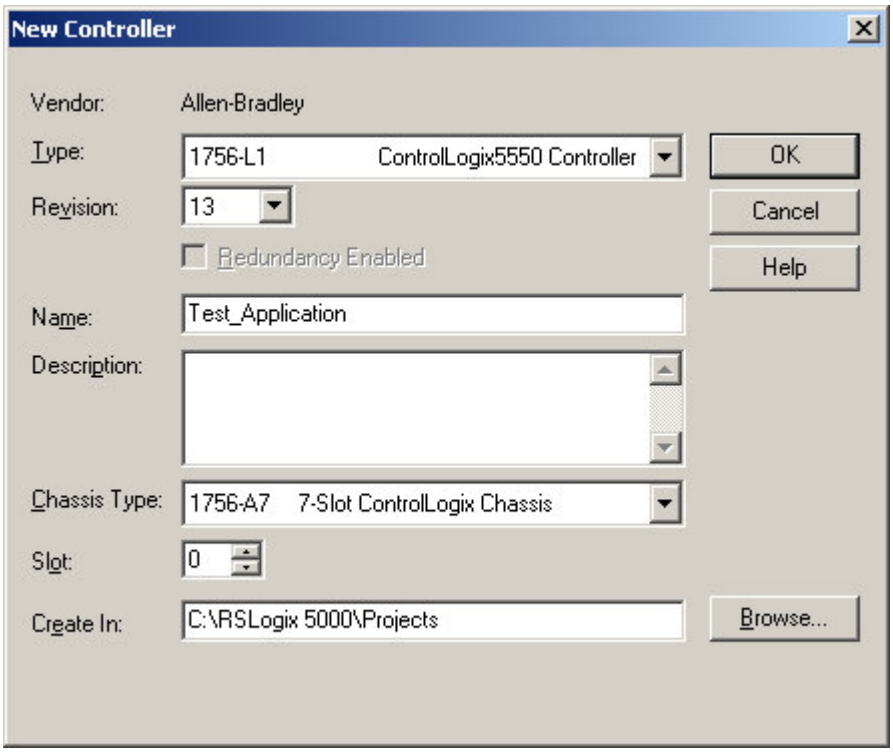
Sample Start-up Project for ACR Controllers and CompactLogix/ControlLogix PLCs

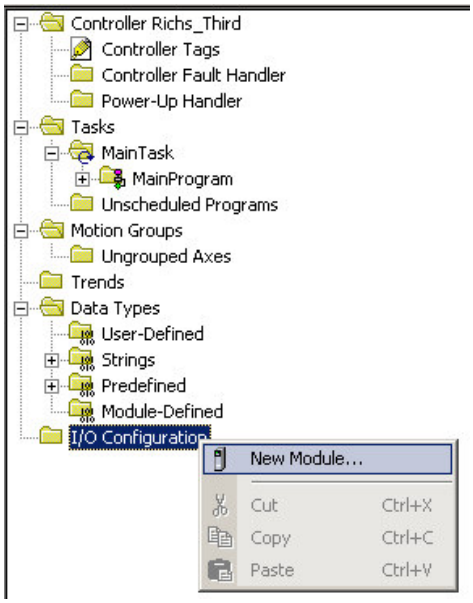
A. Create a New Project (File->New), this will automatically open the "New Controller" dialog box.

B. Select the ControlLogix5550 Controller in Type (1756-L1, 7 slot chassis, Revision 13). Be sure to give the controller a useful name, select the number of slots in chassis and select the correct slot for the Logix5550 processor.



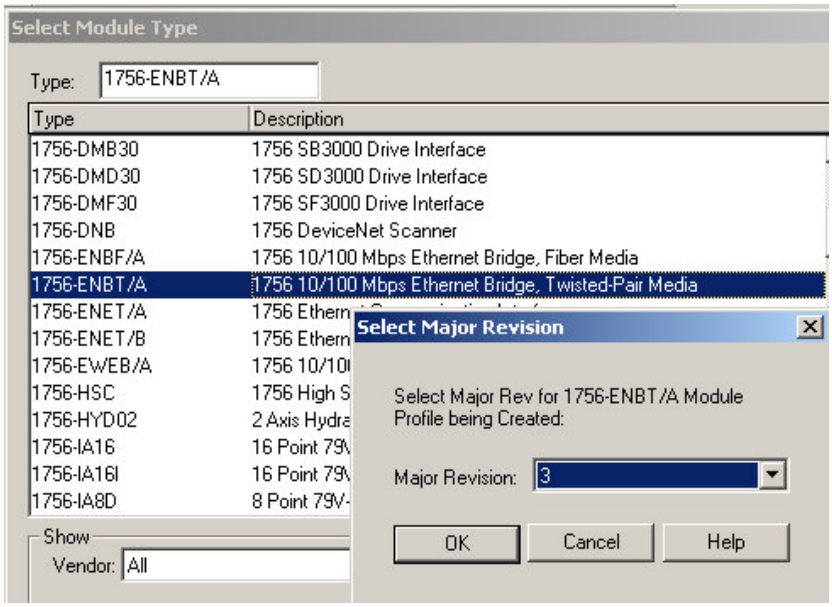
C. Add the Ethernet IP module

1. Right click on I/O Configuration and select "New module"



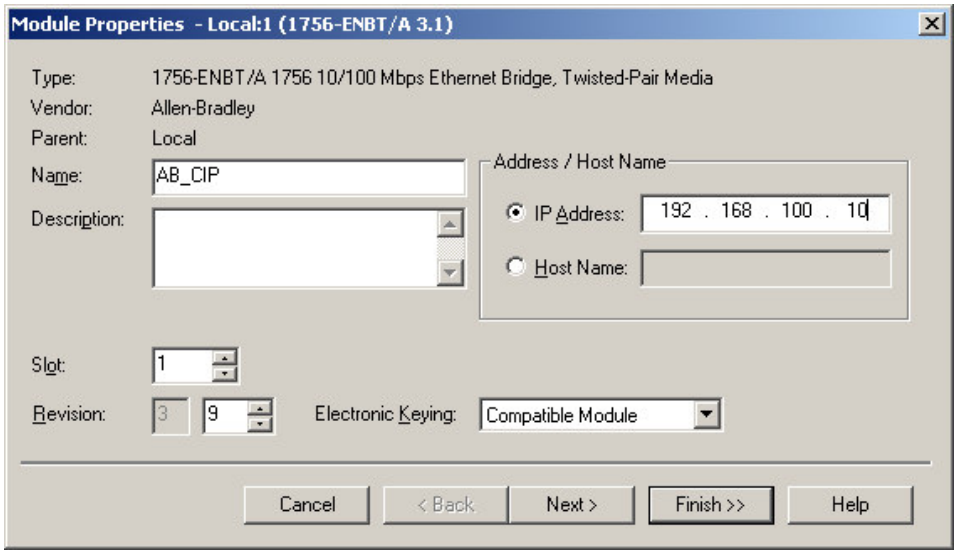
2. Select appropriate Ethernet IP module (1756-ENBT/A) then click on OK (see step 3's illustration.)

3. Select proper Major firmware revision (3.x) then click OK again. Firmware revision is printed on the side of the Ethernet IP module.

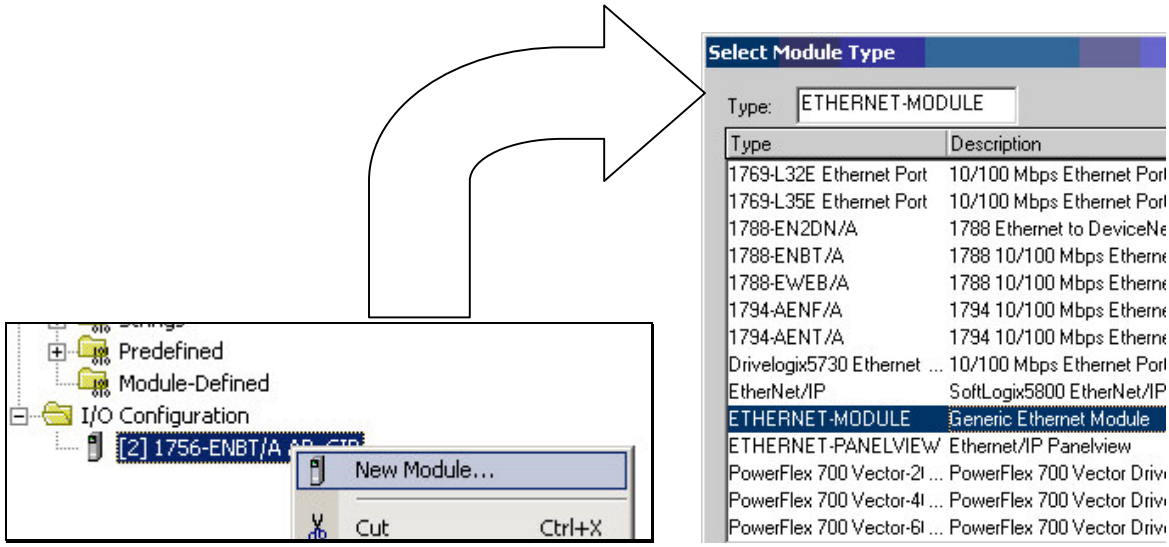


4. Enter a Name (i.e. "AB_CIP",) correct "IP address", rack "Slot" and firmware "Minor" revision for the adapter then click on FINISH. See picture below.

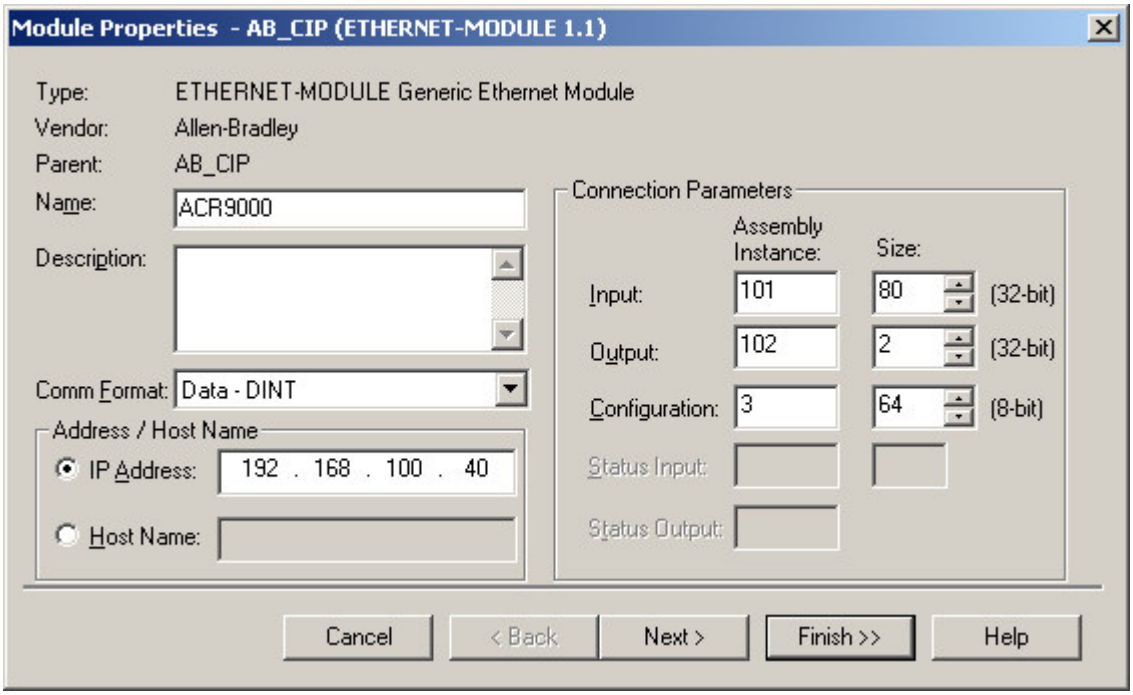
   Name: AB_CIP
   IP Address 192.168.100.10 (read from display on E/IP module)
   Slot: 1 (remember that the slots are numbered starting at slot 0)
   Revision: 3.9


Module Properties - Local:1 (1756-ENBT/A 3.1)
Type:
1756-ENBT/A 1756 10/100 Mbps Ethernet Bridge, Twisted-Pair Media
Vendor:
Allen-Bradley
Parent:
Local
Name:
AB_CIP
Description:
Address / Host Name
IP Address:
192 . 168 . 100 . 10
Host Name:
Slot:
1
Revision:
3
9
Electronic Keying:
Compatible Module
Cancel
< Back
Next >
Finish >>
Help

5. Add ACR9000 as module for AB_CIP module by right-clicking and selecting New Module. Select "ETHERNET-MODULE" and click on OK:



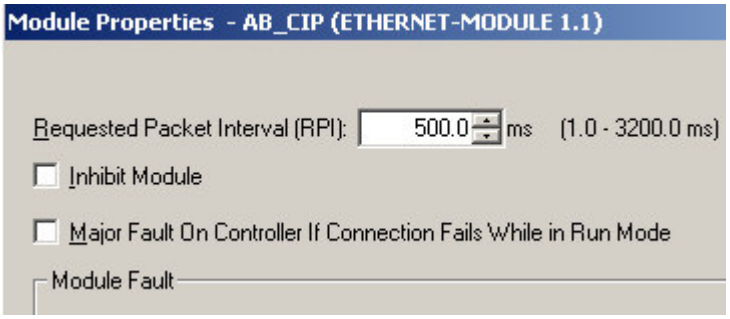
6. Enter the information as shown in the diagram below:



The "Connection Parameters" section is used to configure what information is automatically polled from or written to the ACR9000. The "Assembly Instance" values will never change, but you can change the "size" which is the number of parameters to read/write. The "Input" parameter can be polled information or can be the "special case" in the manual that refers to the "Fast Status" or FSTAT capability of the ACR9000 controller. "Output" refers to data that will be sent to the controller automatically at a set update rate selected on the "next" screen.

"Configuration" refers to the information that is sent back and forth. When done entering items click on NEXT.

7. Set the Requested Packet Interval to a valid value, minimum is 100ms. It is recommended you set this to a "large" value to ensure that all information is transferred. With revision 1.21 of the ACR9000 OS, 300ms was needed to write 6 items and read 3 items. This seems pretty excessive and may change if something is found wrong in the firmware.

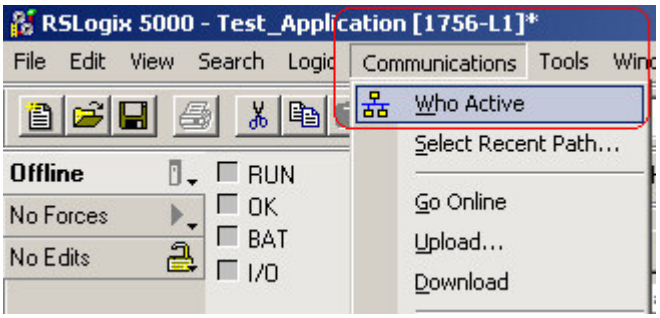


8. Click on FINISH.

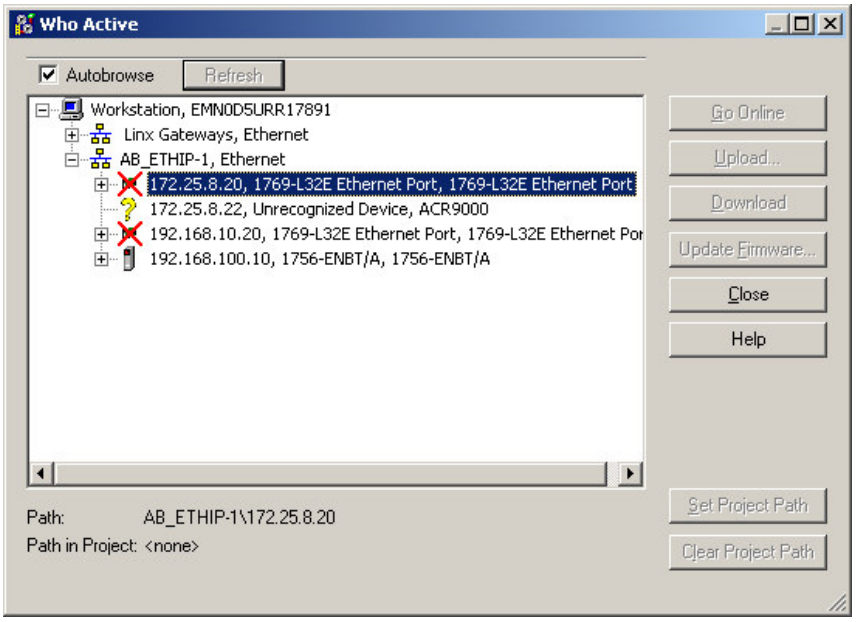
Save the project!!

Go online with the AB controller

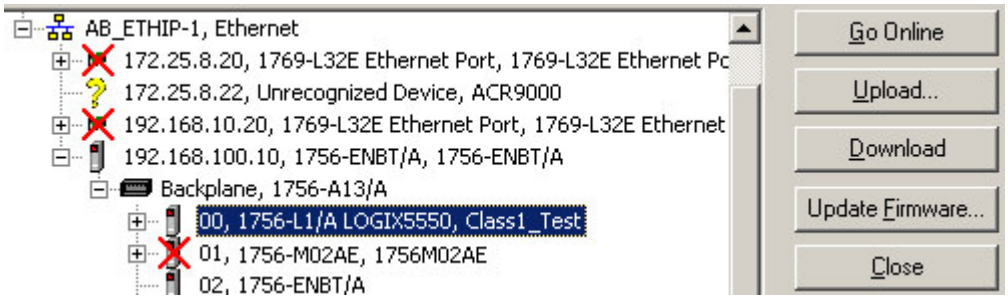
1. Select Communications then Who Active from the menu:



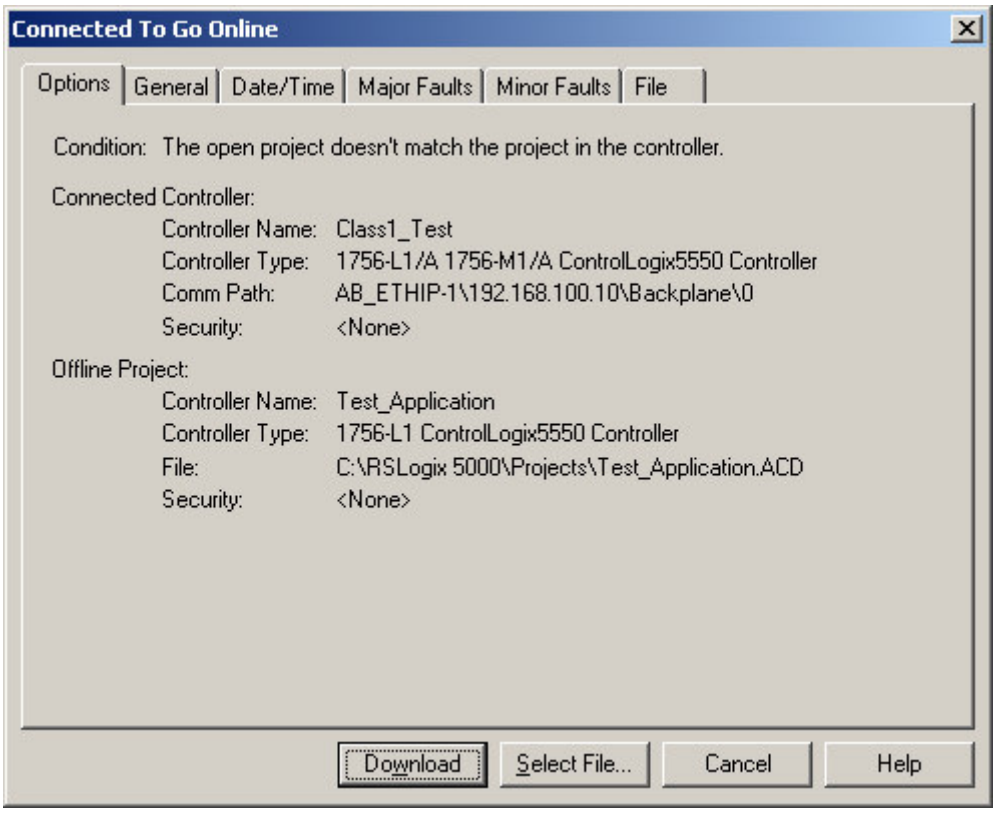
2. If Autobrowse is checked, the Who Active dialog box will show what controllers are on your particular network. (Did I forget to mention to connect the controller to the Ethernet network with your PC and ACR9000?) Note that it may be necessary to open RSLinx and add the Ethernet/IP and/or RS232 drive to talk to the controllers.



3. Click on the plus next to your AB controller's IP address to expand the tree to see the modules installed, select the correct processor's slot (name will show an old name from the current program installed in the processor, "Class1_Test" in this screen shot.) Click on "Go Online"

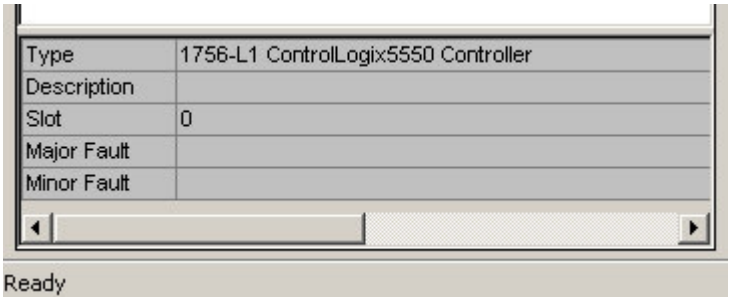

AB_ETHIP-1, Ethernet
172.25.8.20, 1769-L32E Ethernet Port, 1769-L32E Ethernet Pc
172.25.8.22, Unrecognized Device, ACR9000
192.168.10.20, 1769-L32E Ethernet Port, 1769-L32E Ethernet
192.168.100.10, 1756-ENBT/A, 1756-ENBT/A
Backplane, 1756-A13/A
00, 1756-L1/A LOGIX5550, Class1_Test
01, 1756-M02AE, 1756M02AE
02, 1756-ENBT/A
Go Online
Upload...
Download
Update Firmware...
Close

4. You will get a new window showing the existing project of the controller if the projects to not match. You will have to select download before you are allowed to interact with the AB controller.



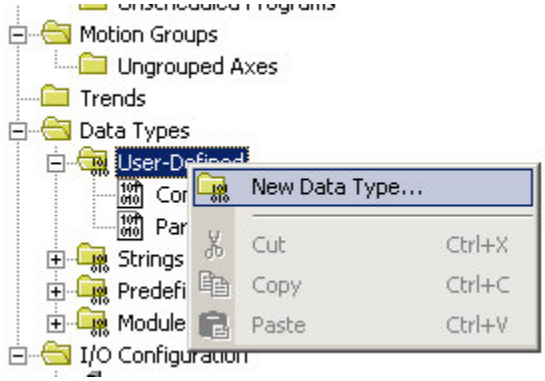
5. Select Download and then Download again when the confirmation dialog box is displayed. Once downloaded and connected, the top of the RSLogix screen will have status information showing the controller and I/O status as well as the project "Path" which is the controller's IP address and slot number-.

6. The lower left portion of the software will show status of whatever is selected in the project tree of left. This will become very useful in troubleshooting the application.

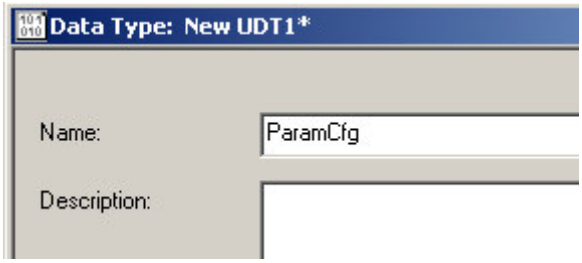

Type
1756-L1 ControlLogix5550 Controller
Description
Slot
0
Major Fault
Minor Fault
Ready

D: Create a User-Defined Data Type called “ParamCfg”. This is used by another User Data Type we will create later (ConfigData):

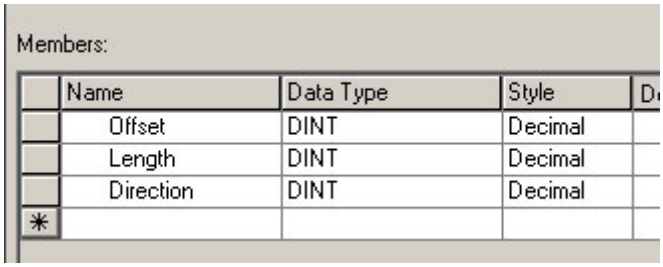
1. Right-click on “User Defined” and select “New Data Type”:



2. Type in the name “ParamCfg”



3. Add three members as shown below. These Members correspond to the CIP specifications for setting the Parameter # (“Offset”), number of parameters to retrieve (“Length”) and whether the data is an input or output to the controller (“Direction”, 1 = input to 9000, 0 = output from 9000.) All three members are Double Integer (DINT.)



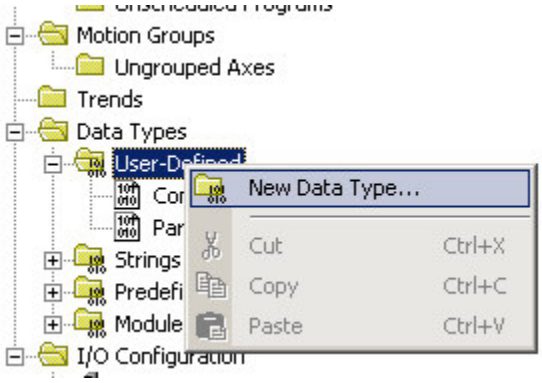
Name	Data Type	Style
Offset	DINT	Decimal
Length	DINT	Decimal
Direction	DINT	Decimal

3. Click on OK or Apply to save the new data type.

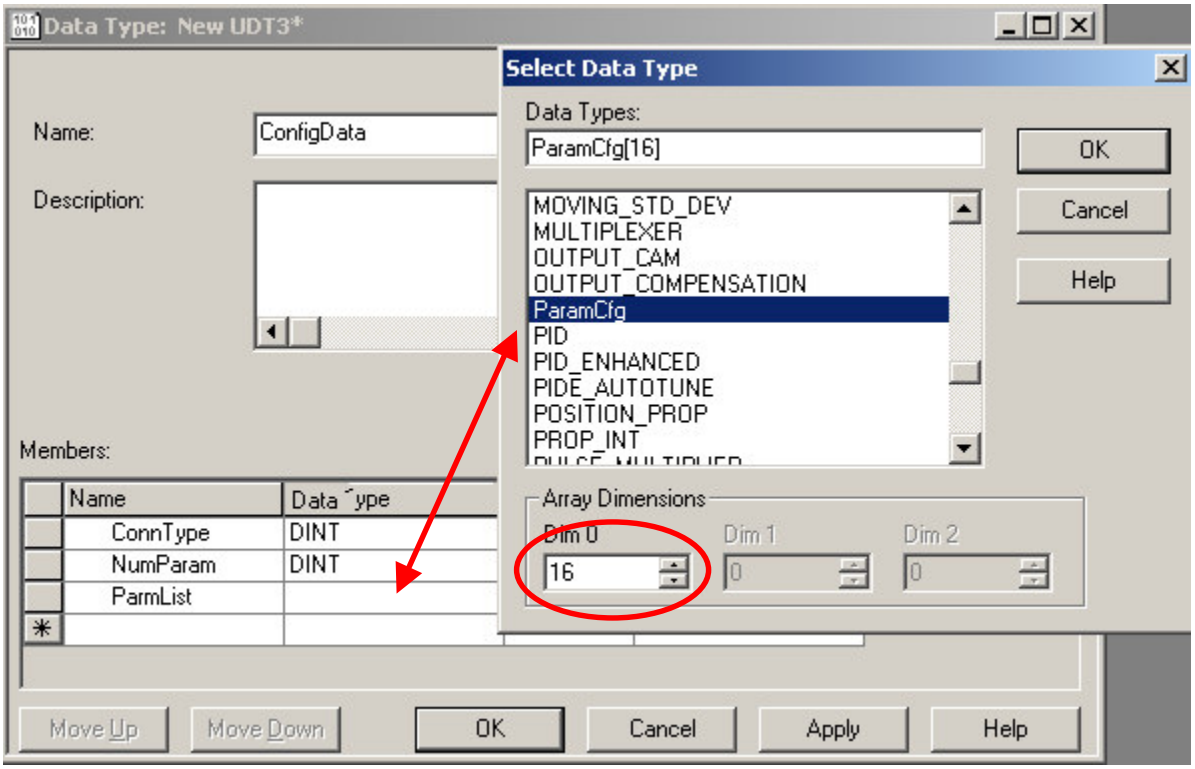
Save the project!!

E: Create another User-Defined Data Type called “ConfigData” as shown below:

1. Right-click on “User Defined” and select “New Data Type”:



2. Type in the name “ConfigData” and enter the Members for the data type (see screenshot):

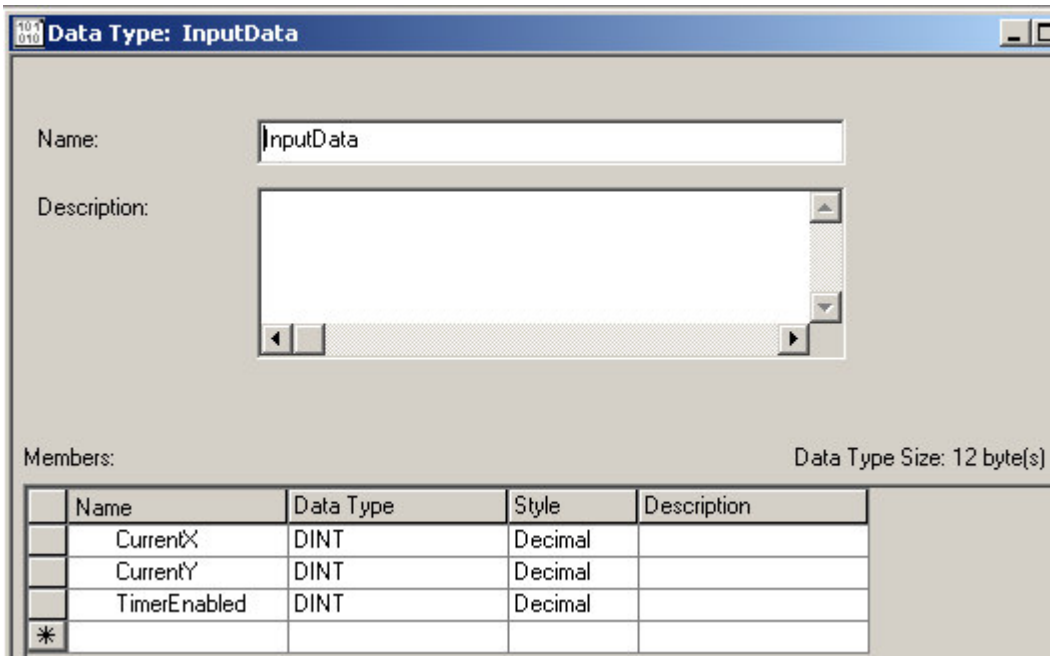


ConnType => DINT
NumParam => DINT
ParmList => ParamCfg(16)

We will assign values for these later on when we configure the data transfer in the ladder program.

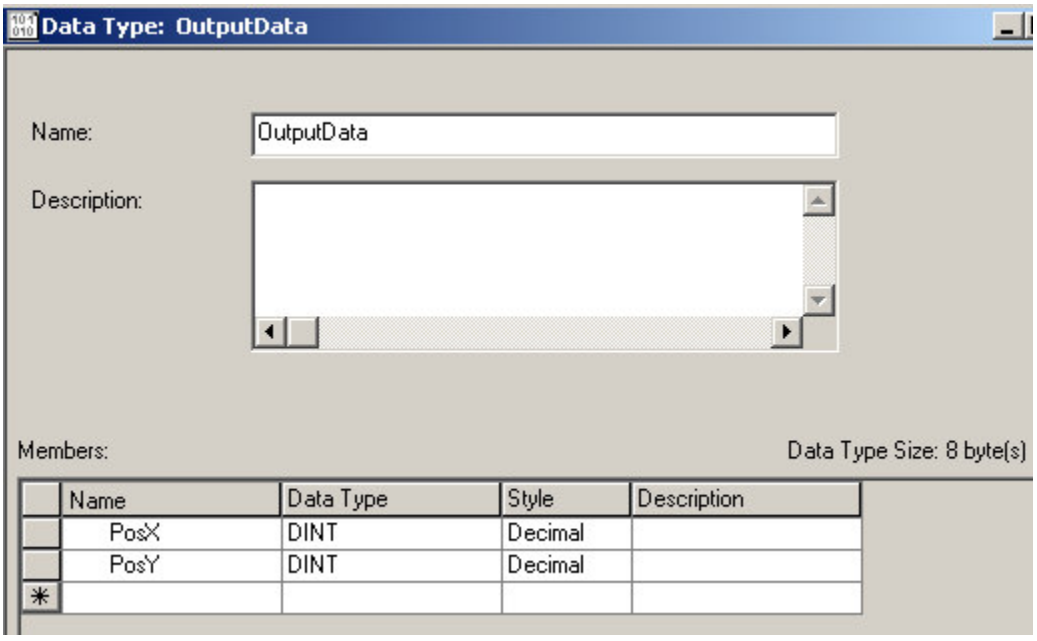
Save the project!!

Create a User-Defined Data Type: InputData



CurrentX, will be used for P12298
CurrentY, will be used for P12544
TimerEnabled, will be used for P4144

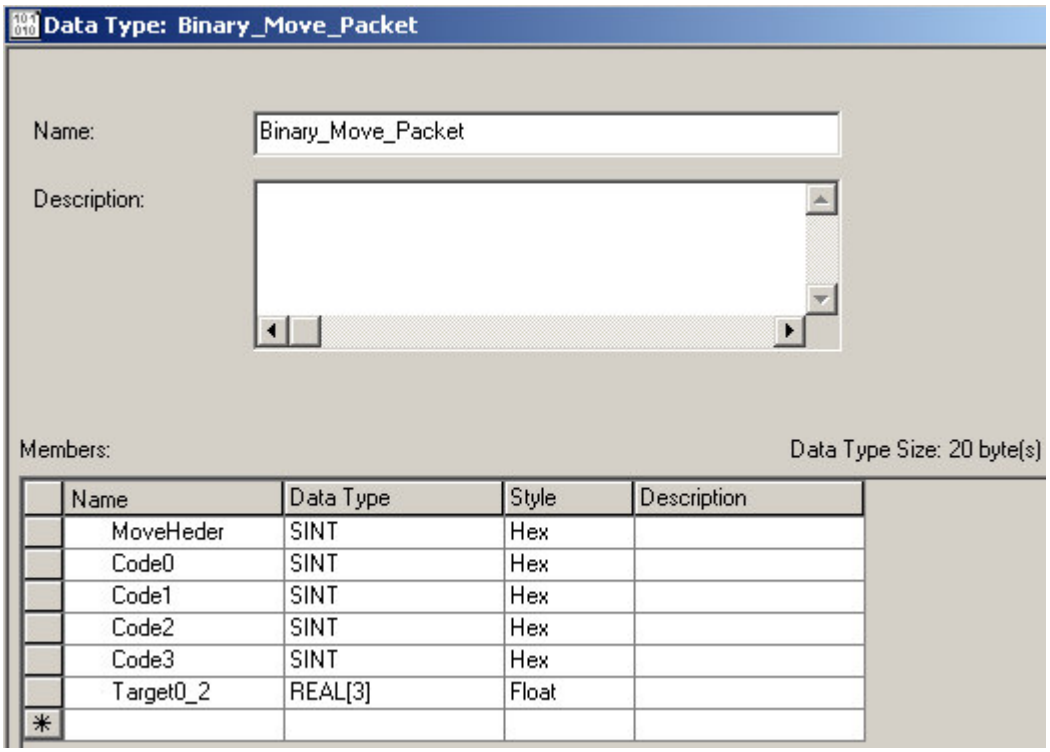
Create a User-Defined Data Type: OutputData



PosX
PosY

Save the project

Create a User-Defined Data Type called Binary_Move_Packet":



Name	Data Type	Style	Description
MoveHeder	SINT	Hex	
Code0	SINT	Hex	
Code1	SINT	Hex	
Code2	SINT	Hex	
Code3	SINT	Hex	
Target0_2	REAL[3]	Float	
*			

This creates a move packet data type to move up to three axes.

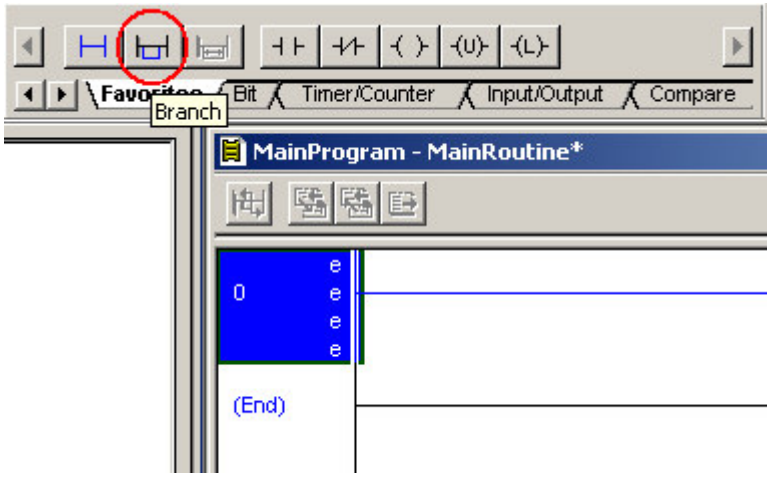
Refer to the ACR Controller Programmer's Guide (Binary Host Interface section) for more information on Binary Move Packets.

Save the project!!

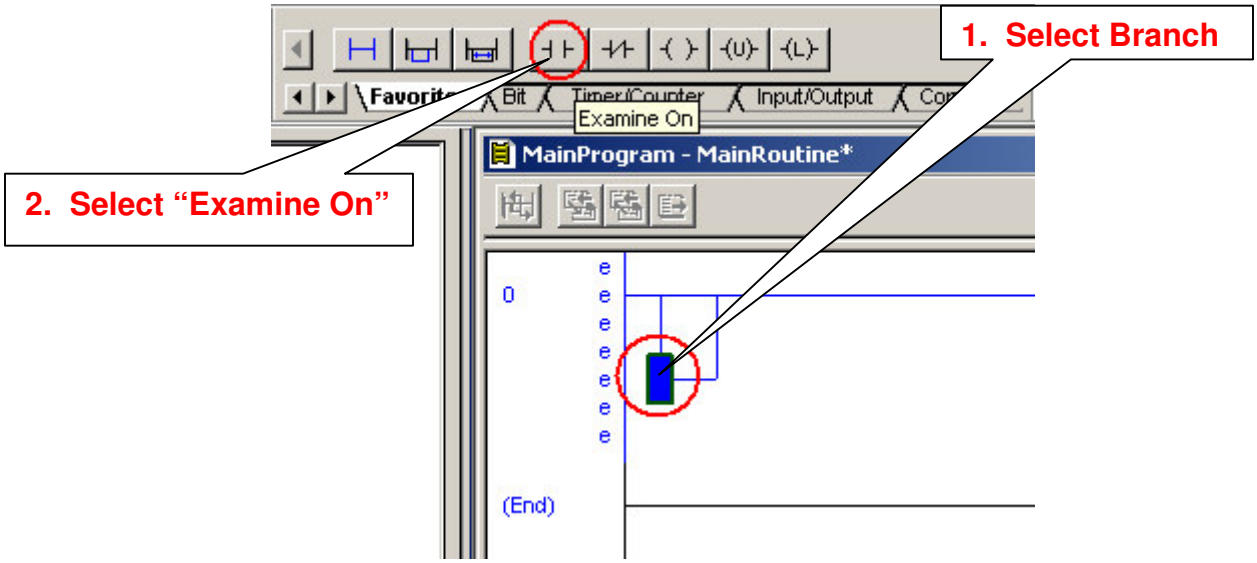
Create a program that will send down the configuration

1. Edit the Main Routine. Expand the project tree to expose Main Routine to double-click to open the PLC editor:

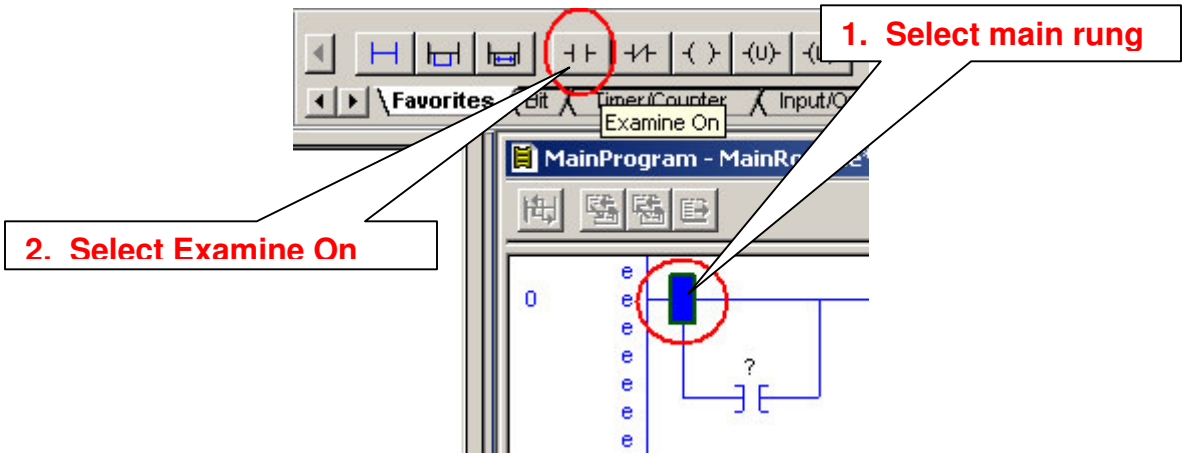
2. Add a branch using buttons above PLC editor:



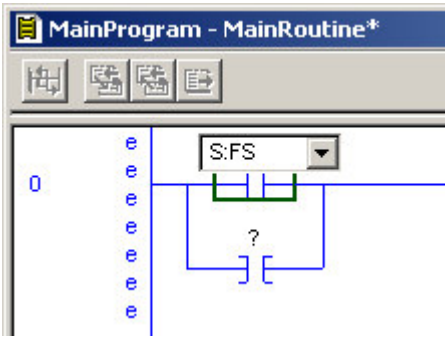
3. Select the branch then click on the “Examine On” button:



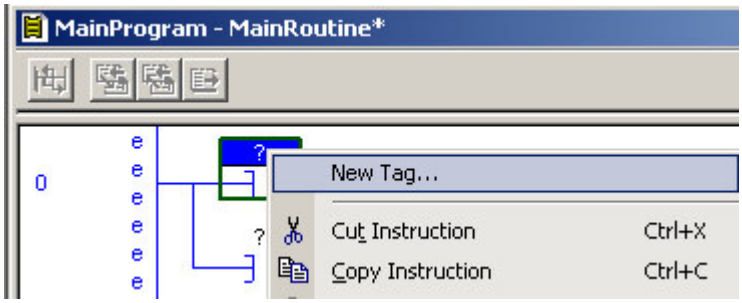
4. Add another parallel "examine on" bit by clicking on main rung then clicking on "Examine On" button:



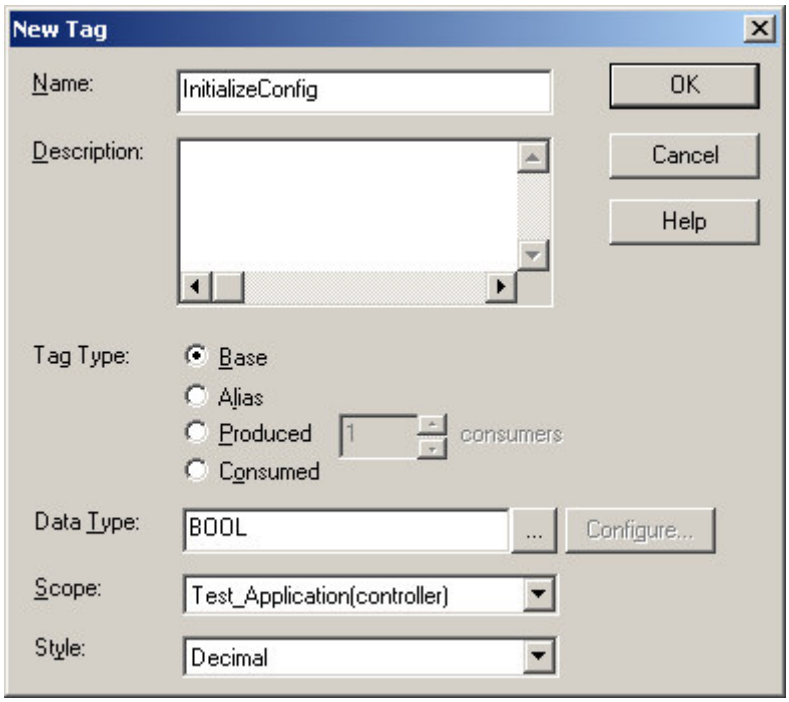
5. Double-click on question mark above top contact and enter "S:FS" which is a keyword for "System: First Scan" which will only be true the first time the PLC runs the program:



6. Right-click on the question mark above the bottom contact and select "New Tag":

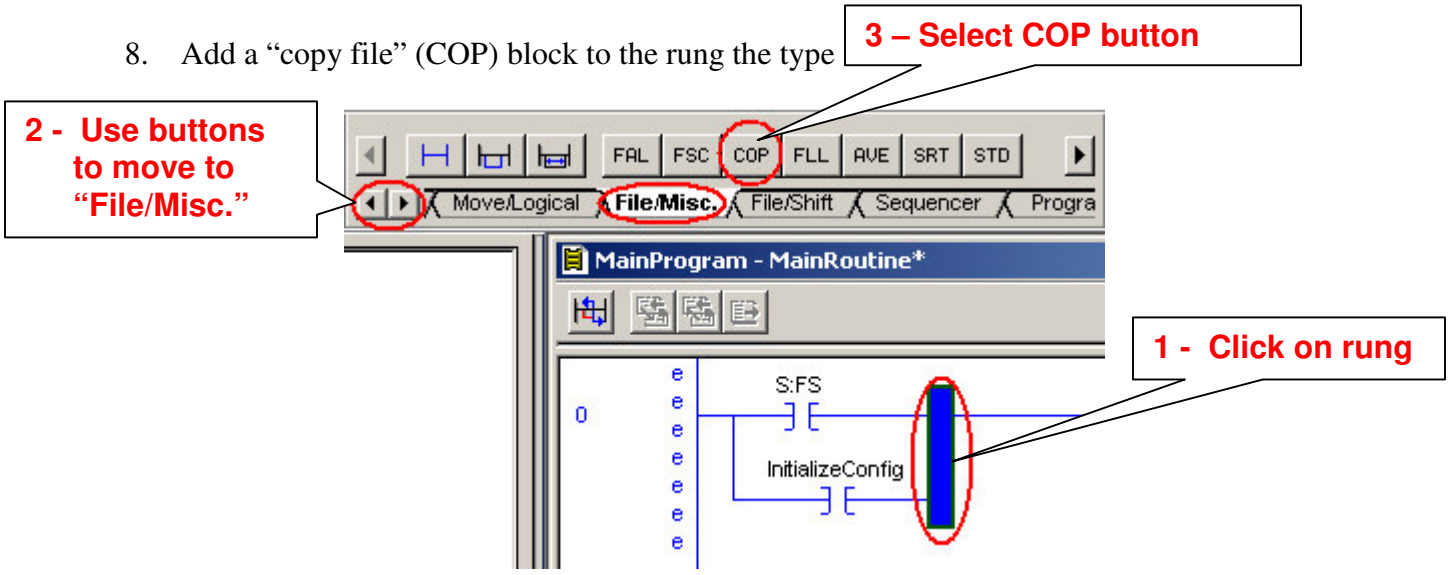

MainProgram - MainRoutine*
0
New Tag...
Cut Instruction Ctrl+X
Copy Instruction Ctrl+C

7. In the New Tag dialog box, type “InitializeConfig” in the Name box and make sure the scope is the controller then click on OK:

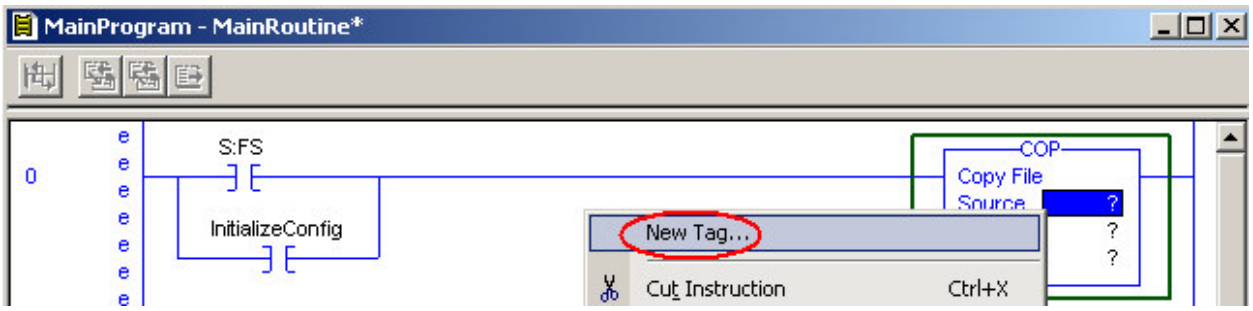


Name: InitializeConfig
Tag Type: Base
Data Type: BOOL
Scope: Controller
Style: Decimal

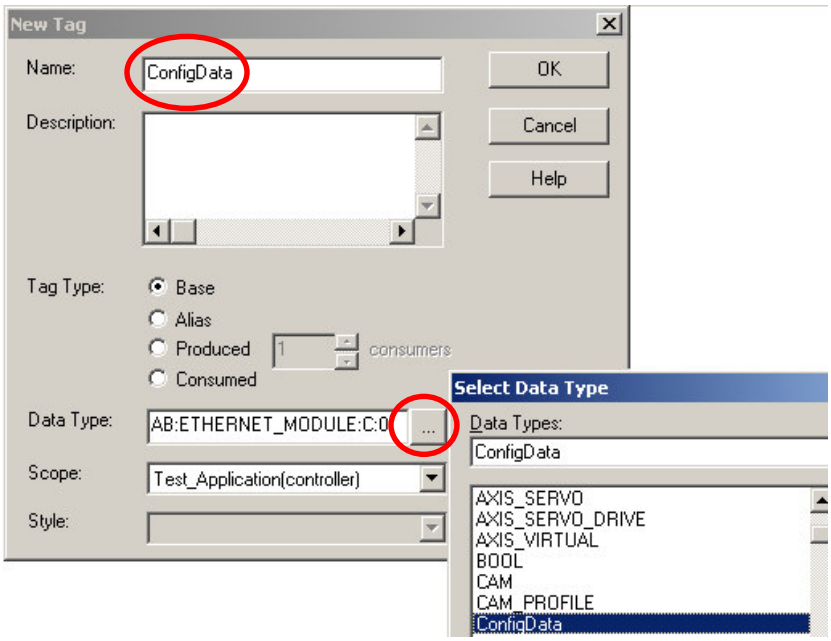
8. Add a “copy file” (COP) block to the rung the type



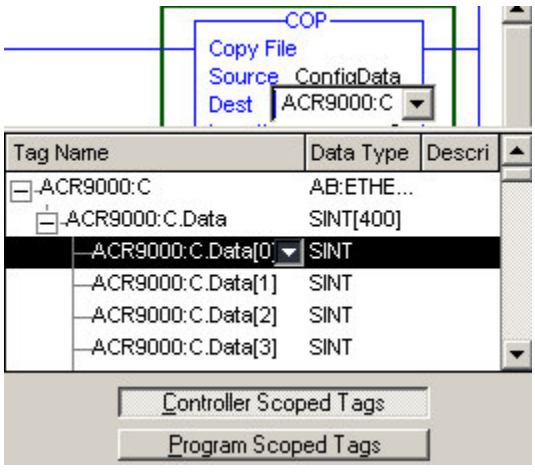
9. Right click on Source in the COP block and select New Tag:



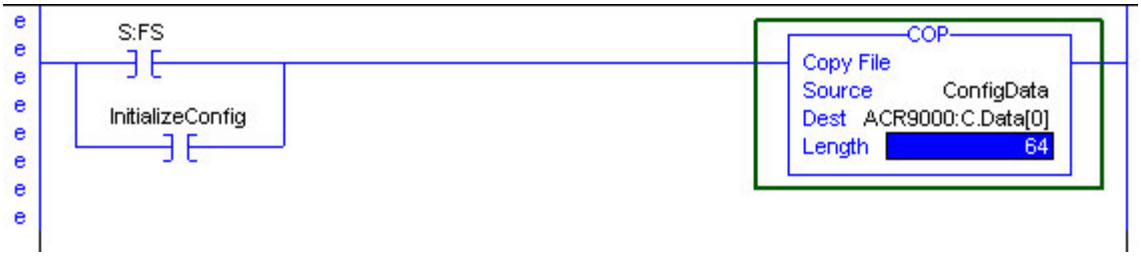
10. In the New Tag dialog box, type "ConfigData" in for the name. Then select the "…" to select the Data Type of "ConfigData" then click on OK twice, once for each dialog box:



11. Select "ACR9000.C.Data[0]" (select via navigation) as the Dest:

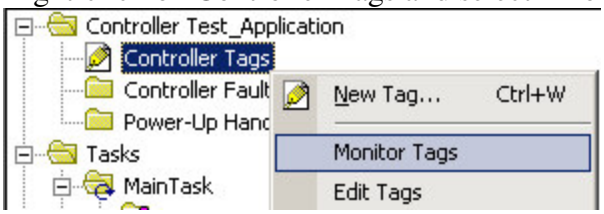


12. Enter Length as “64”. When we are done the ladder should look like:

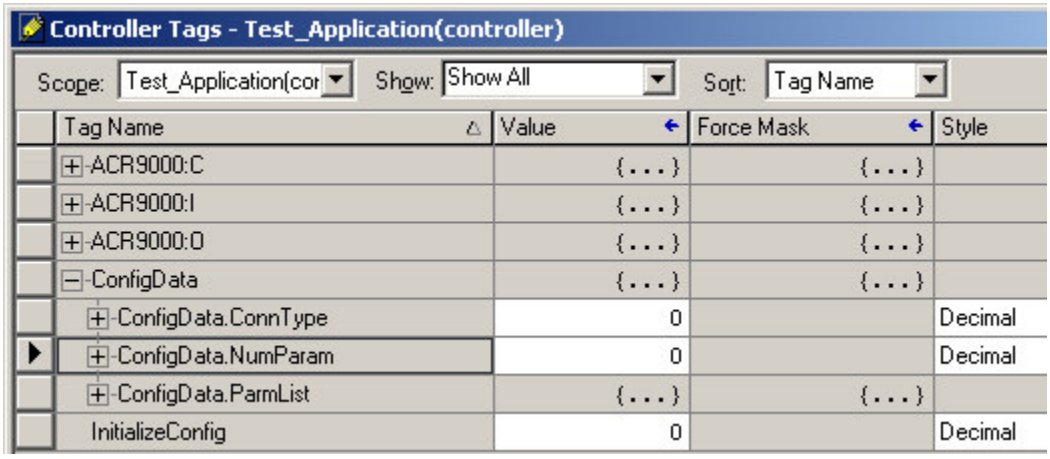


Save The Project!!!

Enter the Class 1 communication configuration information:
Right-click on Controller Tags and select “Monitor Tags”



In the Controller Tags window, expand the ConfigData tag to expose the ConnType, NumParam and ParmList members:



Enter “33” as the ConfigData.ConnType value, then set the ConfigData.NumParam value to the number of parameter groups we intend to have the AB automatically poll and set. This will be the number of groups and not the individual number of items. Remember that we can only set up 16 different groups, maximum. “Offset” is the P number, “Length” is the number of consecutive parameters to read/write, “Direction” set to “0” indicates the PLC will read from the ACR9000, “1” indicates it will write to the parameter in the ACR9000.
In this example, we will set up three different groups. The first group will retrieve P4096 – P4103. We set the ParmList[0].Offset to “4096”, the ParmList[0].Length to “8” and the ParmList[0].Direction to “0”. The second group we will set P4100 – P4103 (User Flag Groups 0 - 3) so we set that ParmList settings to 4100, 4, 1. The third group will be to set P4156 – P4159

(User Flag Groups 4 – 7) so we use 4156, 4, 1 as its settings:

-ConfigData	{...}
-ConfigData.ConnType	33
-ConfigData.NumParam	3
-ConfigData.ParmList	{...}
-ConfigData.ParmList[0]	{...}
-ConfigData.ParmList[0].Offset	4096
-ConfigData.ParmList[0].Length	8
-ConfigData.ParmList[0].Direction	0
-ConfigData.ParmList[1]	{...}
-ConfigData.ParmList[1].Offset	4100
-ConfigData.ParmList[1].Length	4
-ConfigData.ParmList[1].Direction	1
-ConfigData.ParmList[2]	{...}
-ConfigData.ParmList[2].Offset	4156
-ConfigData.ParmList[2].Length	4
-ConfigData.ParmList[2].Direction	1

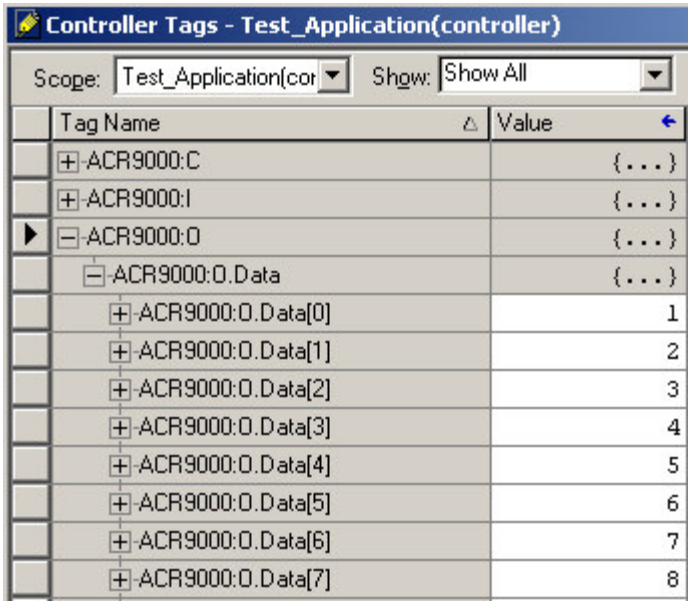
Save the project!!!!

We are now ready to download and actively “monitor” the Input and Output tags…

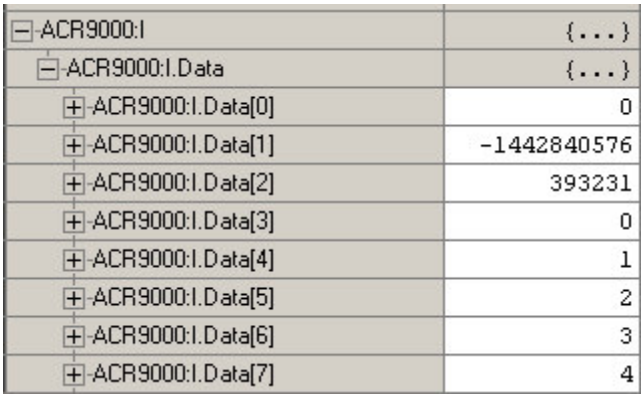
Monitoring the controllers Class 1 input and outputs:

The monitor tag utility will allow us to set and view outputs and inputs of the PLC, including the ACR controller's parameters.

We can actively set the outputs to the controller via the ACR9000:O.Data[n] monitor. Enter numbers into the Value cells of the ACR9000:O.Data[0] through ACR9000:O.Data[7]. These will be immediately written to the controller when the PLC is put into the "RUN" mode via the key or via the Remote -> Run option of the software (if the key is in the REM position.) ACR9000:O.Data[0] corresponds to P4100, ACR9000:O.Data[4] corresponds to P4156.

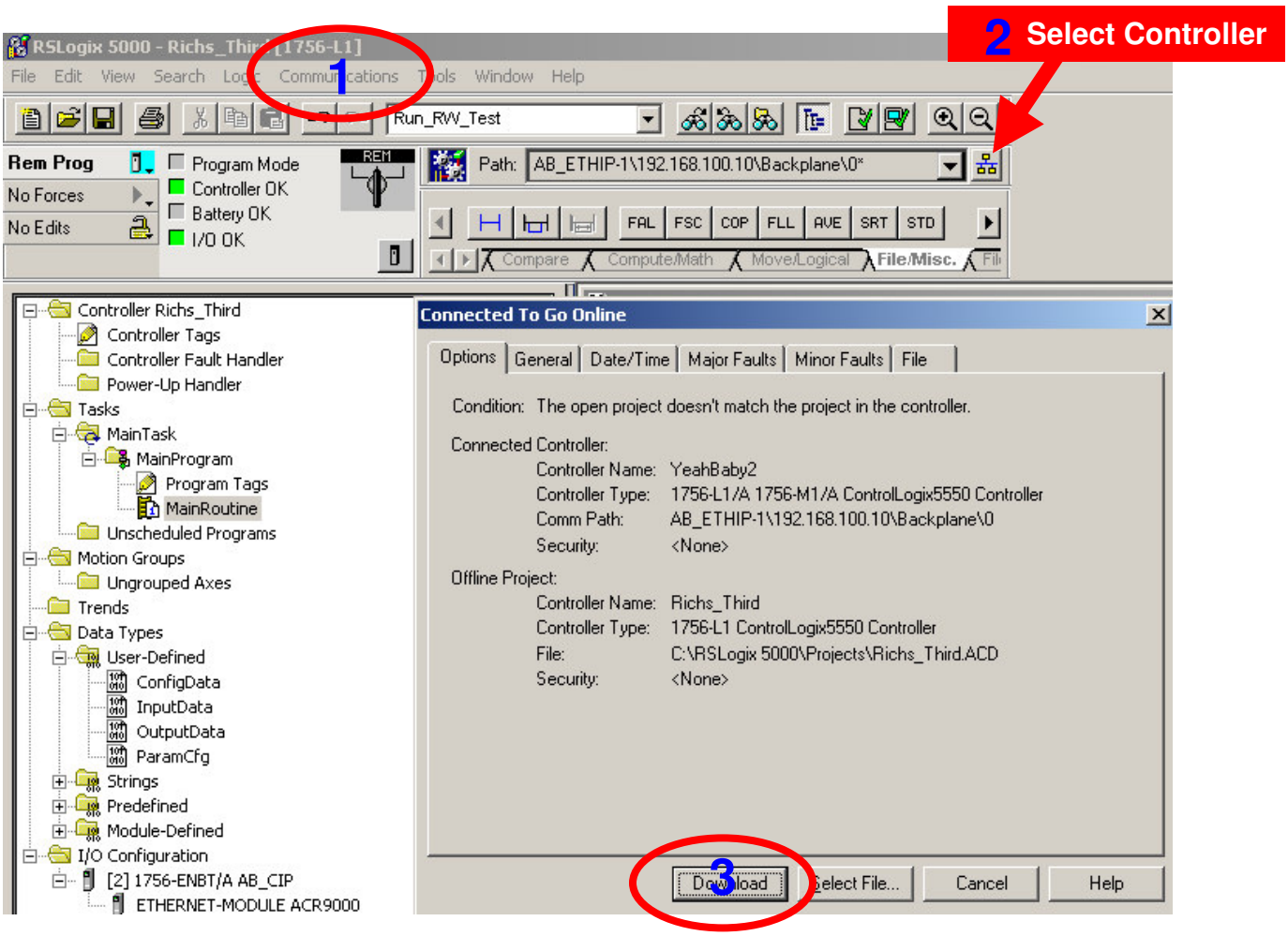


We can also monitor the parameters we have selected to read. In our setup, ACR9000:I.Data[0] corresponds to P4096, Data[1] is P4097, Data[2] is P4098, etc. Notice that we previously set ACR9000:O.Data[0] = 1, which now shows up in ACR9000:I.Data[4] as the same value:

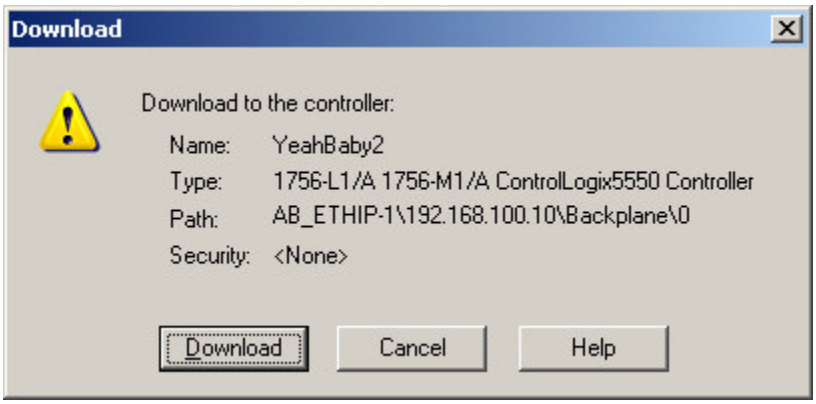


Go online with PLC and download the project:

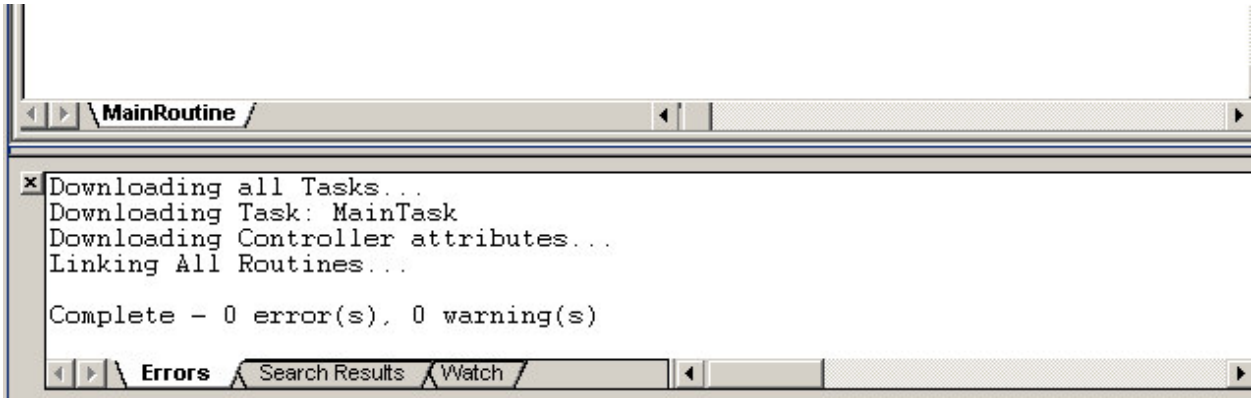
1. Under “Communications” menu, select “Go Online”
2. Select controller path
3. Click on “Download” button



Answer “Download” to the confirmation:



Be sure to look for any errors or warnings and correct them:


MainRoutine
Downloading all Tasks...
Downloading Task: MainTask
Downloading Controller attributes...
Linking All Routines...
Complete - 0 error(s), 0 warning(s)
Errors
Search Results
Watch

Adding communication drivers via RSLinx to communicate to AB PLC's: