

NETWORKING GROUP TUTORIAL

This quick tutorial will set up a group network where all class members can monitor and send data between each other's 5000 family products. This lab will show you how to set up a Modbus TCP driver and how to share variables over a Modbus TCP network. It is assumed that the student has the skills to install drivers, build screen objects, and to add tags to tag tables.

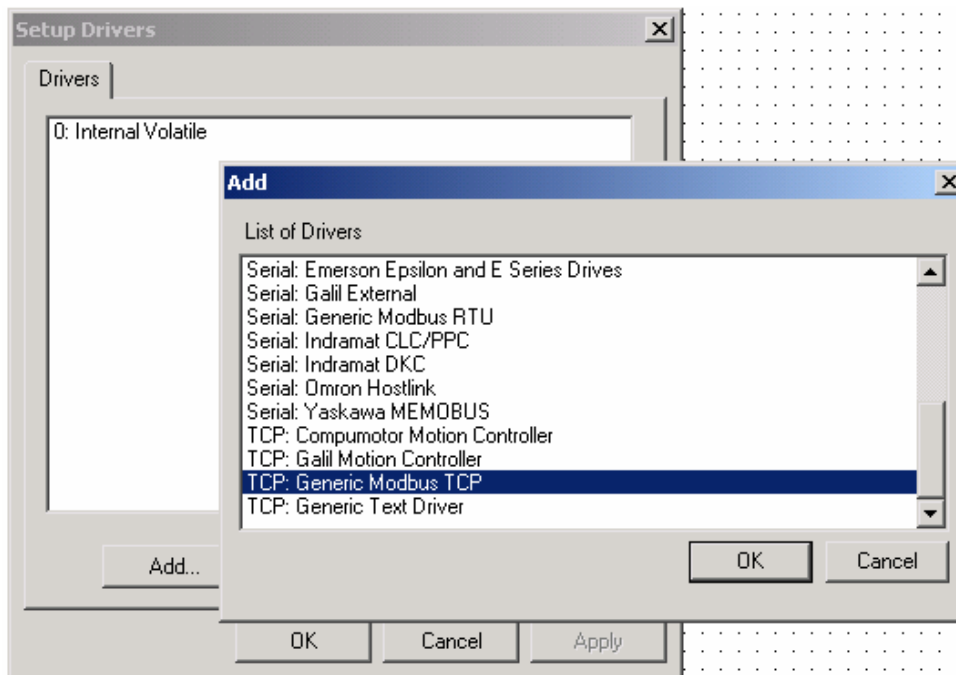
Start a new project. We will add one new driver for every 5400 in the classroom so you can talk to every one of your classmates. The instructor will write on the board the specific IP address of each 5400 in the class.

**STEP
1**

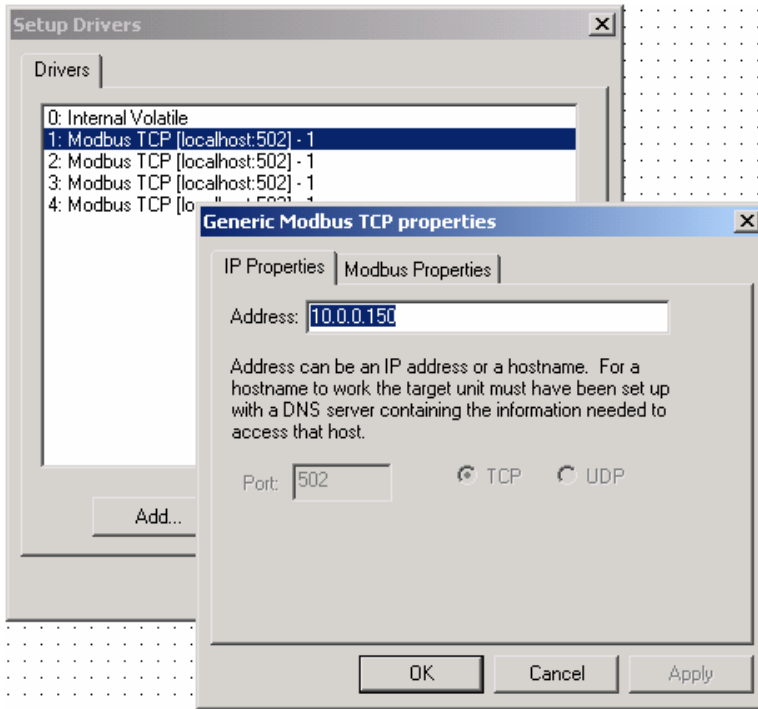
Add TCP Drivers

Choose operator terminal you would like to use. Click on "PROJECT □ HARDWARE" and select the right ET. *Changing terminal type during or after project development will NOT automatically resize or adapt objects on screen.*

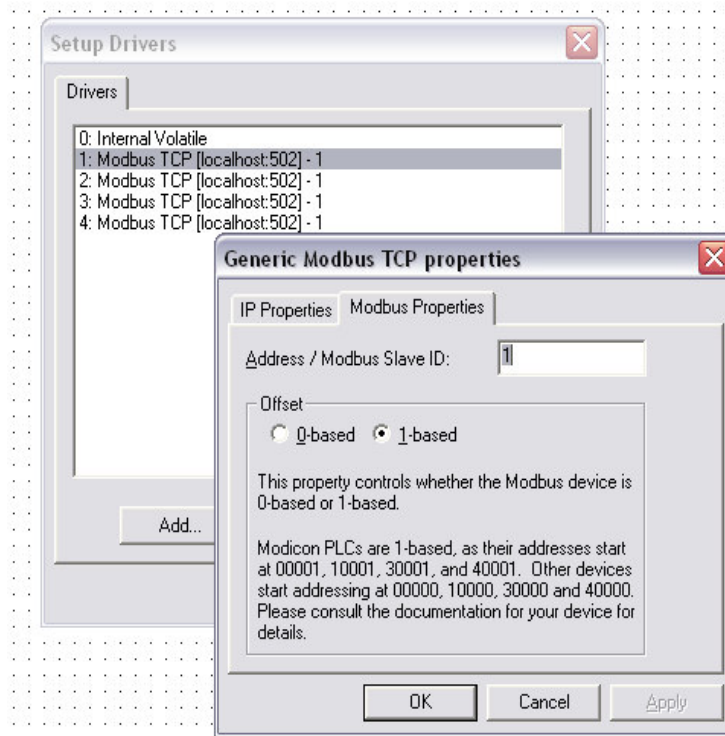
Add a TCP: Generic Modbus TCP driver for each 5000 unit in the class:



Next, configure each driver for the specific IP address that the instructor gives you:



Next, configure the Modbus Properties for each driver. We will need to set the address offset to “1-based” since that is the address model that the 5000 uses for its addressing.



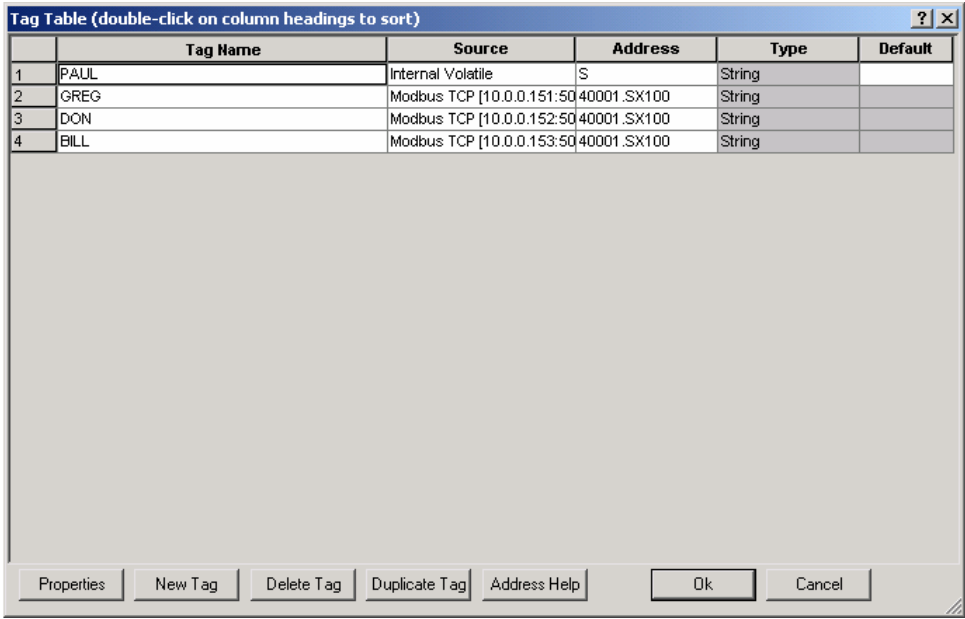
STEP 2	<h1>Add Tags</h1>
-------------------	-------------------

We now need to add tags to the program. You will add tags for your own system, and one additional tag for each other system in the classroom. For example, you are Paul, and Greg, Don and Bill are in the class and have a networked unit on their desk. Also, assume that the following addresses have been assigned:

Paul 10.0.0.150
 Greg 10.0.0.151
 Don 10.0.0.152
 Bill 10.0.0.153

You will set up four tags, one Internal Volatile, and three additional tags, one for each other member of your class. It is important that you associate the driver with the correct IP address with correct tag names for each member's unit. For example, Greg has address 10.0.0.151 therefore the driver for tag "Greg" uses the TCP driver addressed to 10.0.0.151.

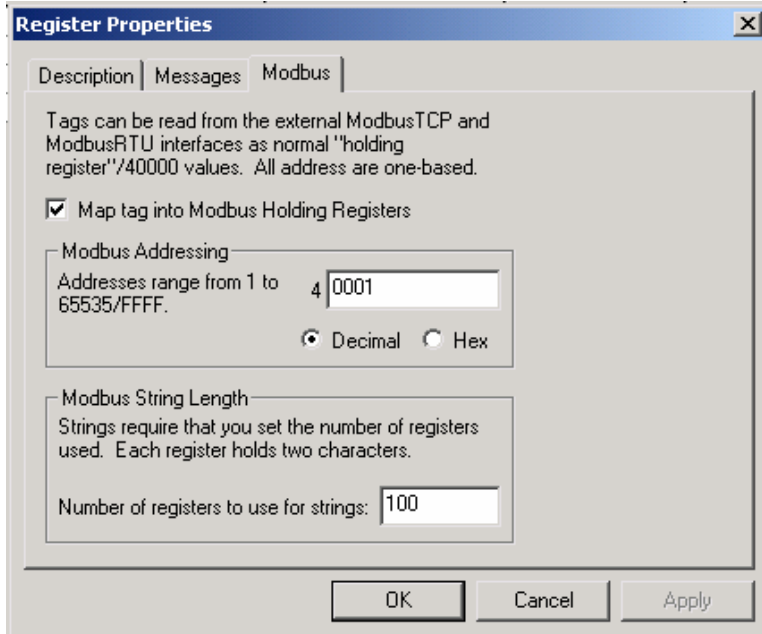
The physical address of the TCP tags that we will use for this example will be a string, at address 40001. Note that we are using the nomenclature "40001.SX100". This means that at address 40001 there are 100 string registers there that have the byte order defined by the X parameter (low order-high order). Your tag table will look something like this:



Tag Table (double-click on column headings to sort)					
	Tag Name	Source	Address	Type	Default
1	PAUL	Internal Volatile	S	String	
2	GREG	Modbus TCP [10.0.0.151:50	40001.SX100	String	
3	DON	Modbus TCP [10.0.0.152:50	40001.SX100	String	
4	BILL	Modbus TCP [10.0.0.153:50	40001.SX100	String	

And note if you are networking other data types, like a Real (Double Real) then you'll use the address notation of "DRX."

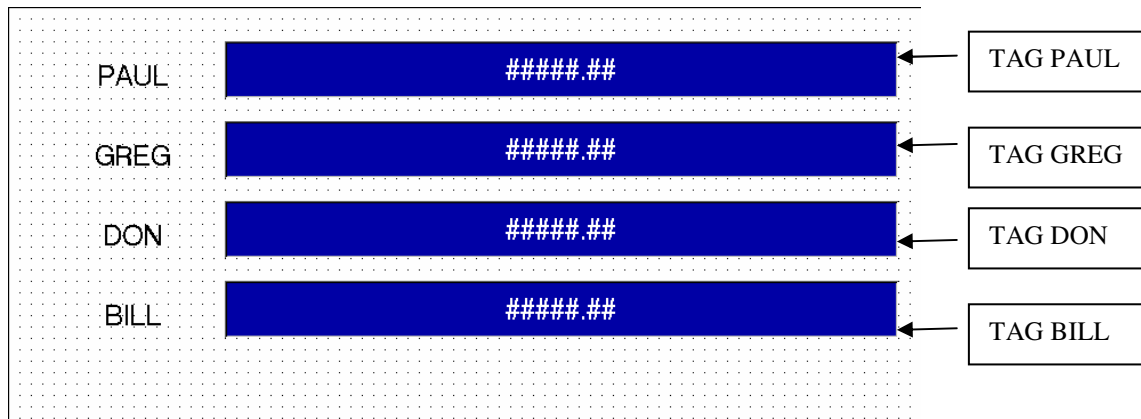
Before you leave the tag table, we have to set the object property of your own tag (internal volatile), tag Paul in this instance. Click on the Modbus tab and check the “Map tag into Modbus Holding Registers” box. Make sure that the address is 40001, Decimal and the Number of registers to use for strings is 100. It is extremely important that all units have the same address and string length.



Click OK to close the property dialog, and again click OK to close the Tag Table dialog box.

STEP 3	<h2>Create Screen</h2>
-------------------	------------------------

You now will have to create a screen that allows data entry for each of the tags you created. It should look something like this:



STEP
4

Let's Try It!

Go ahead and download the program to the unit. Once downloaded, you should be able to enter a tag on your Non-Volatile data entry (Paul in our case), and everyone else should be able to see it. Likewise, you should be able to see everyone else's tags as well.