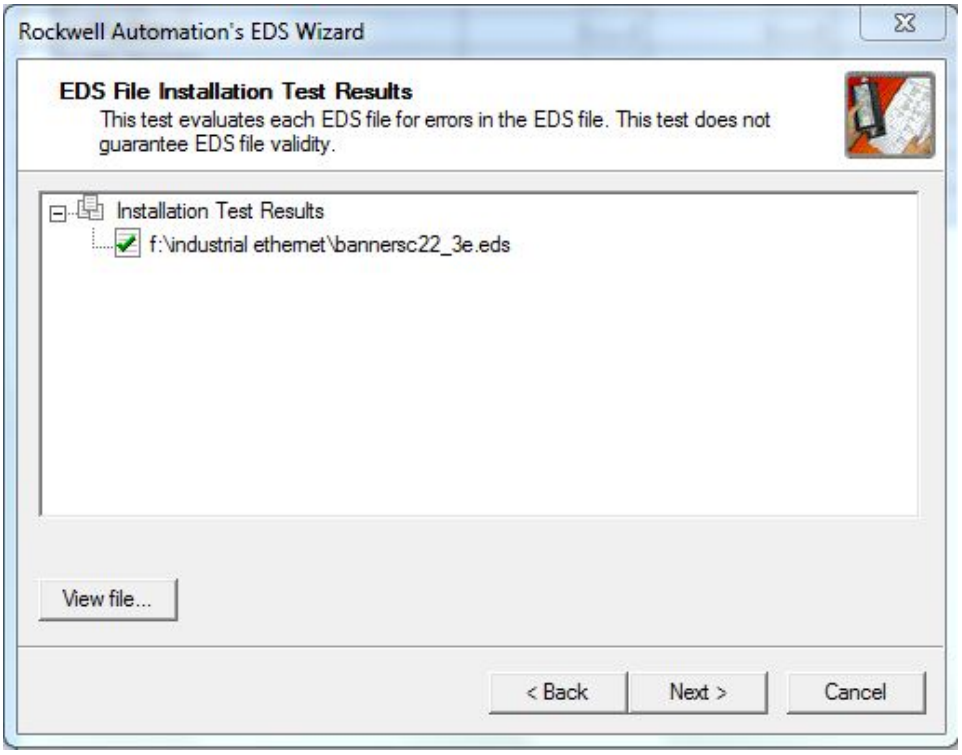
2. Click Next

3. Choose the "Register and EDS file(s)" option

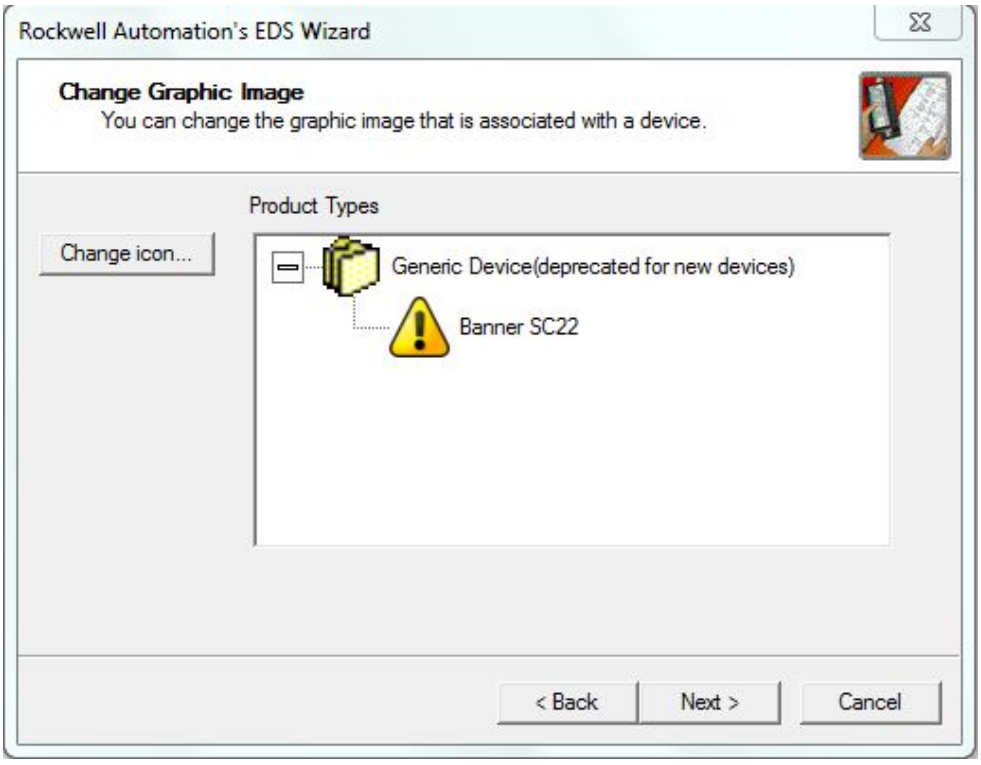
4. Browse to find the EDS file, then click Next.



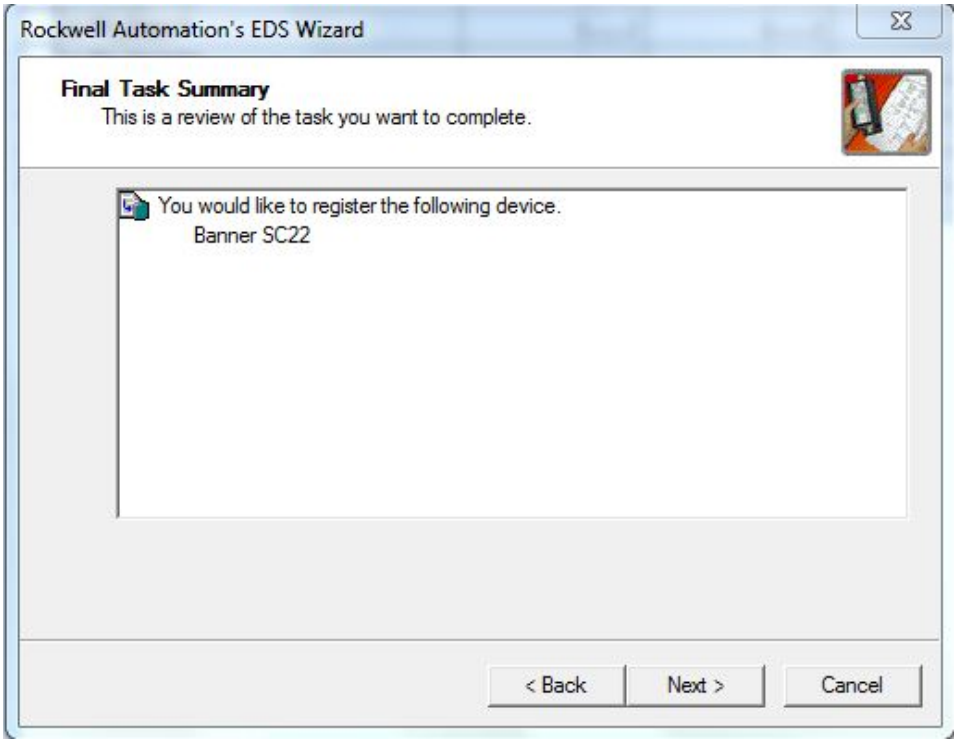
5. The file has been tested and can be registered. Click Next.



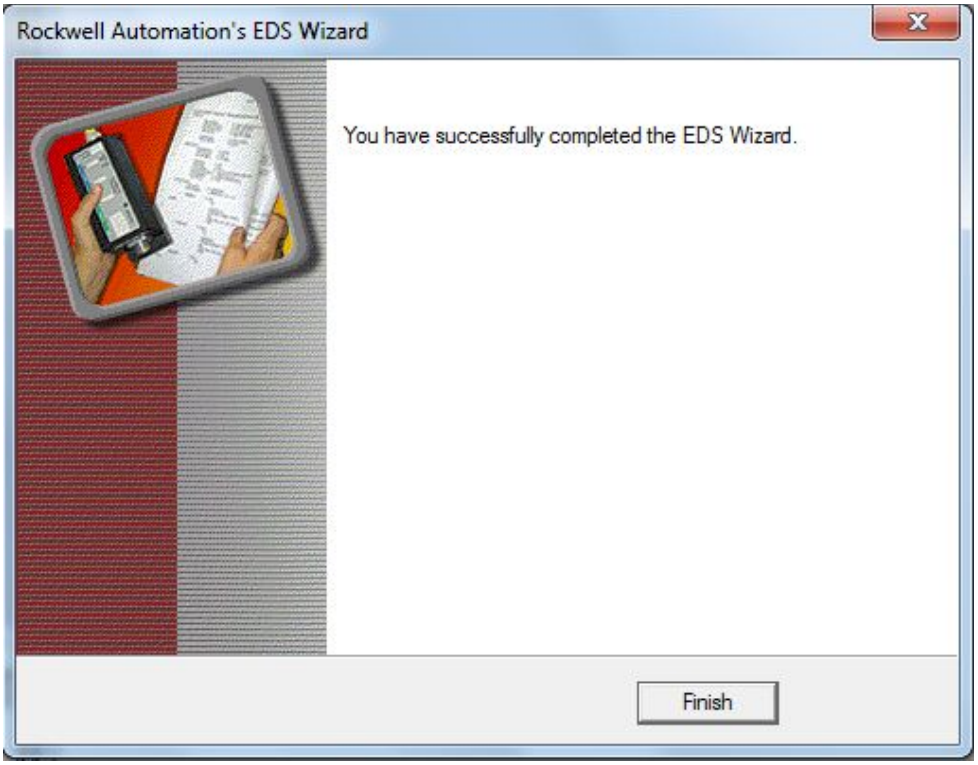
6. Here is the icon associated with the EDS file. Click Next.



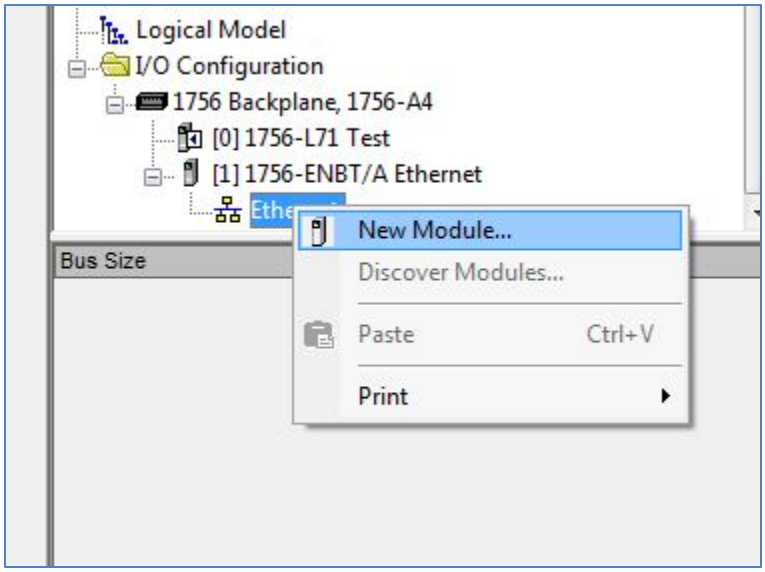
7. Everything looks good. Click Next to register this EDS file.



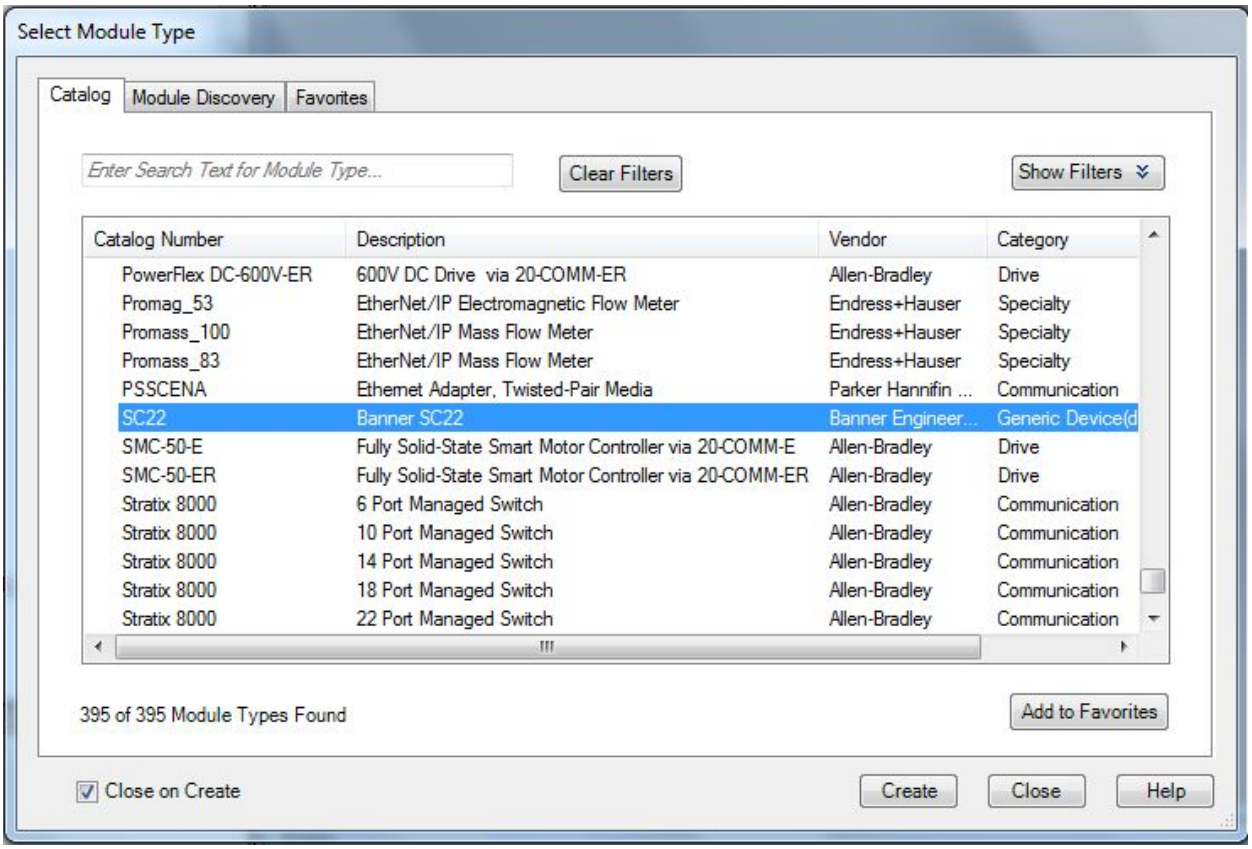
8. Complete! Click Finish. The EDS file is now registered in the Rockwell software.



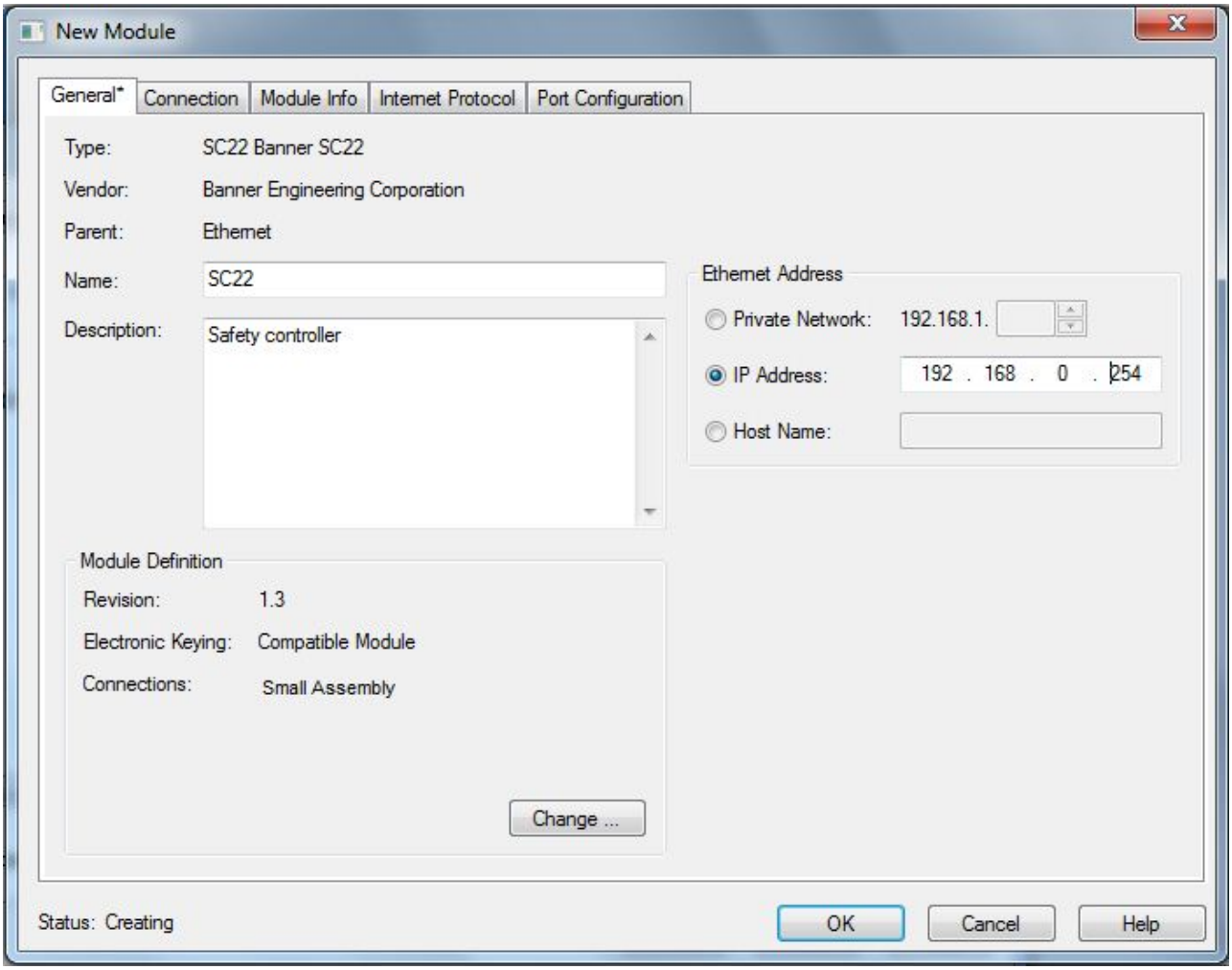
9. Now we will make a new module using the EDS file. Right click on the PLC's Ethernet adapter and choose "New Module".



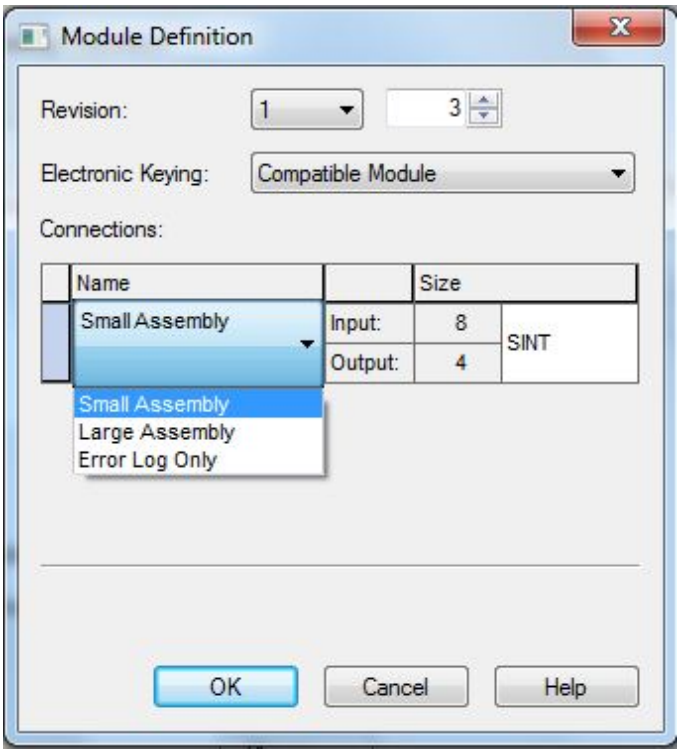
10. From the list, locate "SC22" then click Create.



11. Fill in a name, optional description, and IP address for the safety controller. Then click the "Change" button in the Module Definition box.



12. In the Module Definition window you can select which type of connection you'd like to create. Each of the menu items in the Connections window stands for a fixed grouping of Input and Output Assembly Instances, as defined here. See section 2.3 for more information on the Assembly Instances.

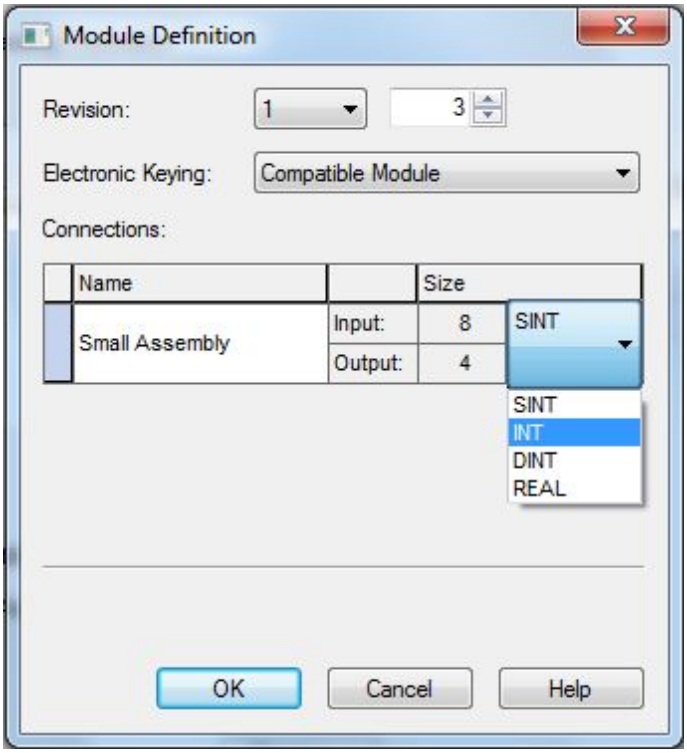


Small Assembly-
O→T PLC Output/Safety Controller Input Assembly 112 (0x70), size 2 16-bit registers
T→O PLC Input/Safety Controller Output Assembly 100 (0x64), size 4 16-bit registers

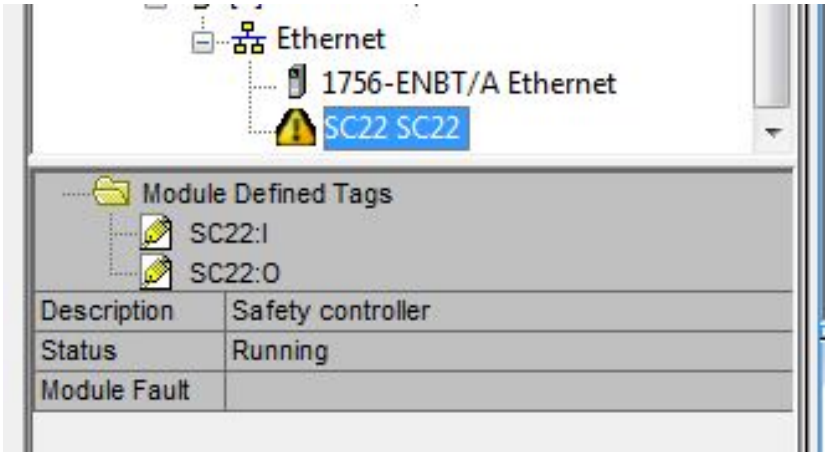
Large Assembly-
O→T PLC Output/Safety Controller Input Assembly 112 (0x70), size 2 16-bit registers
T→O PLC Input/Safety Controller Output Assembly 101 (0x65), size 42 16-bit registers

Error Log Only-
O→T PLC Output/Safety Controller Input Assembly 112 (0x70), size 2 16-bit registers
T→O PLC Input/Safety Controller Output Assembly 102 (0x66), size 140 16-bit registers

13. Make sure to select "INT" as the data type.



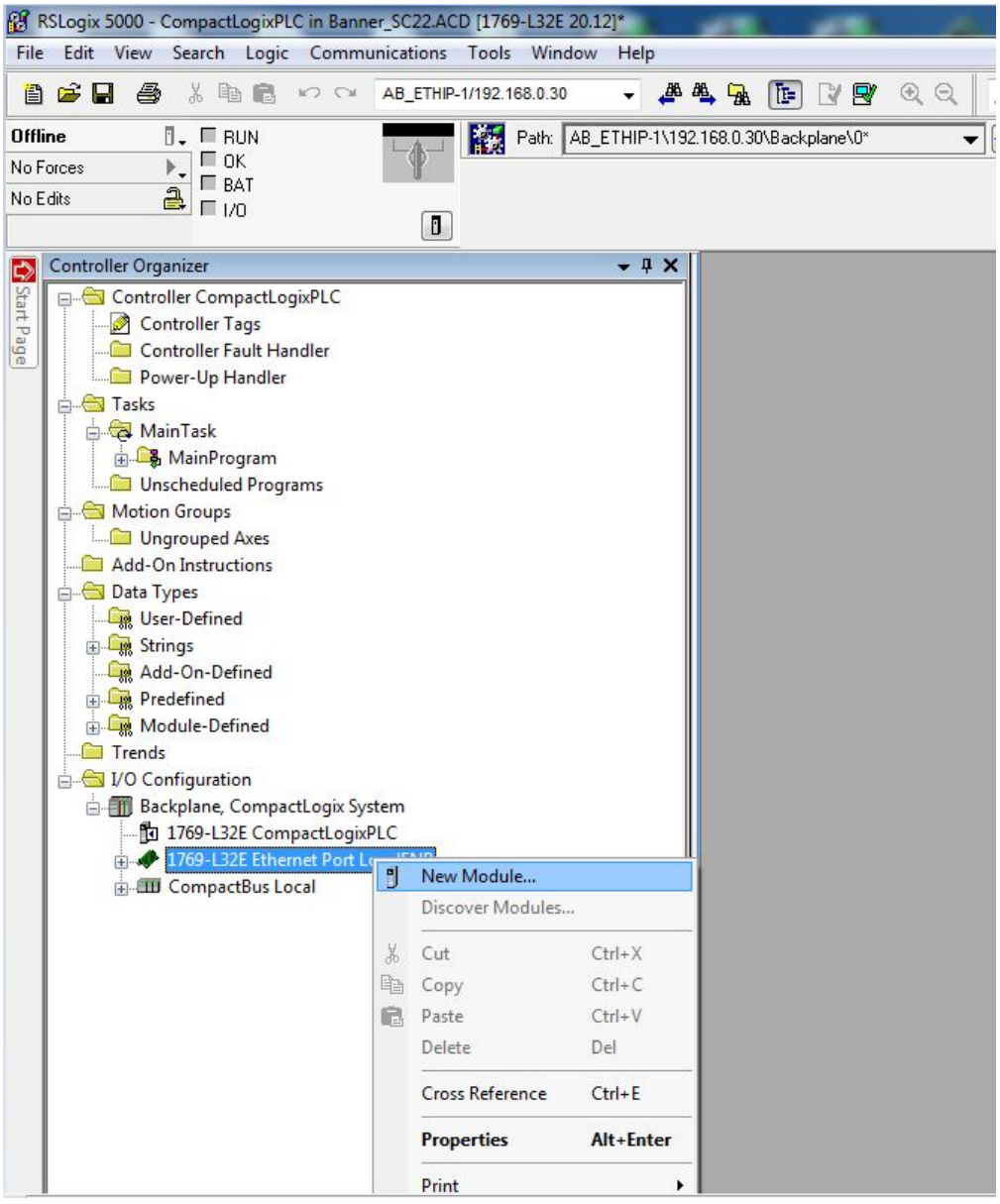
14. Click OK, then OK again and download the program to the PLC. The connection will look like that seen below.



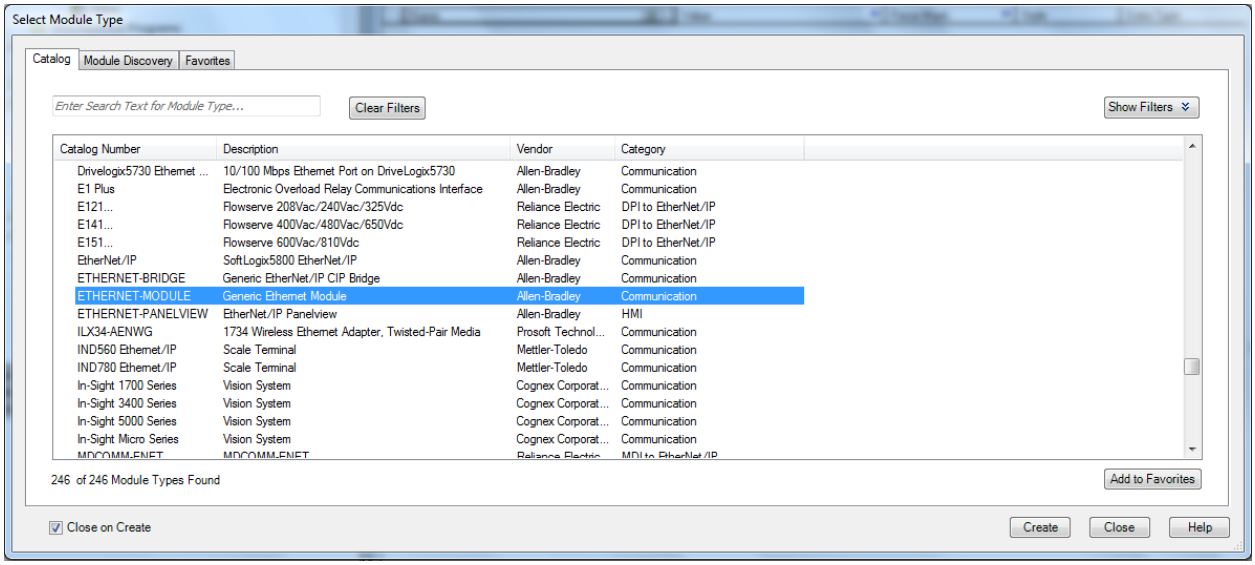
2.2 RSLogix5000 Configuration as Generic Ethernet Module

Another method for creating an implicit Class 1 configuration between a Safety Controller and a ControlLogix family PLC is to configure the Safety Controller as a "Generic Ethernet Module". The following is a sample setup:

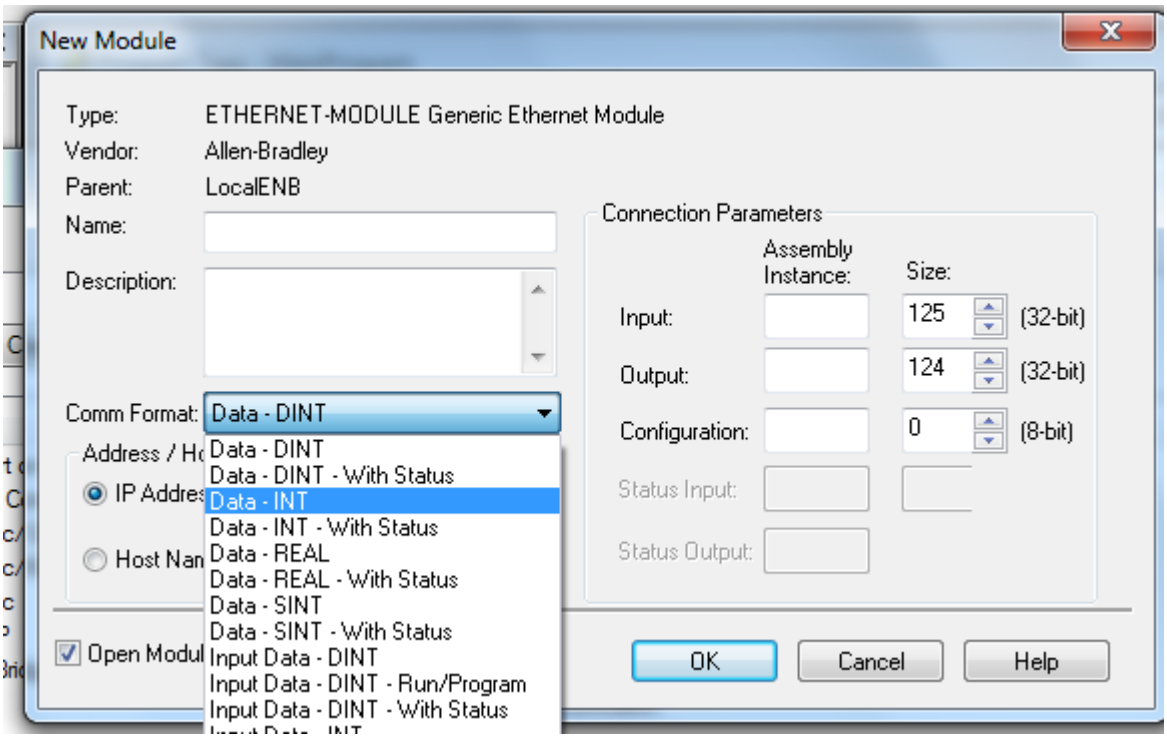
1. Add a module to the PLC's Ethernet card.



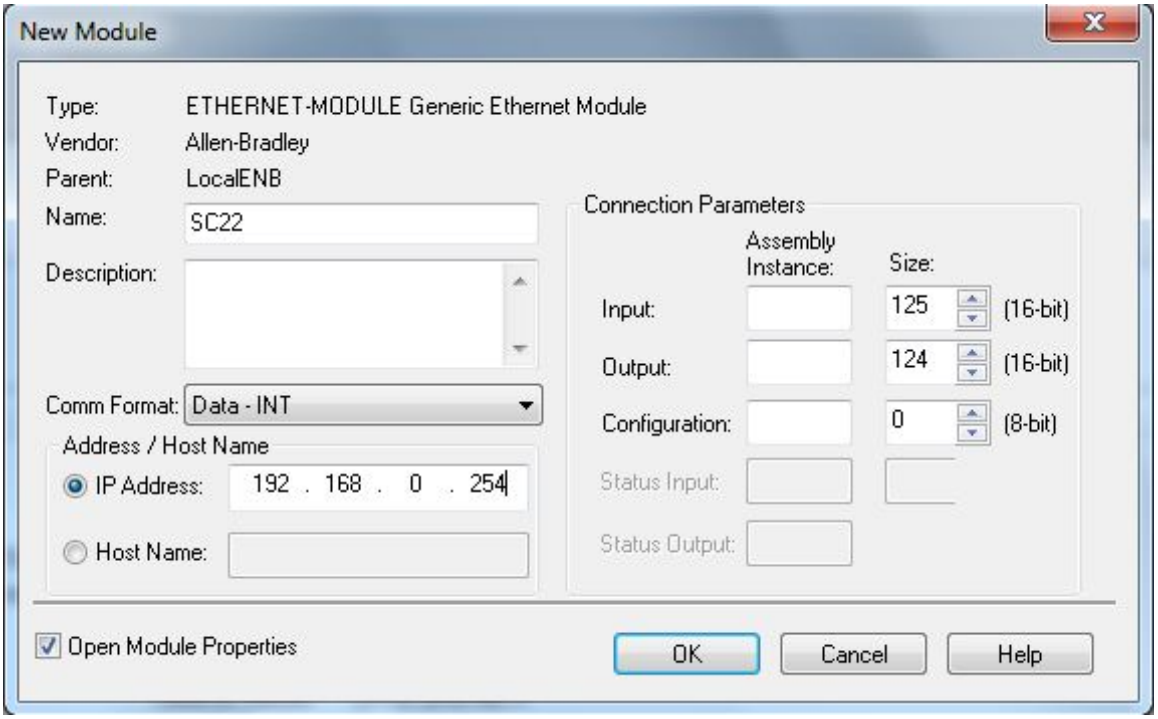
2. Select Module as "Generic Ethernet Module"



3. Change Comm Format to INT (default is DINT)

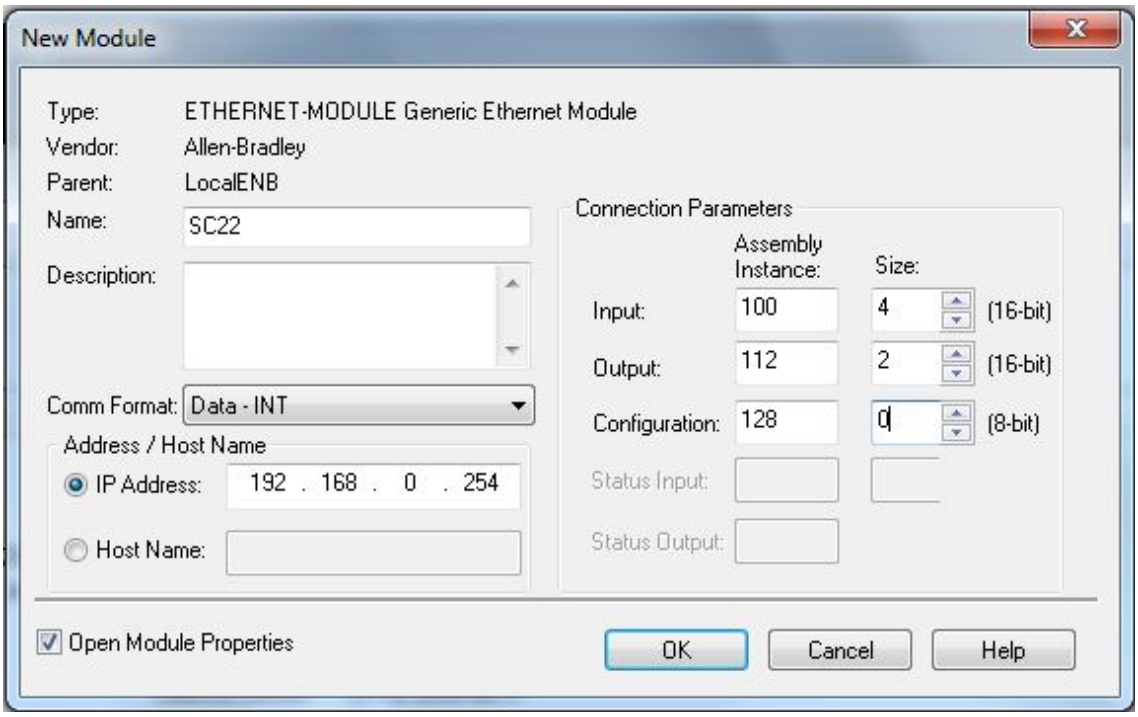


4. Add a module name and the IP address of the Safety Controller (default controller IP address is 192.168.0.254 with a subnet mask of 255.255.255.0).

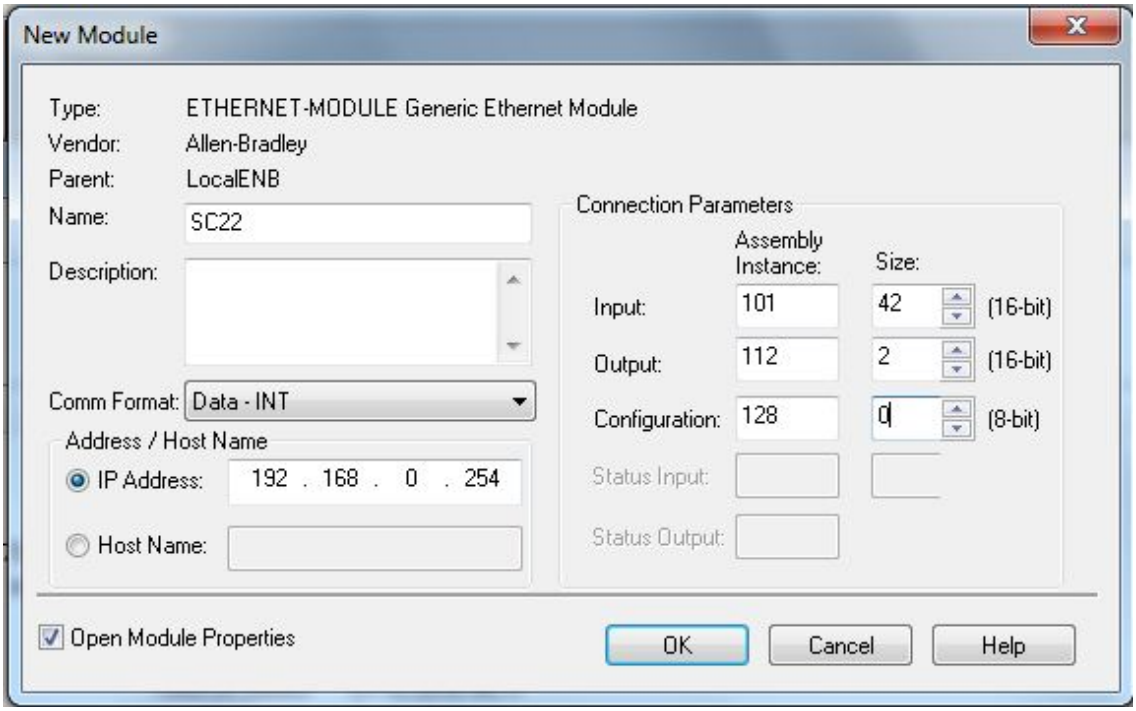


5. Choose one of three possible Assembly Object setups. See section 2.3.3 for more information on each choice.

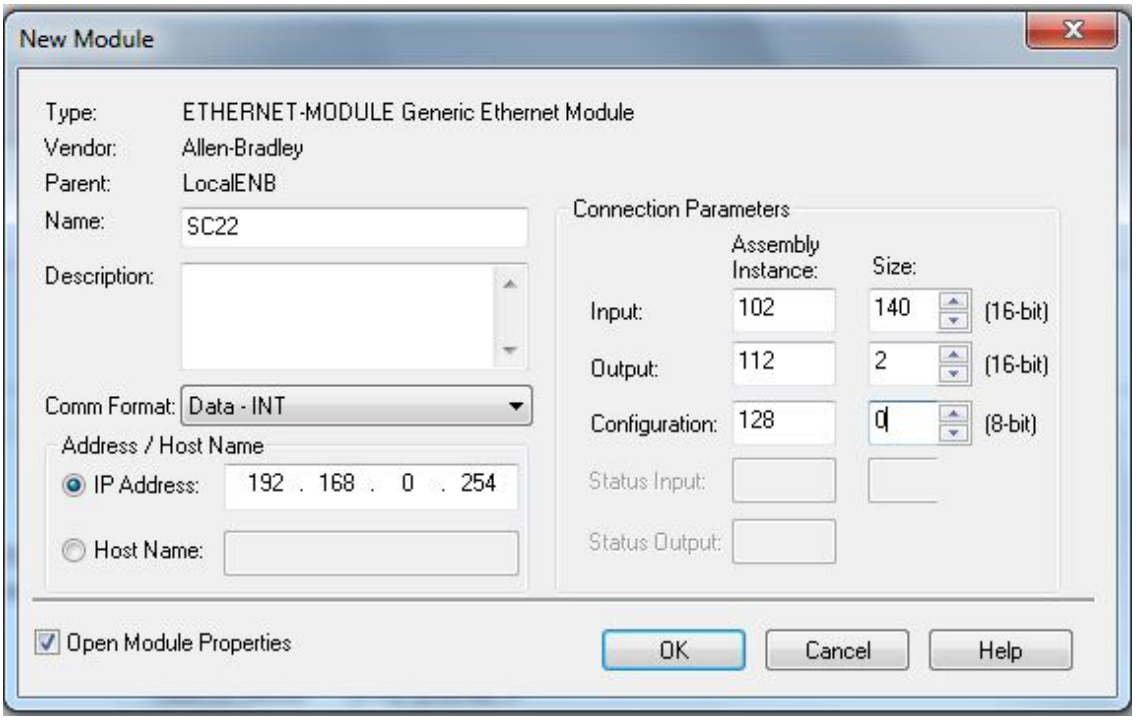
PLC Input Assembly 100 (0x64), size 4 words (16-bit)



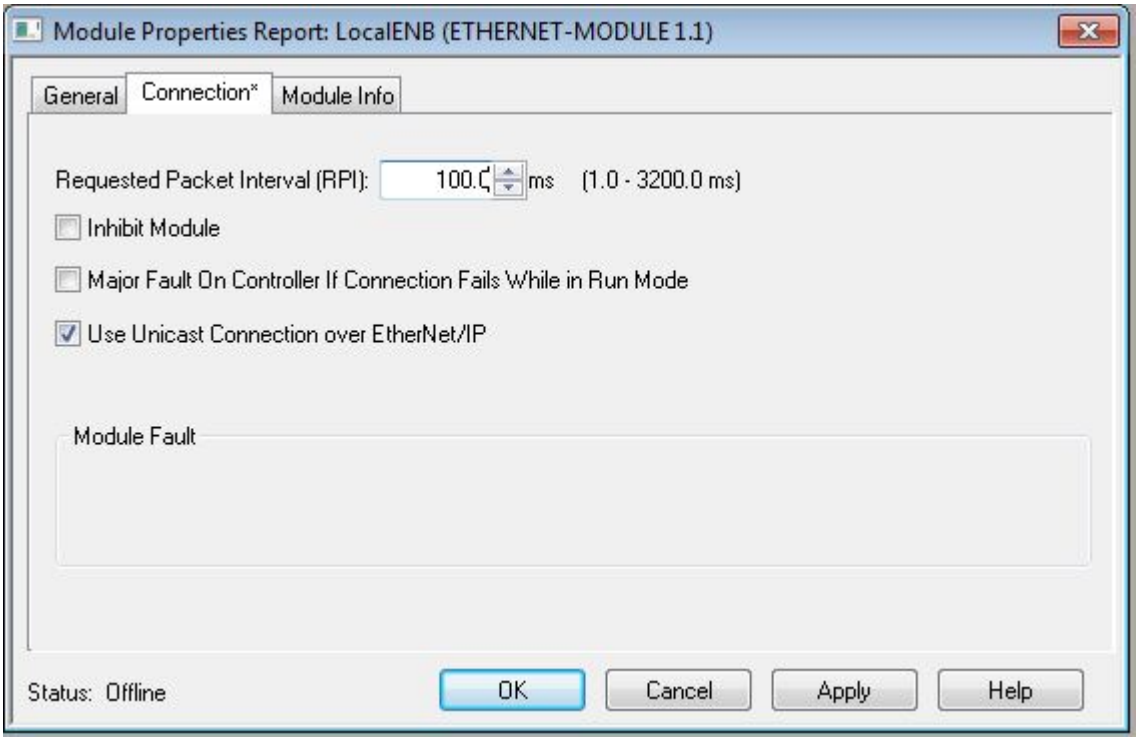
PLC Input Assembly 101 (0x65), size 42 words (16-bit)



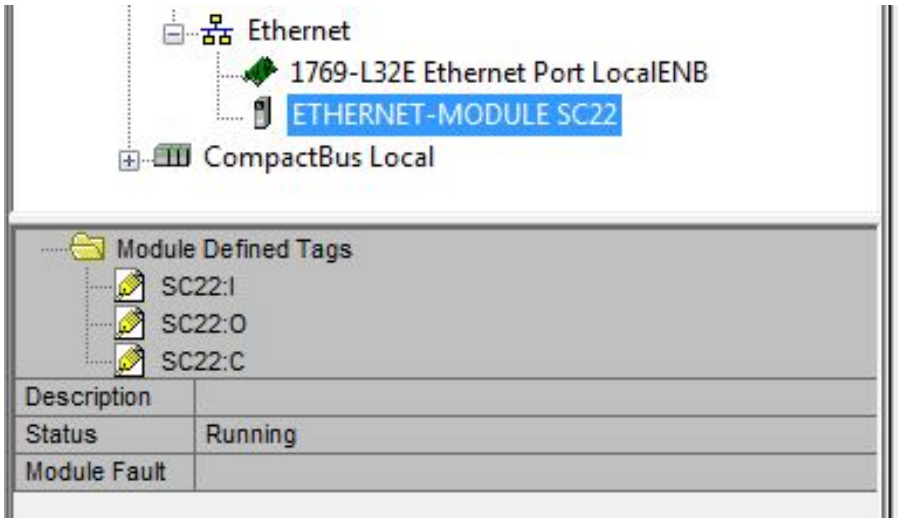
PLC Input Assembly 102 (0x66), size 140 words (16-bit) (Safety Controller Fault Log Only)



6. Set the Connection parameters: RPI and Unicast yes/no. Note that the recommended minimum RPI is 100msec.



7. If the module configuration was successful, the following information should be displayed:



I = Inputs to PLC (outputs from the Safety Controller)
O = Outputs from PLC (inputs to the Safety Controller- not used)
C = Configuration (not used)

8. Locate the memory map in the Controller Tags list. The 4 input words from Assembly Instance 100 are shown below as an example.

⊟	SC22:I	{...}	{...}		AB:ETHERNET_MODULE_
⊟	SC22:I.Data	{...}	{...}	Decimal	INT[4]
⊞	SC22:I.Data[0]	2		Decimal	INT
⊞	SC22:I.Data[1]	16384		Decimal	INT
⊞	SC22:I.Data[2]	0		Decimal	INT
⊞	SC22:I.Data[3]	0		Decimal	INT

In the example pictured above, we see that Virtual Outputs 2 and 31 are ON.
VO2 is word 0, bit 1 → 2^1 = 2; VO31 is word 1, bit 14 → 2^14 = 16,384

2.3 Assembly Objects

2.3.1 Safety Controller Inputs (Outputs from PLC) O→T

The Safety Controller does not use an Input Assembly Object. As some EtherNet/IP clients require one, use PLC Output Assembly Instance 112 (0x70) with a size of two 16-bit registers.

2.3.2 Safety Controller Configuration Assembly Object

The Safety Controller does not use a Configuration Assembly Object. As some EtherNet/IP clients require one, use PLC Configuration Assembly Instance 128 (0x80) with a size of zero 16-bit registers.

2.3.3 Safety Controller Outputs (Inputs to PLC) T→O

There are three choices for Safety Controller Output Assembly Objects. The first and smallest choice includes just information about Virtual Outputs and whether they have faults. The second choice adds to the Virtual Output information some more advanced data like the reason why each of the 3 safety outputs is off, and more descriptive fault information for the Virtual Outputs. The third choice of Output Assembly Object is used exclusively to access the Safety Controller's fault log. All three options are shown below.

PLC Input Assembly Instance 100 (0x64)- 4 Registers
This Assembly Instance includes only basic information about the status of all 32 Virtual Outputs.

PLC Input Assembly Instance 100 (0x64) – Safety Controller Outputs T→O

WORD #	WORD NAME	DATA TYPE
0	VO1 – VO16 (see Flags, section 2.5)	16-bit integer
1	VO17 – VO32 (see Flags, section 2.5)	16-bit integer
2	Fault bits for VO1 – VO16 (see Flags, section 2.5)	16-bit integer
3	Fault bits for VO17 – VO32 (see Flags, section 2.5)	16-bit integer

PLC Input Assembly Instance 101 (0x65)- 42 Registers

This Assembly Instance includes the status of all 32 Virtual Outputs plus advanced information about potential error codes and the status of the 3 safety outputs.

PLC Input Assembly Instance 101 (0x65) – Safety Controller Outputs T→O

WORD #	WORD NAME	DATA TYPE
0	VO1 – VO16 (see Flags, section 2.5)	16-bit integer
1	VO17 – VO32 (see Flags, section 2.5)	16-bit integer
2	Fault bits for VO1 – VO16 (see Flags, section 2.5)	16-bit integer
3	Fault bits for VO17 – VO32 (see Flags, section 2.5)	16-bit integer
4	SO1 is off due to: VO1 – VO16 (see Flags, section 2.5)	16-bit integer
5	SO1 is off due to: VO17 – VO32 (see Flags, section 2.5)	16-bit integer
6	SO2 is off due to: VO1 – VO16 (see Flags, section 2.5)	16-bit integer
7	SO2 is off due to: VO17 – VO32 (see Flags, section 2.5)	16-bit integer
8	SO3 is off due to: VO1 – VO16 (see Flags, section 2.5)	16-bit integer
9	SO3 is off due to: VO17 – VO32 (see Flags, section 2.5)	16-bit integer
10	VO1 Fault Index	16-bit integer
11	VO2 Fault Index	16-bit integer
12	VO3 Fault Index	16-bit integer
13	VO4 Fault Index	16-bit integer
14	VO5 Fault Index	16-bit integer
15	VO6 Fault Index	16-bit integer
16	VO7 Fault Index	16-bit integer
17	VO8 Fault Index	16-bit integer
18	VO9 Fault Index	16-bit integer
19	VO10 Fault Index	16-bit integer
20	VO11 Fault Index	16-bit integer
21	VO12 Fault Index	16-bit integer
22	VO13 Fault Index	16-bit integer
23	VO14 Fault Index	16-bit integer
24	VO15 Fault Index	16-bit integer
25	VO16 Fault Index	16-bit integer
26	VO17 Fault Index	16-bit integer
27	VO18 Fault Index	16-bit integer
28	VO19 Fault Index	16-bit integer
29	VO20 Fault Index	16-bit integer
30	VO21 Fault Index	16-bit integer
31	VO22 Fault Index	16-bit integer
32	VO23 Fault Index	16-bit integer
33	VO24 Fault Index	16-bit integer
34	VO25 Fault Index	16-bit integer
35	VO26 Fault Index	16-bit integer
36	VO27 Fault Index	16-bit integer
37	VO28 Fault Index	16-bit integer
38	VO29 Fault Index	16-bit integer
39	VO30 Fault Index	16-bit integer
40	VO31 Fault Index	16-bit integer
41	VO32 Fault Index	16-bit integer

Virtual Output Fault Index words

Note that not every Virtual Output has an associated Fault Index. See section 1.4 for definitions of fault words.

PLC Input Assembly Instance 102 (0x66)- 140 Registers

This Assembly Instance is used exclusively to access the fault log information on the Safety Controller. Note that this Assembly Instance contains no information about the status of the Virtual Outputs.

The Safety Controller can store 10 faults in the log. Fault #1 is the most recent fault while higher fault numbers represent successively older faults.

PLC Input Assembly Instance 102 (0x66) – Safety Controller Outputs T→O

WORD #	WORD NAME	DATA TYPE
0-1	Fault #1 Time Stamp	32-bit integer
2-9	Fault #1 Name of I/O or System	2-word length + 12-ASCII chars
10	Fault #1 Fault Index	16-bit integer
11	Fault #1 VO number	16-bit integer
12-13	*reserved*	16-bit integer
14-15	Fault #2 Time Stamp	32-bit integer
16-23	Fault #2 Name of I/O or System	2-word length + 12- ASCII chars
24	Fault #2 Fault Index	16-bit integer
25	Fault #2 VO number	16-bit integer
26-27	*reserved*	16-bit integer
28-29	Fault #3 Time Stamp	32-bit integer
30-37	Fault #3 Name of I/O or System	2-word length + 12- ASCII chars
38	Fault #3 Fault Index	16-bit integer
39	Fault #3 VO number	16-bit integer
40-41	*reserved*	16-bit integer
42-43	Fault #4 Time Stamp	32-bit integer
44-51	Fault #4 Name of I/O or System	2-word length + 12- ASCII chars
52	Fault #4 Fault Index	16-bit integer
53	Fault #4 VO number	16-bit integer
54-55	*reserved*	16-bit integer
56-57	Fault #5 Time Stamp	32-bit integer
58-65	Fault #5 Name of I/O or System	2-word length + 12- ASCII chars
66	Fault #5 Fault Index	16-bit integer
67	Fault #5 VO number	16-bit integer
68-69	*reserved*	16-bit integer
70-71	Fault #6 Time Stamp	32-bit integer
72-79	Fault #6 Name of I/O or System	2-word length + 12- ASCII chars
80	Fault #6 Fault Index	16-bit integer
81	Fault #6 VO number	16-bit integer
82-83	*reserved*	16-bit integer
84-85	Fault #7 Time Stamp	32-bit integer
86-93	Fault #7 Name of I/O or System	2-word length + 12- ASCII chars
94	Fault #7 Fault Index	16-bit integer
95	Fault #7 VO number	16-bit integer
96-97	*reserved*	16-bit integer
98-99	Fault #8 Time Stamp	32-bit integer
100-107	Fault #8 Name of I/O or System	2-word length + 12- ASCII chars

108	Fault #8 Fault Index	16-bit integer
109	Fault #8 VO number	16-bit integer
110-111	*reserved*	16-bit integer
112-113	Fault #9 Time Stamp	32-bit integer
114-121	Fault #9 Name of I/O or System	2-word length + 12- ASCII chars
122	Fault #9 Fault Index	16-bit integer
123	Fault #9 VO number	16-bit integer
124-125	*reserved*	16-bit integer
126-127	Fault #10 Time Stamp	32-bit integer
128-135	Fault #10 Name of I/O or System	2-word length + 12-ASCII chars
136	Fault #10 Fault Index	16-bit integer
137	Fault #10 VO number	16-bit integer
138-139	*reserved*	16-bit integer

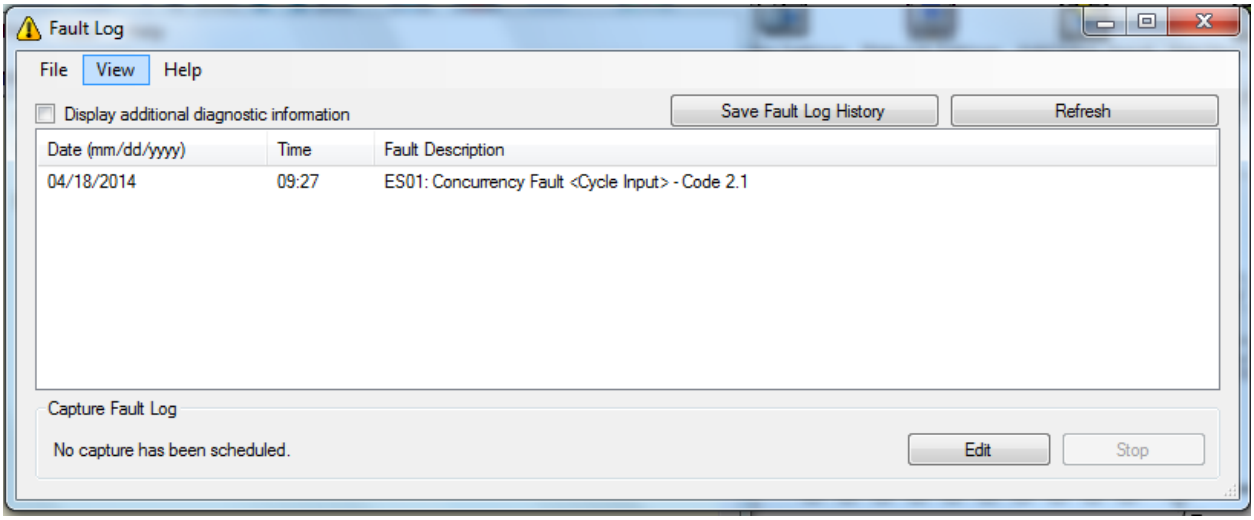
Fault Time Stamp
This is a 32-bit integer representation of the fault time, listed as the number of seconds since powering up the Safety Controller when the fault occurred.

Name of I/O or System
This section gives more information as to the source of the fault. See section 1.3 below for more information.

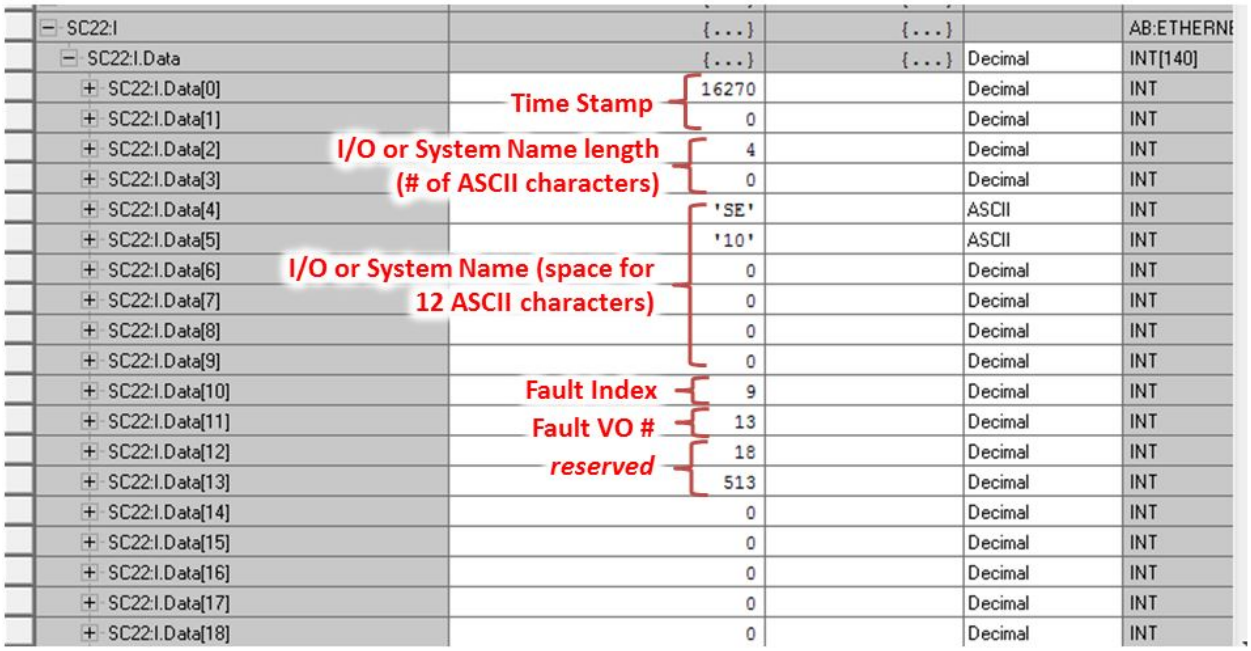
Fault Index
See Chapter 5 for Fault Index Number definitions. The Fault Index Numbers are a 16-bit representation of the floating point Fault Codes.

Fault VO Number
Which of the 32 Virtual Outputs, written as a 16-bit integer, is related to the fault. Note that not every Fault will have a related VO number.

2.4 Fault Examples



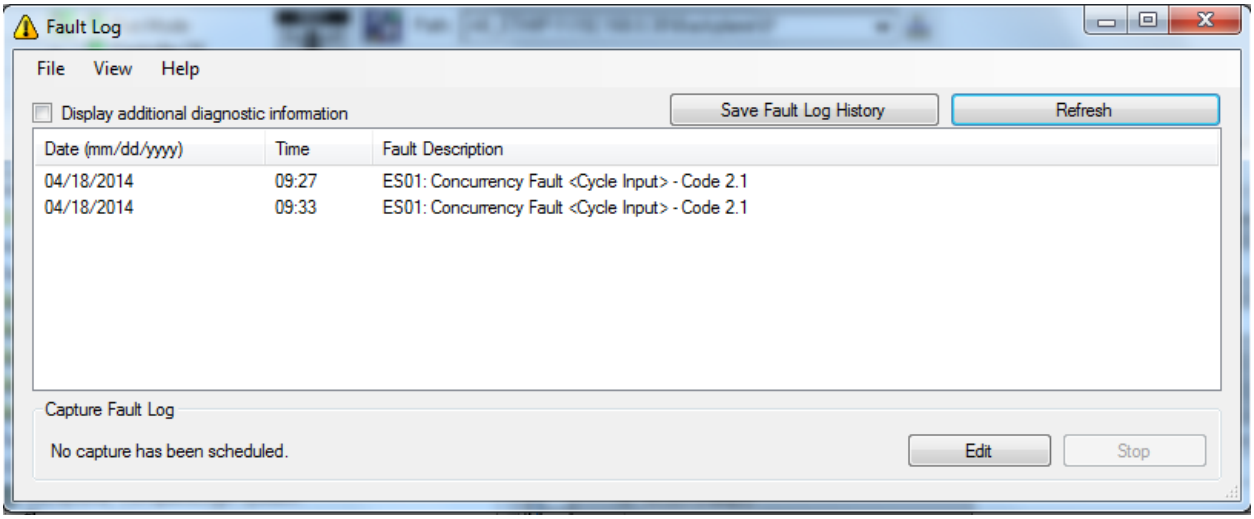
Here is a fault from the SC22-3E software fault log.



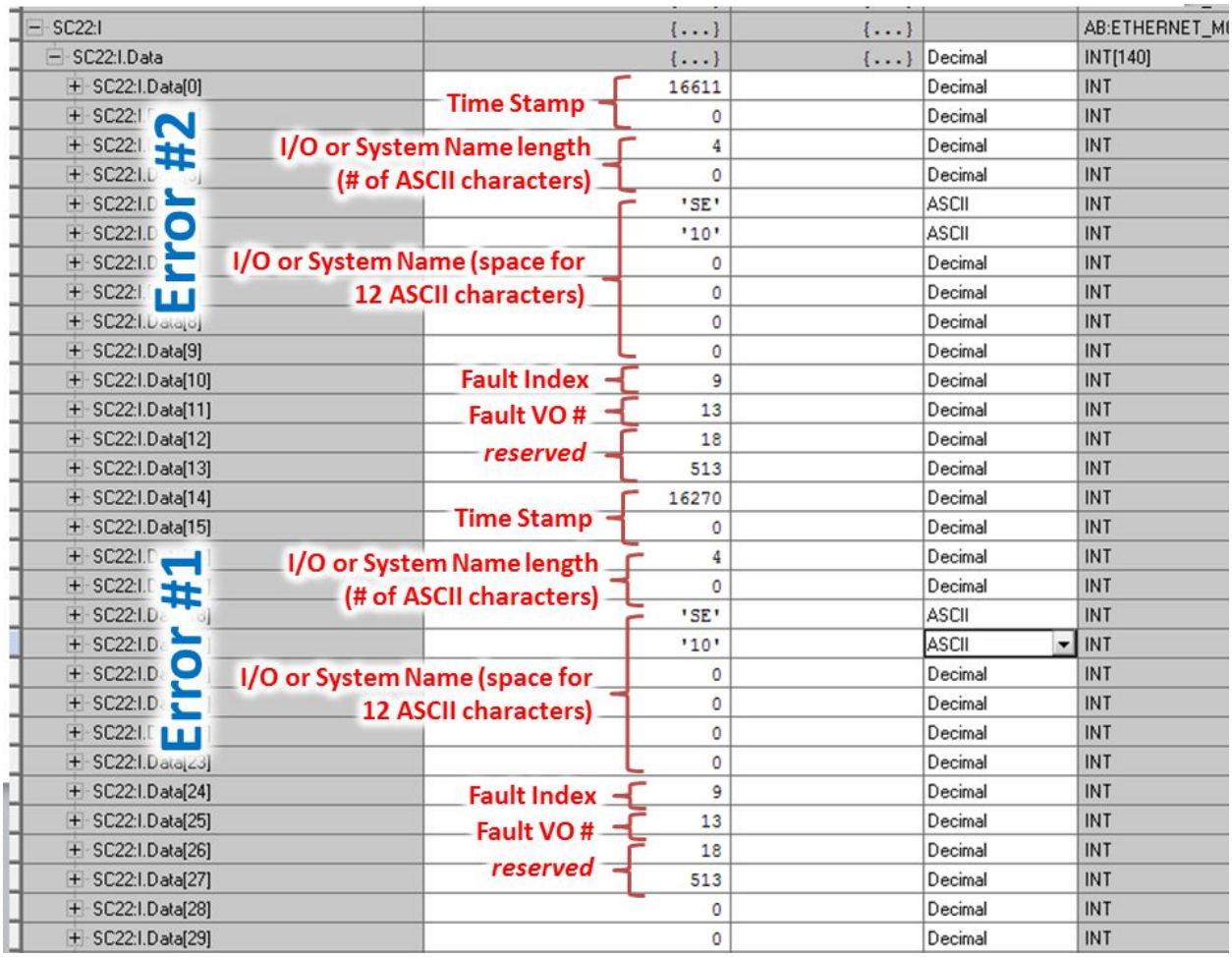
And the same fault as seen in the EtherNet/IP registers.

Note the ControlLogix string format, wherein the ASCII characters are shown, two per register, "backwards". "ES01" becomes "SE" in register 4, followed by "10" in register 5.

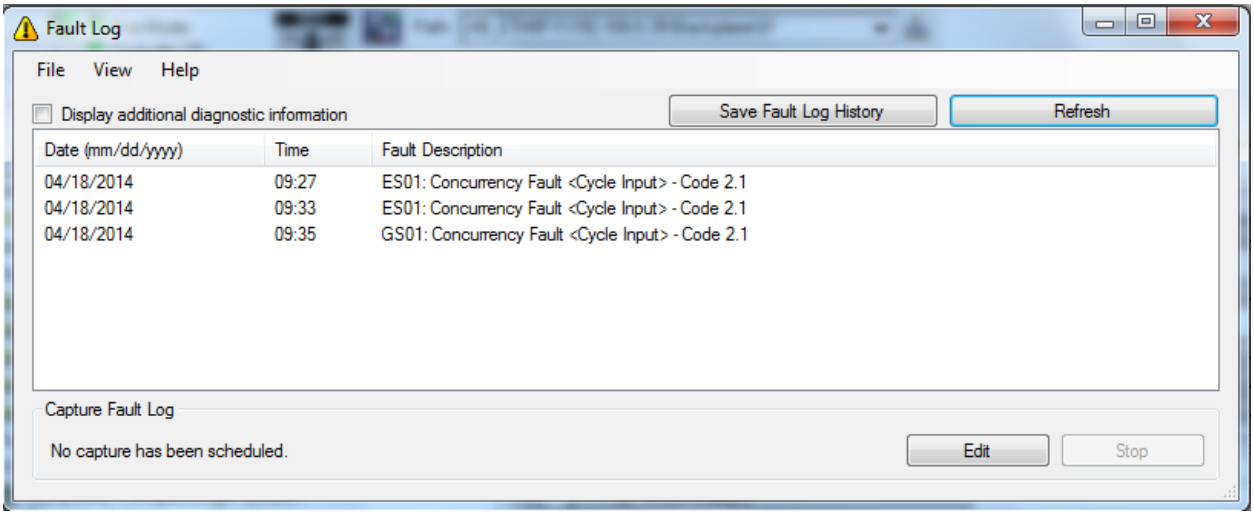
Fault Index #9 = Fault Code 2.1 (Concurrency Fault). See Chapter 5 for Fault Index definitions.



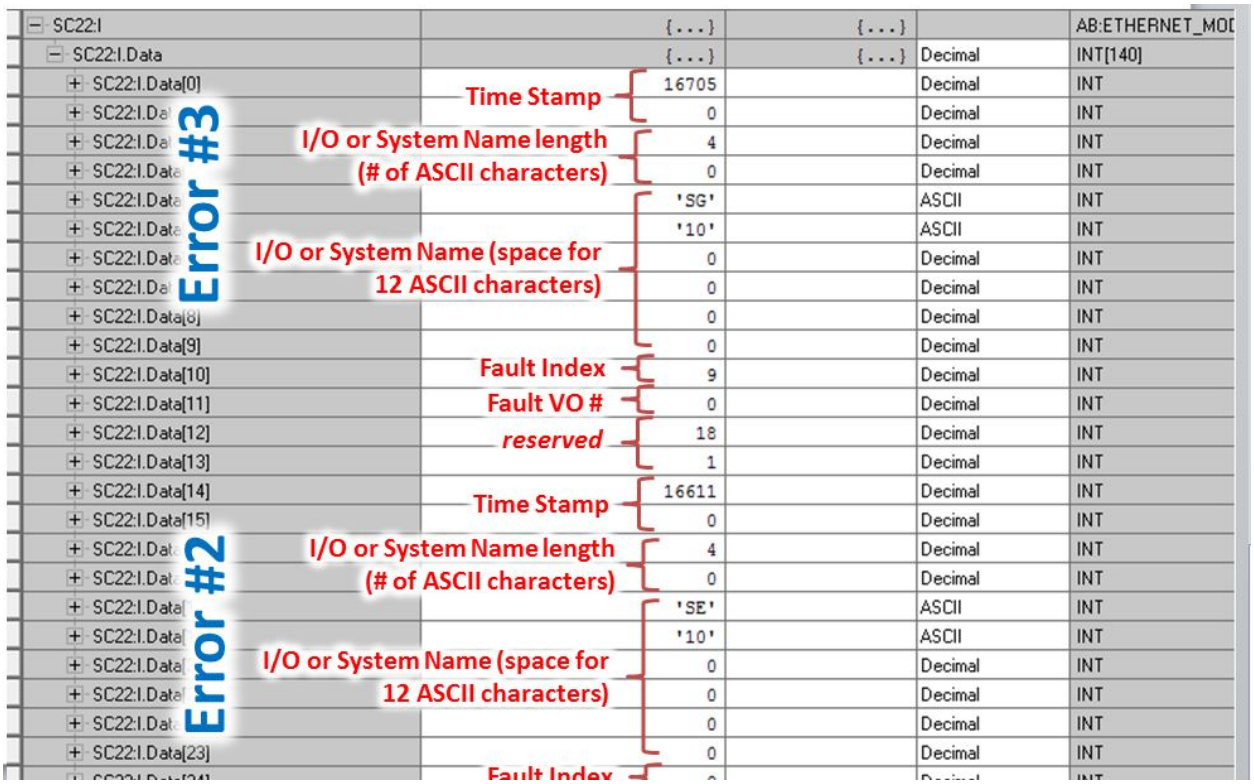
Now we see two faults in the SC22-3E software fault log.



And the two faults in in the PLC registers.



Here is a third fault in the log.



Note how the newest in time error appears at the top of the fault log registers, pushing the other errors down the list.

2.5 Flags

Words 0 through 3, defined below, appear as the first 4 words in both Assembly Instance 100 and 101. Words 4-9 appear only in Assembly Instance 101.

Word #0, Virtual Output 1-16, Bit Position

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO16	VO15	VO14	VO13	VO12	VO11	VO10	VO9	VO8	VO7	VO6	VO5	VO4	VO3	VO2	VO1

Word #1, Virtual Output 17-32, Bit Position

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

Word #2, Fault Flag bits for Virtual Output 1-16, Bit Position

Note that not every Virtual Output has a defined Fault Flag.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

Word #3, Virtual Output 17-32 Fault Flag, Bit Position

Note that not every Virtual Output has a defined Fault Flag.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

Registers 4 through 9 are used, in blocks of two registers each, to communicate information about the three safety outputs. Each two register block covers all 32 virtual output bits that could have turned the safety output in question off. The first block of two such registers is seen below as an example.

Register #4, Safety Output 1 is off due to: Virtual Output 1-16, Bit Position

Note that not every Virtual Output need be configured to turn off Safety Output 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO16	VO15	VO14	VO13	VO12	VO11	VO10	VO9	VO8	VO7	VO6	VO5	VO4	VO3	VO2	VO1

Register #5, Safety Output 1 is off due to: Virtual Output 17-32, Bit Position

Note that not every Virtual Output need be configured to turn off Safety Output 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

The pattern then repeats. Registers 6-7 are linked to Safety Output 2, registers 8-9 are for Safety Output 3.

2.6 RSLogix5000 Configuration (Explicit Messaging)

The XS/SC26-2E supports a number of different Explicit Messaging connections. In addition to the Assembly Instances from the previous section, there are some extra Assembly Instances that can only be accessed via Explicit Messaging.

2.6.1 Choices for Explicit Message Connections

IO Assembly Instances

To get a copy of one of the Assembly Instances from section 2.2.3, use Service Type 14 (Get Attribute Single, hex 0E), Class 4, Instance 100 (0x64) or 101 (0x65) or 102 (0x66), Attribute 3. A successful Explicit Message of this type will return the appropriate Assembly Instance as show in section 2.2.3. See section 2.5.2.1 for an example of this type of connection.

Virtual Output Status

To get the current status of all 32 Virtual Outputs, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x64, Instance 1, Attribute 1. A successful Explicit Message of this type will return a 32-bit integer representing the status of VO1 through VO32. See section 2.5.2.2 for an example of this type of connection.

Virtual Output Fault Bits

To get the current status of all 32 Virtual Output Fault Bits, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x65, Instance 1, Attribute 1. A successful Explicit Message of this type will return a 32-bit integer representing the status of the Fault Bits for VO1 through VO32.

Reason for Safety Output 1 OFF

To learn which Virtual Outputs are responsible for Safety Output 1 being off, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x66, Instance 1, Attribute 1. A successful Explicit Message of this type will return a 32-bit integer representing the array of possible answers, VO1 through VO32.

Reason for Safety Output 2 OFF

To learn which Virtual Outputs are responsible for Safety Output 2 being off, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x68, Instance 1, Attribute 1. A successful Explicit Message of this type will return a 32-bit integer representing the array of possible answers, VO1 through VO32.

Reason for Safety Output 3 OFF

To learn which Virtual Outputs are responsible for Safety Output 3 being off, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x69, Instance 1, Attribute 1. A successful Explicit Message of this type will return a 32-bit integer representing the array of possible answers, VO1 through VO32.

Individual Fault Index Values

To get a specific Fault Index Value for one of the 32 Virtual Outputs, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x6A, Instance 1-32 (choose one), Attribute 1. A successful Explicit Message of this type will return a single 16-bit register representing the Fault Index value for one of the Virtual Outputs.

Individual Fault Log Entry

To get a specific entry from the 10 entry Fault Log, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x6C, Instance 1, Attribute 1-10 (choose one). A successful Explicit Message of this type will return a single 14 register entry from the fault log, as defined below. Note that Attribute = 1 references the most recent entry in the error log, while Attribute = 10 is the oldest entry.

WORD #	WORD NAME	DATA TYPE
0-1	Fault #1 Time Stamp	32-bit integer
2-9	Fault #1 Name of I/O or System	2-word length + 12-ASCII chars
10	Fault #1 Fault Index	16-bit integer
11	Fault #1 VO number	16-bit integer
12-13	*reserved*	16-bit integer

System Information

Some system information can be accessed using Service Code 14 (Get Attribute Single, hex 0E), Class 0x6D, Instance 1, Attribute 1-14 (choose one, see table below). A successful Explicit Message of this type will return the system information seen below (size and data type vary). See section 2.6.2.3 for an example of this type of connection.

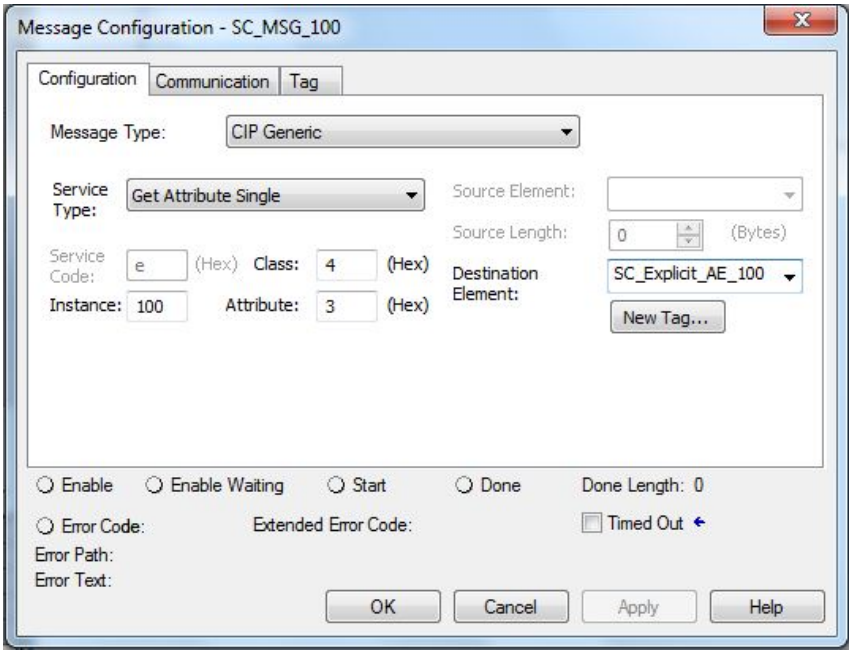
Attribute	System value	Data Type
1	Seconds Since Boot	32-bit integer
2	Operating Mode	16-bit integer
3	Config CRC	16-bit integer
4-8	*reserved*	16-bit integer
9	Reconfig Time	32-bit integer
10	Date Code	2-word length + 8-ASCII chars
11	Configuration Name	2-word length + 16-ASCII chars
12	Device Name	2-word length + 20-ASCII chars
13	Device Model Number	2-word length + 22-ASCII chars
14	Device Serial Number	32-bit integer

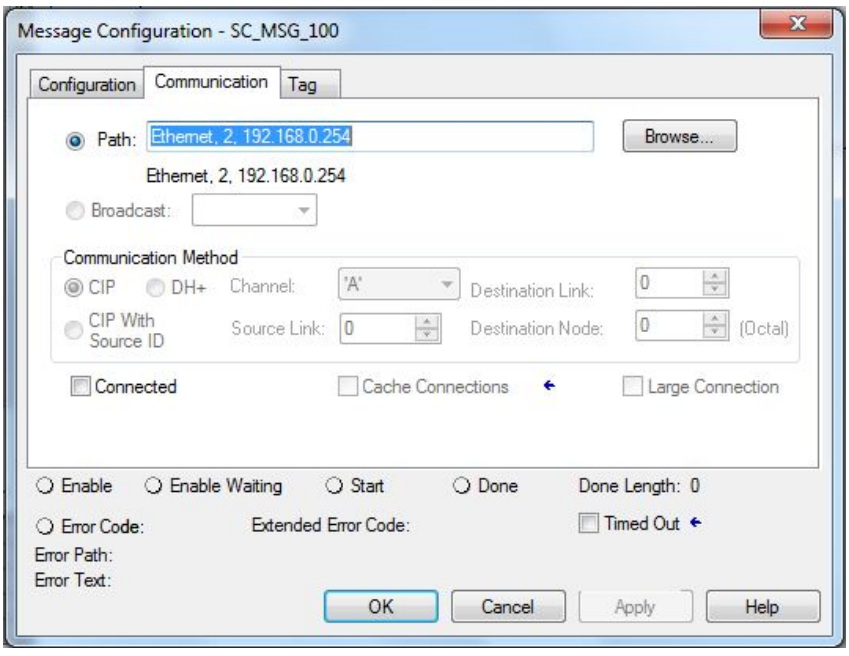
2.6.2 Examples of Explicit Message Connections

2.6.2.1 Assembly Instance 0x64 (100)

To get the 100 (0x64) Assembly Instance, use Service Type 14 (Get Attribute Single, hex 0E), Class 4, Instance 100, Attribute 3. A successful Explicit Message of this type will return all 4 registers of the 100 (0x64) Assembly Instance, as defined in section 2.2.3.

Here is the MSG command for this explicit message.





Here is the user defined array (called **SC_Explicit_AE_100**) showing all 4 registers.

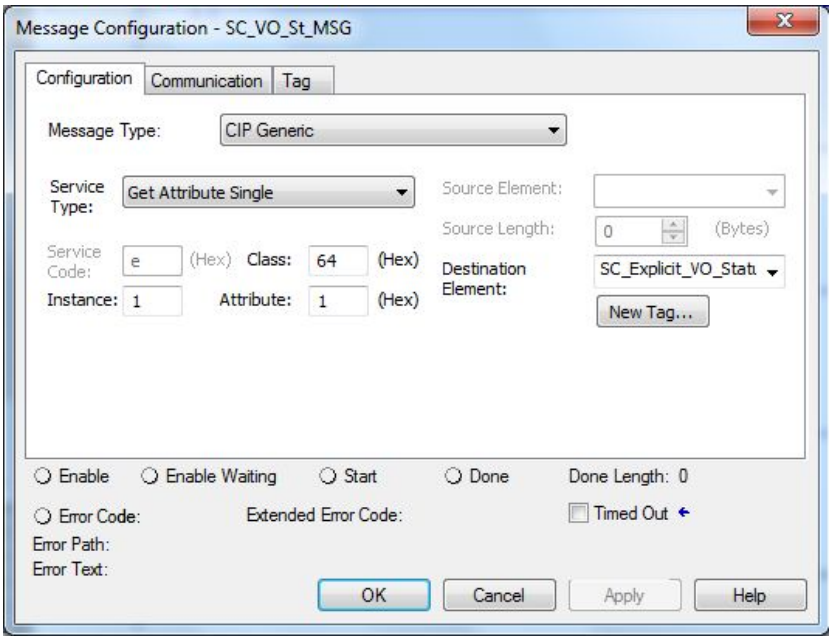
SC_Explicit_AE_100	{...}	{...}	Decimal	INT[4]
SC_Explicit_AE_100[0]	4097		Decimal	INT
SC_Explicit_AE_100[1]	256		Decimal	INT
SC_Explicit_AE_100[2]	0		Decimal	INT
SC_Explicit_AE_100[3]	0		Decimal	INT

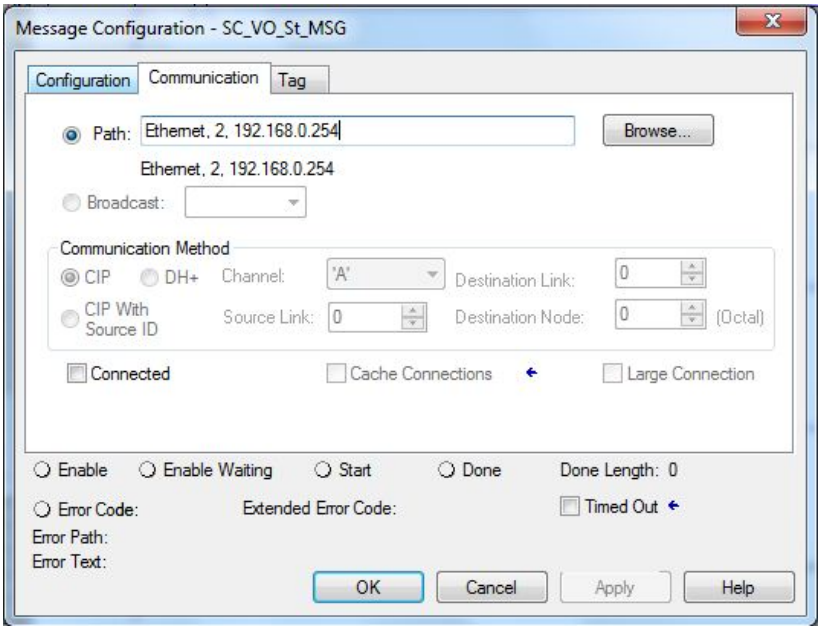
In this example data, we can see that VO1, VO13, and VO25 are currently ON. VO1 is word 0, bit 0 → 2^0 = 1. VO13 is word 0, bit 12 → 2^12 = 4096. 1 + 4096 = 4097. VO25 is word 1, bit 8 → 2^8 = 256.

2.6.2.2 Virtual Output Status

To get the current status of all 32 Virtual Outputs, use Service Type 14 (Get Attribute Single, hex 0E), Class 0x64, Instance 1, Attribute 1. A successful Explicit Message of this type will return a 32-bit integer representing the status of VO1 through VO32.

Here is the MSG command for this explicit message.



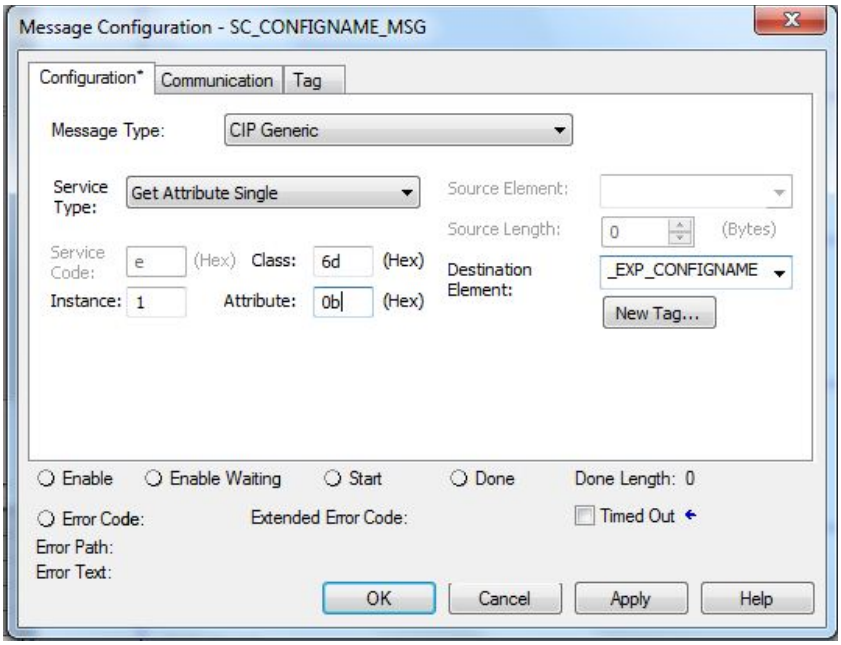


Here is the user defined array (called **SC_Explicit_VO_Status**) showing all the 32-bit integer.

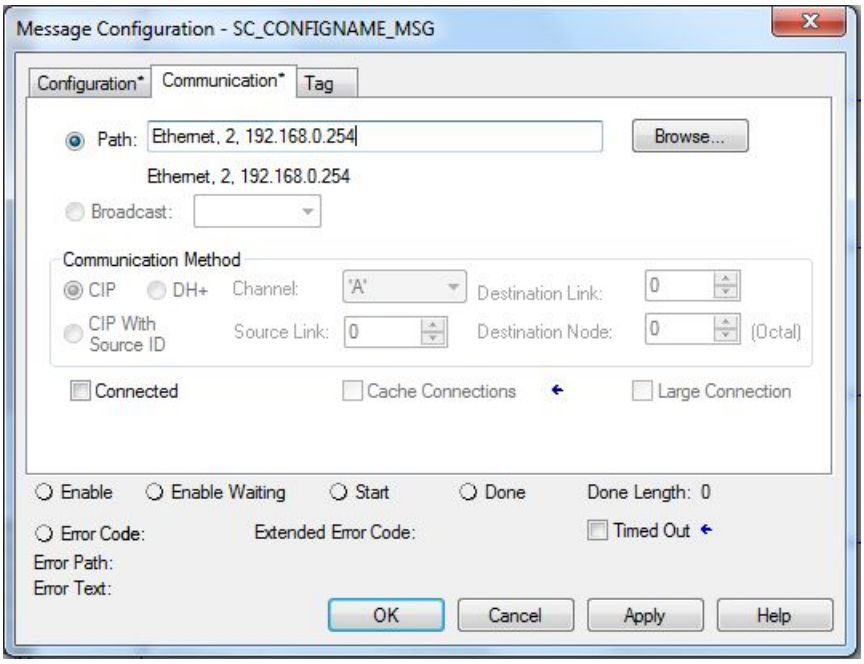
In this case, we see that VO1 and VO24 are ON → 2^0 = 1, 2^24 = 16777216. 1 + 16777216 = 16777217.

2.6.2.3 System Information

Some system information can be accessed using EtherNet/IP Explicit Messages. One such piece of data is the Configuration Name from the safety controller. To get this information, use ServiceType 14 (Get Attribute Single, hex 0E), Class 0x6D, Instance 1, Attribute 11. A successful Explicit Message of this type will return the 32-bit length and ASCII string comprising the SC22's Configuration Name.



Attribute 11 in Hex is 0x0B.



Here is the user defined array (called **SC_EXP_CONFIGNAME**) showing all the information.

SC_EXP_CONFIGNAME	{...}	{...}	Decimal	INT[10]
SC_EXP_CONFIGNAME[0]	8		Decimal	INT
SC_EXP_CONFIGNAME[1]	0		Decimal	INT
SC_EXP_CONFIGNAME[2]	'EN'		ASCII	INT
SC_EXP_CONFIGNAME[3]	' W'		ASCII	INT
SC_EXP_CONFIGNAME[4]	'IF'		ASCII	INT
SC_EXP_CONFIGNAME[5]	'EL'		ASCII	INT
SC_EXP_CONFIGNAME[6]	0		Decimal	INT
SC_EXP_CONFIGNAME[7]	0		Decimal	INT
SC_EXP_CONFIGNAME[8]	0		Decimal	INT
SC_EXP_CONFIGNAME[9]	0		Decimal	INT

Note that the first two registers are a 32-bit integer describing how many ASCII characters are coming in the Config Name. Here that value is "8". ASCII characters are packed, two per register, in the so-called ControlLogix String Format. The Config name here is "New File", but the ControlLogix string format displays those characters in reverse order per line.

Making an explicit message connection from scratch in an Allen-Bradley PLC program requires the following steps:

1. Make a new tag with the Message data type
2. Make a new tag to act as a Destination Element (a 16-bit array large enough to hold the data you'll be requesting).
3. Add a MSG command to your ladder logic (using the Message tag from #1 and the Destination Element from #2). The Class, Instance, and Attribute values depend on the data desired.
4. In the Communication tab of the MSG command, type in the Path to the safety controller:
   e.g. Ethernet, 2, 192.168.0.254
   where
   the "2" is the slot number for the EtherNet/IP card in the PLC rack and the IP Address shown is that of the safety controller

Chapter 3: Modbus/TCP

The Modbus/TCP protocol provides device information using register and coil banks defined by the slave device. This section defines the register and coil banks. By specification, Modbus/TCP uses TCP port 502. The SC22 does not support a Unit ID of 0 (sometimes called Slave ID or Device ID).

The following registers are used to send output values from the safety controller to the PLC. These can be read as Input Registers (30000) using Modbus function code 04 (Read Input Registers). The same values can also be read as Holding Registers (40000) using Modbus function code 03 (Read Holding Registers). The status information for all the virtual outputs and their fault flags, contained in the first 4 registers, can also be read as Inputs (10000) using Modbus function code 02 (Read Input Status) or Coils (00001) using Modbus Function code 01 (Read Coil Status).

Virtual Outputs and Virtual Output Faults
(Coils 00001-00064 or Inputs 10001-10064)
01: Read Coil Status or *02: Read Input Status*

Coil #	Input #	NAME
00001	10001	VO1
00002	10002	VO2
00003	10003	VO3
…	…	…
00031	10031	VO31
00032	10032	VO32

Coil #	Input #	NAME
00033	10033	VO1 Fault
00034	10034	VO2 Fault
00035	10035	VO3 Fault
…	…	…
00063	10063	VO31 Fault
00064	10064	VO32 Fault

Safety Controller Output Registers (Modbus/TCP Input or Holding Registers)
04: Read Input Registers or *03: Read Holding Registers*

Input REG #	Holding REG #	WORD NAME	DATA TYPE
1	1	VO1 – VO16 (see Flags, section 3.1)	16-bit integer
2	2	VO17 – VO32 (see Flags, section 3.1)	16-bit integer
3	3	Fault bits for VO1 – VO16 (see Flags, section 3.1)	16-bit integer
4	4	Fault bits for VO17 – VO32 (see Flags, section 3.1)	16-bit integer
5	5	SO1 is off due to: VO1 – VO16 (see Flags, section 3.1)	16-bit integer
6	6	SO1 is off due to: VO17 – VO32 (see Flags, section 3.1)	16-bit integer
7	7	SO2 is off due to: VO1 – VO16 (see Flags, section 3.1)	16-bit integer
8	8	SO2 is off due to: VO17 – VO32 (see Flags, section 3.1)	16-bit integer
9	9	SO3 is off due to: VO1 – VO16 (see Flags, section 3.1)	16-bit integer
10	10	SO3 is off due to: VO17 – VO32 (see Flags, section 3.1)	16-bit integer
11	11	VO1 Fault Index	16-bit integer
12	12	VO2 Fault Index	16-bit integer
13	13	VO3 Fault Index	16-bit integer
14	14	VO4 Fault Index	16-bit integer
15	15	VO5 Fault Index	16-bit integer
16	16	VO6 Fault Index	16-bit integer
17	17	VO7 Fault Index	16-bit integer
18	18	VO8 Fault Index	16-bit integer
19	19	VO9 Fault Index	16-bit integer
20	20	VO10 Fault Index	16-bit integer
21	21	VO11 Fault Index	16-bit integer

22	22	VO12 Fault Index	16-bit integer
23	23	VO13 Fault Index	16-bit integer
24	24	VO14 Fault Index	16-bit integer
25	25	VO15 Fault Index	16-bit integer
26	26	VO16 Fault Index	16-bit integer
27	27	VO17 Fault Index	16-bit integer
28	28	VO18 Fault Index	16-bit integer
29	29	VO19 Fault Index	16-bit integer
30	30	VO20 Fault Index	16-bit integer
31	31	VO21 Fault Index	16-bit integer
32	32	VO22 Fault Index	16-bit integer
33	33	VO23 Fault Index	16-bit integer
34	34	VO24 Fault Index	16-bit integer
35	35	VO25 Fault Index	16-bit integer
36	36	VO26 Fault Index	16-bit integer
37	37	VO27 Fault Index	16-bit integer
38	38	VO28 Fault Index	16-bit integer
39	39	VO29 Fault Index	16-bit integer
40	40	VO30 Fault Index	16-bit integer
41	41	VO31 Fault Index	16-bit integer
42	42	VO32 Fault Index	16-bit integer
43-106	43-106	*reserved*	16-bit integer
107-8	107-8	Fault #1 Time Stamp	32-bit integer
109-16	109-16	Fault #1 Name of I/O or System	2-word length + 12-ASCII chars
117	117	Fault #1 Fault Index	16-bit integer
118	118	Fault #1 VO number	16-bit integer
119-20	119-20	*reserved*	16-bit integer
121-22	121-22	Fault #2 Time Stamp	32-bit integer
123-30	123-30	Fault #2 Name of I/O or System	2-word length + 12- ASCII chars
131	131	Fault #2 Fault Index	16-bit integer
132	132	Fault #2 VO number	16-bit integer
133-34	133-34	*reserved*	16-bit integer
135-36	135-36	Fault #3 Time Stamp	32-bit integer
137-44	137-44	Fault #3 Name of I/O or System	2-word length + 12- ASCII chars
145	145	Fault #3 Fault Index	16-bit integer
146	146	Fault #3 VO number	16-bit integer
147-48	147-48	*reserved*	16-bit integer
149-50	149-50	Fault #4 Time Stamp	32-bit integer
151-58	151-58	Fault #4 Name of I/O or System	2-word length + 12- ASCII chars
159	159	Fault #4 Fault Index	16-bit integer
160	160	Fault #4 VO number	16-bit integer
161-62	161-62	*reserved*	16-bit integer
163-64	163-64	Fault #5 Time Stamp	32-bit integer
165-72	165-72	Fault #5 Name of I/O or System	2-word length + 12- ASCII chars
173	173	Fault #5 Fault Index	16-bit integer
174	174	Fault #5 VO number	16-bit integer

175-76	175-76	*reserved*	16-bit integer
177-78	177-78	Fault #6 Time Stamp	32-bit integer
179-86	179-86	Fault #6 Name of I/O or System	2-word length + 12- ASCII chars
187	187	Fault #6 Fault Index	16-bit integer
188	188	Fault #6 VO number	16-bit integer
189-190	189-190	*reserved*	16-bit integer
191-92	191-92	Fault #7 Time Stamp	32-bit integer
193-200	193-200	Fault #7 Name of I/O or System	2-word length + 12- ASCII chars
201	201	Fault #7 Fault Index	16-bit integer
202	202	Fault #7 VO number	16-bit integer
203-4	203-4	*reserved*	16-bit integer
205-6	205-6	Fault #8 Time Stamp	32-bit integer
207-14	207-14	Fault #8 Name of I/O or System	2-word length + 12- ASCII chars
215	215	Fault #8 Fault Index	16-bit integer
216	216	Fault #8 VO number	16-bit integer
217-18	217-18	*reserved*	16-bit integer
219-20	219-20	Fault #9 Time Stamp	32-bit integer
221-28	221-28	Fault #9 Name of I/O or System	2-word length + 12- ASCII chars
229	229	Fault #9 Fault Index	16-bit integer
230	230	Fault #9 VO number	16-bit integer
231-32	231-32	*reserved*	16-bit integer
233-34	233-34	Fault #10 Time Stamp	32-bit integer
235-42	235-42	Fault #10 Name of I/O or System	2-word length + 12-ASCII chars
243	243	Fault #10 Fault Index	16-bit integer
244	244	Fault #10 VO number	16-bit integer
245-46	245-46	*reserved*	16-bit integer
247-48	247-48	Seconds Since Boot	32-bit integer
249	249	Operating Mode	16-bit integer
250	250	Config CRC	16-bit integer
251-55	251-55	*reserved*	16-bit integer
256-57	256-57	Reconfig Time	32-bit integer
258-63	258-63	Date Code	2-word length + 8-ASCII chars
264-73	264-73	Configuration Name	2-word length + 16-ASCII chars
274-285	274-285	Device Name	2-word length + 20-ASCII chars
286-298	286-298	Device Model Number	2-word length + 22-ASCII chars
299-300	299-300	Device Serial Number	32-bit integer
301-348	301-348	*reserved*	16-bit integer
349-51	349-51	MAC ID (to be displayed in hex)	16-bit integer

3.1 Flags

Registers 1 through 4, defined below, appear as the first 4 words in register map. The information in these registers can be read as Input Registers (30000) using Modbus function code 04 (Read Input Registers). The same values can also be read as Holding Registers (40000) using Modbus function code 03 (Read Holding Registers).

Virtual Output 1-16
PLC Input register 30001 or Holding Register 40001, also Inputs 10001-16 or Coils 00001-16

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
VO16	VO15	VO14	VO13	VO12	VO11	VO10	VO9	VO8	VO7	VO6	VO5	VO4	VO3	VO2	VO1

Virtual Output 17-32
PLC Input register 30002 or Holding Register 40002, also Inputs 10017-32 or Coils 00017-32

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

Virtual Output Fault 1-16
PLC Input register 30003 or Holding Register 40003, also Inputs 10033-48 or Coils 00033-48

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
VO16 fault	VO15 fault	VO14 fault	VO13 fault	VO12 fault	VO11 fault	VO10 fault	VO9 fault	VO8 fault	VO7 fault	VO6 fault	VO5 fault	VO4 fault	VO3 fault	VO2 fault	VO1 fault

Virtual Output Fault 17-32
PLC Input register 30004 or Holding Register 40004, also Inputs 10049-64 or Coils 00049-64

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
VO32 fault	VO31 fault	VO30 fault	VO29 fault	VO28 fault	VO27 fault	VO26 fault	VO25 fault	VO24 fault	VO23 fault	VO22 fault	VO21 fault	VO20 fault	VO19 fault	VO18 fault	VO17 fault

Registers 5 through 10 are used, in blocks of two registers each, to communicate information about the three safety outputs. Each two register block covers all 32 virtual output bits that could have turned the safety output in question off. The first block of two such registers is seen below as an example.

Safety Output 1 is off due to: Virtual Output 1-16
PLC Input register 30005 or Holding Register 40005
Note that not every Virtual Output need be configured to turn off Safety Output 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO16	VO15	VO14	VO13	VO12	VO11	VO10	VO9	VO8	VO7	VO6	VO5	VO4	VO3	VO2	VO1

Safety Output 1 is off due to: Virtual Output 17-32
PLC Input register 30006 or Holding Register 40006
Note that not every Virtual Output need be configured to turn off Safety Output 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

The pattern then repeats. Registers 7-8 are linked to Safety Output 2, registers 9-10 are for Safety Output 3.

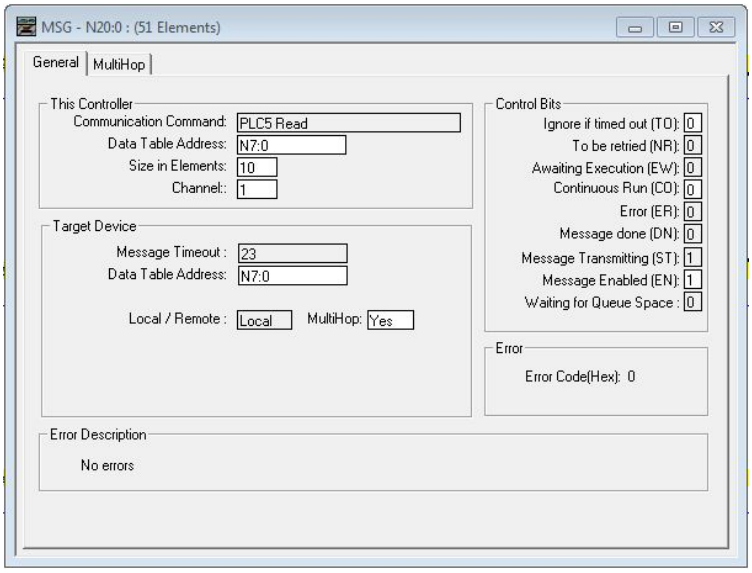
Chapter 4: PCCC

Allen-Bradley's PLC5 and SLC 500 family of devices use PCCC communications protocol. The safety controller will support these PLCs using an input register array. The term "Input" is from the point of view of the PLC.

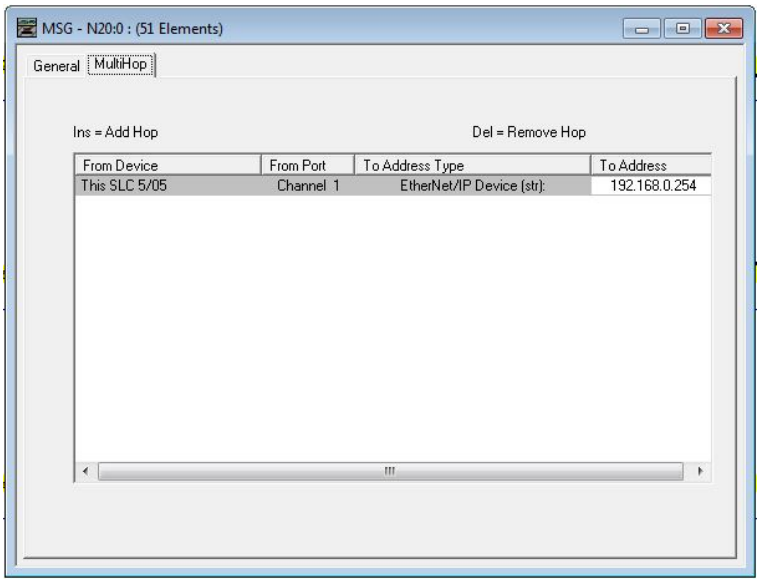
4.1 PLC Configuration

The images below represent a typical configuration:

1. Read. Message command reading from N7 table on safety controller



2. Read. IP Address of the safety controller is entered here.



4.2 Outputs from Safety Controller (Inputs to PLC)

The Output registers are used to push output values from the safety controller to the PLC. MSG (message) commands are used to Read (N7) from the controller.

N7 REGS

REG #	WORD NAME	DATA TYPE
0	VO1 – VO16 (see Flags, section 4.3)	16-bit integer
1	VO17 – VO32 (see Flags, section 4.3)	16-bit integer
2	Fault bits for VO1 – VO16 (see Flags, section 4.3)	16-bit integer
3	Fault bits for VO17 – VO32 (see Flags, section 4.3)	16-bit integer
4	SO1 is off due to: VO1 – VO16 (see Flags, section 4.3)	16-bit integer
5	SO1 is off due to: VO17 – VO32 (see Flags, section 4.3)	16-bit integer
6	SO2 is off due to: VO1 – VO16 (see Flags, section 4.3)	16-bit integer
7	SO2 is off due to: VO17 – VO32 (see Flags, section 4.3)	16-bit integer
8	SO3 is off due to: VO1 – VO16 (see Flags, section 4.3)	16-bit integer
9	SO3 is off due to: VO17 – VO32 (see Flags, section 4.3)	16-bit integer
10	VO1 Fault Index	16-bit integer
11	VO2 Fault Index	16-bit integer
12	VO3 Fault Index	16-bit integer
13	VO4 Fault Index	16-bit integer
14	VO5 Fault Index	16-bit integer
15	VO6 Fault Index	16-bit integer
16	VO7 Fault Index	16-bit integer
17	VO8 Fault Index	16-bit integer
18	VO9 Fault Index	16-bit integer
19	VO10 Fault Index	16-bit integer
20	VO11 Fault Index	16-bit integer
21	VO12 Fault Index	16-bit integer
22	VO13 Fault Index	16-bit integer
23	VO14 Fault Index	16-bit integer
24	VO15 Fault Index	16-bit integer
25	VO16 Fault Index	16-bit integer
26	VO17 Fault Index	16-bit integer
27	VO18 Fault Index	16-bit integer
28	VO19 Fault Index	16-bit integer
29	VO20 Fault Index	16-bit integer
30	VO21 Fault Index	16-bit integer
31	VO22 Fault Index	16-bit integer
32	VO23 Fault Index	16-bit integer
33	VO24 Fault Index	16-bit integer
34	VO25 Fault Index	16-bit integer
35	VO26 Fault Index	16-bit integer
36	VO27 Fault Index	16-bit integer
37	VO28 Fault Index	16-bit integer
38	VO29 Fault Index	16-bit integer
39	VO30 Fault Index	16-bit integer
40	VO31 Fault Index	16-bit integer
41	VO32 Fault Index	16-bit integer
42-105	*reserved*	16-bit integer

106-7	Fault #1 Time Stamp	32-bit integer
108-15	Fault #1 Name of I/O or System	2-word length + 12-ASCII chars
116	Fault #1 Fault Index	16-bit integer
117	Fault #1 VO number	16-bit integer
118-19	*reserved*	16-bit integer
120-21	Fault #2 Time Stamp	32-bit integer
122-29	Fault #2 Name of I/O or System	2-word length + 12- ASCII chars
130	Fault #2 Fault Index	16-bit integer
131	Fault #2 VO number	16-bit integer
132-33	*reserved*	16-bit integer
134-35	Fault #3 Time Stamp	32-bit integer
136-43	Fault #3 Name of I/O or System	2-word length + 12- ASCII chars
144	Fault #3 Fault Index	16-bit integer
145	Fault #3 VO number	16-bit integer
146-47	*reserved*	16-bit integer
148-49	Fault #4 Time Stamp	32-bit integer
150-57	Fault #4 Name of I/O or System	2-word length + 12- ASCII chars
158	Fault #4 Fault Index	16-bit integer
159	Fault #4 VO number	16-bit integer
160-61	*reserved*	16-bit integer
162-63	Fault #5 Time Stamp	32-bit integer
164-71	Fault #5 Name of I/O or System	2-word length + 12- ASCII chars
172	Fault #5 Fault Index	16-bit integer
173	Fault #5 VO number	16-bit integer
174-75	*reserved*	16-bit integer
176-77	Fault #6 Time Stamp	32-bit integer
178-85	Fault #6 Name of I/O or System	2-word length + 12- ASCII chars
186	Fault #6 Fault Index	16-bit integer
187	Fault #6 VO number	16-bit integer
188-89	*reserved*	16-bit integer
190-93	Fault #7 Time Stamp	32-bit integer
194-99	Fault #7 Name of I/O or System	2-word length + 12- ASCII chars
200	Fault #7 Fault Index	16-bit integer
201	Fault #7 VO number	16-bit integer
202-3	*reserved*	16-bit integer
204-5	Fault #8 Time Stamp	32-bit integer
206-13	Fault #8 Name of I/O or System	2-word length + 12- ASCII chars
214	Fault #8 Fault Index	16-bit integer
215	Fault #8 VO number	16-bit integer
216-17	*reserved*	16-bit integer
218-19	Fault #9 Time Stamp	32-bit integer
220-27	Fault #9 Name of I/O or System	2-word length + 12- ASCII chars

228	Fault #9 Fault Index	16-bit integer
229	Fault #9 VO number	16-bit integer
230-31	*reserved*	16-bit integer
232-33	Fault #10 Time Stamp	32-bit integer
234-41	Fault #10 Name of I/O or System	2-word length + 12-ASCII chars
242	Fault #10 Fault Index	16-bit integer
243	Fault #10 VO number	16-bit integer
244-45	*reserved*	16-bit integer
246-47	Seconds Since Boot	32-bit integer
248	Operating Mode	16-bit integer
249	Config CRC	16-bit integer
250-54	*reserved*	16-bit integer
255-56	Reconfig Time	32-bit integer
257-62	Date Code	2-word length + 8-ASCII chars
263-72	Configuration Name	2-word length + 16-ASCII chars
273-284	Device Name	2-word length + 20-ASCII chars
285-297	Device Model Number	2-word length + 22-ASCII chars
298-99	Device Serial Number	32-bit integer
300-47	*reserved*	16-bit integer
348-50	MAC ID (to be displayed in hex)	16-bit integer

4.3 Flags

Words 0 through 3, defined below, appear as the first 4 words in the N7 table. They are used to communicate the status and fault flags for all 32 Virtual Outputs.

Word #0, Virtual Output 1-16, Bit Position

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO16	VO15	VO14	VO13	VO12	VO11	VO10	VO9	VO8	VO7	VO6	VO5	VO4	VO3	VO2	VO1

Word #1, Virtual Output 17-32, Bit Position

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

Word #2, Fault Flag bits for Virtual Output 1-16, Bit Position

Note that not every Virtual Output has a defined Fault Flag.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

Word #3, Virtual Output 17-32 Fault Flag, Bit Position

Note that not every Virtual Output has a defined Fault Flag.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

Registers 4 through 9 are used, in blocks of two registers each, to communicate information about the three safety outputs. Each two register block covers all 32 virtual output bits that could have turned the safety output in question off. The first block of two such registers is seen below as an example.

Register #4, Safety Output 1 is off due to: Virtual Output 1-16, Bit Position

Note that not every Virtual Output need be configured to turn off Safety Output 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO16	VO15	VO14	VO13	VO12	VO11	VO10	VO9	VO8	VO7	VO6	VO5	VO4	VO3	VO2	VO1

Register #5, Safety Output 1 is off due to: Virtual Output 17-32, Bit Position

Note that not every Virtual Output need be configured to turn off Safety Output 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VO32	VO31	VO30	VO29	VO28	VO27	VO26	VO25	VO24	VO23	VO22	VO21	VO20	VO19	VO18	VO17

The pattern then repeats. Registers 6-7 are linked to Safety Output 2, registers 8-9 are for Safety Output 3.

Chapter 5: Error Codes

5.1 Fault Index Numbers

The Fault Index Numbers are 16-bit integers that correspond to the Safety Controller's floating-point style fault codes. See Chapter 9 of the SC22 Instruction Manual for more information about troubleshooting.

Fault Index	Fault Code	Error Message	Remedy
0	0	Input Fault	Cycle Input
1	1.1	Output Fault	Check for shorts
2	1.2	Output Fault	Check for shorts
3	1.3	Internal Fault	Contact Banner Engineering
4	1.4	Internal Fault	Contact Banner Engineering
5	1.5	Output Fault	Check Output Wiring
6	1.6	Internal Fault	Contact Banner Engineering
7	1.7	Output Fault	Check for shorts
8	1.8	Internal Fault	Contact Banner Engineering
9	2.1	Concurrency Fault	Cycle Input
10	2.2	Simultaneity Fault	Cycle Input
11	2.3	Concurrency Fault	Cycle Input
12	2.4	Simultaneity Fault	Cycle Input
13	2.5	Concurrency Fault	Cycle Input
14	2.6	Simultaneity Fault	Cycle Input
15	2.7	Internal Fault	Contact Banner Engineering
16	2.8	Input Fault	Check Input Terminals
17	2.9	Input Fault	Check Input Terminals
18	2.1	Input Fault	Check Input Terminals
19	2.11	Input Fault	Check Input Terminals
20	2.12	Input Fault	Check Input Terminals
21	2.13	Input Fault	Check Input Terminals
22	2.14	Input Fault	Check Input Terminals
23	2.15	Open Lead	Check Input Terminals
24	2.16	Input Fault	Check Input Terminals
25	2.17	Input Fault	Check Input Terminals
26	2.18	Input Fault	Check Input Terminals
27	2.19	Open Lead	Check Input Terminals
28	2.2	Input Fault	Check Input Terminals
29	2.21	Open Lead	Check Input Terminals
30	2.22	Input Fault	Check Input Terminals
31	2.23	Input Fault	Check Input Terminals
32	3.1	EDM Fault	Check EDM Terminals

33	3.2	EDM Fault	Check EDM Terminals
34	3.3	EDM Fault	Check EDM Terminals
35	3.4	EDM Fault	Check EDM Terminals
36	3.5	EDM Fault	Check EDM Terminals
37	3.6	EDM Fault	Check EDM Terminals
38	3.7	EDM Fault	Check EDM Terminals
39	4.1	Supply Voltage Low	Check Power Supply
40	4.2	Internal Fault	Contact Banner Engineering
41	4.3	Internal Fault	Contact Banner Engineering
42	4.4	Internal Fault	Contact Banner Engineering
43	4.5	Internal Fault	Contact Banner Engineering
44	4.6	Internal Fault	Contact Banner Engineering
45	4.7	Internal Fault	Contact Banner Engineering
46	4.8	Internal Fault	Contact Banner Engineering
47	4.9	Internal Fault	Contact Banner Engineering
48	4.1	Internal Fault	Contact Banner Engineering
49	4.11	Internal Fault	Contact Banner Engineering
50	4.12	Configuration Timeout	Check Configuration
51	4.13	Configuration Timeout	Check Configuration
52	4.14	Configuration Unconfirmed	Confirm Configuration
53	4.15	Internal Fault	Contact Banner Engineering
54	4.16	Internal Fault	Contact Banner Engineering
55	4.17	Internal Fault	Contact Banner Engineering
56	4.18	Internal Fault	Contact Banner Engineering
57	4.19	Internal Fault	Contact Banner Engineering
58	4.2	Unassigned Terminal in Use	Check Terminals
59	4.21	Internal Fault	Contact Banner Engineering
60	4.22	Internal Fault	Contact Banner Engineering
61	4.23	Internal Fault	Contact Banner Engineering
62	4.24	Internal Fault	Contact Banner Engineering
63	4.25	Internal Fault	Contact Banner Engineering
64	4.26	Internal Fault	Contact Banner Engineering
65	4.27	Internal Fault	Contact Banner Engineering
66	4.28	Internal Fault	Contact Banner Engineering
67	4.29	Internal Fault	Contact Banner Engineering
68	4.3	Internal Fault	Contact Banner Engineering
69	4.31	Internal Fault	Contact Banner Engineering
70	4.32	Internal Fault	Contact Banner Engineering
71	5.1	Mute Lamp Fault	Check Lamp and Wiring
72	5.2	Mute Lamp Fault	Check for shorts
73	5.3	Internal Fault	Contact Banner Engineering

74	6.xx	Internal Fault	Load New Configuration
75	4.33	Internal Fault	Contact Banner Engineering
76	1.9	AVM Input Fault	Perform a System Reset
77	1.1	AVM Input Fault	Perform a System Reset
78	2.24	Input activated while bypassed	Perform a system reset
79	2.25	Input Fault	Monitoring timer expired before AVM closed
80	2.26	Input Fault	AVM input not closed when output turned on