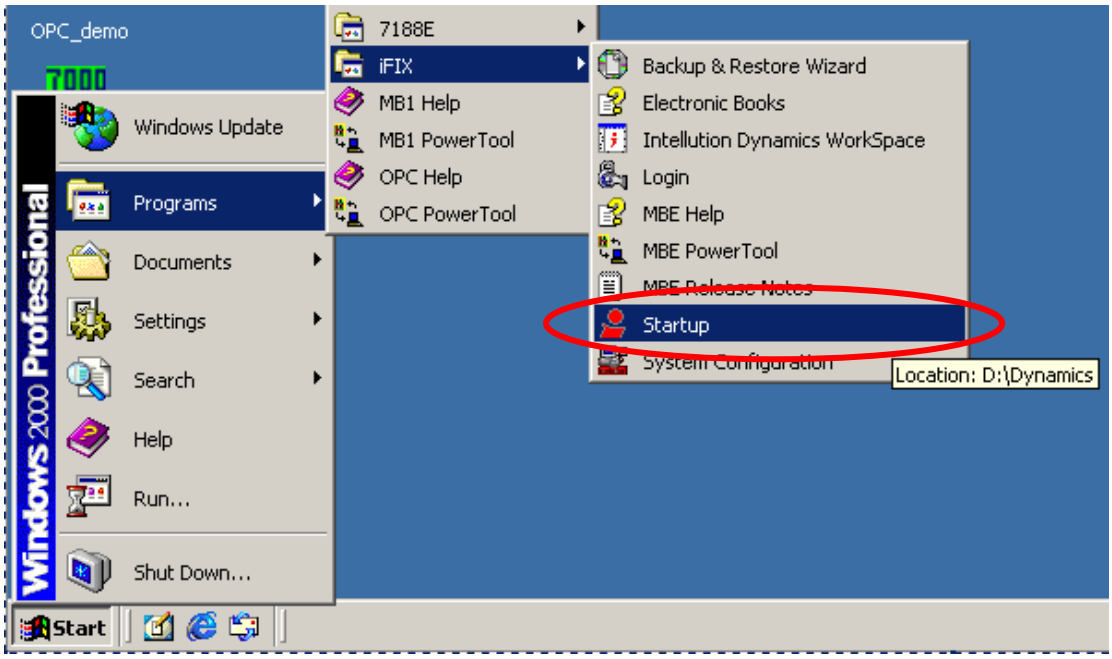
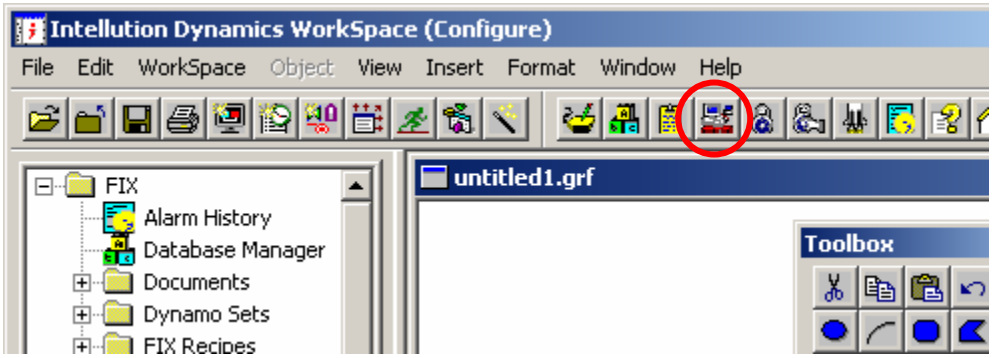
The connection solution of iFIX with ET-7000 series modules by using Modbus-TCP protocol

iFix is a kind of powerful SCADA software, and is generally implemented in the application of industrial control. In the following section, we will show you about how to use iFIX ver2.1 to connect ET-7000 series modules by using Modbus-TCP protocol, and guide you to finish a simple project step by step.(Attention: Before you use Modbus-TCp protocol, you must nstall the Modbus-TCP drivers-"MBE" first.)

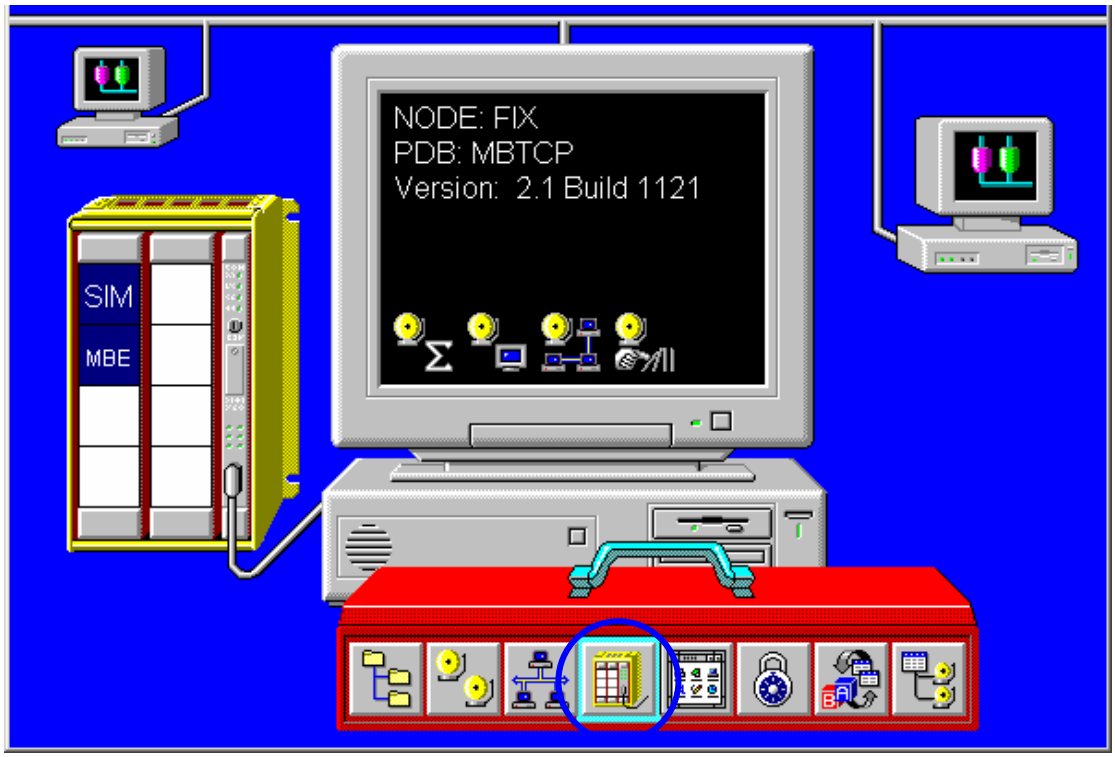
Step 1: Startup the iFix SCADA software



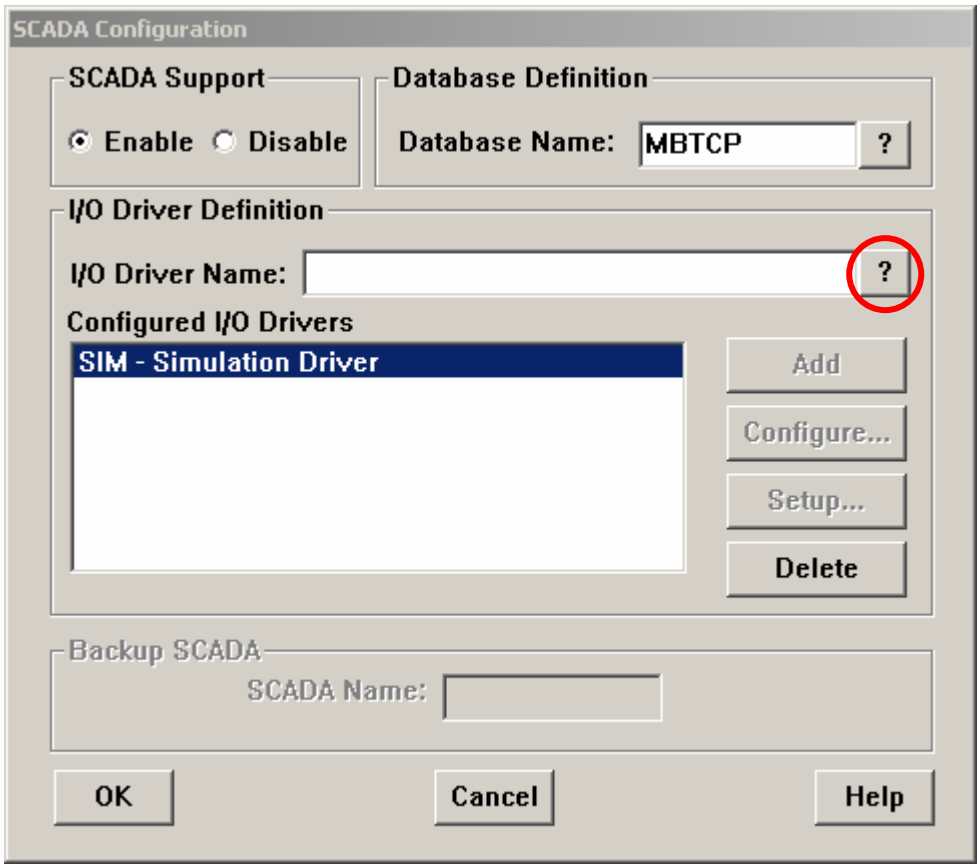
Step2: Click button to set the configuration parameters of a device.



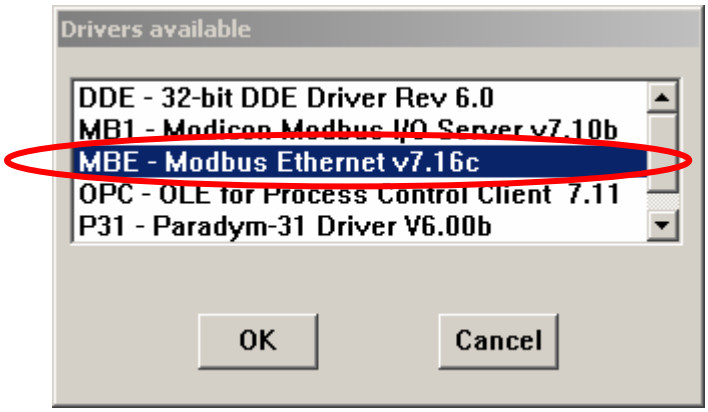
Step3: Click button to set the communication parameters and I/O devices.



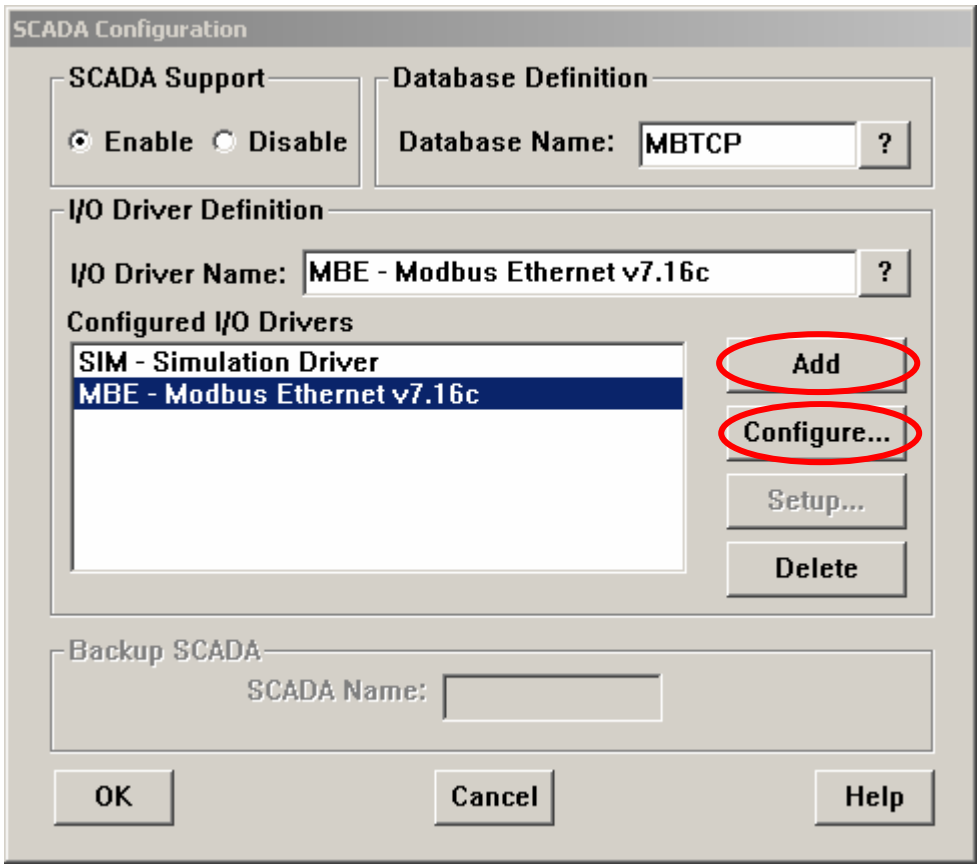
Step4: Click button to select the communication method.



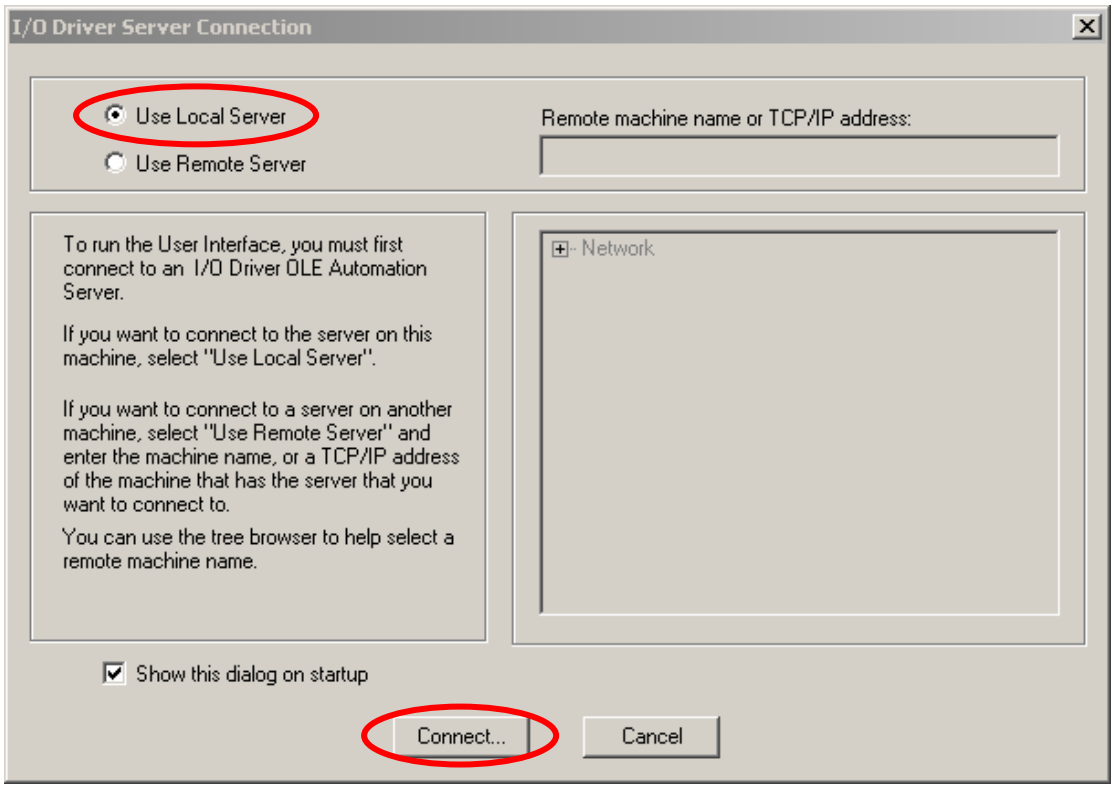
Step5: Select the item “MBE-Modbus Ethernet v7.16” in the list box, and click Ok the finish it**.**



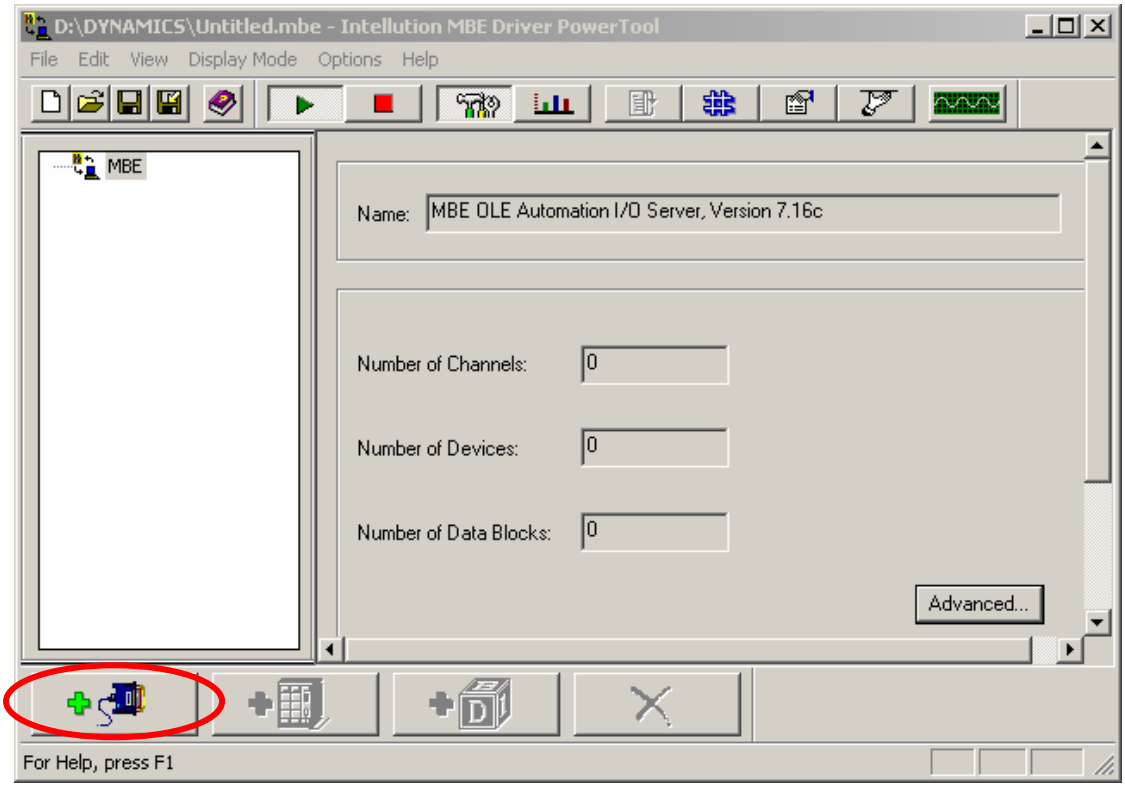
Step6: Click “Add” button to add MBE - Modbus Ethernet v7.16 into configured I/O Drivers, and then click “Configure…” button to do the I/O drivers parameters setting.

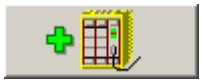


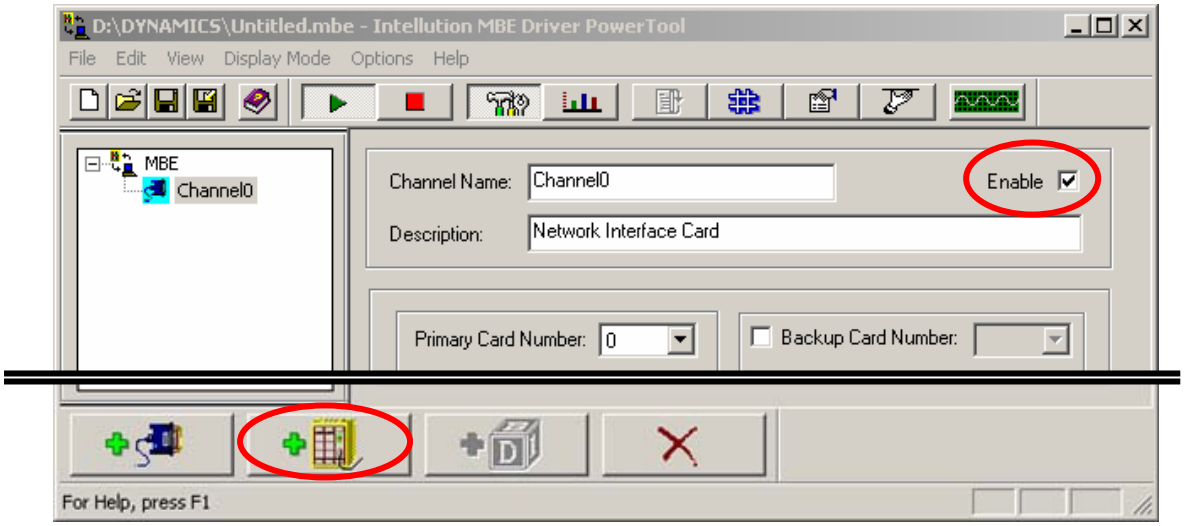
Step7: Select "Use Local Server" and Click "Connect…" button to continue the other setting.



Step8: Click button to build a new connection channel.

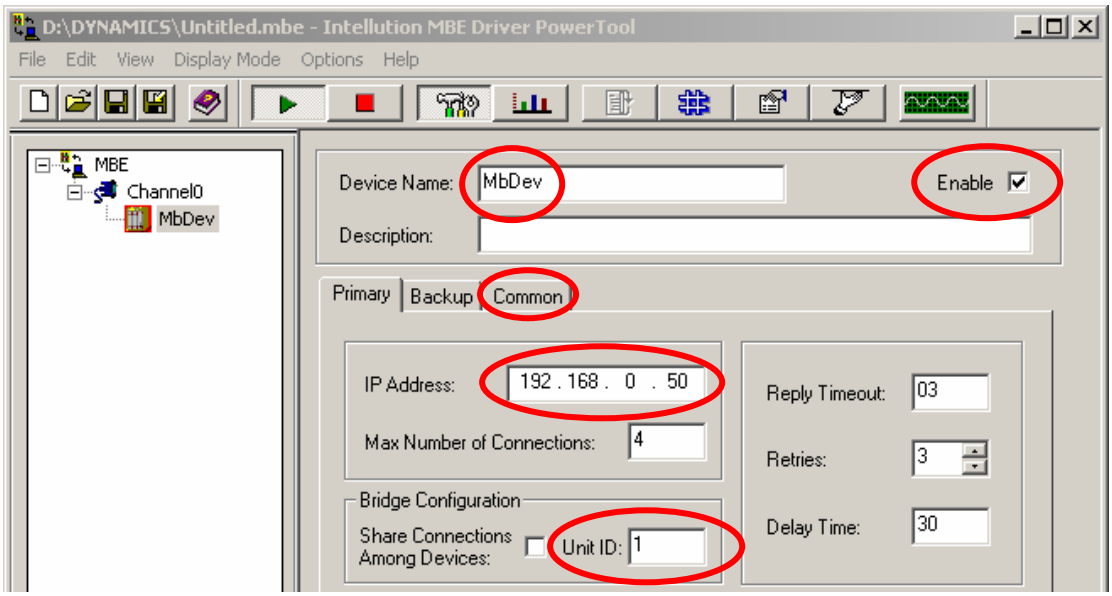


Step9: Check the "Enable" check box to enable the Channel0, and then click  button to add a new device.

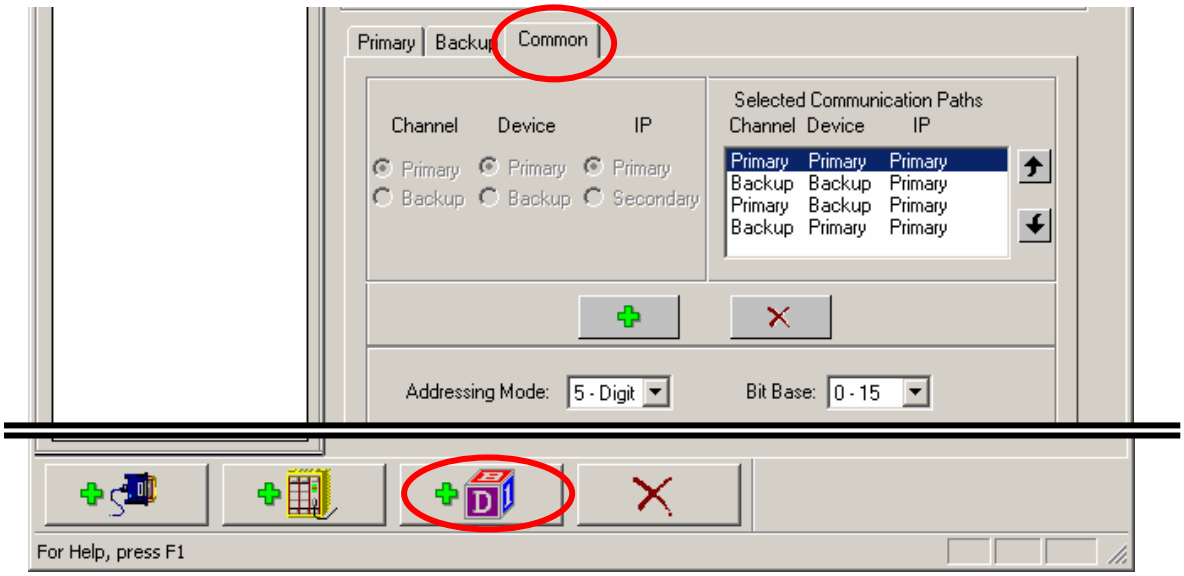


Step10: Fill the device name which you want in the field of Device Name, we use "MbDev" to be the name of this device here. Fill the IP address of ET-7000 in the field of "IP Address". If the ET-7000 has Modbus-TCP firmware inside. The IP address of ET-7000

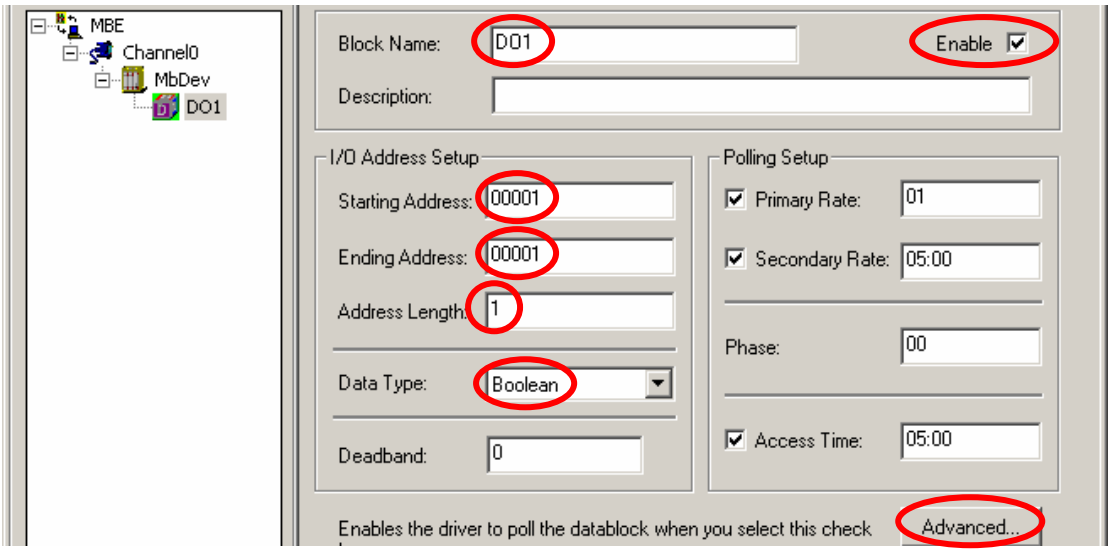
will be showed on the LED in the left hand side of ET-7000. Fill the station number of ET-7000 in the filed of “Unit ID”. The station number is the switch value in the right hand side of ET-7000.After finish the setting described above, click the “Enable” check box to enable this device. Then click the “Common” tag to modify the other parameters of MbDev.



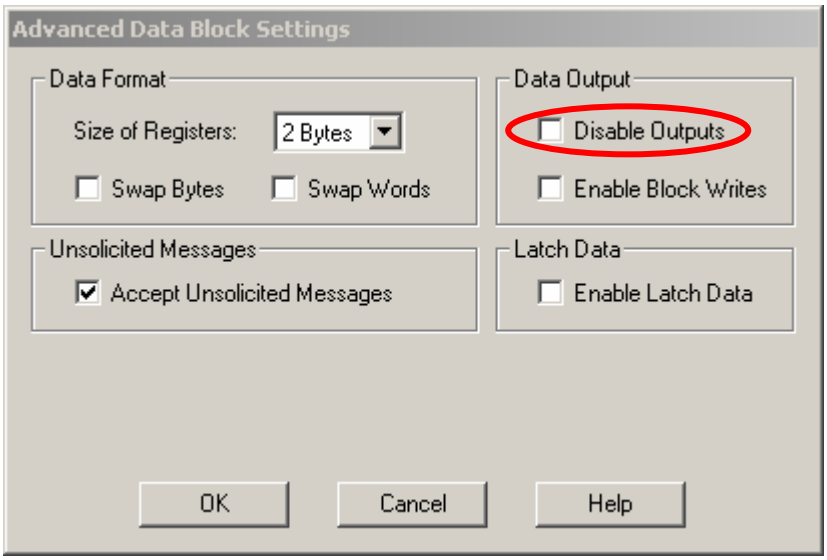
Step11: Select “5-Digit” in the “Addressing Mode” combo box to rule the address length, then click button to build a new block.



Step12: Fill the block name which you want in the field of “Block Name”, we use “DO1” to be the name of first block here. Select “Boolean” for the data type. Fill the “Starting address”, “Ending address”, and address length in corresponding fields. Then click “Enable” check box, and click “Advanced… ” button to make sure the correct setting of other parameters.(Attention: see the note of Step17)

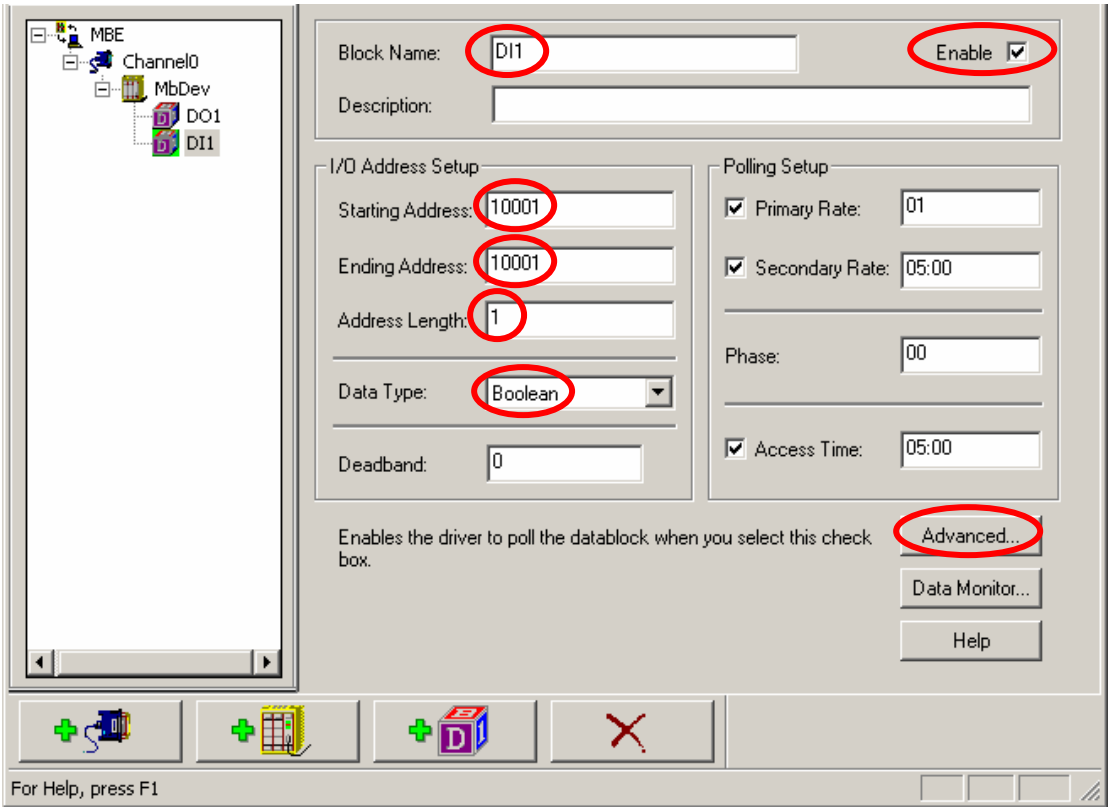


Step13: Because this block is used for output channels, you should make sure that the “Disable Output” check box to be blank. Then, click “ok” button to finish this dialog, and click button to add second block.

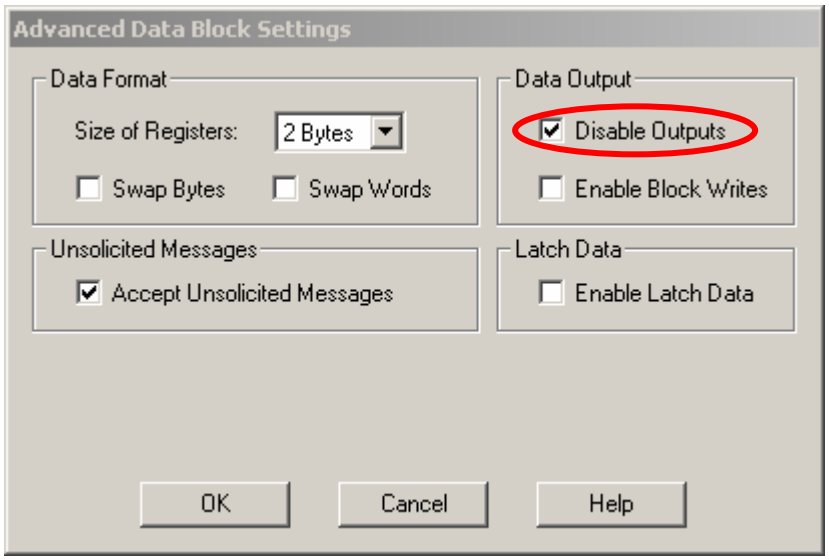


Step14: Following the “Step 12” as a model to set the parameters of second block to be DI1 block. Then click “Enable” check box,

and click “Advanced… ” button to make sure the correct setting of other parameters.(Attention: see the note of Step17)

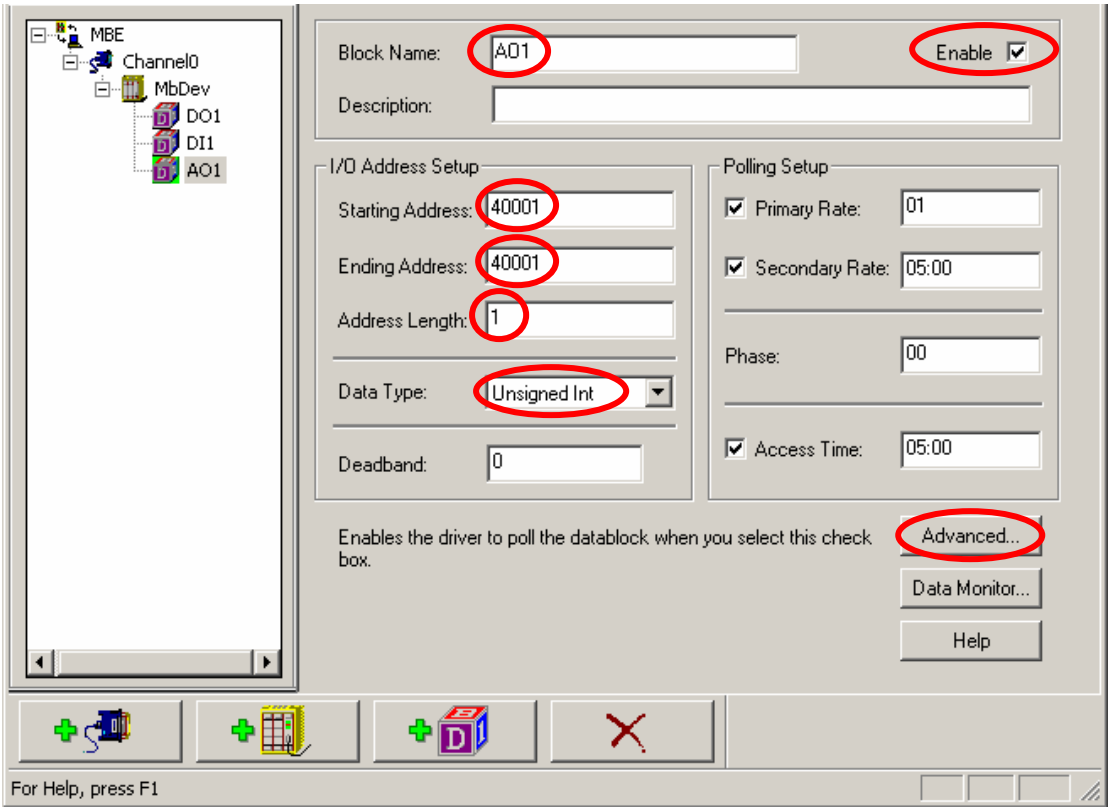


Step15: This block is used for input channels, so you should make sure that the “Disable Output” check box to be checked. Then, click “ok” button to finish this dialog, and click button to add second block.

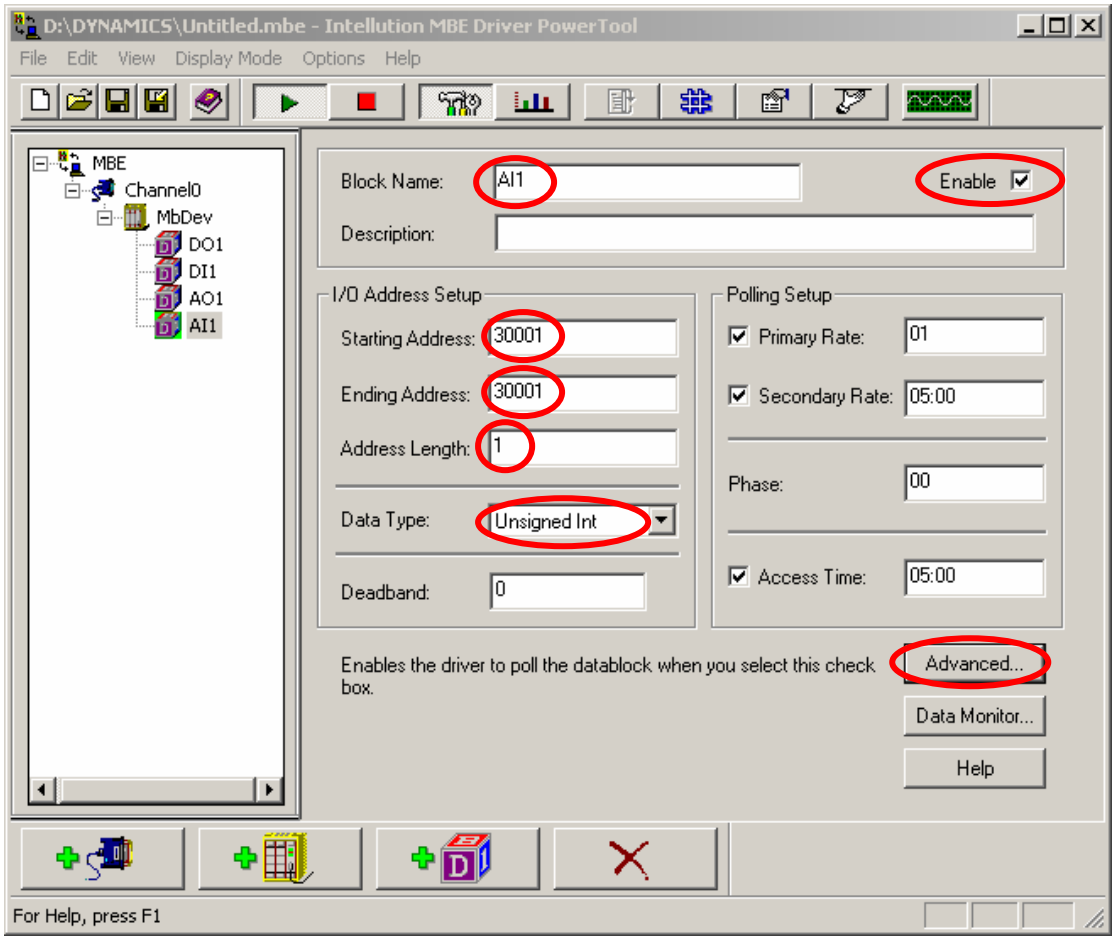


Step16: Following the “Step 12 and Step13” as a model to set the parameters of 3rd block to be AO1 block. After finish the

parameter setting of 3rd block, add the 4th block for AI channel.(Attention: see the note of Step17)

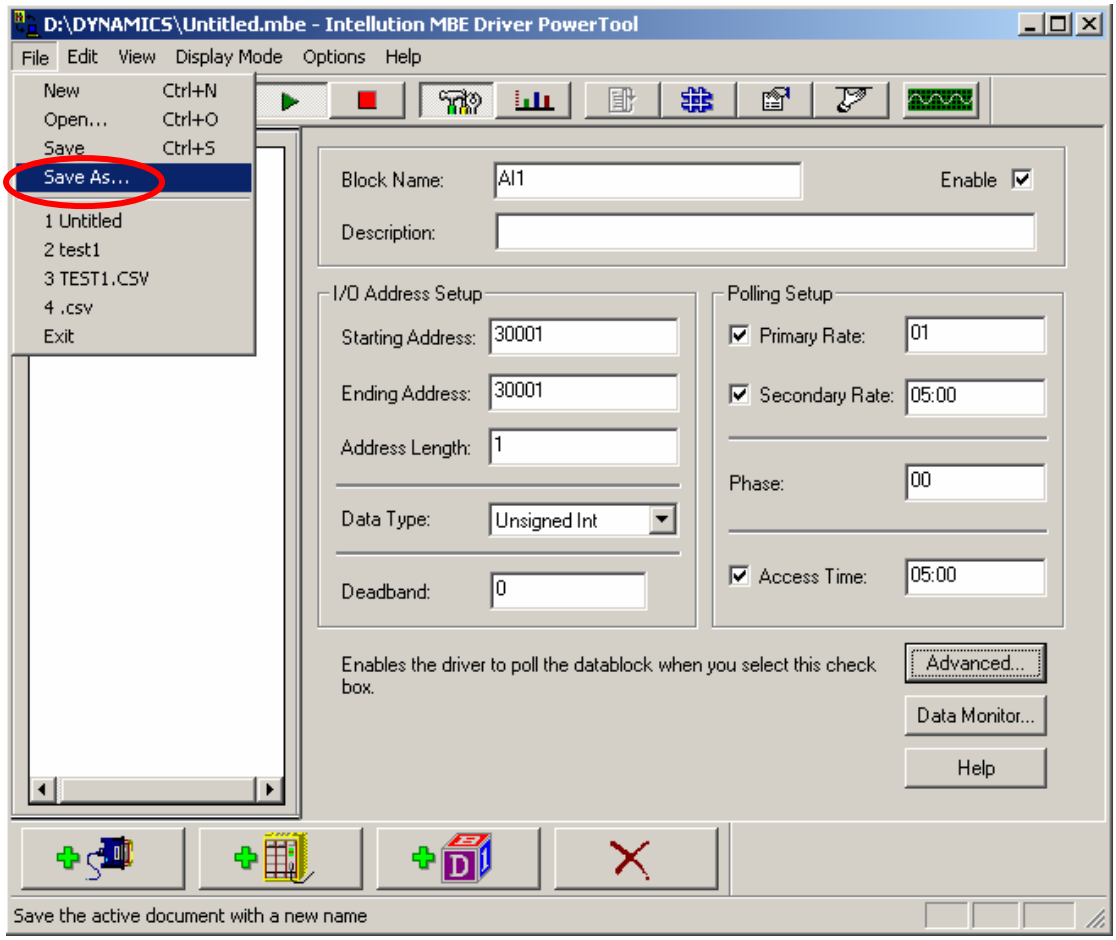


Step17: Following the “Step 12 and Step13” as a model to set the parameters of 3rd block to be AI1 block.

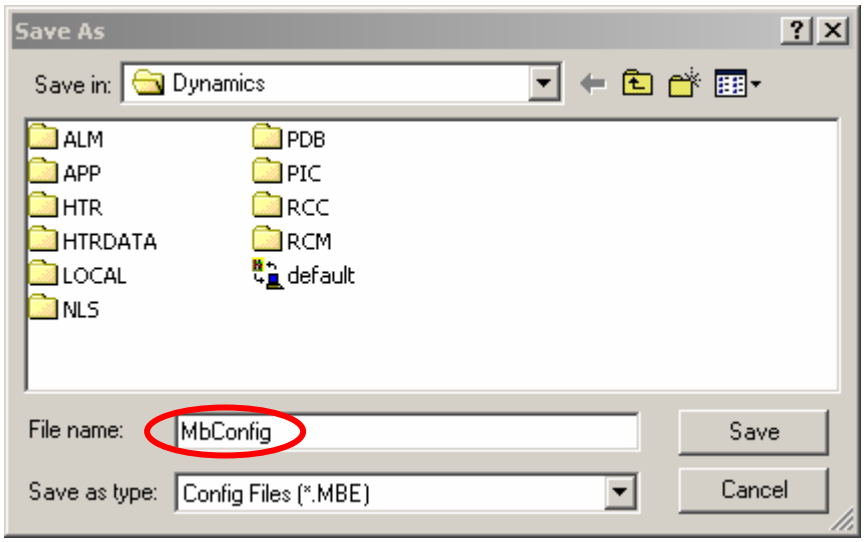


Note: In Modbus protocol, the type of digital output is set by “0”, digital input is set by “1”, analog output is set by “4” and analog input is set by “3”. The data type of DI and DO use Boolean to present, and data type of AI and AO use unsigned integral to present. The “Starting address” begin at “0001”. So if you want to use 5th channel of total digital input channels in ET-7000 series modules, you must give the “Starting address” to be “10005”, and select Boolean to be the data type. The “Ending address” is the same with “Starting address”. If you want to use the 15th channel of total analog input channels in ET-7000 series modules, you must set “30015” to be “Starting address”, and Select “Unsigned Int” to be the data type. The “Ending address” is the same with “Starting address”.

Step18: After finishing all parameter setting, click “Save As” to save the result.

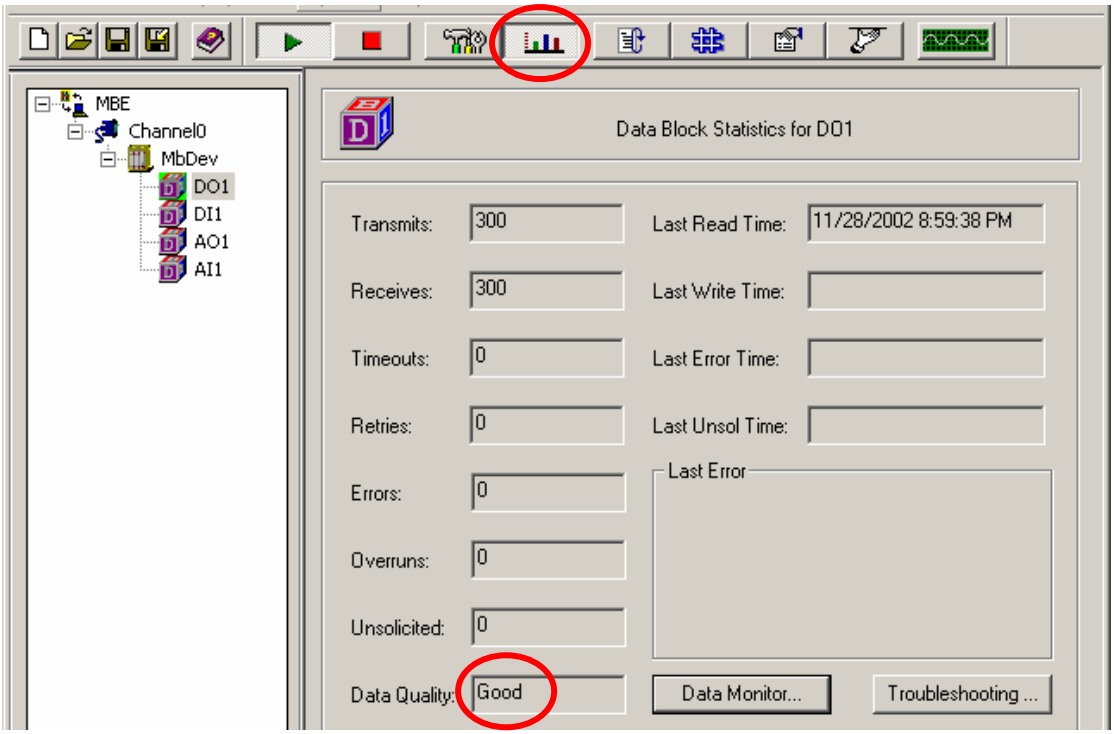


Step18: The file name is called “MbConfig”. Click “Save” to the file.

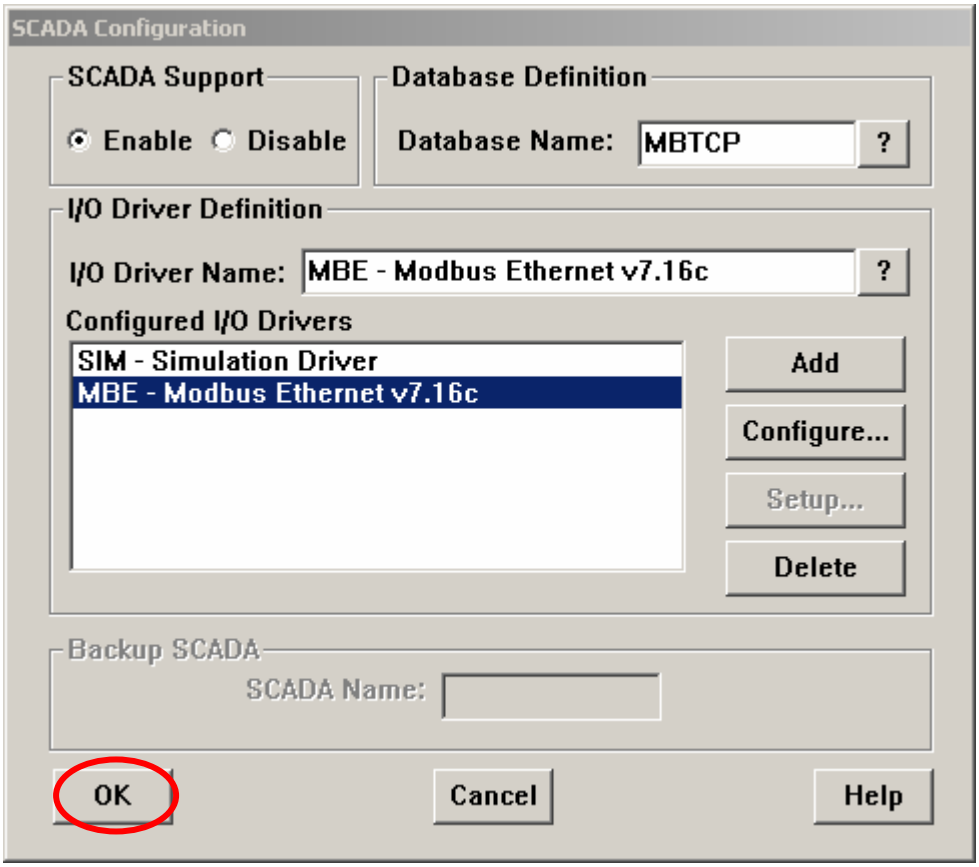


Step19: After finishing parameter setting, you can click button to see the connection state. If the “Data Quality” shows “Good”.

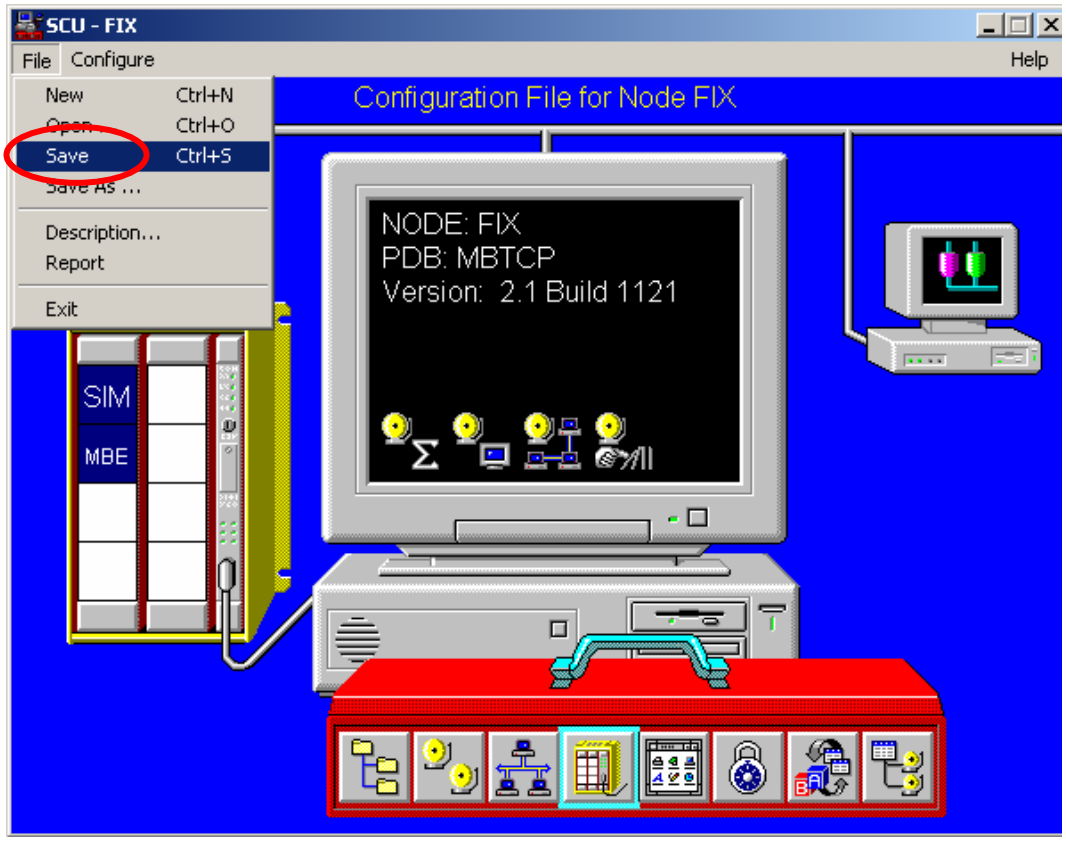
That means the connection state is good.



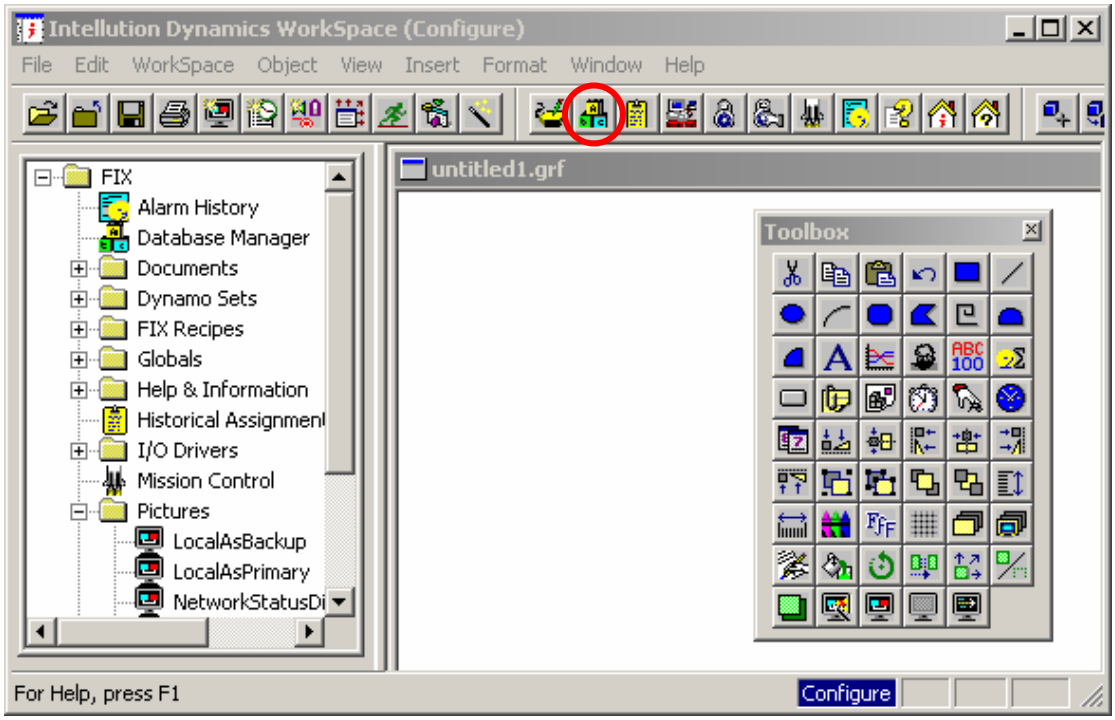
Step20: Click “OK” to finish the configuration of I/O driver.



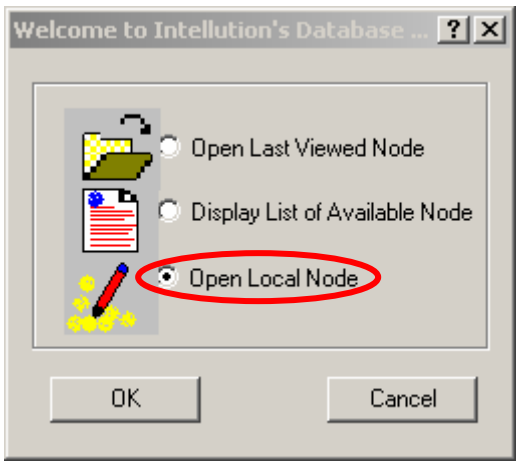
Step21: Click “Save” to store the result of system configuration.



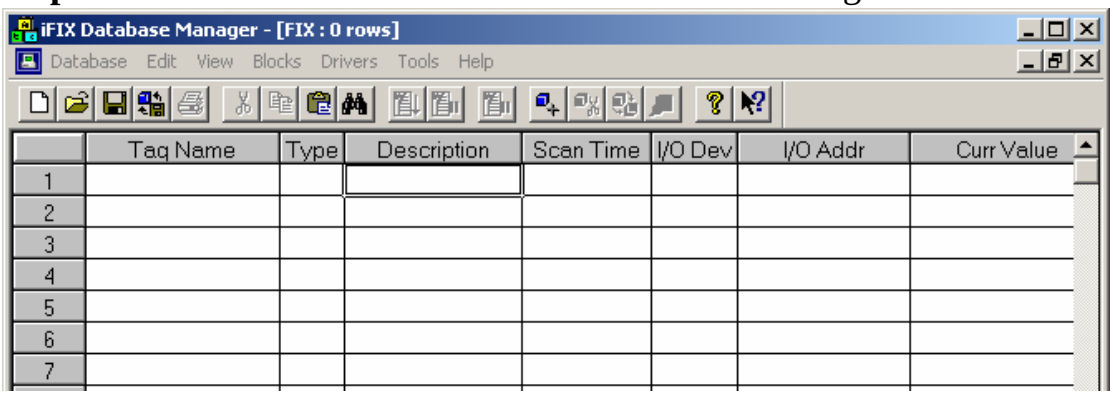
Step22: Click button to build the database.



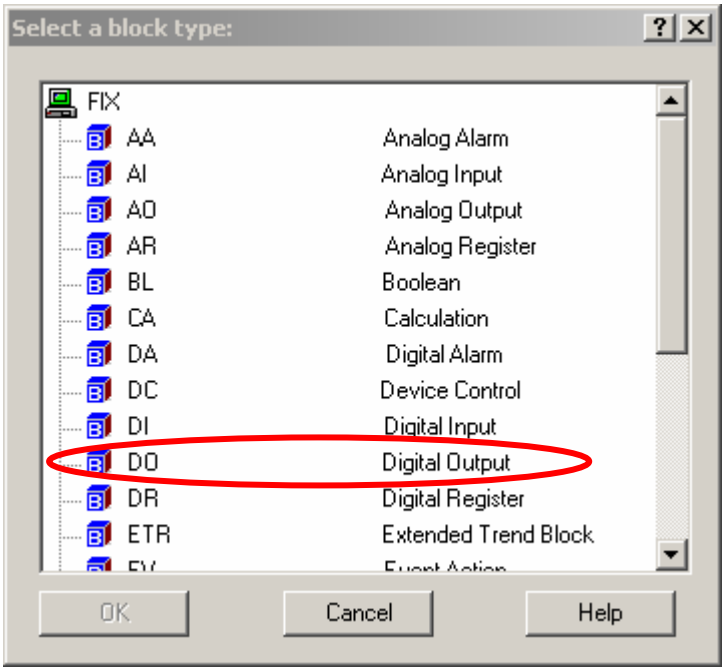
Step23: Select “Open Local Node” to continute.



Step24: Double click in the blank filed to build a new tag.

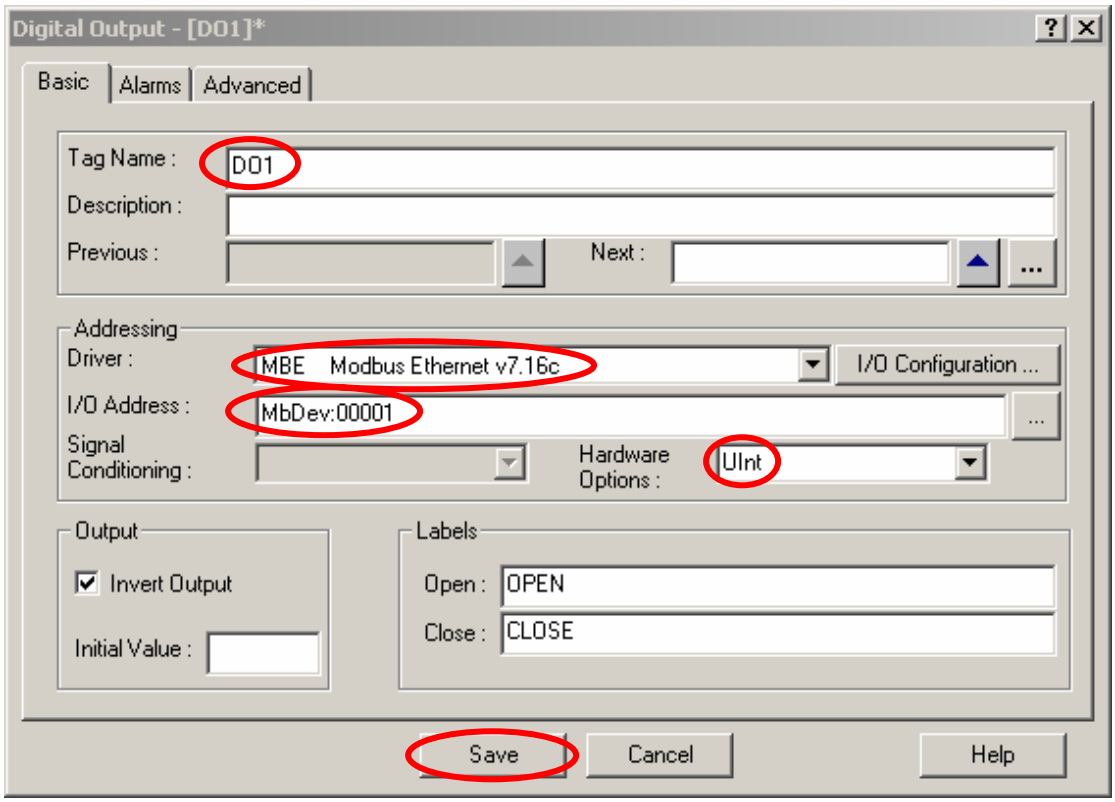


Step25: Select “DO” to build a digital output tag.

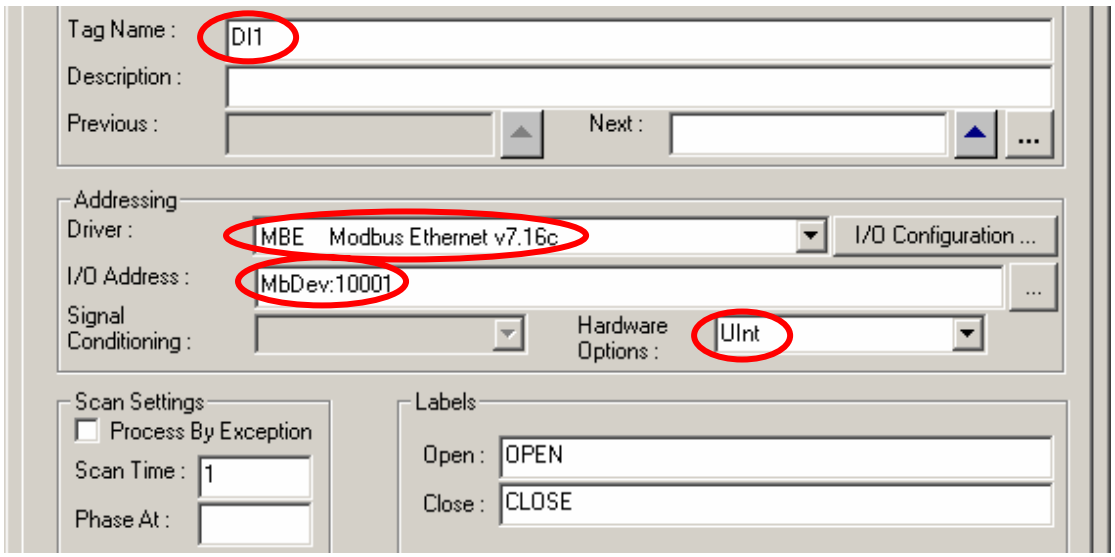


Step26: Creat a tag name which you want and fill it in the “Tag Name”

field. Select Driver to be “MBE Modbus Ethernet v7.16c”, and “I/O Address” to be “MbDev:00001”. The first part of “I/O Address” is device name set in step10, and second part is block address set in step12. Select “UIni” to be “Hardware Options”, and click “Invert Output” check box to inverse the output value. Then click “Save” button to save this tag.

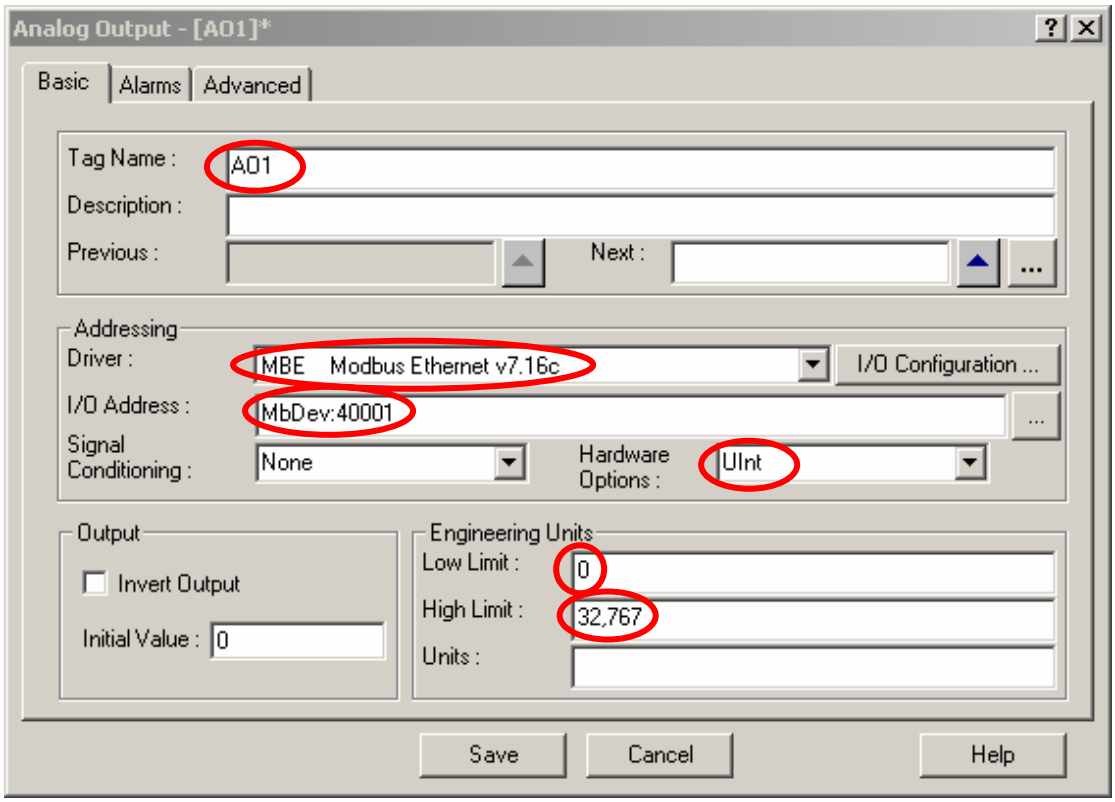


Step27: Following the “Step25” and “Step26” as a model, Select “DI” to build a new digital input tag, and set parameters as following.

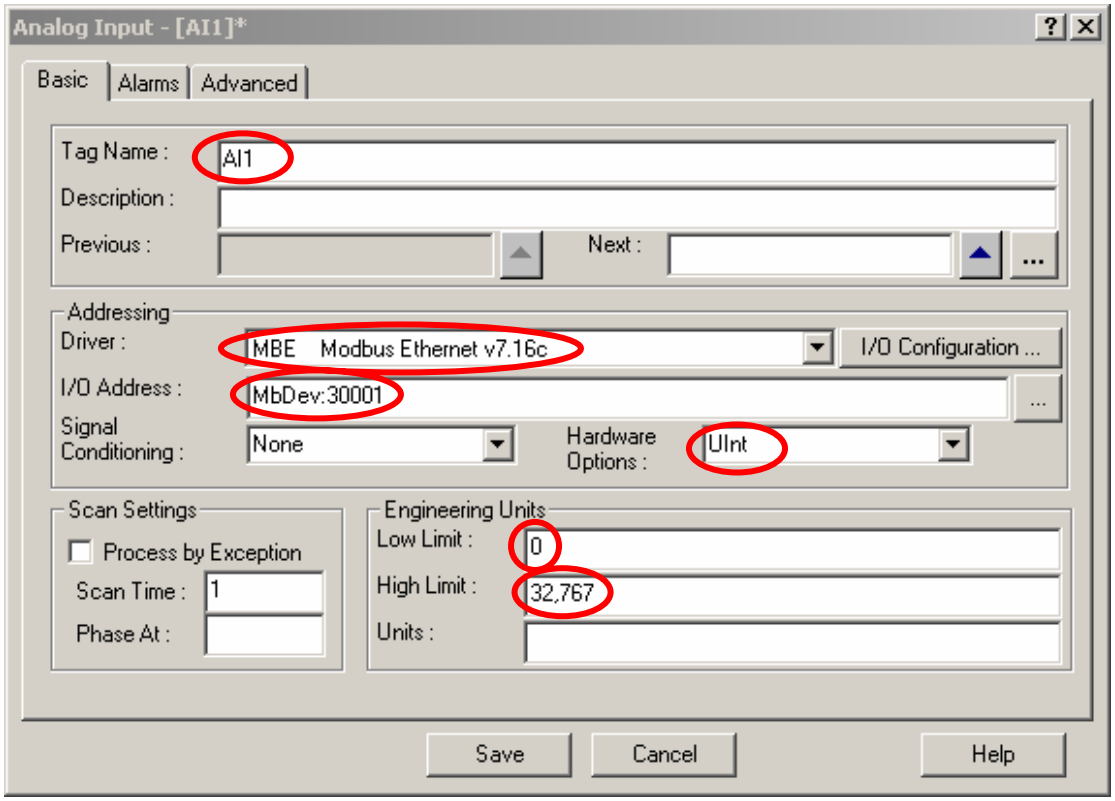


Step27: Follow “Step25” and “Step26” as a model, select “AO” to build a

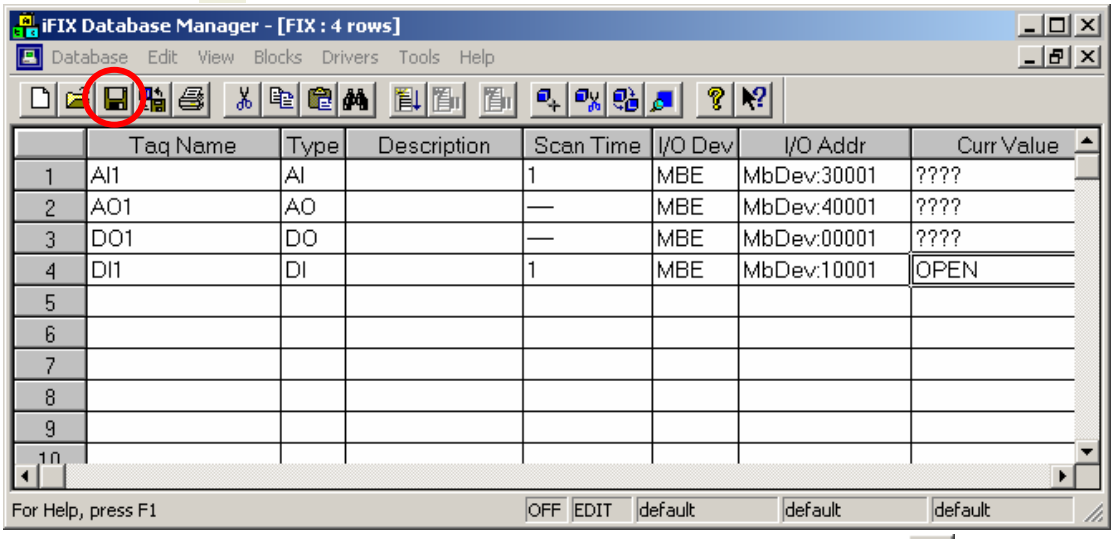
new analog output tag, and set parameters as following. Be careful that the value of “High Limit” and “Low Limit” fields to be “32767” and “0”.



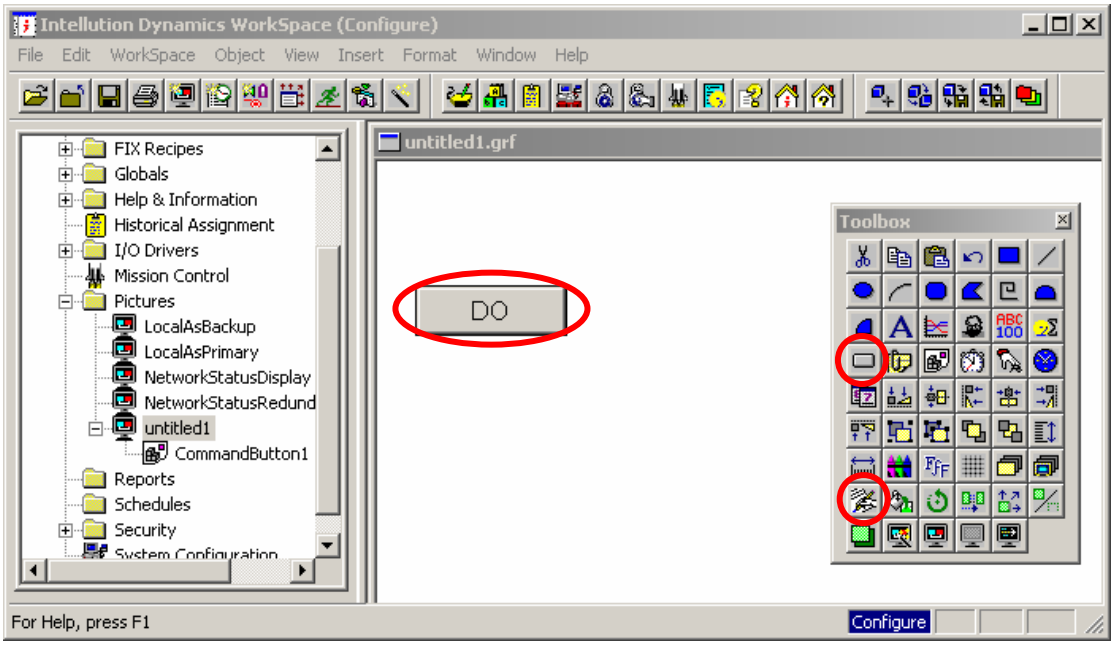
Step28: Set parameters as following by the same way for AI tag.



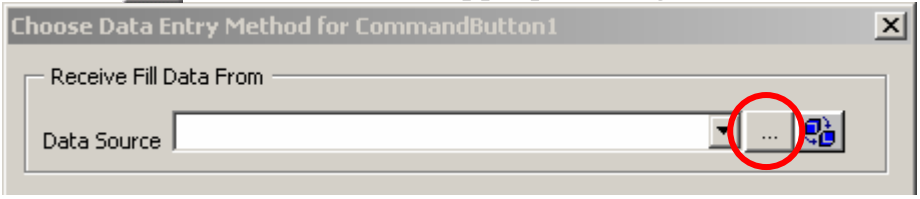
Step29: After finishing all tag setting, the database will show as bellow. Click button to save this database.



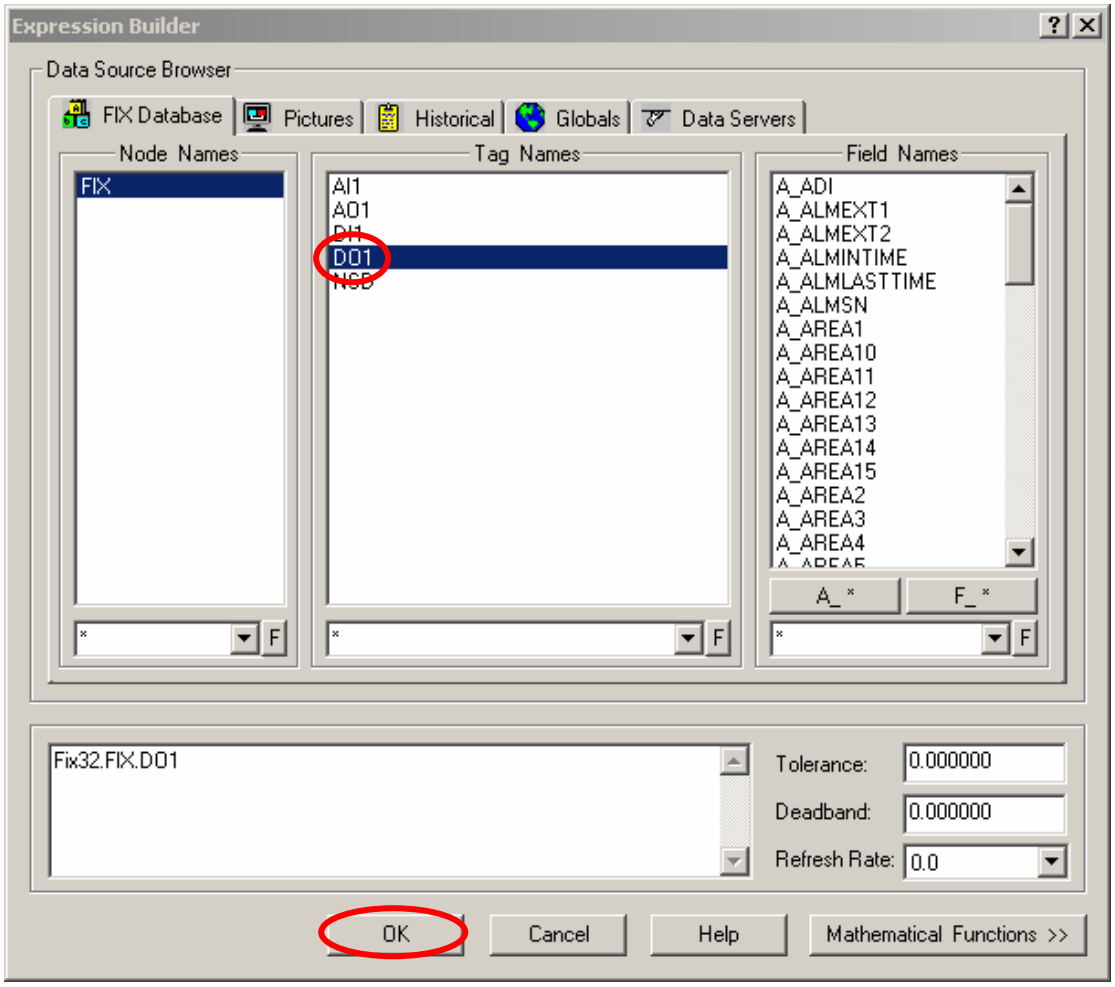
Step30: Build fore objects in a new picture. First, click button to build a "Push Button" object. When the "Push Button" object is put on the picture, double click on this object and input the words-"DO". Afterward, click button to set the action of this object.



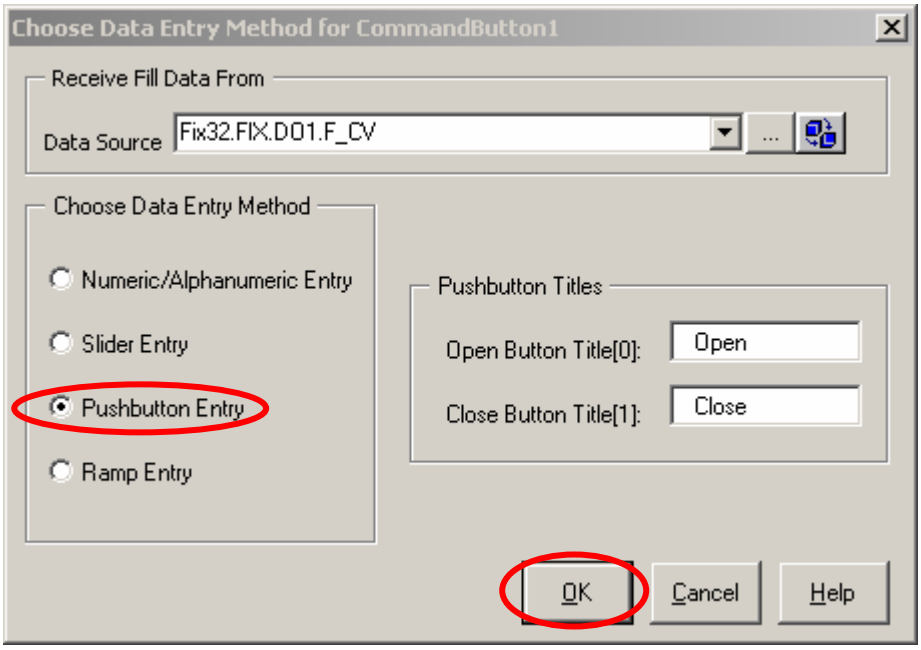
Step31: Click ... button to select appropriate tag in the database.



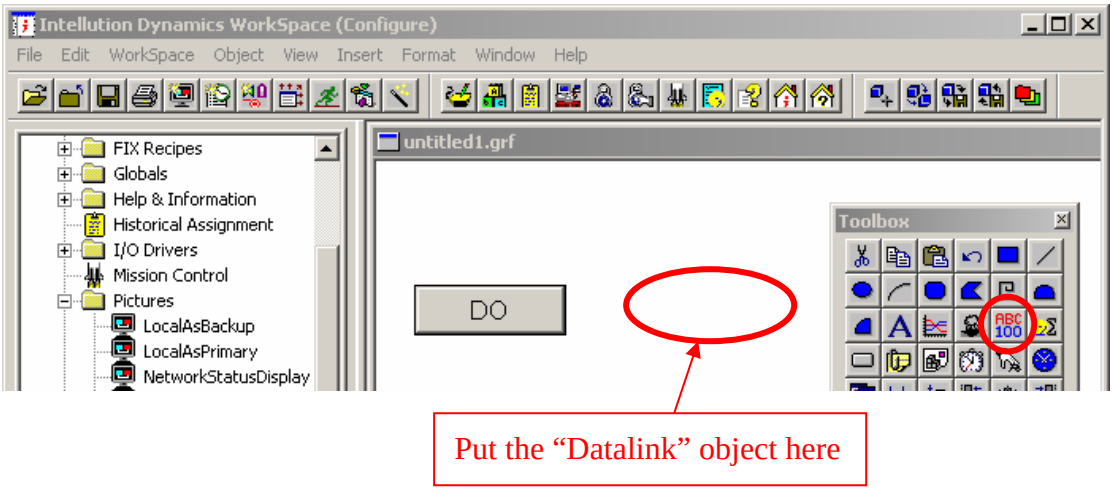
Step32: Select “DO1” in the “Tag Name” list box. Then click “Ok”.



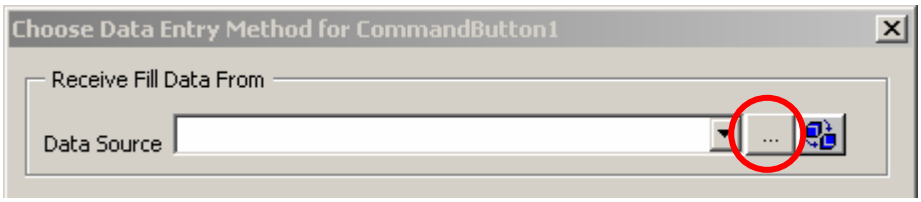
Step33: Select “Pushbutton Entry” to be the output mode of “Push Button” object, and then click “OK” to store the result.



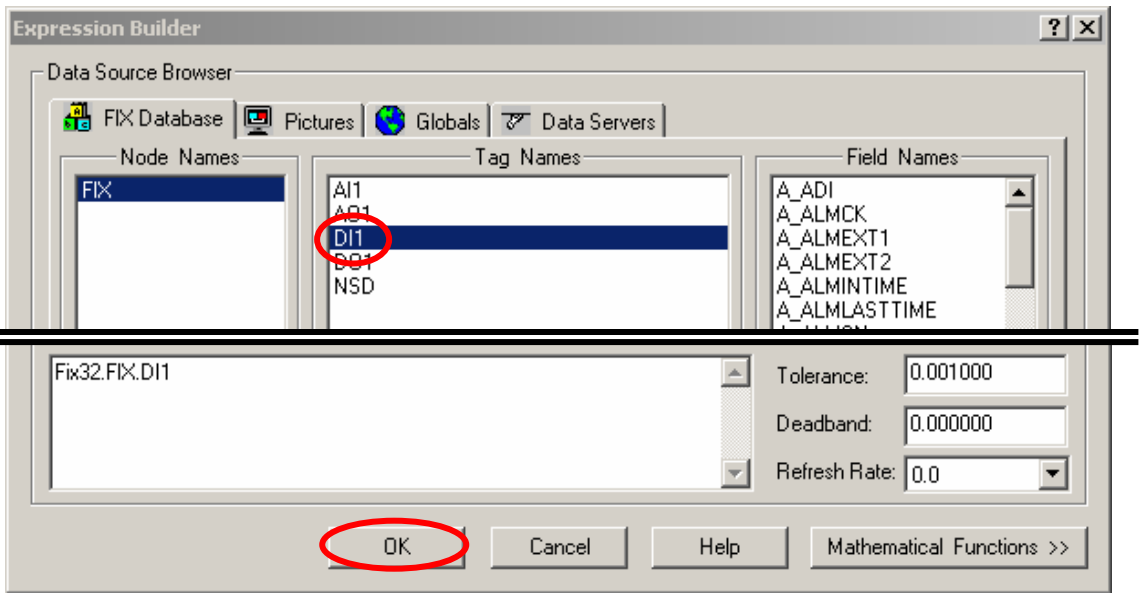
Step34: Click to build a “Datalink” object to show the value of a digital input channel.



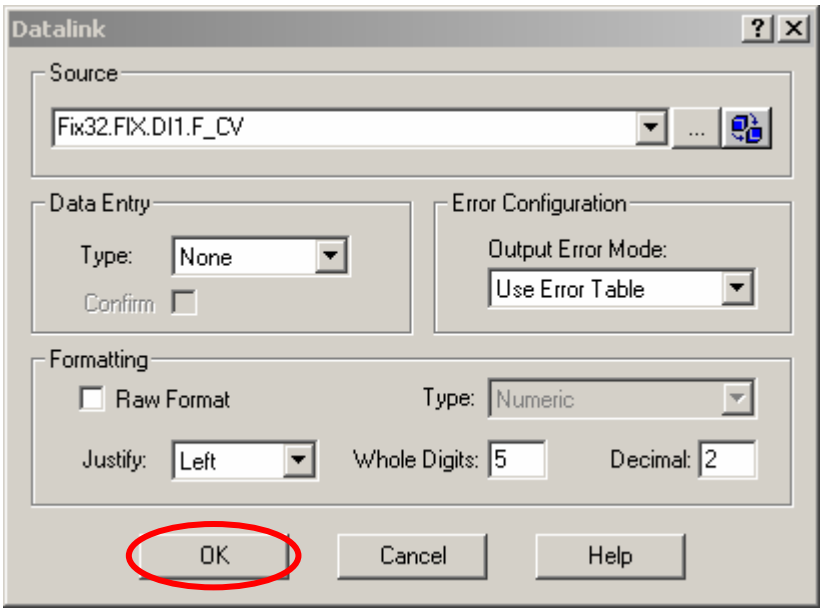
Step35: Click button to select appropriate data source in the database.



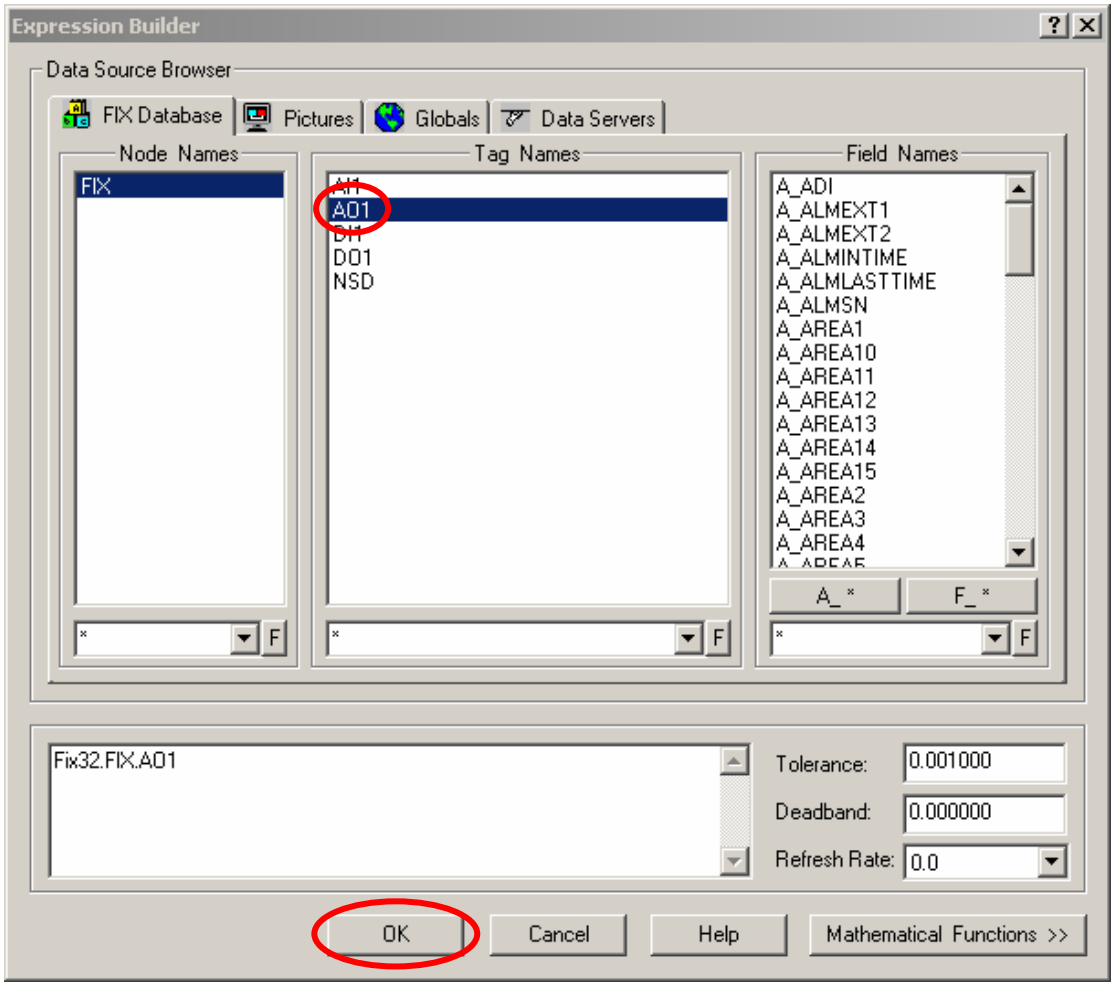
Step36: Select “DI1” in the “Tag Name” list box. Then click “Ok”.



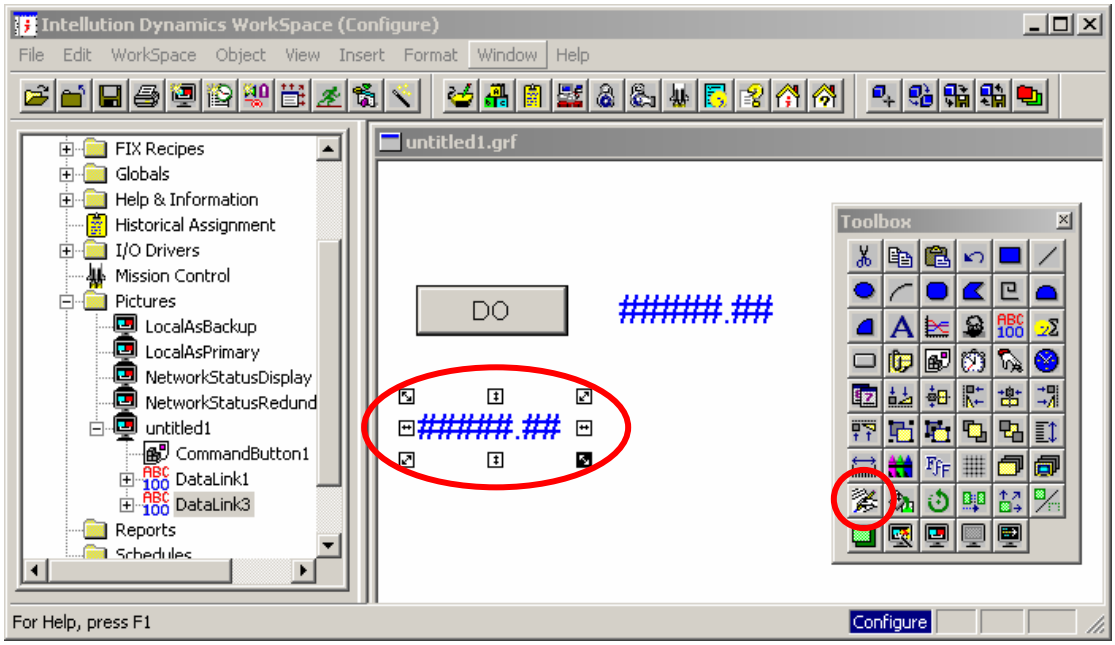
Step36: Click “Ok” to continue the other setting.



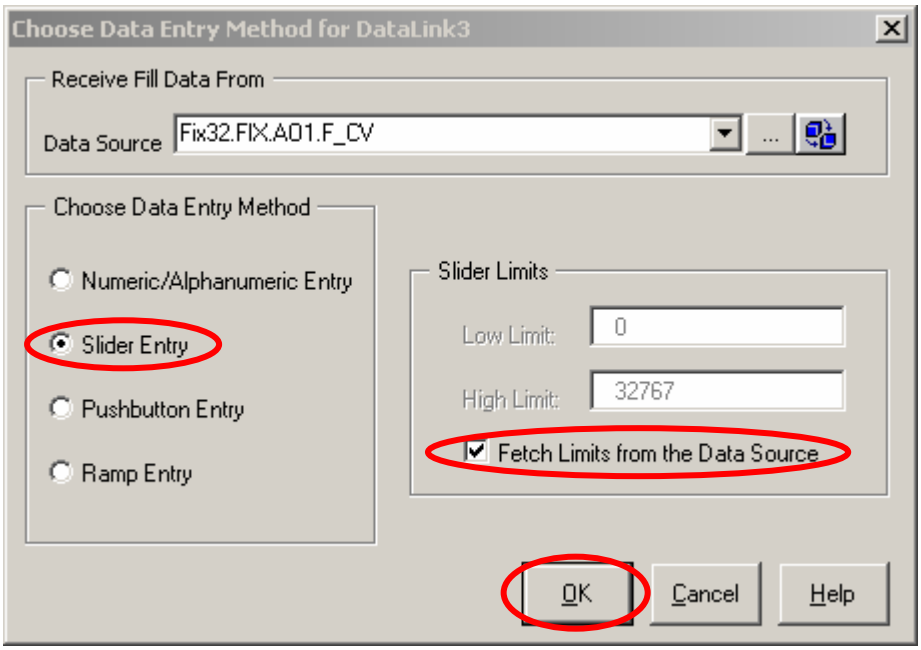
Step37: Build a new “Datalink” object in the picture by the same way ,and put the “Datalink” below the “Push Button” object. Select “AO1” to be the data source.



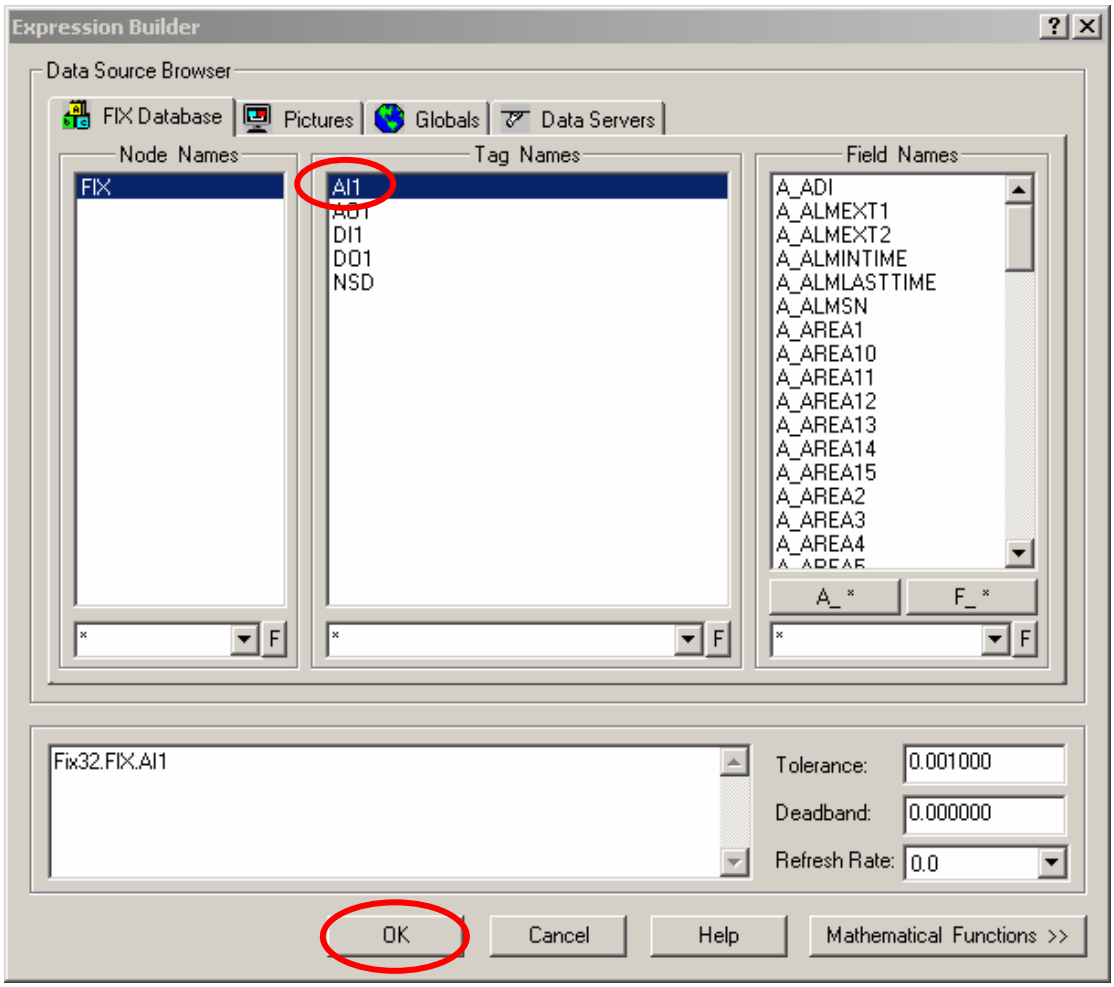
Step38: Select “Datalink” object which is below “Push Button” object. Click button to set the action of this “Datalink” object.



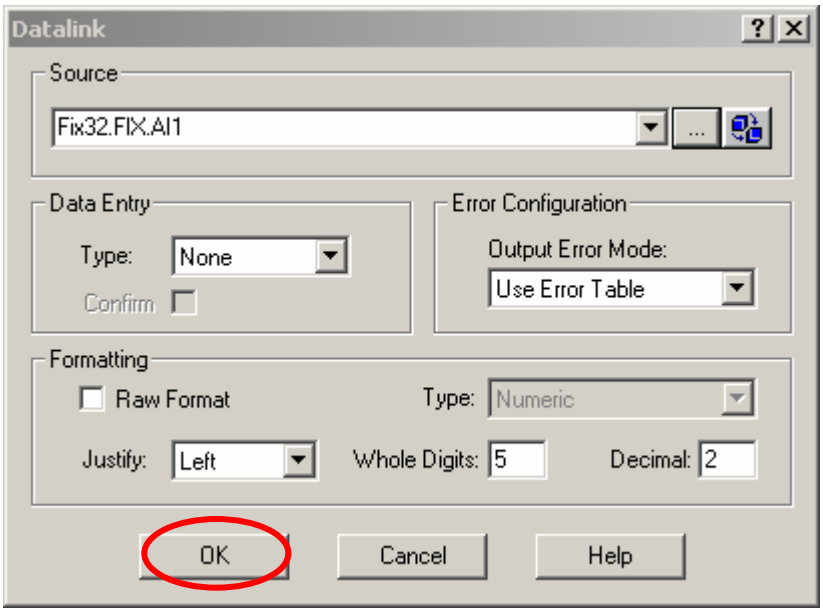
Step39: Select “Slider Entry” to set the output mode of AO object to be a slider. Check the “Fetch Limits from the Data Source” check box.



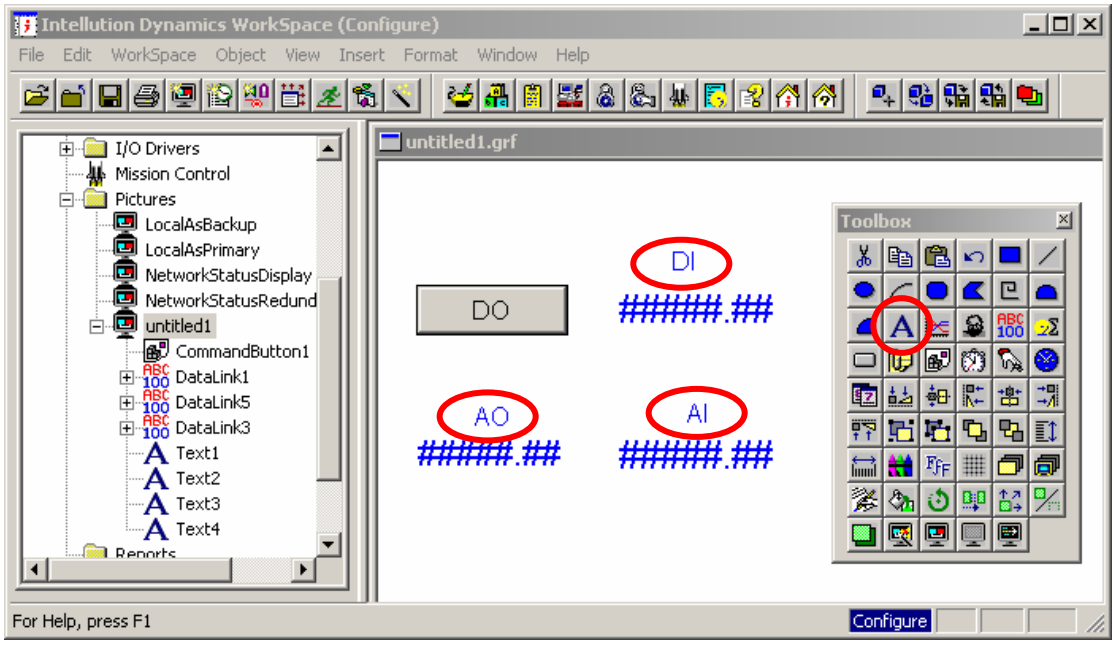
Step40: Follow “Step37” as a model to build a new “Datalink” object in the picture ,and select “AI1” to be the data source.



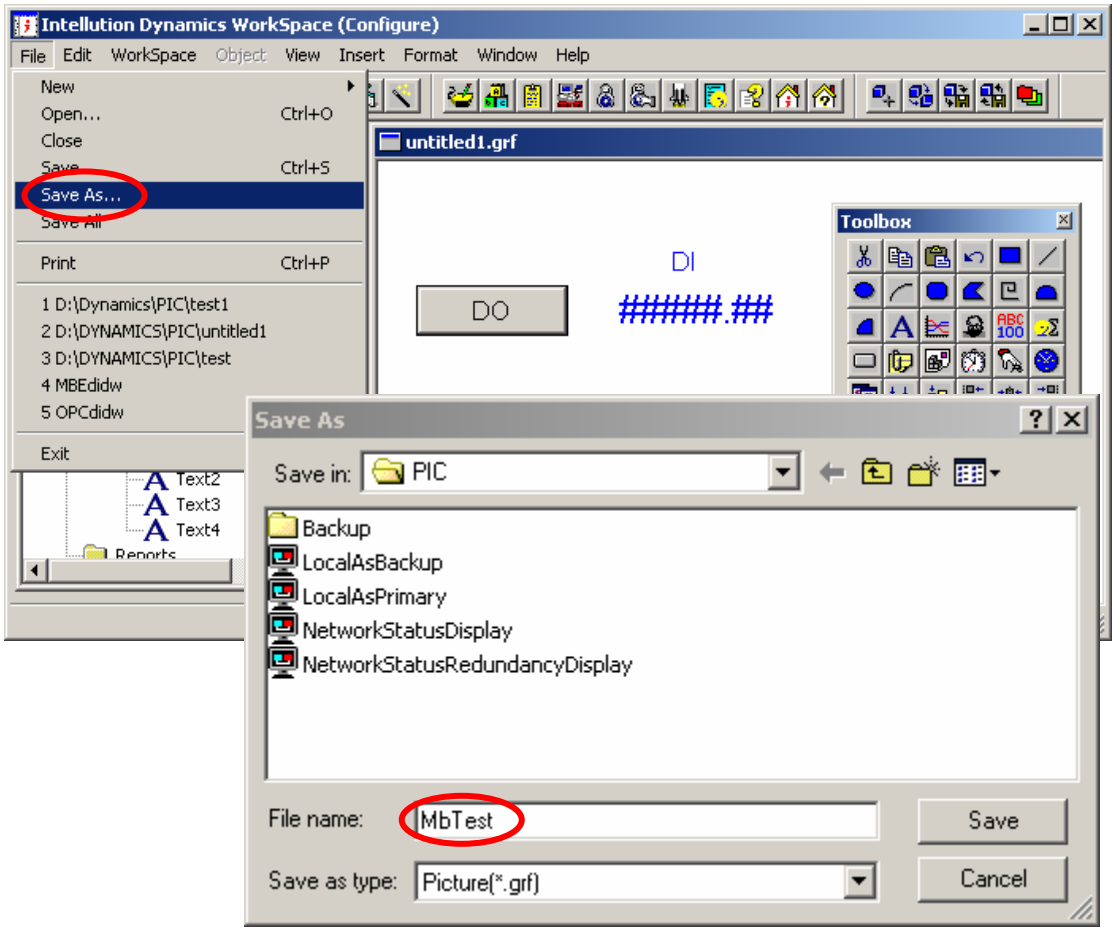
Step41: After finishing all parameter setting, click “Ok” to continue.



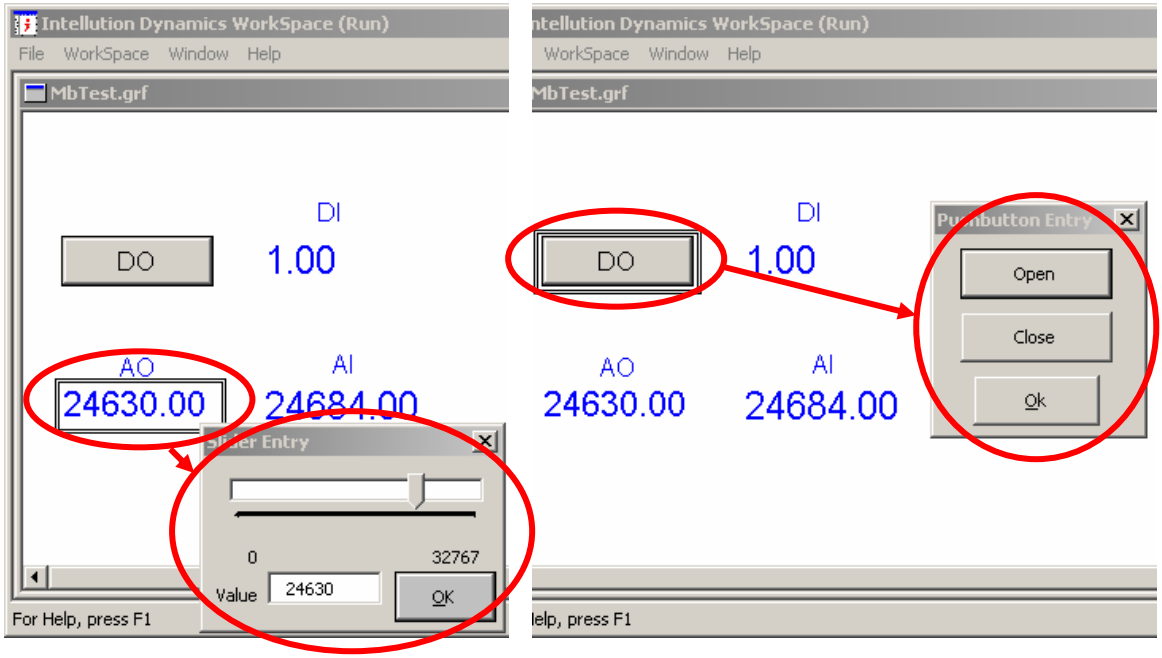
Step42: Click A button to create three “Text” object, and input the words “DI”, “AO”, and “AI” receptively in the appropriate place.



Step43: Click “Save as” to save this page as the name-“MbTest”.



Step44: Run this project and **c**lick output object to set the digital and analog output value that you want as below.


Intellution Dynamics WorkSpace (Run)
File WorkSpace Window Help
MbTest.grf
DI
DO
1.00
AO
AI
24630.00
24684.00
Slider Entry
0
32767
Value
24630
OK
For Help, press F1
WorkSpace Window Help
MbTest.grf
DI
DO
1.00
AO
AI
24630.00
24684.00
Pushbutton Entry
Open
Close
Ok
elp, press F1