NAPOPC_XPE DA Server

User's Manual

[For ICP DAS XPAC]

(Supports 7000, 8000, 87000 series modules and modbus devices)

Table of Contents

1 NAPOPC_XPE DA Server 4
1.1 Installing NAPOPC_XPE DA Server 5
1.2 Uninstalling NAPOPC_XPE DA Server 7
1.3 File 7
1.4 Searching Modules 10
1.5 Monitoring Devices 14
1.6 Adding A New Device 15
1.6.1 Adding A New I-7K/I-8K/I-87K/ZB-2K I/O Module 15
1.6.2 Adding A New RPC Controller 18
1.6.3 Adding A New FRnet I/O 20
1.6.4 Adding A New Modbus TCP Controller 22
1.6.5 Adding A New Modbus RTU Controller 24
1.6.6 Adding A New Modbus ASCII Controller 27
1.7 Adding A New Group 30
1.8 Adding A New Tag 30
1.8.1 Adding A New Tag For I/O Module 31
1.8.2 Adding A New Tag For Controller 32
1.8.3 Scaling Settings 34
1.9 Adding Multi Tags For Modbus Device 35
1.10 Expand/ Shrink Devices 36
1.11 Read/Write Tags 37
1.12 Editing A Device/Group/Tag Properties 38
1.13 Deleting A Device/Group/Tag 39
1.14 Generating Tags 40
1.15 Configuring Initial Status 40
1.16 Help 41
1.17 About 42
2 Quick Start 43
3 Connect To OPC Server 44
3.1 Optimize Your Communication 44
3.2 VB5 Client Demo Program 47
3.3 .Net Client Demo Program 50
3.4 LabVIEW 52
3.5 NATIONAL INSTRUMENTS 55
3.6 WIZCON 60
3.7 iFix 66
3.8 InduSoft 72
3.9 Citect SCADA 79
4 Remote Accessing 99
4.1 System Requirement 100
4.2 Configuring DCOM 101
4.2.1 Configuring On the Server Site (XPAC) 102
4.2.2 Configuring On the Server Site (PC) 111
4.2.3 Configuring On the Server Site (WinPAC) 120
4.2.4 Configuring On the Client Site (PC) 122
4.2.5 Configuring On the Client Site (XPAC) 131
5 OPC To Modbus Slave 141
5.1 OPC To Modbus TCP Slave 141

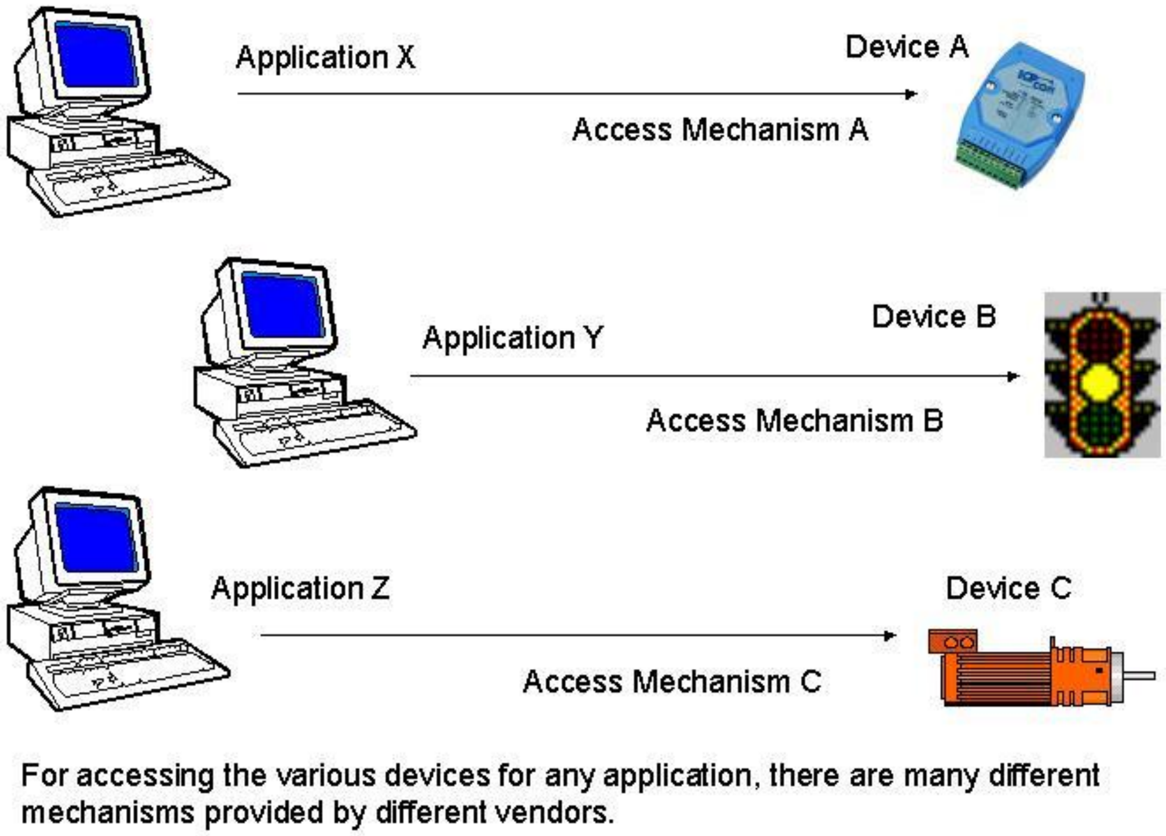
5.1.1 Configuring OPC2MBTCP 141
5.1.2 Startup OPC2MBTCP 143
5.2 OPC To Modbus RTU/ASCII Slave 144
5.2.1 Configuring OPC2MBRTU 144
5.2.2 Startup OPC2MBRTU 146
6 Writing Client Program with VB 148
6.1 Programming with VB5 148
6.1.1 Overview of OPC & VB 148
6.1.2 Tools You Will Need to Build Your VB Client 149
6.1.3 Building Your VB Client – Step By Step 149
6.2 Programming with .Net 152
6.2.1 Limitations about .Net client programming 152
6.2.2 Tools – You Need to Build Your .Net Client 153
6.2.3 Building Your VB.Net Client – Step By Step 153
6.2.4 Building Your VC#.Net Client – Step By Step 156
7 NAPOPC_XPE Changes List 160
7.1 Features of NAPOPC_XPE version 1.01 160
7.1.1 Customized Module/Device Polling Time 160
7.1.2 Multi-Thread Communication 160
7.1.3 Support Remote Procedure Call with Quicker/UPC 160
7.1.4 Switch of Single-Thread and Multi-Thread 161
7.2 Features of NAPOPC_XPE version 1.02 161
7.2.1 Bug Fix 161
7.3 Features of NAPOPC_XPE version 1.10 161
7.3.1 Support ET-7000 Search 161
7.3.2 Support ZB-2K I/O 161
7.3.3 Support FRnet Module 161
7.3.4 Support Account Selection 161
7.3.5 UI Modification 162
7.4 Features of NAPOPC_XPE version 1.12 162
7.4.1 Support FRnet module via RPC communication 162
8 Reference 163

1 NAPOPC_XPE DA Server

The NAPOPC_XPE DA Server is one set of ICP DAS OPC solution. It works on XPAC(**Based on Windows XP Embedded Operation System**) and uses an Explorer-style user interface to display a hierarchical tree of modules and groups with their associated tags. A group can be defined as a subdirectory containing one or more tags. A module may have many subgroups of tags and all tags belong to their module. (**The "OPC" stands for "OLE for Process Control" and the "DA" stands for "Data Access"**)

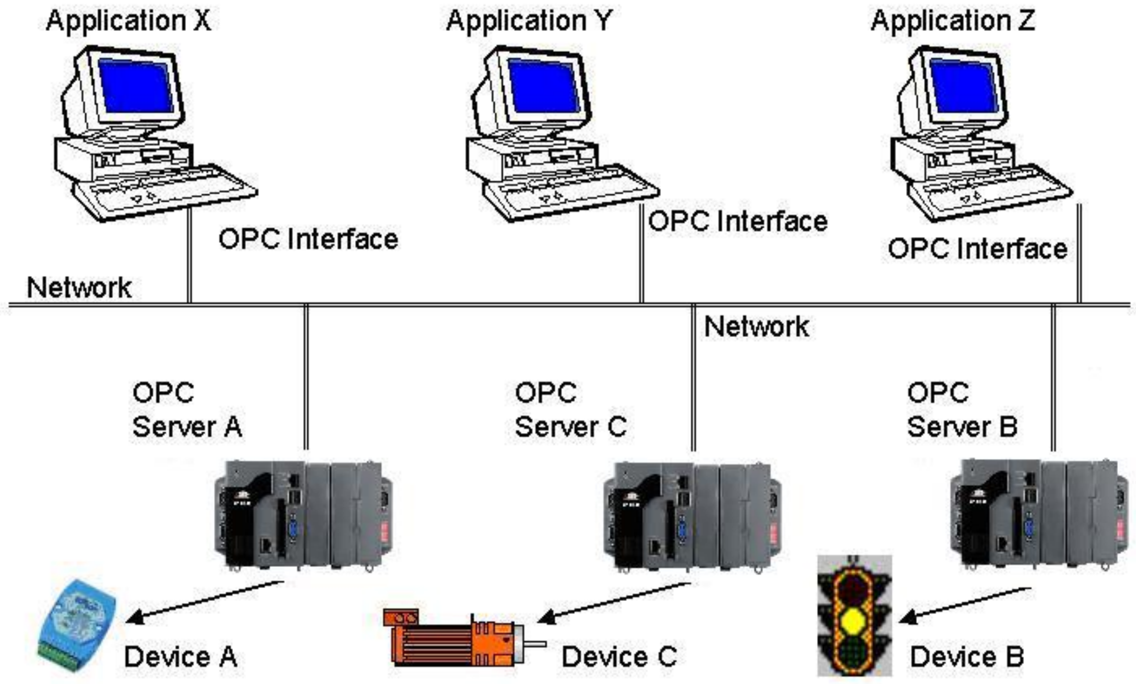
Furthermore, NAPOPC_XPE DA Server package provides not only traditional **"OPC"** interface but also two **"Modbus Slave"** interfaces to communicate with modbus device. The two interfaces play the role of bridge between OPC client and modbus slave to transform the OPC tags to modbus address. Users can accept its result of auto-mapping or adjust the modbus address by themselves.

The following two figures show the difference between traditional system architecture and the OPC architecture.



For accessing the various devices for any application, there are many different mechanisms provided by different vendors.

Figure 1-0-1 Traditional architecture used to access a device.



Different vendors provide both different devices and the appropriate OPC Server. To access each device for any application, there is only one common mechanism through the "OPC Interface".

Figure 1-0-2 Using the OPC architecture to access a device.

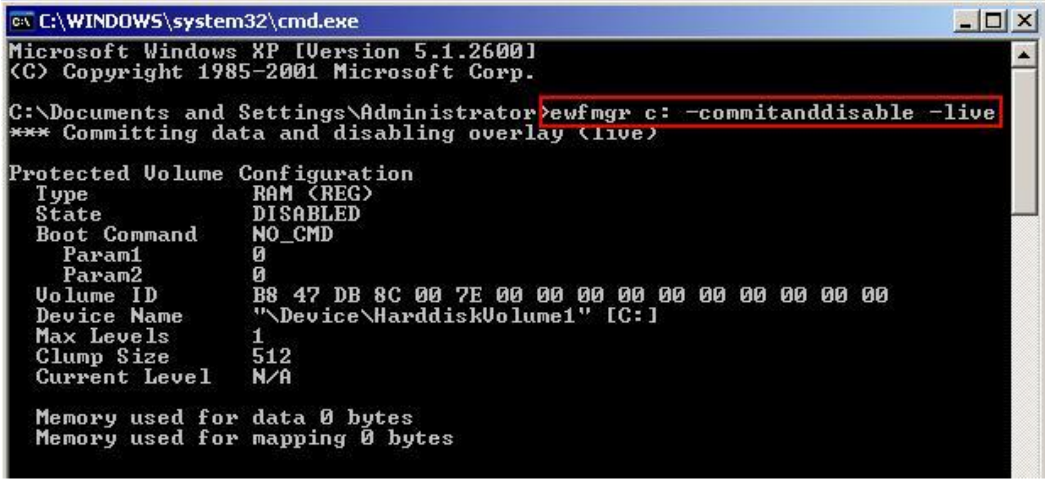
The main program of NAPOPC_XPE DA Server is "NAPOPCSvr_XPE.exe". It automatically calls the "DCON_PC.DLL", "IOCtrl.DLL", "XPacSDK.DLL", "pac_i8017hw.DLL", "pac_i8024w.DLL", "pac_i8172W.DLL" and "UART.DLL" functions on demand.

1.1 Installing NAPOPC_XPE DA Server

You can get the software from ftp://ftp.icpdas.com/pub/cd/xp-8000/tools/napopcsvr/ or download it from http://opc.icpdas.com/download.htm.

Step 1:

Enter the command **"ewfmgr c: -commitanddisable –live"** at console.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>ewfmgr c: -commitanddisable -live
*** Committing data and disabling overlay (live)

Protected Volume Configuration
  Type             RAM (REG)
  State            DISABLED
  Boot Command     NO_CMD
    Param1         0
    Param2         0
  Volume ID        B8 47 DB 8C 00 7E 00 00 00 00 00 00 00 00 00 00
  Device Name      "\Device\HarddiskVolume1" [C:]
  Max Levels       1
  Clump Size       512
  Current Level    N/A

  Memory used for data 0 bytes
  Memory used for mapping 0 bytes
```

Step 2:

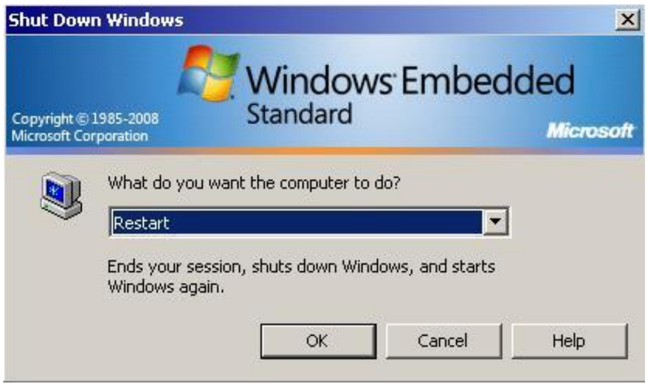
Copy NAPOPC_XPE DA Server.exe to a specific folder in XPAC.

Step 3:

Double click NAPOPC_XPE DA Server.exe and follow the installing wizard to finish the installation.

Step 4:

Restart XPAC.



Shut Down Windows

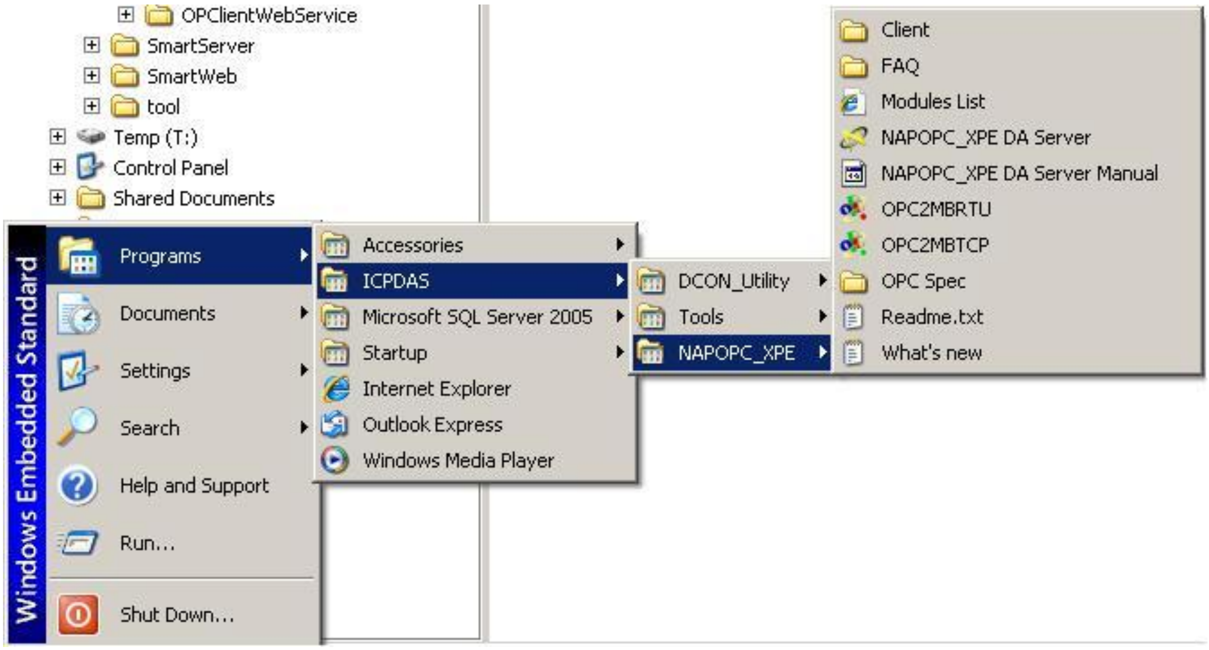
What do you want the computer to do?

Restart

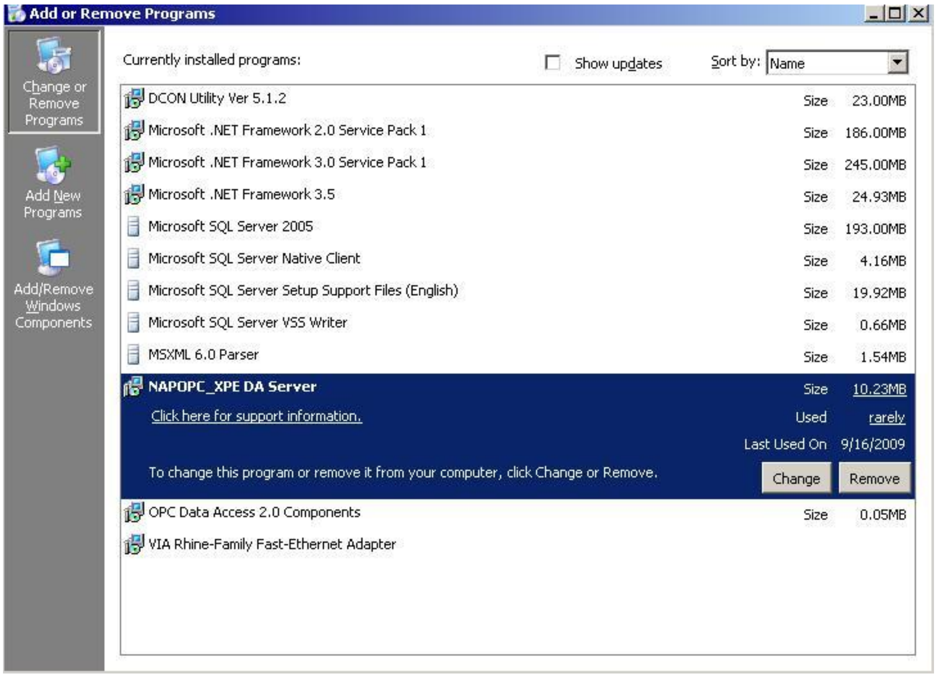
Ends your session, shuts down Windows, and starts Windows again.

Windows Embedded Standard
Copyright © 1985-2008 Microsoft Corporation
Microsoft
OK
Cancel
Help

After you complete the above steps, you can start the NAPOPC_XPE DA Server by clicking the "NAPOPC_XPE DA Server" as below.



1.2 Uninstalling NAPOPC_XPE DA Server



Currently installed programs:

Program		
DCON Utility Ver 5.1.2	Size	23.00MB
Microsoft .NET Framework 2.0 Service Pack 1	Size	186.00MB
Microsoft .NET Framework 3.0 Service Pack 1	Size	245.00MB
Microsoft .NET Framework 3.5	Size	24.93MB
Microsoft SQL Server 2005	Size	193.00MB
Microsoft SQL Server Native Client	Size	4.16MB
Microsoft SQL Server Setup Support Files (English)	Size	19.92MB
Microsoft SQL Server VSS Writer	Size	0.66MB
MSXML 6.0 Parser	Size	1.54MB
NAPOPC_XPE DA Server	Size	10.23MB
Click here for support information.	Used	rarely
	Last Used On	9/16/2009
To change this program or remove it from your computer, click Change or Remove.		
OPC Data Access 2.0 Components	Size	0.05MB
VIA Rhine-Family Fast-Ethernet Adapter		

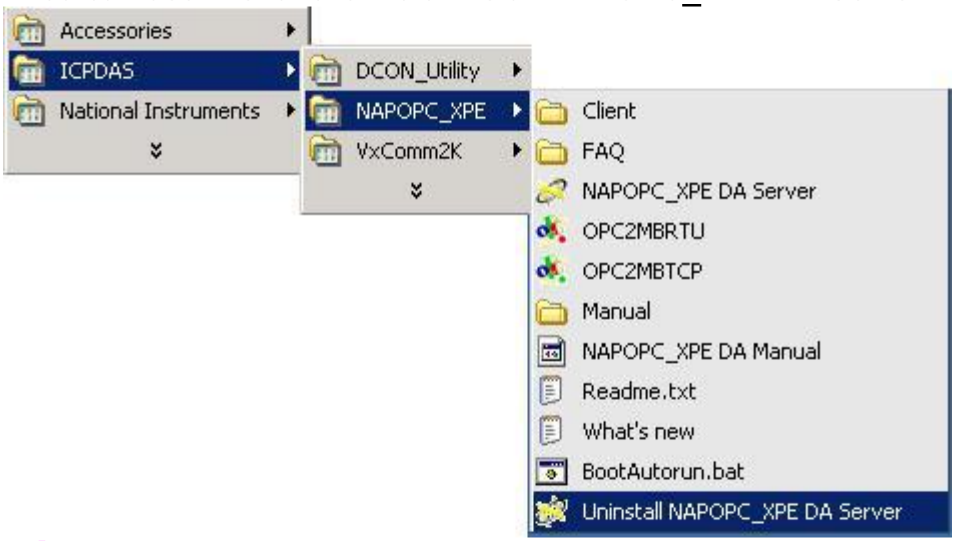
Step 1:

Go to "Add or Remove Programs" dialog and select NAPOPC_XPE DA Server

Step 2:

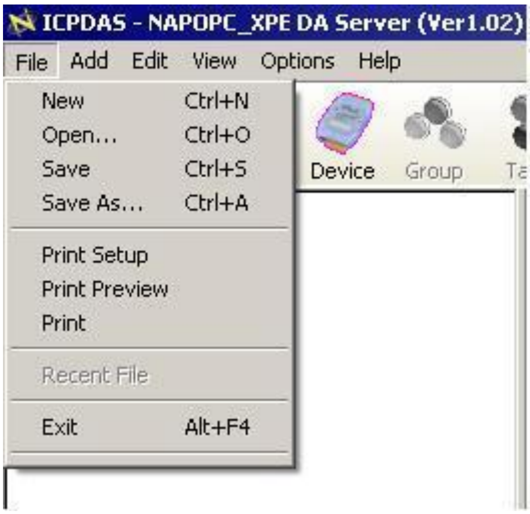
Press "Remove" button to remove NAPOPC_XPE DA Server

NAPOPC_XPE DA Server v1.12 or later version provides "Uninstall" function as below. You can use this function to uninstall NAPOPC_XPE DA Server.



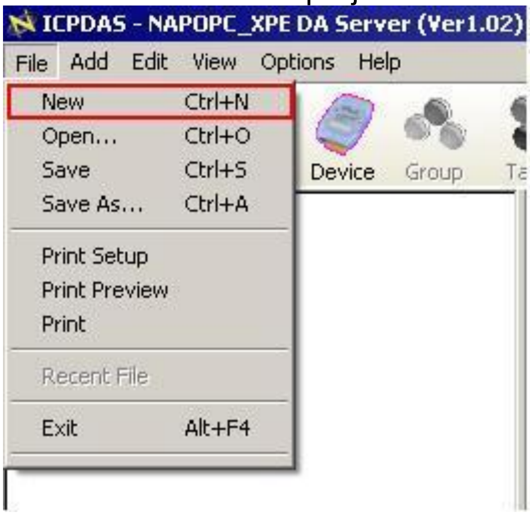
1.3 File

All configuration settings can be saved into configuration file by clicking the "File/ Save" and "File/ Save As …" menu item. The OPC server will automatically load the last configuration file with every launch.



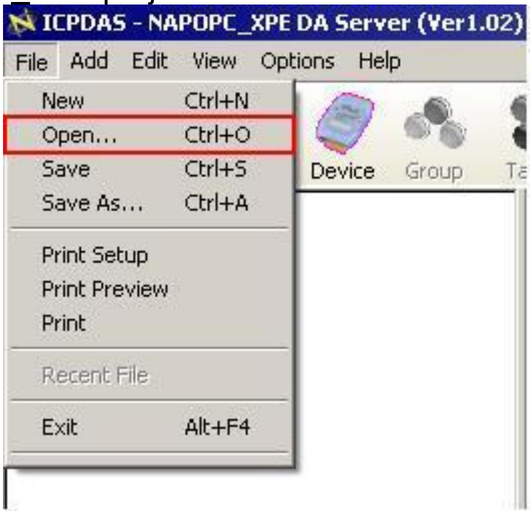
New:

Clean current project and create a new project



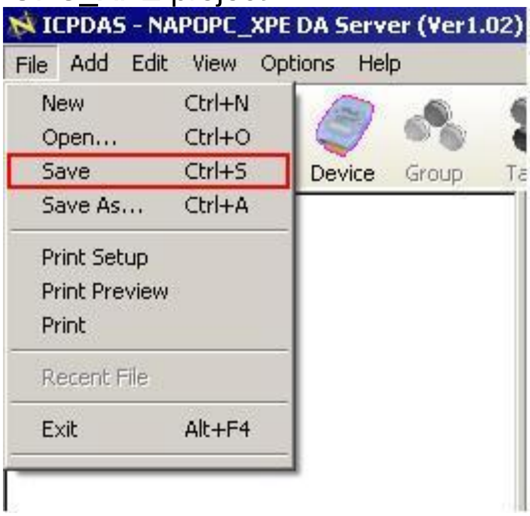
Open:

Load old NAPOPC_XPE project



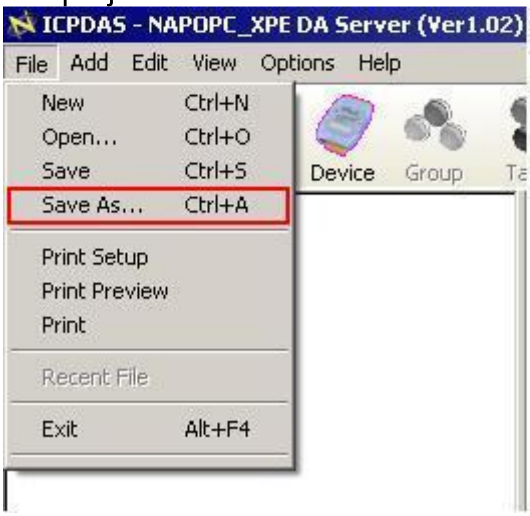
Save:

Save current NAPOPC_XPE project



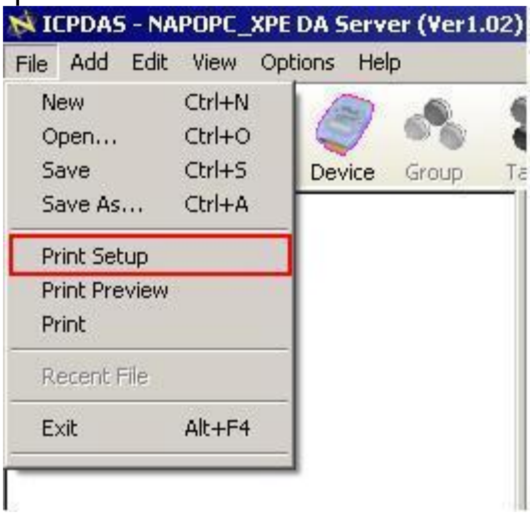
Save as…:

Save NAPOPC_XPE project as a new one



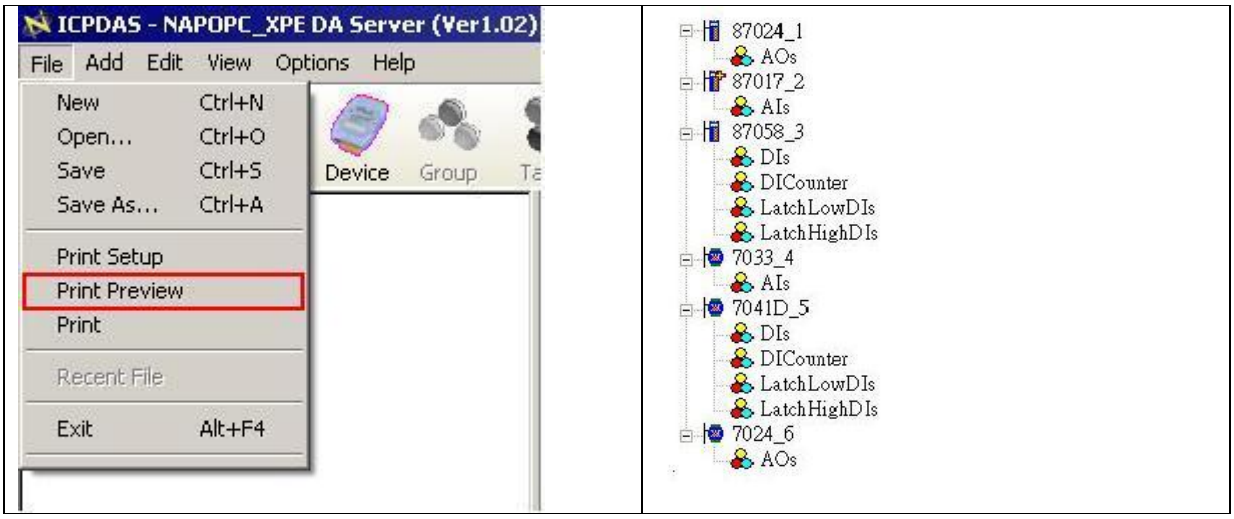
Print Setup:

Choose and setup printer



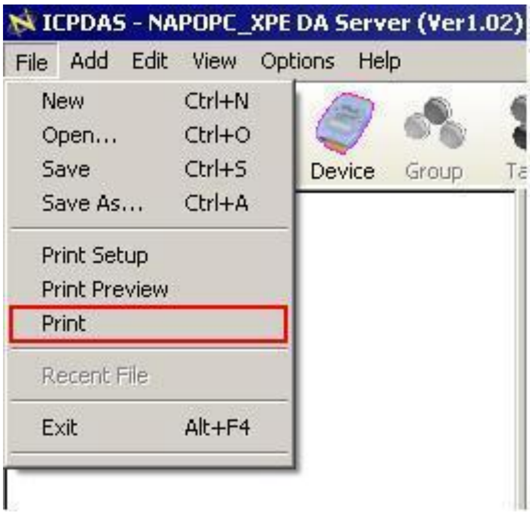
Print Preview:

Preview current modules list



Print

Print current modules list

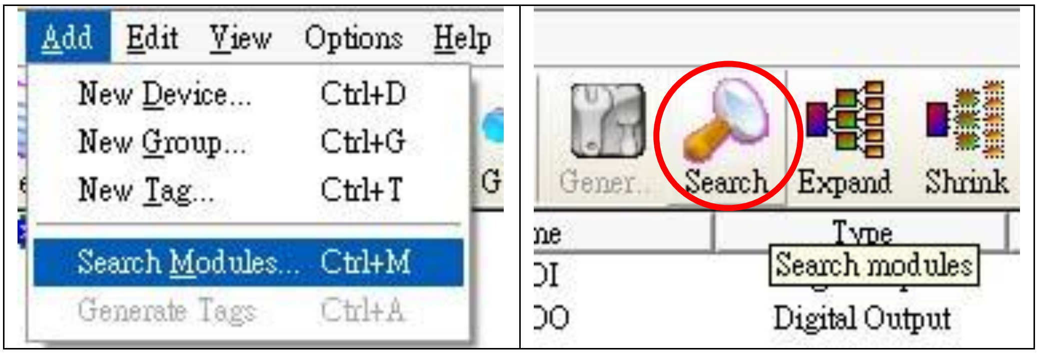


1.4 Searching Modules

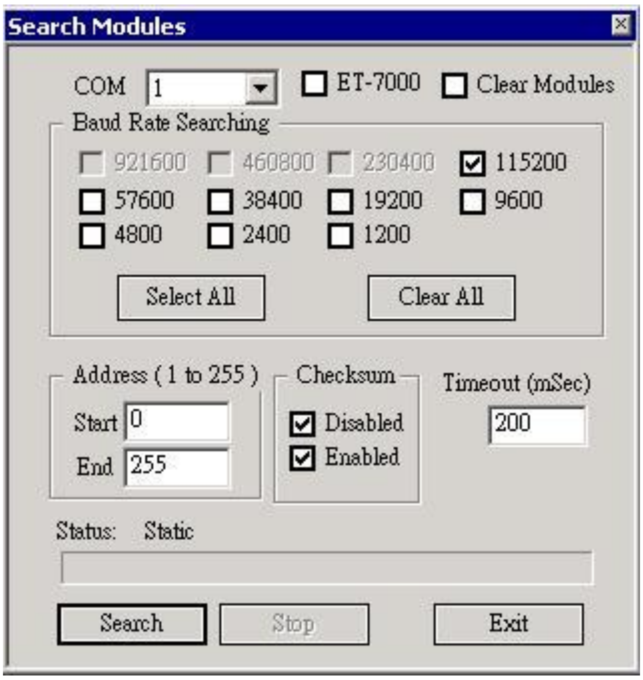
The "Search Modules…" function lets you configure the OPC server automatically. It searches the RS-232 and RS-485 network to find modules and then generates tags automatically. This function generates AI/AO, DI/DO, Latched DI and Counter tags.

NOTE:
For complete module support, please update "\ICPDAS\NAPOPC_XPE \module.ini","\ICPDAS\NAPOPC_ XPE \module_ET.ini", and "\ICPDAS\ NAPOPC_ XPE \module_FRnet.ini" from http://opc.icpdas.com/download.htm frequently.

Step 1: Click on the "Add/ Search Modules…" menu item or the 🔍 icon to search for modules.



Step 2: The "Search Modules" dialog box pops up.



COM 1

□ ET-7000 □ Clear Modules

□ 921600 □ 460800 □ 230400 ☑ 115200
□ 57600 □ 38400 □ 19200 □ 9600
□ 4800 □ 2400 □ 1200

Address (1 to 255)
Start 0
End 255

Checksum
☑ Disabled
☑ Enabled

Timeout (mSec) 200

Status: Static

COM Port:

Specifies which "COM Port" number is search. The default value is 1 and the valid range is from 1 to 255. Please verify the "COM Port" number that the RS-232 or RS-485 network is connected.

ET-7000:

If this field is checked, NAPOPC can search not only the modules communicating via COM port but also ET-7000 modules via Ethernet automatically.

Clear Modules:

Modules can be added many times. If this field is checked, it removes all modules from the list window before searching. Checking this box prevents adding a duplicate module. The default setting is "not checked".

Baud Rate Searching:

Specifies which "Baud Rate" will be look for. The default setting is “9600".

Naturally, if multiple baud rates are checked, the search will be longer. The computer system must close and then reopen the COM ports to communicate with modules when searching for multiple baud rates. This also reduces communication performance. Thus, using the same baud rate and COM port number for every module is highly recommended.

Select All:

Sets all of the "Baud Rate" be checked. Please refer to the above "Baud Rate Searching" section.

Clear All:

Sets all of the "Baud Rate" be unchecked (nothing to search). Please refer to the above "Baud Rate Searching" section.

Address/Start:

Specifies the starting address. The default value is 1 and the valid range is from 1 to 255. It won't search for an address below these settings.

Address/End:

Specifies the ending address. The default value is 255 and the valid range is from 1 to 255. It won't search for an address greater than these settings.

Checksum/Disabled:

If this field is checked, modules are searched with no checksum. If both the "Disabled" and "Enabled" fields were unchecked, the search would be undefined.

Checksum/Enabled:

If this field is checked, it searches modules with checksum. If both the "Disabled" and "Enabled" fields were unchecked, again, the search would be undefined.

Timeout:

Specifies the communicate timeout value for each module. The default value is 500 (equal to 0.5 Seconds), measured in millisecond(s) [0.001 Second(s)]. After a module has been found, this timeout value will also be recorded for further use.

Users can reduce this value to shorten the search time. Be careful. A shorter search time may cause communication failure.

Status:

It shows the searching status (includes: progress in %, Address in "A:??”, Baud-Rate in "B:????", Checksum in "S:?" and Error-Code in "EC:??"). The timeout error code is 15. In most cases, it indicates no module has responded to the current command.

Search:

After setting the above options, click this button to search. The window will be closed automatically when completed.

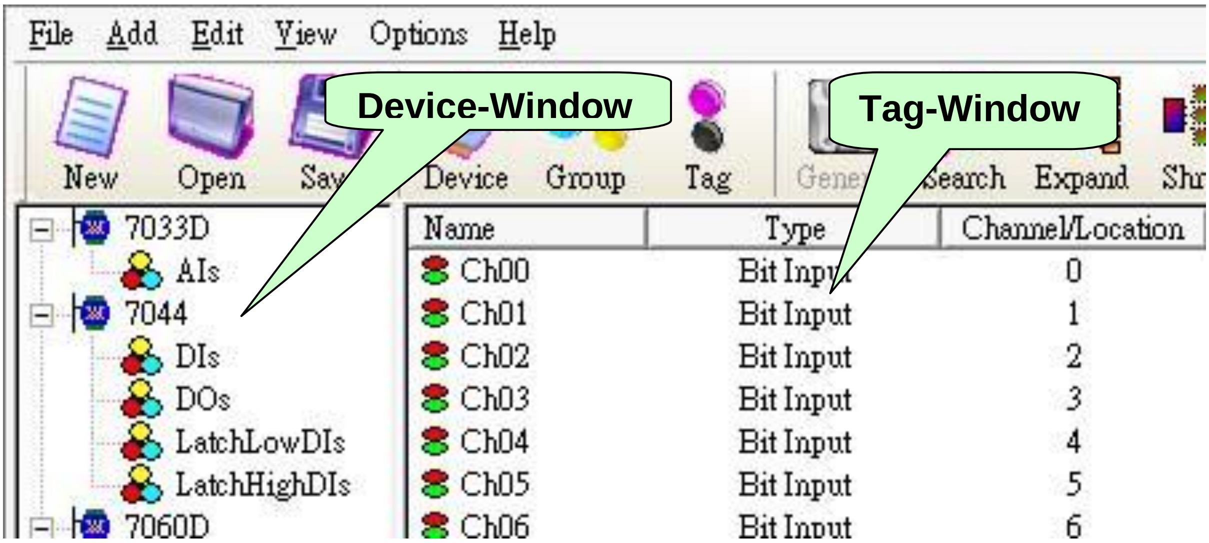
Stop:

During the search, users can click the button to stop. The window will stay on the screen after the search is cancelled.

Exit:

Users can click the button to close the window.

Step 3: After the search, the discovered modules will be listed on the Device-Window (left side). Users can also see the tags on the Tag-Window (right side) generated by the "Search Modules…" function automatically.

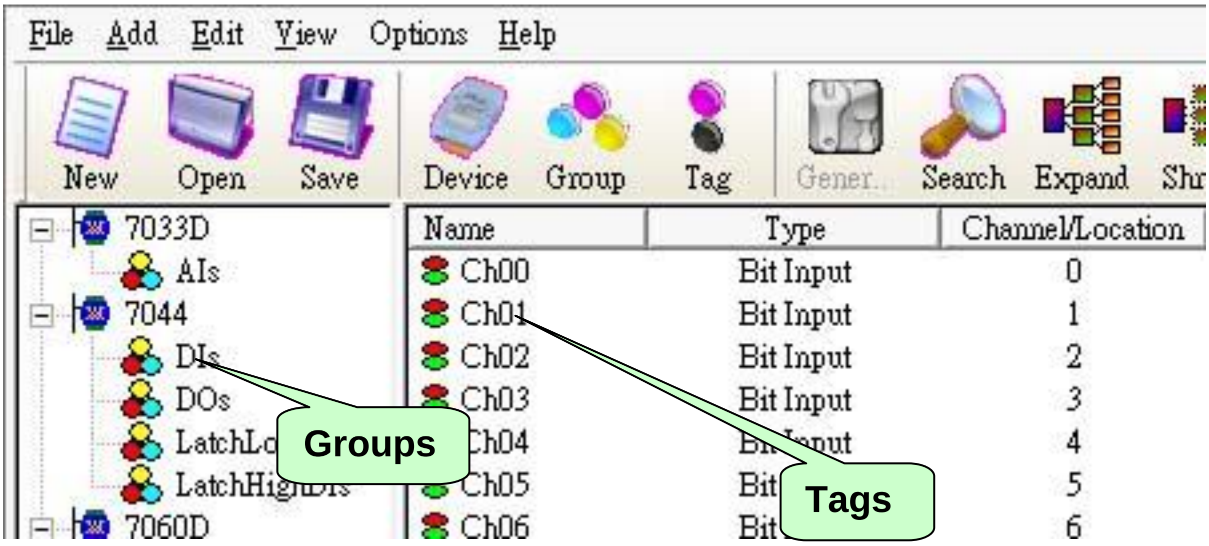


- 7033D
  - AIs
- 7044
  - DIs
  - DOs
  - LatchLowDIs
  - LatchHighDIs
- 7060D

Name	Type	Channel/Location
Ch00	Bit Input	0
Ch01	Bit Input	1
Ch02	Bit Input	2
Ch03	Bit Input	3
Ch04	Bit Input	4
Ch05	Bit Input	5
Ch06	Bit Input	6

The "Search Modules…" function generates "Digital Input", "Digital Output" "Bit Input" or "Bit Output" tags.

The "Digital Input" and "Digital Output" tags use one communication to read the status of all channels, while the "Bit Input" and "Bit Output" tags use one communication to read only one-channel status. The "Digital Input" and "Digital Output" tags have better performance than the "Bit Input" and "Bit Output" tags. Using the "Digital Input" and "Digital Output" tags to access modules is highly recommended.



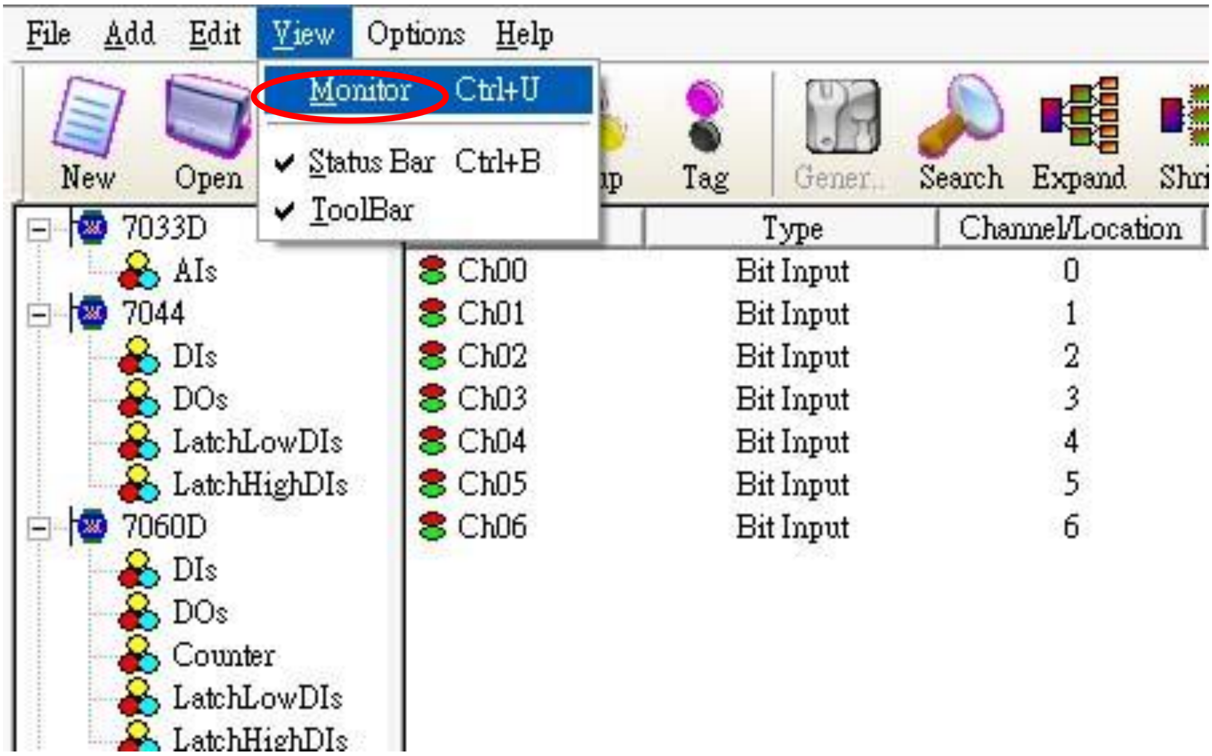
- 7033D
  - AIs
- 7044
  - DIs
  - DOs
  - LatchLowDIs
  - LatchHighDIs
- 7060D

Name	Type	Channel/Location
Ch00	Bit Input	0
Ch01	Bit Input	1
Ch02	Bit Input	2
Ch03	Bit Input	3
Ch04	Bit Input	4
Ch05	Bit Input	5
Ch06	Bit Input	6

1.5 Monitoring Devices

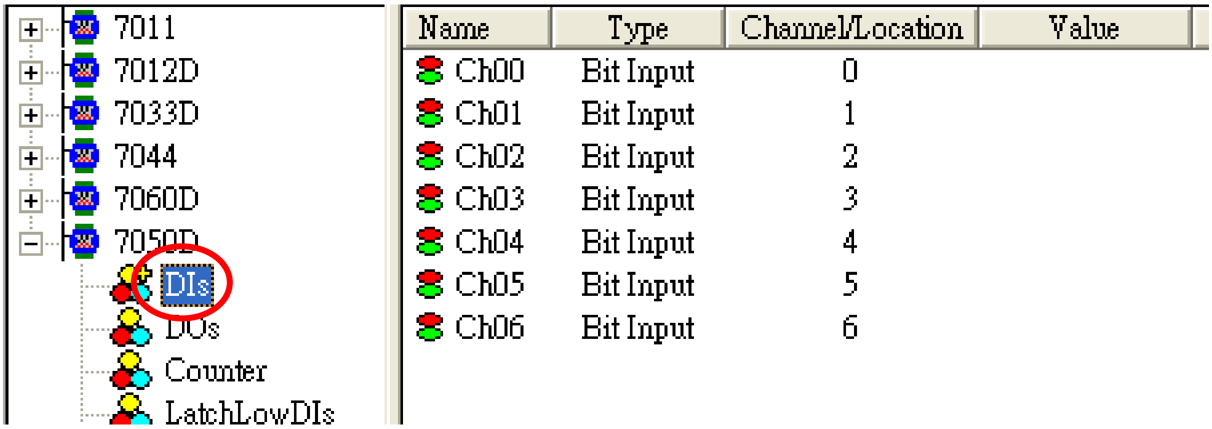
Use the "Monitor" function to see values of tags by checking the "View/ Monitor" menu item. Uncheck the item to stop monitoring.

Step 1: Click the "View/ Monitor" menu item or the icon to enable monitor.



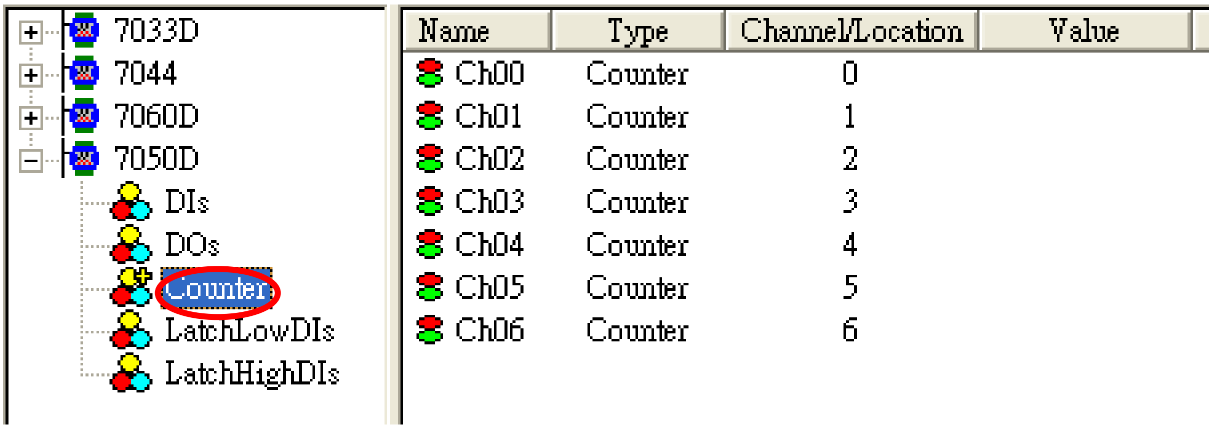
	Type	Channel/Location
Ch00	Bit Input	0
Ch01	Bit Input	1
Ch02	Bit Input	2
Ch03	Bit Input	3
Ch04	Bit Input	4
Ch05	Bit Input	5
Ch06	Bit Input	6

Step 2: Select the "DIs" group in the Device-Window (left side) to monitor its own Bit -Input tags.



Name	Type	Channel/Location	Value
Ch00	Bit Input	0	
Ch01	Bit Input	1	
Ch02	Bit Input	2	
Ch03	Bit Input	3	
Ch04	Bit Input	4	
Ch05	Bit Input	5	
Ch06	Bit Input	6	

Step 3: Select the "7050D" module on the Device-Window to monitor its own Digital-Input and Digital-Output tags.

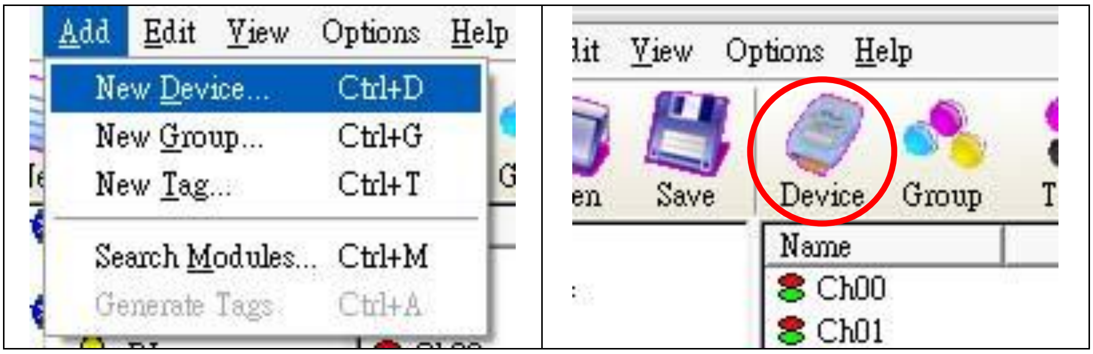


Name	Type	Channel/Location	Value
Ch00	Counter	0	
Ch01	Counter	1	
Ch02	Counter	2	
Ch03	Counter	3	
Ch04	Counter	4	
Ch05	Counter	5	
Ch06	Counter	6	

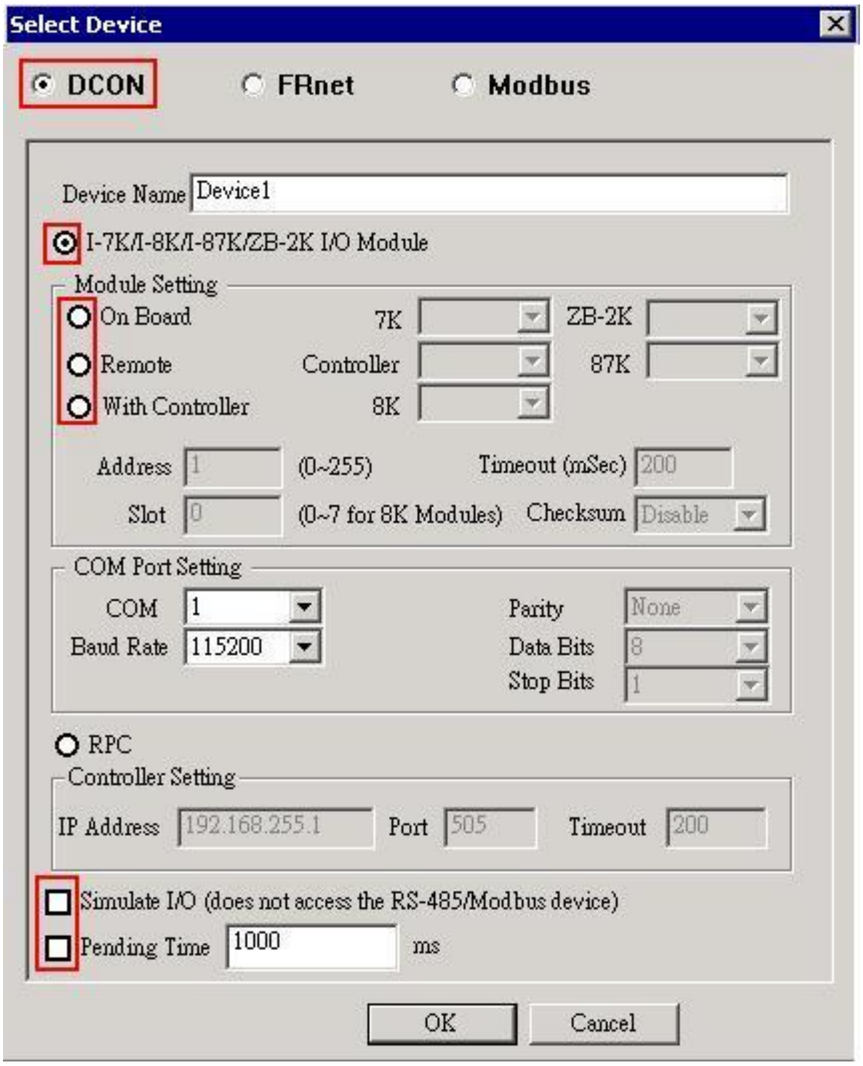
1.6 Adding A New Device

1.6.1 Adding A New I-7K/I-8K/I-87K/ZB-2K I/O Module

Step 1: Click on the "Add/ New Device…" menu item or the icon to add a new module.



Step 2: The "Select Device" dialog box pops up.
Step 3: Click on the "DCON" radio button.
Step 4: Click on the "I-7K/I-8K/I-87K/ZB-2K I/O Modules" radio button.
Step 5: Click on the "On Board", "Remote" or "With Controller" radio button.



Select Device

● DCON ○ FRnet ○ Modbus

Device Name Device1

● I-7K/I-8K/I-87K/ZB-2K I/O Module

Module Setting

○ On Board 7K ZB-2K

○ Remote Controller 87K

○ With Controller 8K

Address 1 (0~255) Timeout (mSec) 200

Slot 0 (0~7 for 8K Modules) Checksum Disable

COM Port Setting

COM 1 Parity None

Baud Rate 115200 Data Bits 8

Stop Bits 1

○ RPC

Controller Setting

IP Address 192.168.255.1 Port 505 Timeout 200

□ Simulate I/O (does not access the RS-485/Modbus device)

□ Pending Time 1000 ms

Device Name:

Names with spaces or punctuation such as "|!.," cannot be used within a module name. The clients use the "Device Name" and "Tags" to access its value. The "Device Name" can not be the same as any other module.

7K/ 87K/ 8K/ZB-2K Controller Module ID:

User can click on the Combo Box to select a Module ID.

Address:

Specifies a Module Address for this module. The default value is 1 and the valid range is between 1 to 255.

This field is disabled for the 8000 sub-devices. It will use the 8000 main-device's address.

Timeout:

Specifies timeout (Response time) value for this module. The default value is 200 ms. A smaller timeout value may cause communication failure and a greater timeout value may reduce the performance of the client program.

This field is disabled for the 8000 sub-devices and it will use the 8000 main-device's timeout value.

Slot:

The 8000 main-device ("With Controller") has 4 or 8 slots for the 8000 sub-device to plug in. This "slot" field indicates the slot number that the 8000 sub-device is using. The valid range is from 0 to 7.

The XPAC ("On Board") has 3 or 7 slots for the 8000 sub-device to plug in. This "slot" field indicates the slot number that the 8000 sub-device is using. The valid range is from 1 to 7.

This field is disabled for 8000 main-device and 7000 series modules.

Checksum:

This checksum field must match the hardware setting. A mismatch will always cause a communication failure with this module.

This field is disabled for the 8000 sub-devices and it will use the 8000 main-device's checksum.

COM Port:

Specifies the COM port to be used. Please verfiy which COM port number that the RS-232 / RS-485 network is using. Wrong settings will always cause communication failure.

This field is disabled for the 8000 sub-devices. It will use the 8000 main controller unit COM port setting.

Baud Rate:

Specifies the baud rate to be used. Verify the module's current baud rate. A wrong setting will always cause communication failure for this module.

This field is disabled for the 8000 sub-devices. It will use the 8000 main-controller unit baud rate.

Simulate I/O:

The "Simulate I/O" checkbox switches to a simulator of reading I/O. Since the simulator does not open the TCP/IP port, it is an easy way to work with the server, to configure tags or to connect clients without requiring any hardware.

Pending Time:

Minimum interval time between two access. To activate this function, NAPOPC_XPE can work under optimized communication performance. If this module only needs to be accessed 1 time per 5 seconds. You can set pending time as 5000 ms. NAPOPC_XPE will automatically spread time resource to other modules which are connected with each other.

OK:

Click on the "OK" button to add the new module setting.

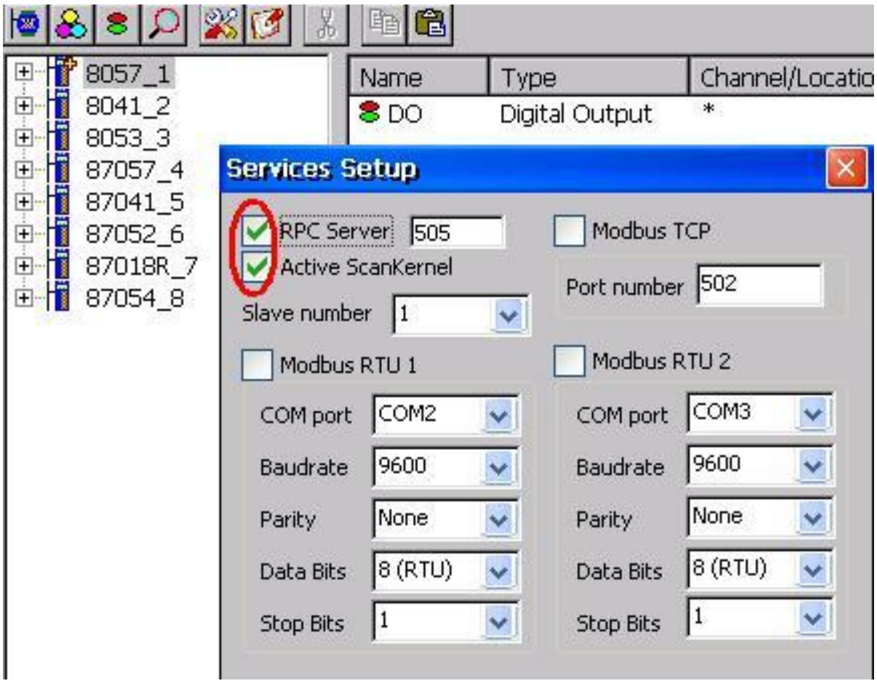
Cancel:

Click on the "Cancel" button to avoid any changes.

Step 6: Click on the "OK" button to add this new module.

1.6.2 Adding A New RPC Controller

Before adding a new "RPC" controller, please check "RPC Server" and "Active ScanKernel" at "Services Setup" dialog of "NAPOPC_CE5/CE6" in WinCon/ WinPAC/ ViewPAC.



8057_1
8041_2
8053_3
87057_4
87041_5
87052_6
87018R_7
87054_8

Name	Type	Channel/Locatio
DO	Digital Output	*

Services Setup

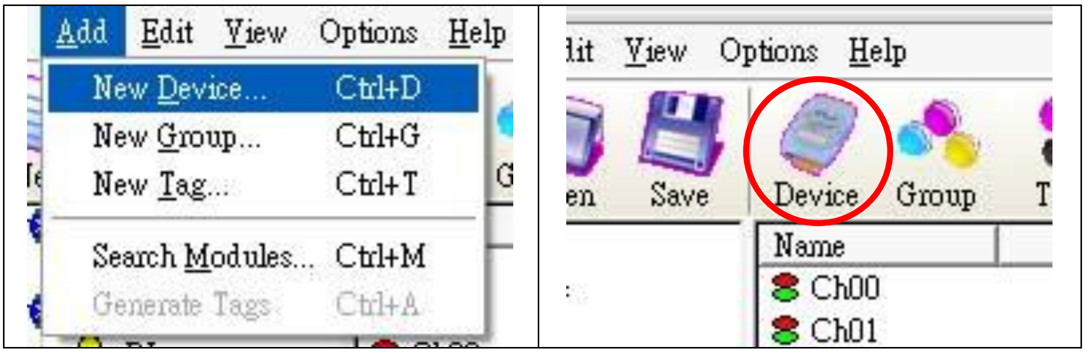
☑ RPC Server 505
☑ Active ScanKernel
Slave number 1

□ Modbus TCP
Port number 502

□ Modbus RTU 1
COM port COM2
Baudrate 9600
Parity None
Data Bits 8 (RTU)
Stop Bits 1

□ Modbus RTU 2
COM port COM3
Baudrate 9600
Parity None
Data Bits 8 (RTU)
Stop Bits 1

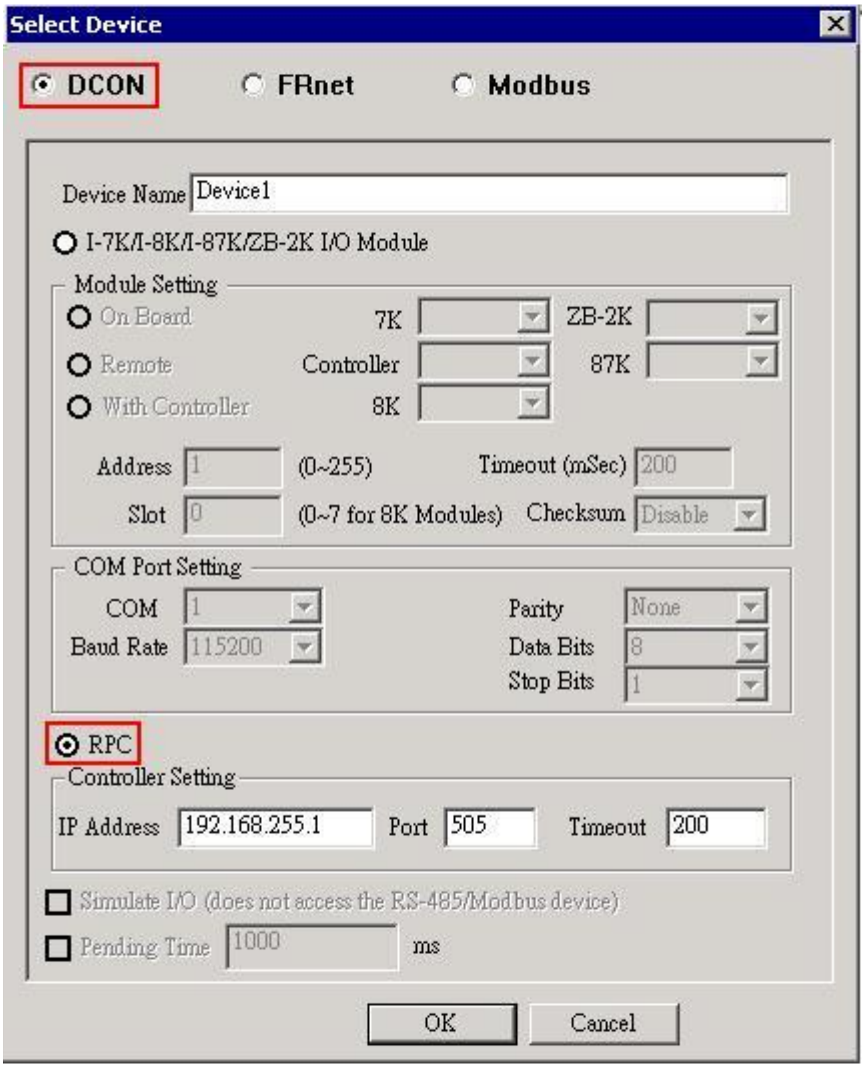
Step 1: Click on the "Add/ New Device…" menu item or the icon to add a new module.



Step 2: The "Select Device" dialog box pops up.

Step 3: Click on the "DCON" radio button.

Step 4: Click on the "RPC" radio button.



Select Device

● DCON ○ FRnet ○ Modbus

Device Name: Device1

○ I-7K/I-8K/I-87K/ZB-2K I/O Module

Module Setting

○ On Board 7K ZB-2K

○ Remote Controller 87K

○ With Controller 8K

Address: 1 (0~255) Timeout (mSec): 200

Slot: 0 (0~7 for 8K Modules) Checksum: Disable

COM Port Setting

COM: 1 Parity: None

Baud Rate: 115200 Data Bits: 8

Stop Bits: 1

● RPC

Controller Setting

IP Address: 192.168.255.1 Port: 505 Timeout: 200

□ Simulate I/O (does not access the RS-485/Modbus device)

□ Pending Time: 1000 ms

Device Name:

Names with spaces or punctuation such as “|!.,” cannot be used within a module name. The clients use the "Device Name" and "Tags" to access its value. The "Device Name" can not be the same as any other module.

Timeout:

Specifies timeout (Response time) value for this controller. The default value is 200 ms. A smaller timeout value may cause communication failure and a larger timeout value may reduce the performance of the client program.

Port:

You have to set up the value with “505” for communicating with NAPOPC_CE5 or NAPOPC_CE6.

IP Address:

The uniqe IP address of your NAPOPC_CE5 or NAPOPC_CE6.

OK:

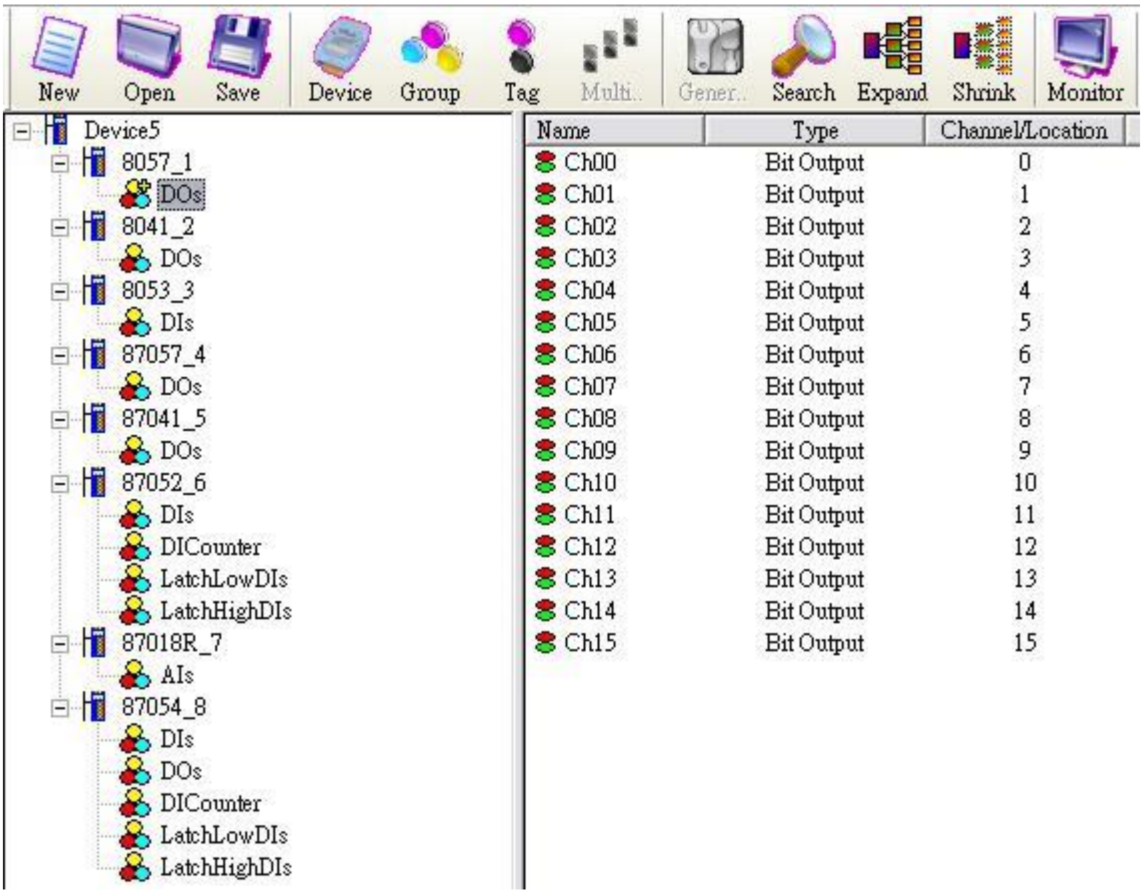
Click on the "OK" button to add the new controller setting.

Cancel:

Click on the "Cancel" button to avoid any changes.

Step 5: Click on the "OK" button to add this new device.

Step 6: After clicking on the "OK" button, NAPOPC_XPE will automatically synchronize and generate the modules of "NAPOPC_CE5/CE6".

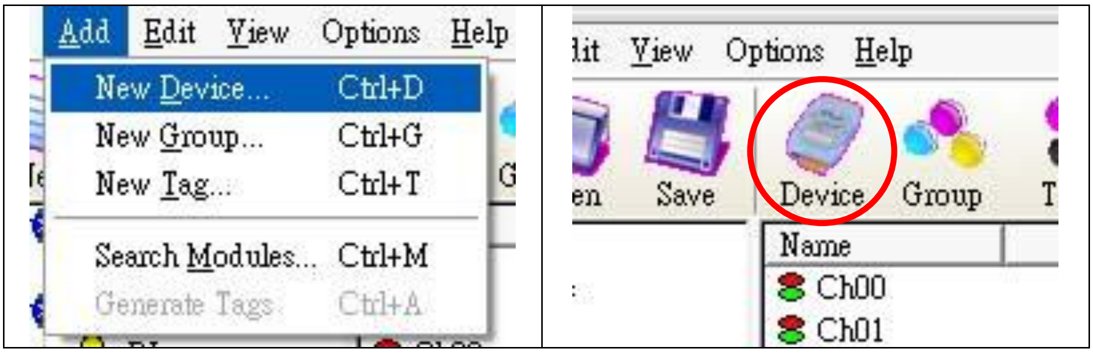


```
Device5
  8057_1
    DOs
  8041_2
    DOs
  8053_3
    DIs
  87057_4
    DOs
  87041_5
    DOs
  87052_6
    DIs
    DICounter
    LatchLowDIs
    LatchHighDIs
  87018R_7
    AIs
  87054_8
    DIs
    DOs
    DICounter
    LatchLowDIs
    LatchHighDIs
```

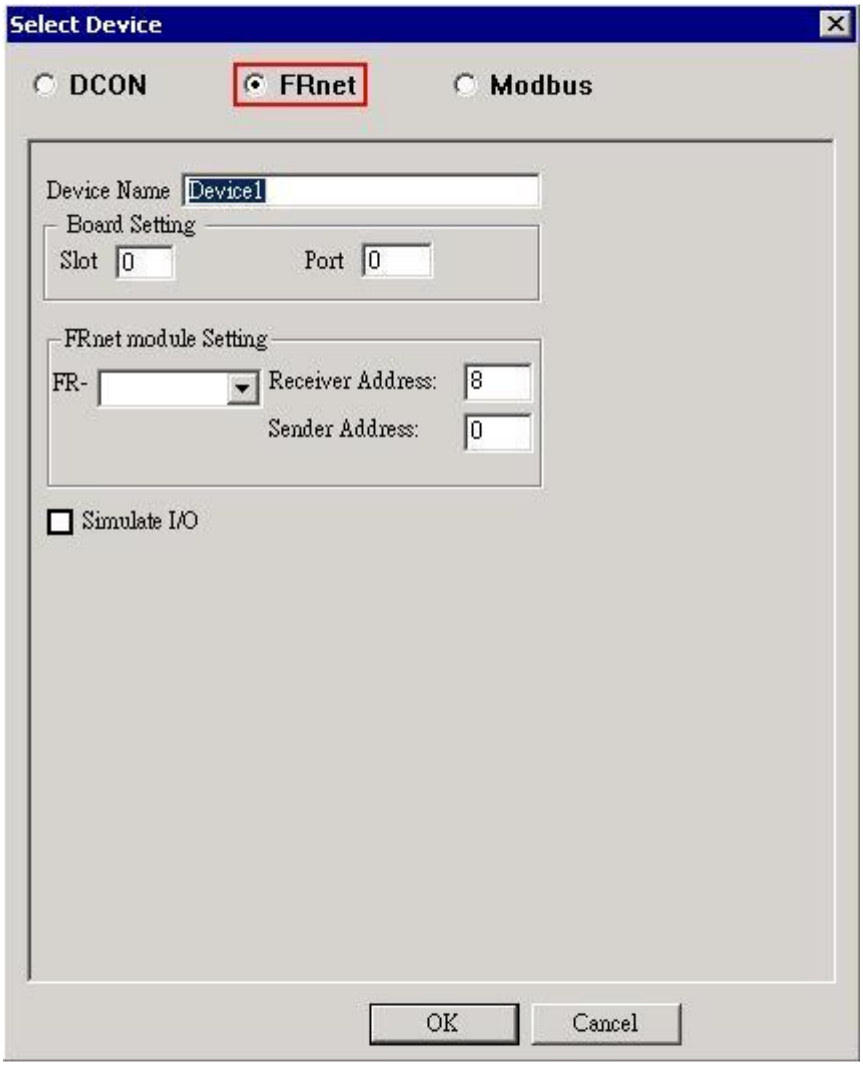
Name	Type	Channel/Location
Ch00	Bit Output	0
Ch01	Bit Output	1
Ch02	Bit Output	2
Ch03	Bit Output	3
Ch04	Bit Output	4
Ch05	Bit Output	5
Ch06	Bit Output	6
Ch07	Bit Output	7
Ch08	Bit Output	8
Ch09	Bit Output	9
Ch10	Bit Output	10
Ch11	Bit Output	11
Ch12	Bit Output	12
Ch13	Bit Output	13
Ch14	Bit Output	14
Ch15	Bit Output	15

1.6.3 Adding A New FRnet I/O

Step 1: Click on the "Add/ New Device…" menu item or the icon to add a new module.



Step 2: The "Select Device" dialog box pops up.
Step 3: Click on the "FRnet" radio button.



Select Device

○ DCON ● FRnet ○ Modbus

Device Name: Device1

Board Setting

Slot: 0

Port: 0

FRnet module Setting

FR-:

Receiver Address: 8

Sender Address: 0

□ Simulate I/O

Device Name:

Names with spaces or punctuation such as “|!.,” cannot be used within a module name. The clients use the "Device Name" and "Tags" to access its value. The "Device Name" can not be the same as any other module.

Slot:

The XPAC has 3 or 7 slots for the 8000 sub-device to plug in. This “slot” field indicates the slot number that the 8000 sub-device is using. The valid range is from 1 to 7.

Port:

The “Port” indicates the port number(0 or 1) of I-8172. Every FRnet I/O modules have to use I-8172 as FRnet communication module. Please refer to the I-8172 manual for more information.

FRnet Module ID:

User can click on the Combo Box to select a FRnet module ID.

Receiver Address:

FRnet communication needs correct hardware configurations for the sender address (SA) and receiver address (RA) on the host controller and the remote module in the network. Please refer to the FRnet manual for more information.

Sender Address:

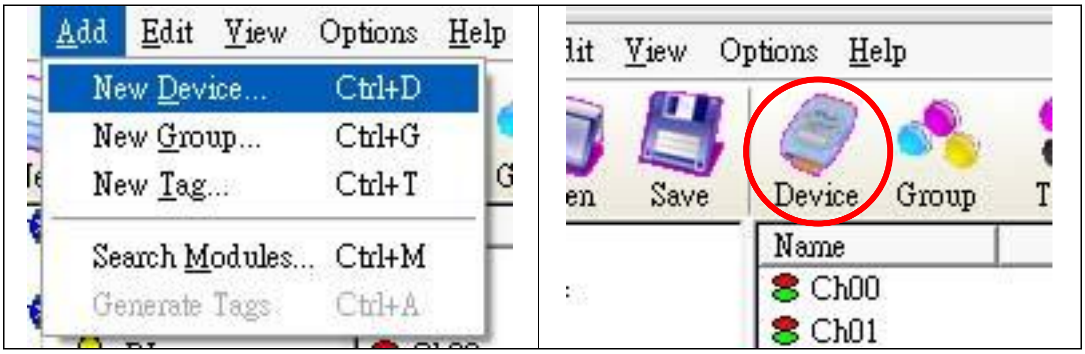
FRnet communication needs correct hardware configurations for the sender address (SA) and receiver address (RA) on the host controller and the remote module in the network. Please refer to the FRnet manual for more information.

Simulate I/O:

The "Simulate I/O" checkbox switches from reading I/O from the module to running a simulator. Since the simulator does not open the COM port, it is an easy way to work with the server, to configure tags or to connect clients without requiring any hardware.

1.6.4 Adding A New Modbus TCP Controller

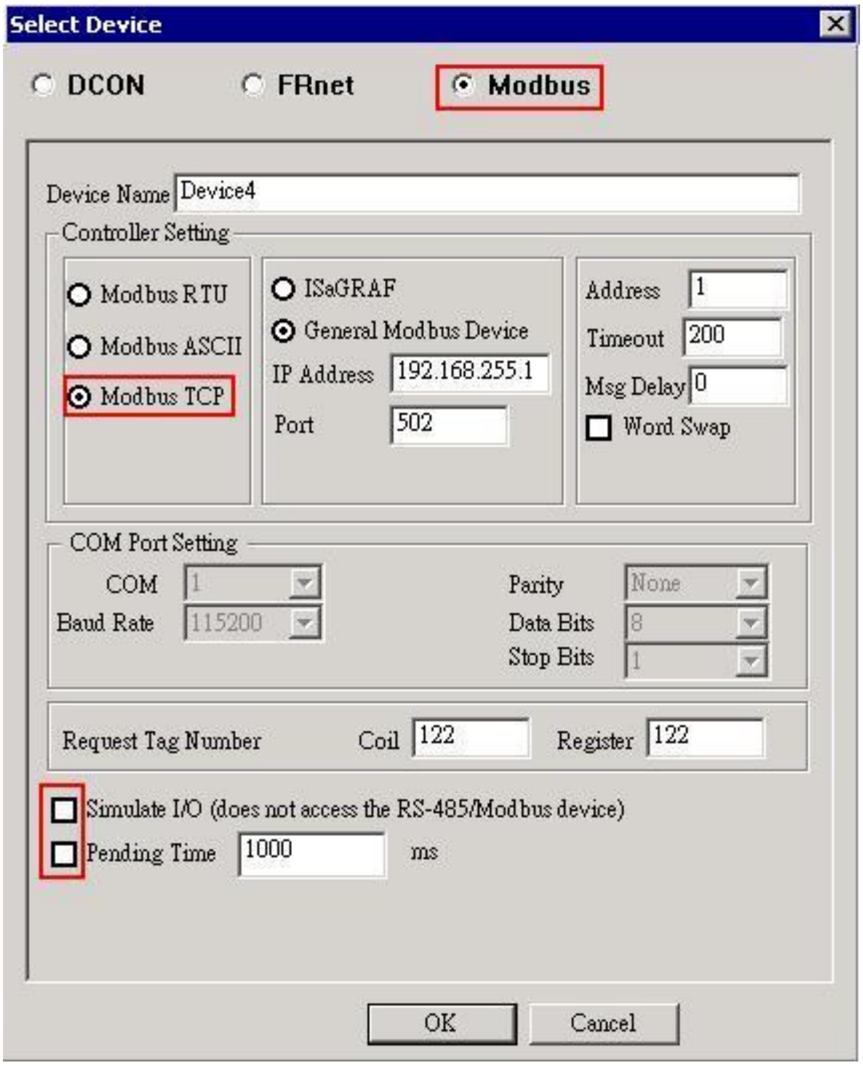
Step 1: Click on the "Add/ New Device…" menu item or the icon to add a new module.



Step 2: The "Select Device" dialog box pops up.

Step 3: Click on the "Modbus" radio button.

Step 4: Click on the "Modbus TCP" radio button.



Select Device

○ DCON ○ FRnet ● Modbus

Device Name: Device4

Controller Setting

○ Modbus RTU
○ Modbus ASCII
● Modbus TCP

○ ISaGRAF
● General Modbus Device
IP Address: 192.168.255.1
Port: 502

Address: 1
Timeout: 200
Msg Delay: 0
□ Word Swap

COM Port Setting

COM: 1
Baud Rate: 115200
Parity: None
Data Bits: 8
Stop Bits: 1

Request Tag Number Coil: 122 Register: 122

□ Simulate I/O (does not access the RS-485/Modbus device)
□ Pending Time 1000 ms

Device Name:

Names with spaces or punctuation such as “|!.,” cannot be used within a module name. The clients use the "Device Name" and "Tags" to access its value. The "Device Name" can not be the same as any other module.

ISaGRAF:

Connect ISaGRAF controller

General Modbus Device:

Connect general modbus device

IP Address:

The uniqe IP address of your Modbus TCP controller.

Port:

You have to set up the value with “502” for communicating with ICP DAS Modbus TCP controller.

Address:

Specifies a Address for this controller. The default value is 1 and the valid range is between 1 to 247.

Timeout:

Specifies timeout (Response time) value for this controller. The default value is 200 ms. A smaller timeout value may cause communication failure.

Msg Delay:

Specifies message delay value for this controller. The default value is 0 ms. A smaller msg delay value may have a higher system loading, but it will have a faster data exchange speed.

Word Swap:

The "Word Swap" checkbox switches the interpretation of 4 Byte values. Sometimes we need to make the checkbox "TRUE" in order to achieve the purpose of Lo-Hi/Hi-Lo communication.

Request Tag Number:

The "Request Tag Number" sets tag value numbers that each command will get from device. For ISaGRAF, it should less than 124 for coil and register. For 7188MTCP, it should less than 498 for coil and 127 for register. The default numbers are both 122. (For Modbus standard, it can't greater than 2000 for coil and 127 for register.

Simulate I/O:

The "Simulate I/O" checkbox switches to a simulator of reading I/O. Since the simulator does not open the TCP/IP port, it is an easy way to work with the server, to configure tags or to connect clients without requiring any hardware.

Pending Time:

Minimum interval time between two access. To activate this function, NAPOPC_XPE can work under optimized communication performance. If this module only needs to be accessed 1 time per 5 seconds. You can set pending time as 5000 ms. NAPOPC_XPE will automatically spread time resource to other modules which are connected with each other.

OK:

Click on the "OK" button to add the new controller setting.

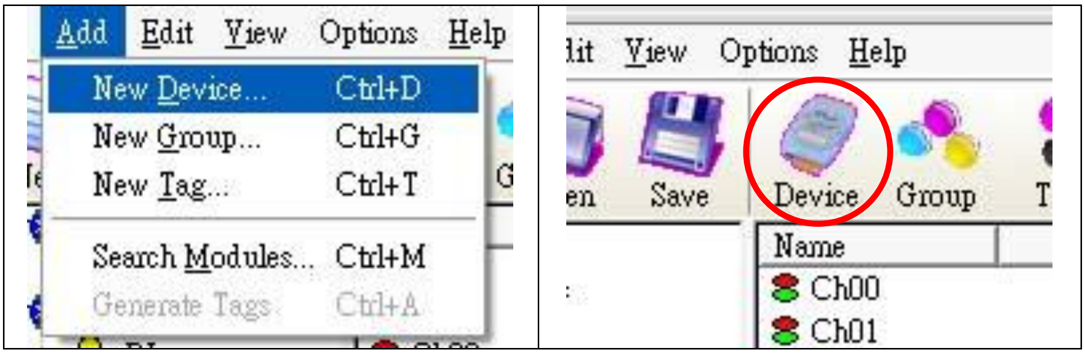
Cancel:

Click on the "Cancel" button to avoid any changes.

Step 5: Click on the "OK" button to add this new device.

1.6.5 Adding A New Modbus RTU Controller

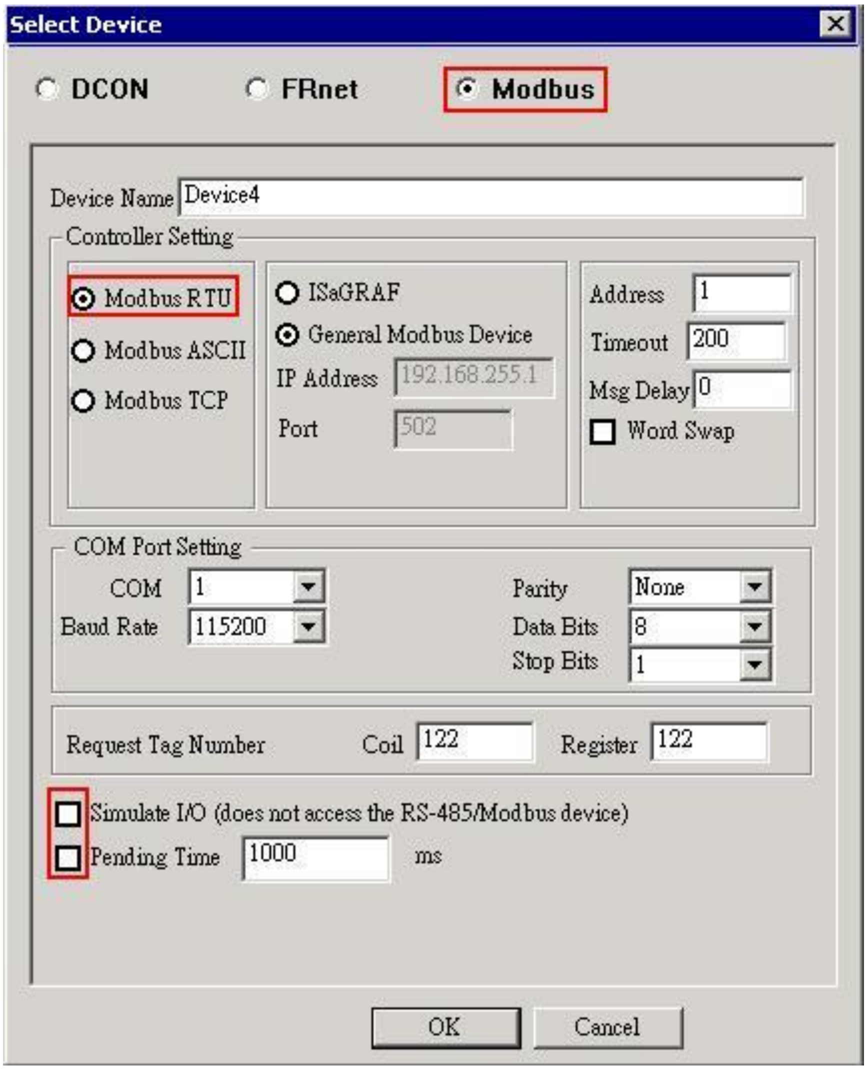
Step 1: Click on the "Add/ New Device…" menu item or the icon to add a new module.



Step 2: The "Select Device" dialog box pops up.

Step 3: Click on the "Modbus" radio button.

Step 4: Click on the "Modbus RTU" radio button.



○ DCON ○ FRnet ● Modbus

Device Name: Device4

Controller Setting

● Modbus RTU
○ Modbus ASCII
○ Modbus TCP

○ ISaGRAF
● General Modbus Device
IP Address: 192.168.255.1
Port: 502

Address: 1
Timeout: 200
Msg Delay: 0
□ Word Swap

COM Port Setting

COM: 1
Baud Rate: 115200
Parity: None
Data Bits: 8
Stop Bits: 1

Request Tag Number Coil: 122 Register: 122

□ Simulate I/O (does not access the RS-485/Modbus device)

□ Pending Time 1000 ms

Device Name:

Names with spaces or punctuation such as “|!.,” cannot be used within a module name. The clients use the "Device Name" and "Tags" to access its value. The "Device Name" can not be the same as any other module.

ISaGRAF:

Connect ISaGRAF controller

General Modbus Device:

Connect general modbus device

Address:

Specifies a Address for this controller. The default value is 1 and the valid range is between 1 to 247.

Timeout:

Specifies timeout (Response time) value for this controller. The default value is 200 ms. A smaller timeout value may cause communication failure and a larger timeout value may reduce the performance of the client program.

Msg Delay:

Specifies message delay value for this controller. The default value is 0 ms. A smaller msg delay value may have a higher system loading, but it will have a faster data exchange speed.

Word Swap:

The "Word Swap" checkbox switches the interpretation of 4 Byte values. Sometimes we need to make the checkbox "TRUE" in order to achieve the purpose of Lo-Hi/Hi-Lo communication.

COM Port:

Specifies the COM port to be used. Please verfiy which COM port number that the RS-232 / RS-485 network is using. Wrong settings will always cause communication failure.

Baud Rate:

Specifies the baud rate to be used. Verify the module's current baud rate. A wrong setting will always cause communication error for this controller.

Parity:

Specifies the parity scheme to be used. It is one of the following values.

Value	Description
None	No parity
Even	Even
Odd	Odd

Data Bits:

Specifies the number of bits in the bytes transmitted and received.

Stop Bits:

Specifies the number of stop bits to be used. It is one of the following values.

Value	Description
1	1 stop bit
2	2 stop bits
1.5	1.5 stop bits

Request Tag Number:

The "Request Tag Number" sets tag value numbers that each command will get from device. For ISaGRAF, it should less than 124 for coil and register. For 7188MTCP, it should less than 498 for coil and 127 for register. The default numbers are both 122. (For Modbus standard, it can't greater than 2000 for coil and 127 for register.

Simulate I/O:

The "Simulate I/O" checkbox switches to a simulator of reading I/O. Since the simulator does not open the TCP/IP port, it is an easy way to work with the server, to configure tags or to connect clients without requiring any hardware.

Pending Time:

Minimum interval time between two access. To activate this function, NAPOPC_XPE can work under optimized communication performance. If this module only needs to be accessed 1 time per 5 seconds. You can set pending time as 5000 ms. NAPOPC_XPE will automatically spread time resource to other modules which are connected with each other.

OK:

Click on the "OK" button to add the new controller setting.

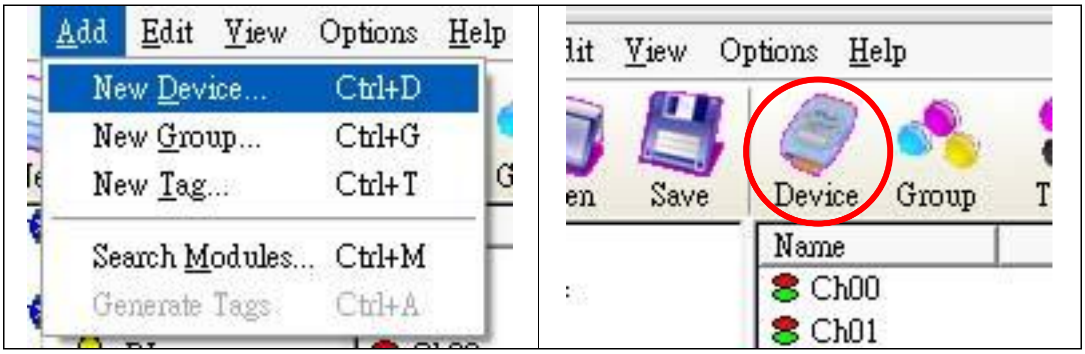
Cancel:

Click on the "Cancel" button to avoid any changes.

Step 5: Click on the "OK" button to add this new device.

1.6.6 Adding A New Modbus ASCII Controller

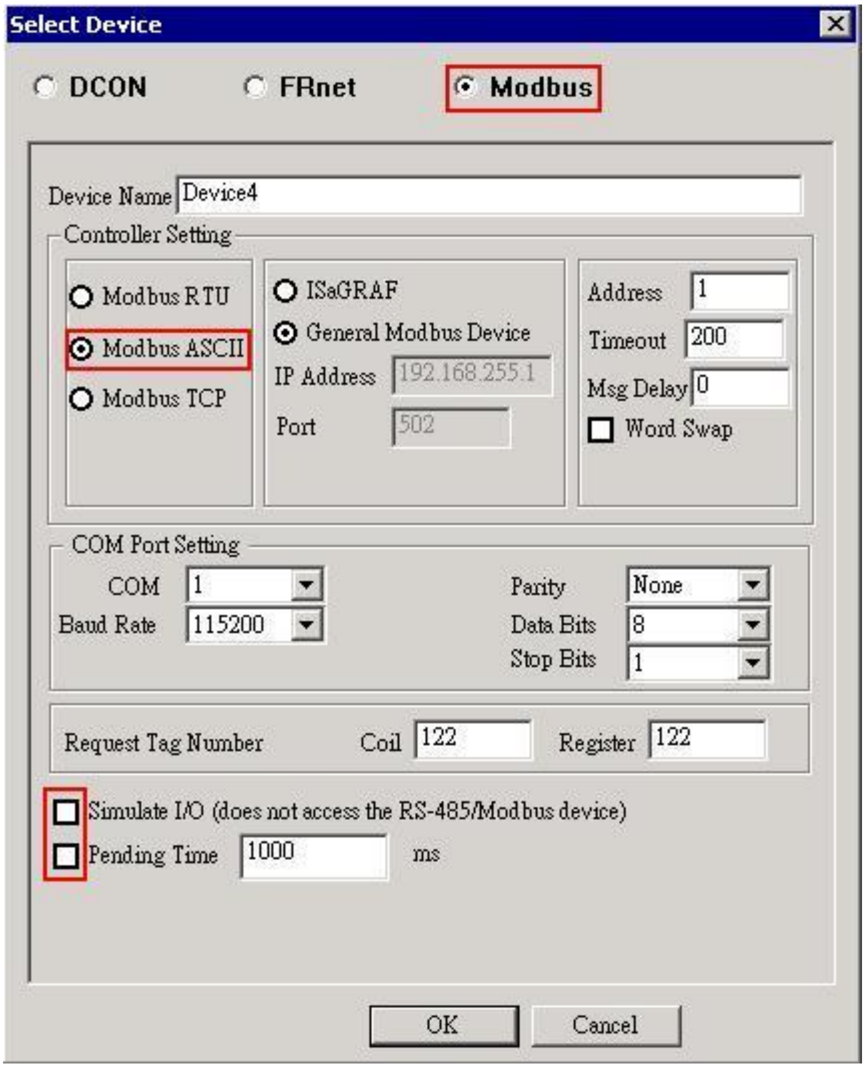
Step 1: Click on the "Add/ New Device…" menu item or the icon to add a new module.



Step 2: The "Select Device" dialog box pops up.

Step 3: Click on the "Modbus" radio button.

Step 4: Click on the "Modbus ASCII" radio button.



Select Device

○ DCON ○ FRnet ● Modbus

Device Name: Device4

Controller Setting

○ Modbus RTU
● Modbus ASCII
○ Modbus TCP

○ ISaGRAF
● General Modbus Device
IP Address: 192.168.255.1
Port: 502

Address: 1
Timeout: 200
Msg Delay: 0
□ Word Swap

COM Port Setting

COM: 1
Baud Rate: 115200
Parity: None
Data Bits: 8
Stop Bits: 1

Request Tag Number Coil: 122 Register: 122

□ Simulate I/O (does not access the RS-485/Modbus device)
□ Pending Time 1000 ms

Device Name:

Names with spaces or punctuation such as "|!.," cannot be used within a module name. The clients use the "Device Name" and "Tags" to access its value. The "Device Name" can not be the same as any other module.

ISaGRAF:

Connect ISaGRAF controller

General Modbus Device:

Connect general modbus device

Address:

Specifies a Address for this controller. The default value is 1 and the valid range is between 1 to 247.

Timeout:

Specifies timeout (Response time) value for this controller. The default value is 200 ms. A smaller timeout value may cause communication failure and a larger timeout value may reduce the performance of the client program.

Msg Delay:

Specifies message delay value for this controller. The default value is 0 ms. A smaller msg delay value may have a higher system loading, but it will have a faster data exchange speed.

Word Swap:

The "Word Swap" checkbox switches the interpretation of 4 Byte values. Sometimes we need to make the checkbox "TRUE" in order to achieve the purpose of Lo-Hi/Hi-Lo communication.

COM Port:

Specifies the COM port to be used. Please verfiy which COM port number that the RS-232 / RS-485 network is using. Wrong settings will always cause communication failure.

Baud Rate:

Specifies the baud rate to be used. Verify the module's current baud rate. A wrong setting will always cause communication error for this controller.

Parity:

Specifies the parity scheme to be used. It is one of the following values.

Value	Description
None	No parity
Even	Even
Odd	Odd

Data Bits:

Specifies the number of bits in the bytes transmitted and received.

Stop Bits:

Specifies the number of stop bits to be used. It is one of the following values.

Value	Description
1	1 stop bit
2	2 stop bits
1.5	1.5 stop bits

Request Tag Number:

The "Request Tag Number" sets tag value numbers that each command will get from device. For ISaGRAF, it should less than 124 for coil and register. For 7188MTCP, it should less than 498 for coil and 127 for register. The default numbers are both 122. (For Modbus standard, it can't greater than 2000 for coil and 127 for register.

Simulate I/O:

The "Simulate I/O" checkbox switches to a simulator of reading I/O. Since the simulator does not open the TCP/IP port, it is an easy way to work with the server, to configure tags or to connect clients without requiring any hardware.

Pending Time:

Minimum interval time between two access. To activate this function, NAPOPC_XPE can work under optimized communication performance. If this module only needs to be accessed 1 time per 5 seconds. You can set pending time as 5000 ms. NAPOPC_XPE will automatically spread time resource to other modules which are connected with each other.

OK:

Click on the "OK" button to add the new controller setting.

Cancel:

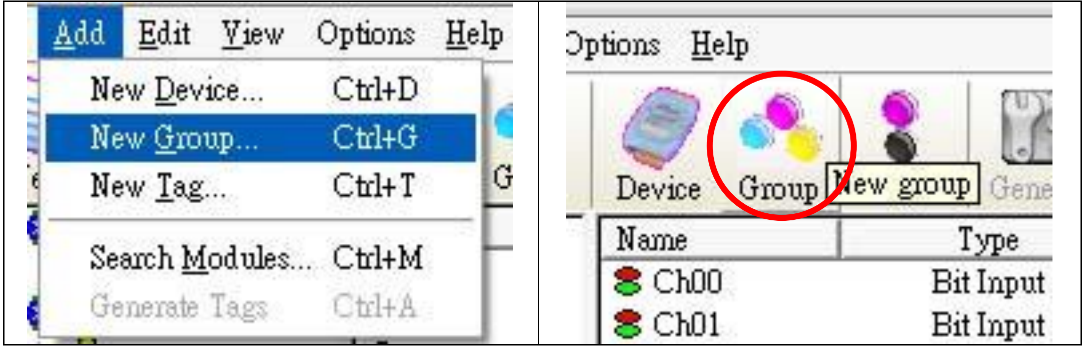
Click on the "Cancel" button to avoid any changes.

Step 5: Click on the "OK" button to add this new device.

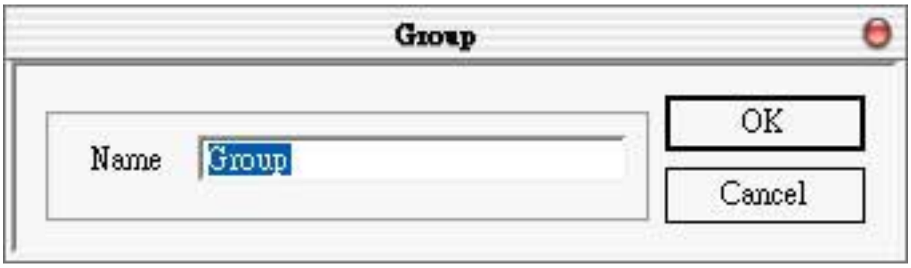
1.7 Adding A New Group

If the device you add is "RPC", you do not need to add groups manually. NAPOPC_XPE will automatically synchronize and generate the modules of "RPC".

Step 1: Click on the "Add/ New Group" menu item or the icon to add a new group.



Step 2: The "Group" dialog box pops up.



Name: Group

Name:

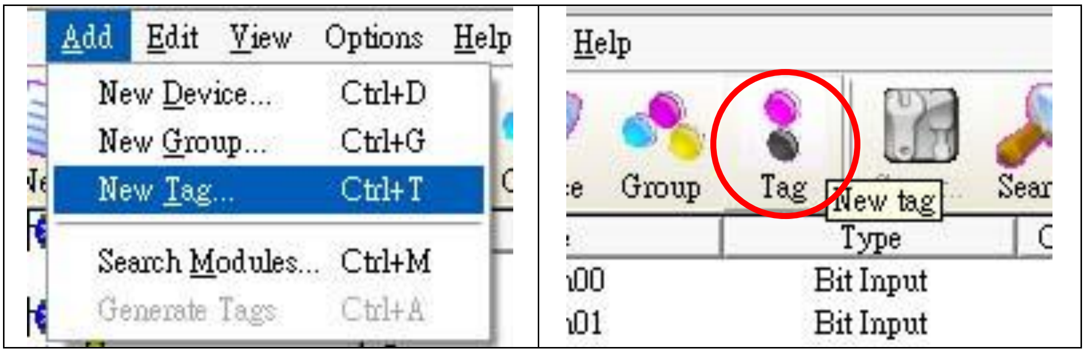
A "Group Name" may have any name, but avoid names with spaces or punctuation such as "|!.,". The "Group Name" must not be used twice. A group can be defined as a subdirectory containing one or more tags. A device may have many subgroups of tags. All tags belong to their module when they are scanned to perform I/O.

1.8 Adding A New Tag

If the device you add is "RPC", you do not need to add tags manually. NAPOPC_XPE will automatically synchronize and generate the modules of "RPC".

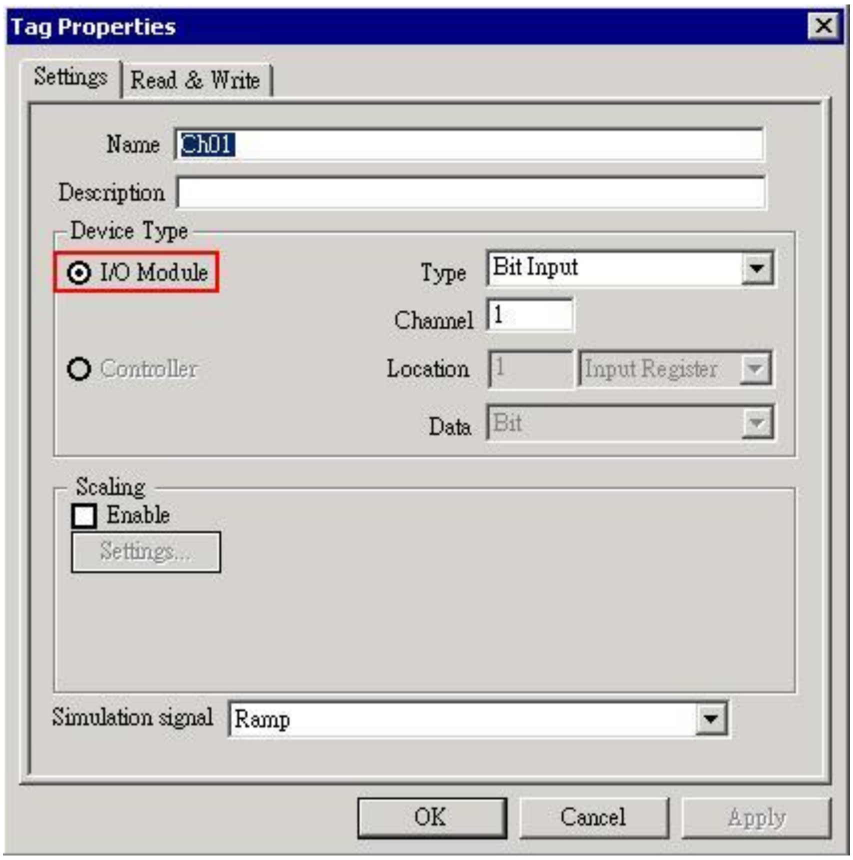
1.8.1 Adding A New Tag For I/O Module

Step 1: Click on the "Add/ New Tag" menu item or the icon to add a new tag.



Step 2: The "Tag Properties" dialog box pops up.

Step 3: Choose the "Settings" page. Because the tag belongs to the module-type device, the "I/O Modules" radio button is active.



Name: Ch01

Description:

Device Type

● I/O Module

○ Controller

Type: Bit Input

Channel: 1

Location: 1 Input Register

Data: Bit

Scaling

□ Enable

Settings...

Simulation signal: Ramp

Name:

Any "Tag Name" may be used, but avoid names with spaces or punctuation such as "|!.,". The clients will use the "Device Name" and "Tags" to access its value. Hence the "Tag Name" cannot be a duplicate of another tag in the same group.

Description:

Specifies the description text for this tag. This can be blank.

Type:

Specifies the command to be used for this tag. Different modules support different commands. For commands, please refer to a "MODULES.HTM" file in \\ICPDAS\NAPOPC_XPE folder

Channel:

Specifies the channel number to be used for this tag. The "Digital Input" and "Digital Output" tags do not use this channel setting, because all channels are read with one communication.

Simulation signal:

The valid signal is SINE, RAMP and RANDOM. This field is validated when the module uses simulation I/O. Please refer to the "Adding A New Device" section.

OK:

Click on the "OK" button to add the new tag setting.

Cancel:

Click on the "Cancel" button to avoid any changes.

Scaling:

Enable:

Check this check-box to enable the "Settings…" button.

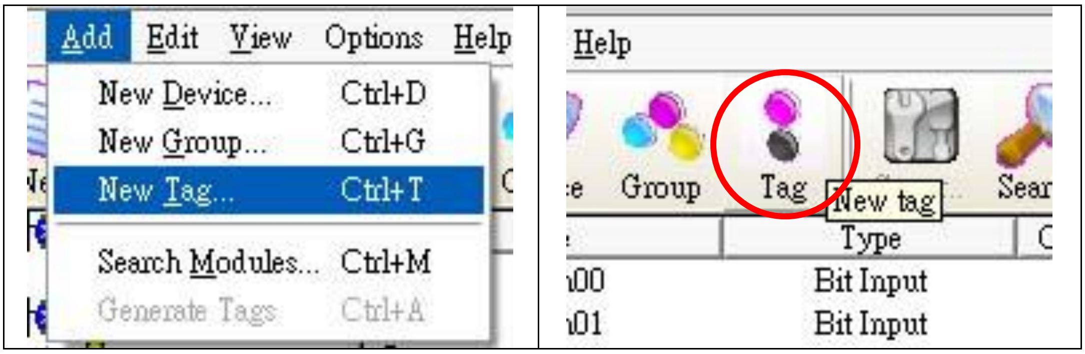
Settings:

Click on this button to set the scaling feature.

For more information, please refer to the section "1.8.3 Scaling Settings".

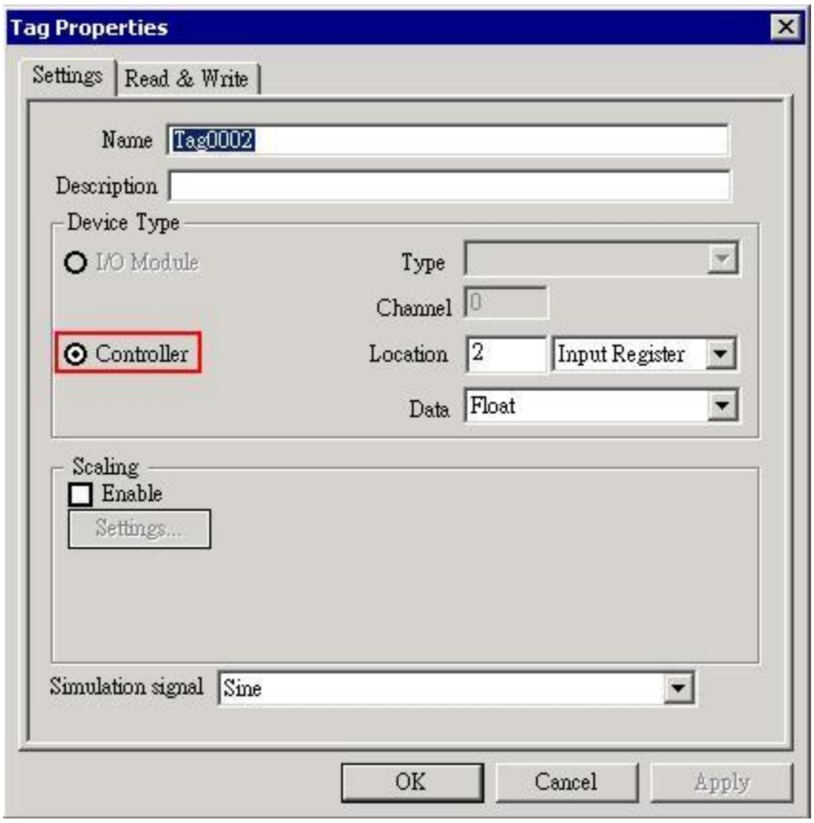
1.8.2 Adding A New Tag For Controller

Step 1: Click on the "Add/ New Tag" menu item or the icon to add a new tag.



Step 2: The "Tag Properties" dialog box pops up.

Step 3: Choose the "Settings" page. Because the tag belongs to the controller-type device, the "Controller" radio button is active.



Name: Tag0002

Description:

Device Type: ○ I/O Module ● Controller

Type:

Channel: 0

Location: 2 Input Register

Data: Float

Scaling: □ Enable

Simulation signal: Sine

Name:

Any "Tag Name" may be used, but avoid names with spaces or punctuation such as “|!.,”. The clients will use the "Device Name" and "Tags" to access its value. Hence the "Tag Name" cannot be a duplicate of another tag in the same group.

Description:

Specifies the description text for this tag. This can be blank.

Data:

Specifies the data type of this tag which’s location type is “Input Register” or “Output Register”. NAPOPC_XPE Server support five kinds of data type which are “Short”, “Long”, “Float”, “Word”, and “DWord”.

Data Type	Definition	Range
Short	16-bit signed integer	-32768~32767
Long	32-bit signed integer	-2147483648~2147483647
Float	Floating-point variable	-1.7E-308~1.7E+308
Word	16-bit unsigned integer	0~65535
DWord	32-bit unsigned integer	0~4294967295

The data type of “Input Coil” or “Output Coil” is “Bool”.

Location:

Specifies the tag address. It must be the same with the the variable address in the controller. Besides, you have to choose the location type. After you choose the location number, there are four location types you can choose.They are ”Input Coil”, “Output Coil”, “Input Register”, and “Output

Register". When you monitor controller device(see 1.5 Monitoring Device), the "Channel/Location" field will show a value according to the location and location type as below.

Location Type	Range
Output Coil	000001 - 065536
Input Coil	100001 - 165536
Input Register	300001 - 365536
Output Register	400001 - 465536

Simulation signal:

The valid signal is SINE, RAMP and RANDOM. This field is validated when the module uses simulation I/O. Please refer to the "Adding A New Device" section.

OK:

Click on the "OK" button to add the new tag setting.

Cancel:

Click on the "Cancel" button to avoid any changes.

Scaling:

Enable:

Check this check-box to enable the "Settings…" button.

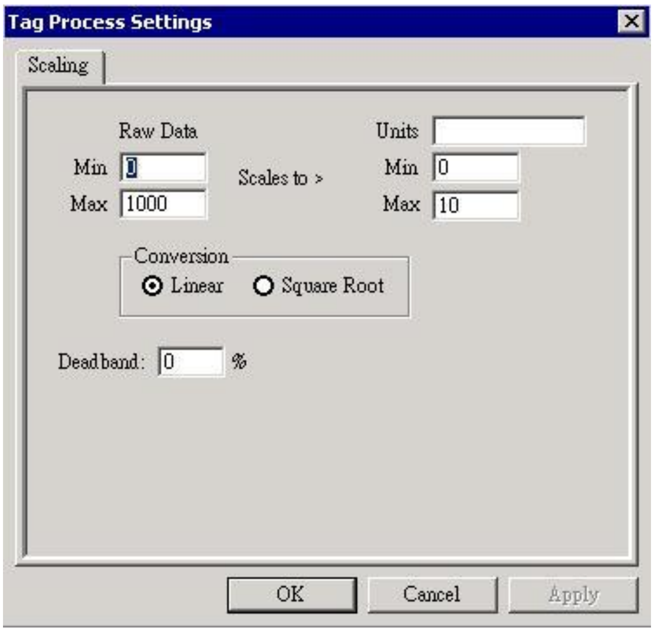
Settings:

Click on this button to set the scaling feature.

For more information, please refer to the section "1.8.3 Scaling Settings".

1.8.3 Scaling Settings

In general, the "Scaling" feature is only useful for the "floating-point" data type.



Raw Data Min: 0
Raw Data Max: 1000
Units:
Min: 0
Max: 10
Conversion: ● Linear ○ Square Root
Deadband: 0 %

Raw Data:

Min: The original Minimum value. ([MinRaw])
Max: The original Maximum value. ([MaxRaw])

Scales to:

Units: The unit of the scaled value. (Just for reference only.)
Min: The scaled Minimum value. ([MinScale])
Max: The scaled Maximum value. ([MaxScale])

Conversion:

Linear:
Scaled Value = ((Original Value – [MinRaw]) / ([MaxRaw] – [MinRaw]))
* ([MaxScale] – [MinScale]) + [MinScale]

Square Root:
Scaled Value = ((sqrt (Original Value) – [MinRaw]) * ([MaxScale] – [MinScale]))
/ sqrt ([MaxRaw] – [MinRaw]) + [MinScale]

Deadband (%):

In general, keep "0" in this field.
For more information, please refer to the "4.5.1.6 Percent Deadband" section in the "OPCDA20_Cust.PDF" manual, page 68.

OK:

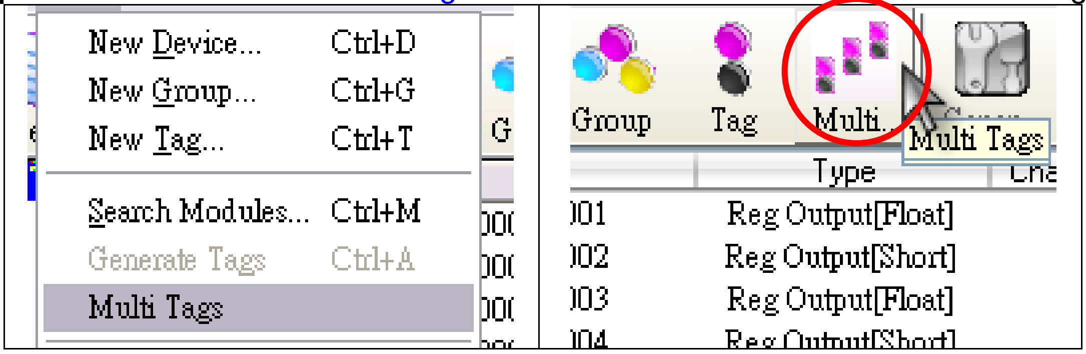
Click the "OK" button to save these settings.

Cancel:

Click the "Cancel" button to avoid any changes.

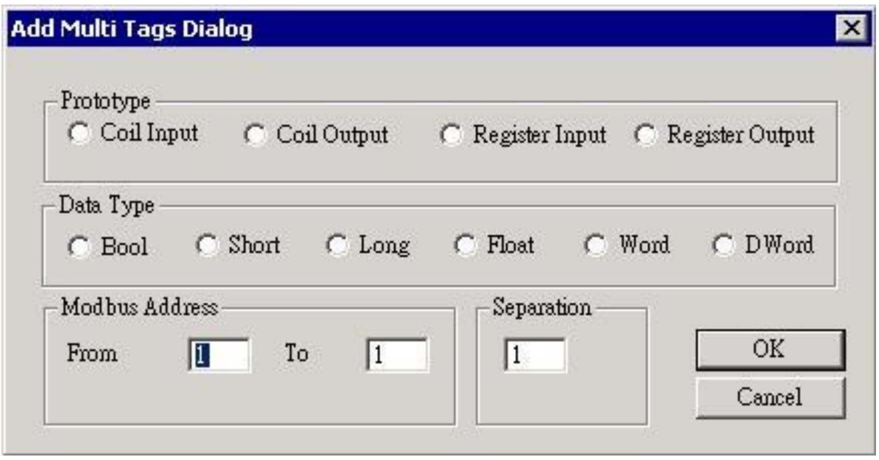
1.9 Adding Multi Tags For Modbus Device

This function only works when the device's protocol is Modbus.
Step 1: Click on the "Add/ Multi Tags" menu item or the icon to add a new tag.



Step 2: The "Add Multi Tags Dialog" dialog box pops up.

Step 3: Choose correct "Prototype" ,"Data Type" and key in Modbus address.



Prototype:

There are four kinds of prototype for modbus tag. "Coil Input", "Coil Output", "Register Input" and "Register Output".

Data Type:

"Bool" : 8 bits, True or False
"Short" : 16 bits, -32768 ~ 32767
"Long" : 32 bits, -2147483648. ~ 2147483647
"Float" : 32 bits, float numbers
"Word" : 16 bits, 0 ~ 65535
"DWORD" : 32 bits, 0 ~ 4294967295

Modbus Address:

"From" : modbus address number of start tag, 1 ~ 65535
"To" : modbus address number of end tag. 1 ~ 65535

Separation:

Separation numbers between each tag. 1 ~ 100

OK:

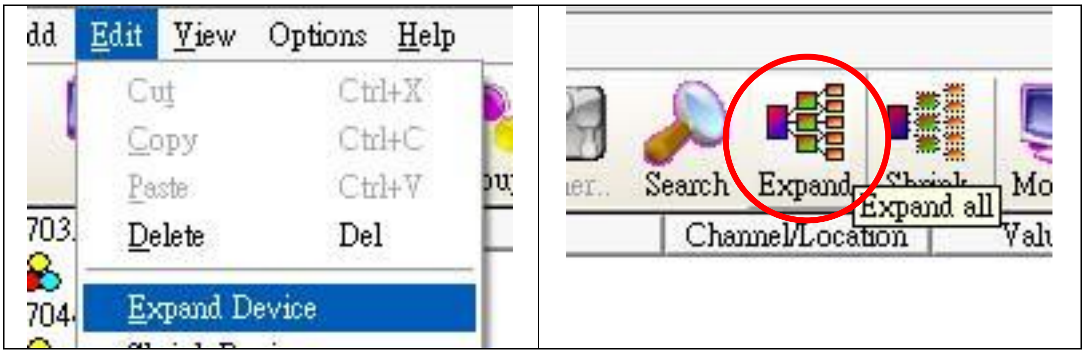
Click on the "OK" button to add the new tag setting.

Cancel:

Click on the "Cancel" button to avoid any changes.

1.10 Expand/ Shrink Devices

Click on the "Edit/ Expand device (Shrink device)" menu item or the Expand (Shrink) icon to expand(shrink) all devices.



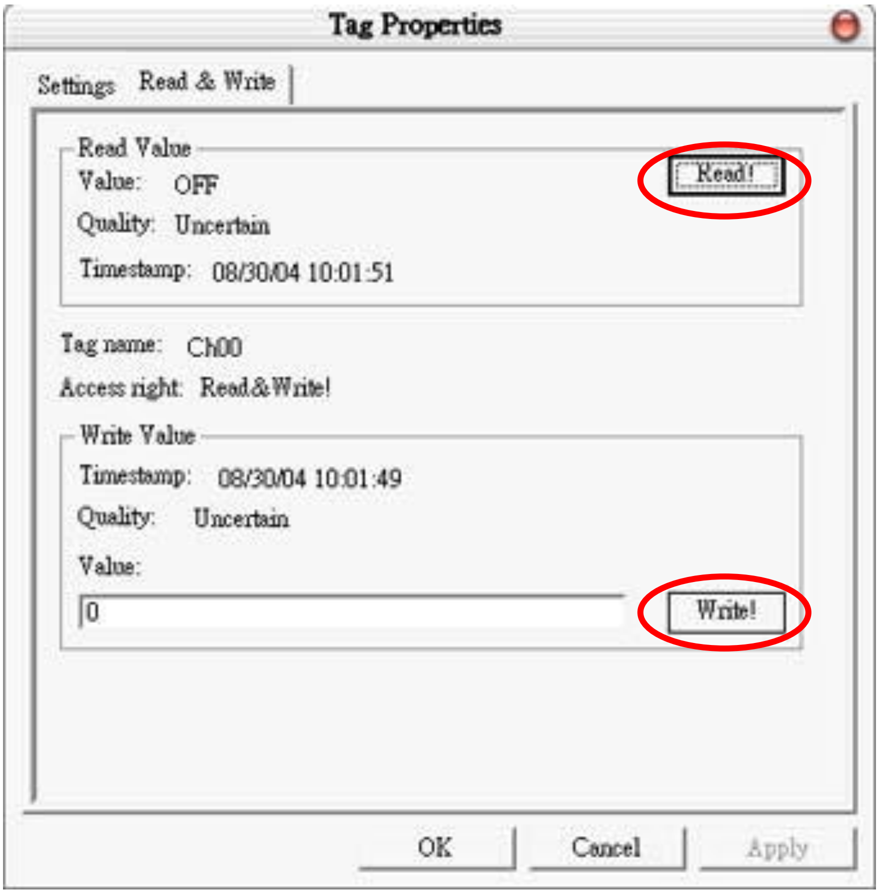
1.11 Read/Write Tags

First, you have to use the "Monitor" function to see values of tags by checking the "View/ Monitor" menu item. Select a tag and right click the mouse button. Then select the "Properties..." option. Choose the "Read & Write" page to read/write the tag.

Step 1: Click the "View/ Monitor" menu item to enable monitor.

Step 2: Select a tag and right click the mouse button. Then select the "Properties.." option.

Step 3: Select the "Read & Write" page. You can see the "Tag name" and "Access right" at the first. If the access right is "Read only!", the write function is disable.



Read Value
Value: OFF
Quality: Uncertain
Timestamp: 08/30/04 10:01:51

Tag name: Ch00
Access right: Read&Write!

Write Value
Timestamp: 08/30/04 10:01:49
Quality: Uncertain
Value: 0

Read Value/Value:

You can press the "Read!" button to read the tag value as you saw on the "Tag-Window".

Read Value/Quality:

Three kinds of qualities, "Good", "Bad", and "Uncertain", would be shown. If the communication status is good, the quality shows "Good". If the communication status has something wrong, the quality shows "Bad". And the other situation is "Uncertain". Maybe you do not click the "View/ Monitor" menu item to enable monitor etc.

Read Value/Timestamp:

It shows the time, when you read the tag.

Tag name:

It is the same with the "Name" at the "Settings" page. You can modify it at the "Settings" page.

Access right:

Two kinds of aceess right, "Read Only!" and "Read&Write!", would be shown. The access right depends on what kind of tag property it is. Please refer to the "1.7 Adding A New Tag"

Write Value/Timestamp:

It shows the time, when you write the tag.

Write Value/Quality:

Three kinds of qualities, "Good", "Bad", and "Uncertain", would be shown. If the communication status is good, the quality shows "Good". If the communication status has something wrong, the shows "Bad". And the other situation is "Uncertain". Maybe you do not click the "View/ Monitor" menu item to enable monitor etc.

Write Value/Value:

You can press the "Write!" button to write the value you key-in to the tag. If the tag data type is "Boolean" the write value "0" means "OFF" and the write value "not 0" means "ON".

1.12 Editing A Device/Group/Tag Properties

To edit a existing Device(/Group/Tag), just select the Device(/Group/Tag) and right click the mouse button. Then select the "Properties…" option.

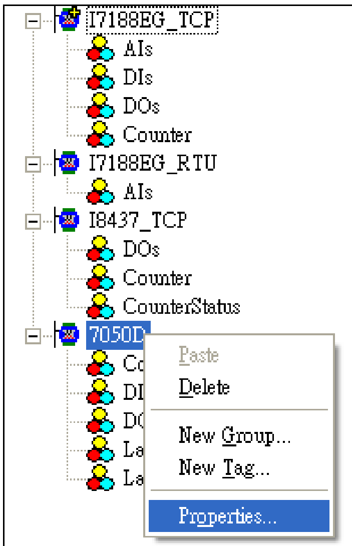


Figure 1-12-1.

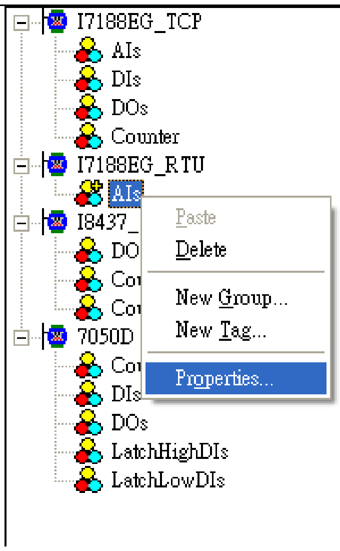


Figure 1-12-2.

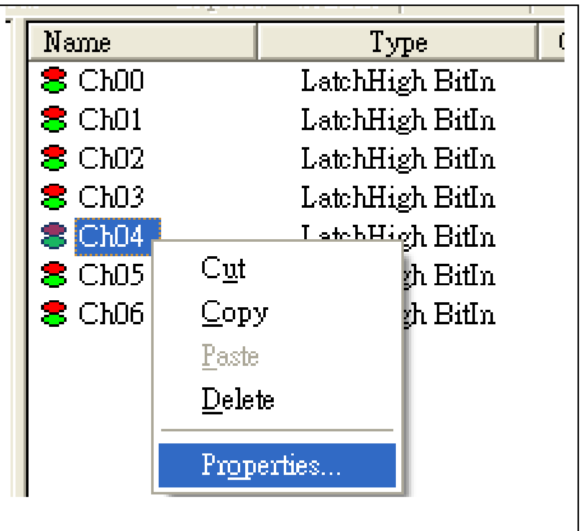


Figure 1-12-3.

Figure 1-12-1. Select a device and right click the mouse button. Then select the "Properties.." option.

Figure 1-12-2. Select a group and right click the mouse button. Then select the "Properties.." option.

Figure 1-12-3. Select a tag and right click the mouse button. Then select the "Properties.." option.

1.13 Deleting A Device/Group/Tag

To delete a existing Device/Group/Tag, just select the Device(/Group/Tag) and right click the mouse button. Then select the "Delete…" option or the icon.

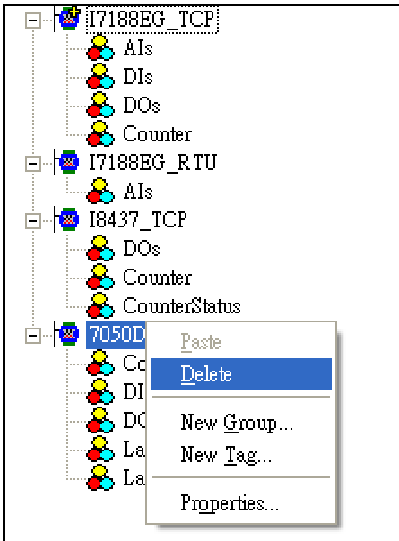


Figure 1-13-1.

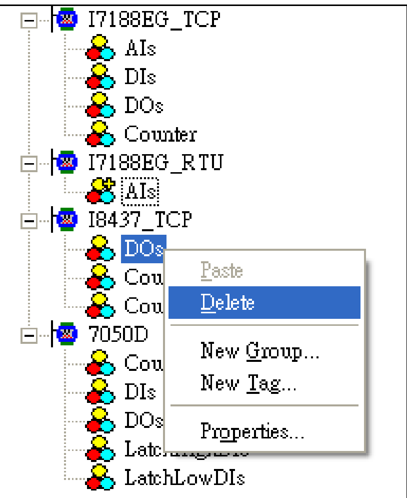


Figure 1-13-2.

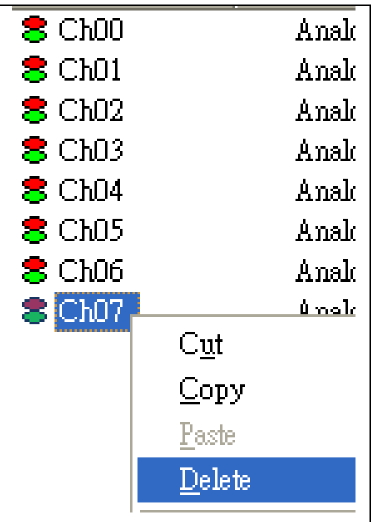


Figure 1-13-3.

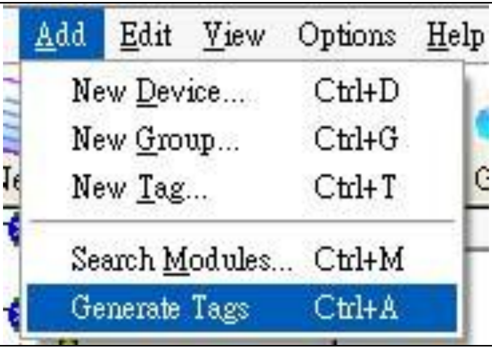
Figure 1-13-1. Select a device and right click the mouse button. Then select the "Delete" option.
Figure 1-13-2. Select a group and right click the mouse button. Then select the "Delete" option.
Figure 1-13-3. Select a tag and right click the mouse button. Then select the "Delete" option.

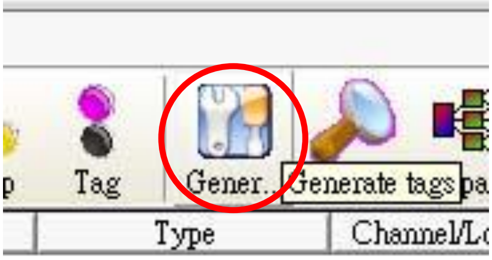
1.14 Generating Tags

This function lets you easily create tags of specific module and test it in the OPC server under simulation mode. It is only valid when the selected device has no "Group" or “Tag” such as a new device.

Step 1: Select a device you want to generate tags.

Step 2: Click on the "Add/ Generate Tags" menu item or the icon to generate tags.

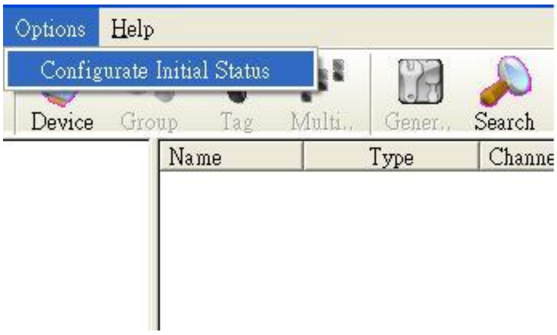




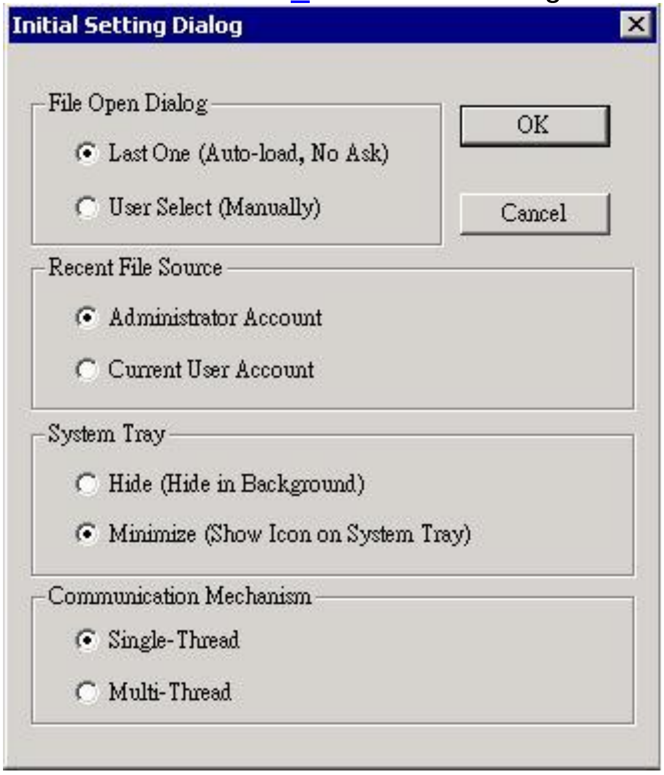
1.15 Configuring Initial Status

This function lets you configurate initial status of NAPOPC_XPE server when client softwares connect to it. The ““Recent File Source” options lets user save related information in registry under “Administrator account” or “Current user account”. Some DCOM application needs to adjust this option for specific scenario. The “File Open Dialog” option lets user load a file automatically or manually. The “System Tray” option can hide or minimize program of NAPOPC_XPE server. And “Communication Mechanism” option lets user define the communication behavior of NAPOPC_XPE. Generally, “Multi-Thread” is the best choice for high performace. However, for some particular OPC clients which can not work smoothly under “Multi-Thread” communication, user can choose “Single-Thread” instead. This function, Configurate Initial Status, can be selected only when “Monitor” function isn’t running.

Step 1: Click on the "Options/ Configurate Initial Status" menu item to open “Initial Setting Dialog”.



Step 2: Select initial status of NAPOPC_XPE at this dialog


Initial Setting Dialog

File Open Dialog
● Last One (Auto-load, No Ask)
○ User Select (Manually)

Recent File Source
● Administrator Account
○ Current User Account

System Tray
○ Hide (Hide in Background)
● Minimize (Show Icon on System Tray)

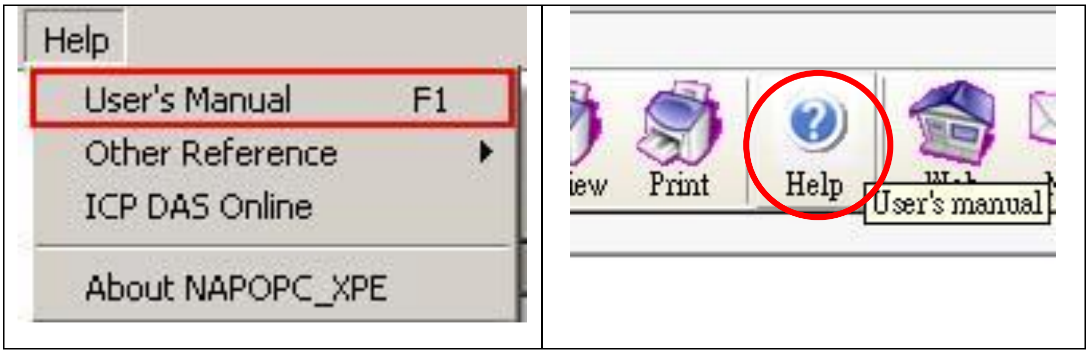
Communication Mechanism
● Single-Thread
○ Multi-Thread

OK
Cancel

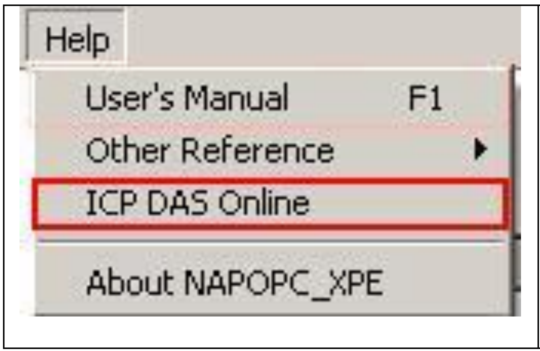
1.16 Help

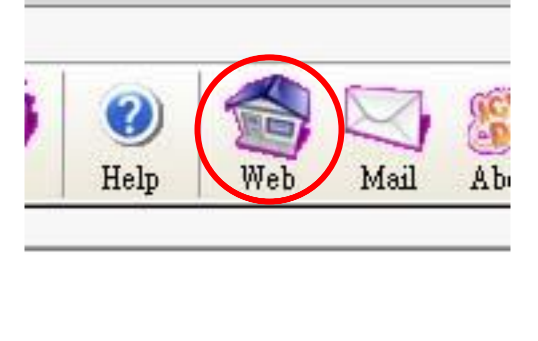
Refer to the user’s manual by Checking the "Help/ User’s Manual" menu item. All PDF formatted files are best view using Acrobat Reader 5 or newer. You can install it from our CD or download a free copy from Adobe's Web Site.

Clicking on the "Help/ User's Manual" menu item or the icon refer to the user’s manual.



Welcome to visit our web site by checking the "Help/ ICP DAS Online" menu item or pressing the icon.



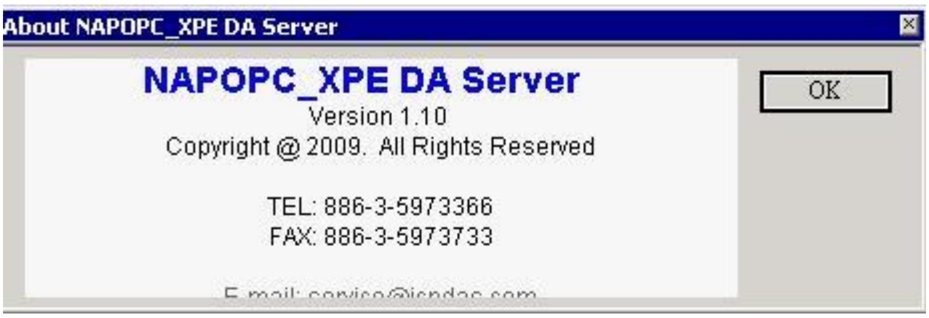


1.17 About

Click on the "Help/ About NAPOPC_XPE" menu item or the icon to see the "About NAPOPC_XPE DA Server" dialog box. It shows the version number of NAPOPC_XPE DA Server.

Step 1: Click on the "Help/ About NAPOPC_XPE" menu item.

Step 2: The "About NAPOPC_XPE DA Server" window pops up.



About NAPOPC_XPE DA Server

NAPOPC_XPE DA Server

Version 1.10

Copyright @ 2009. All Rights Reserved

TEL: 886-3-5973366

FAX: 886-3-5973733

E-mail: service@icpdas.com

2 Quick Start

Please follow these steps:

[Configure NAPOPC_XPE Server]

1. Wire Modules or Controllers.
   Wire modules in the RS-232 / RS-485 network. (Refer to "\CD \Napdos\7000\manual \GetStart.PDF" manual.)
   Wire controllers to your PC.

2. Configure Modules or Controllers.
   Use DCON Utility to set modules. (Refer to "\CD \Napdos\7000\manual \GetStart.PDF" manual.)
   Use ISaGRAF to configure the I-7188EG/XG or I-8xx7.

3. Install the OPC server.
   Install the NAPOPC_XPE on XPAC.

4. Run the OPC Server.
   Launch the OPC server by executing " C:\ICPDAS\NAPOPC_XPE\NAPOPCSvr_XPE.exe"

5. Search Modules.
   Refer to the "1.4 Search Modules…" section to search modules in the RS-485 network.

6. Add a new controller
   Refer to the "1.6 Adding A New Device" section to add a I/O or add a new modbus RTU, modbus ASCII or modbus TCP controller.

7. Save Configuration.
   Save the configuration by clicking "File/Save" menu item.

8. Close OPC server.
   Close OPC Server by clicking "File/Exit" menu item.

[Connect to NAPOPC_ST Server]

1. Connect to OPC server.
   Run your client program and connect to our OPC server by linking "NAPOPC.Svr" or " NAPOPC.Svr.1".
   (Please refer to user's manual of your client software provided by your vendor.)
   This forces the system to run the OPC server automatically in background.

3 Connect To OPC Server

The OPC is defined by the OPC Foundation, and any client program supporting OPC can connect to OPC server (for example: Lab VIEW v5.0 and WIZCON 7.51). Any computer language supporting the COM mechanism can also connect to the OPC server directly through the COM interface.

The first section shows you how to optimize your communication. And the following sections show you how to connect to OPC server by using client program provided by Factory Soft, Inc, Lab VIEW, Server Explorer provided by National Instruments, WIZCON, iFix, InduSoft and CitectSCADA. To connect to OPC server by other OPC client, please refer to your OPC client user's manual.

3.1 Optimize Your Communication

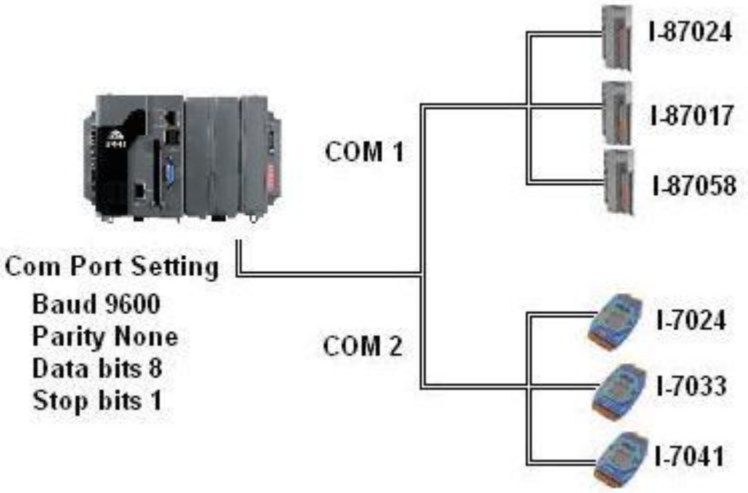
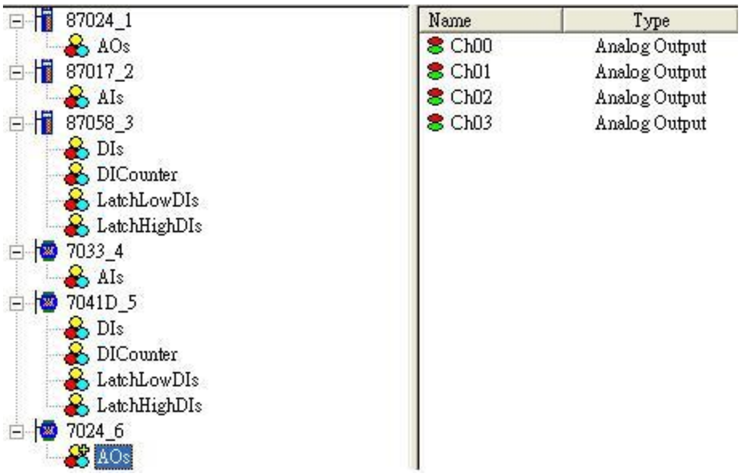


Figure 3.1.1 Communication architecture of I/O modules

Figure 3.1.1 shows the communication architecture of I/O modules. NAPOPC_XPE server accesses to I-87024/I-87017/I-87058/I-7024/I-7033/I-7041 via serial COM port. Suppose we need the interval time of accessing I-87024 and I-7024 is 1 sec and the interval time of I-7041 and I-87058 is 3 sec. But, we want to update I-7033 and I-87017 every 100 ms. For this purpose, we can achieve it by seven steps as below.

Step 1: First of all, we try to connect all modules on COM1 and to auto search these modules.

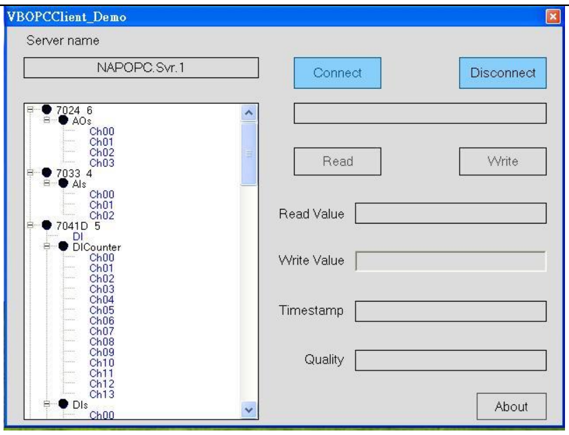


Name	Type
Ch00	Analog Output
Ch01	Analog Output
Ch02	Analog Output
Ch03	Analog Output

Step 2: Connect OPC client to NAPOPC_XPE Server. At the status bar of NAPOPC_XPE window, it shows average scan time Avg Scan time: 968 ms when clicking any module.

We will find the average scan time is 968 ms.

	COM1					
Module	I-87024	I-87017	I-87058	I-7033	I-7041	I-7024
Avg Scan Time	**968 ms**					



Server name: NAPOPC.Svr.1

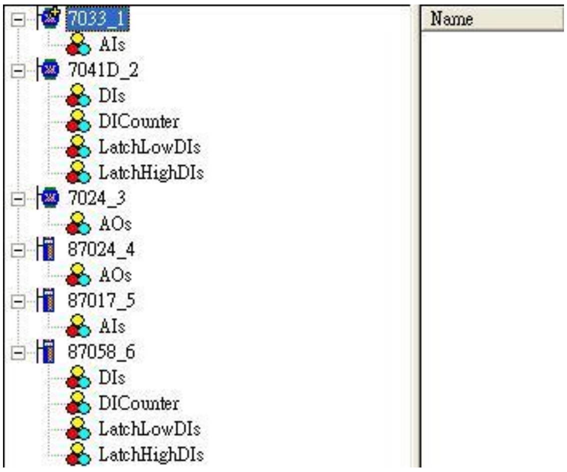
Read Value

Write Value

Timestamp

Quality

Step 3: We divide these modules into two groups. I-87024/I-87017/I-87058 connects to COM1. I-7033/I-7041/I-7024 connects to COM2. And we search again.



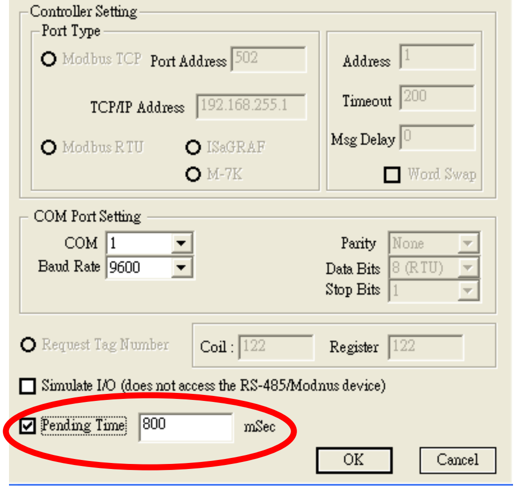
Step 4: Connect OPC client to NAPOPC_XPE server again. We can discover the average scan time separately when clicking each module. We will find the average scan time of COM1 is 391 ms, and of COM2 is 516 ms.

	COM 1			**COM2**		
Module	I-87024	I-87017	I-87058	I-7033	I-7041	I-7024
Avg Scan Time	**391 ms**			**516 ms**		

Step 5: Now we can set pending time to each module as below.

	I-7033	I-7041	I-7024
Pending Time	—	**2000**	**800**

	I-87024	I-87017	I-87058
Pending Time	**800**	—	**2000**



Controller Setting

Port Type

○ Modbus TCP Port Address 502

TCP/IP Address 192.168.255.1

○ Modbus RTU ○ ISaGRAF ○ M-7K

Address 1

Timeout 200

Msg Delay 0

□ Word Swap

COM Port Setting

COM 1

Baud Rate 9600

Parity None

Data Bits 8 (RTU)

Stop Bits 1

○ Request Tag Number Coil : 122 Register 122

□ Simulate I/O (does not access the RS-485/Modnus device)

☑ Pending Time 800 mSec

Step 7: Discover the average scan time of COM1 and COM2. We can find it is at our target. The average scan time of COM1 is 62 ms. The average scan time of COM2 is 31 ms.

	COM 1			**COM2**		
Module	I-87024	I-87017	I-87058	I-7033	I-7041	I-7024
Avg Scan Time	**62 ms**			**31 ms**		

3.2 VB5 Client Demo Program

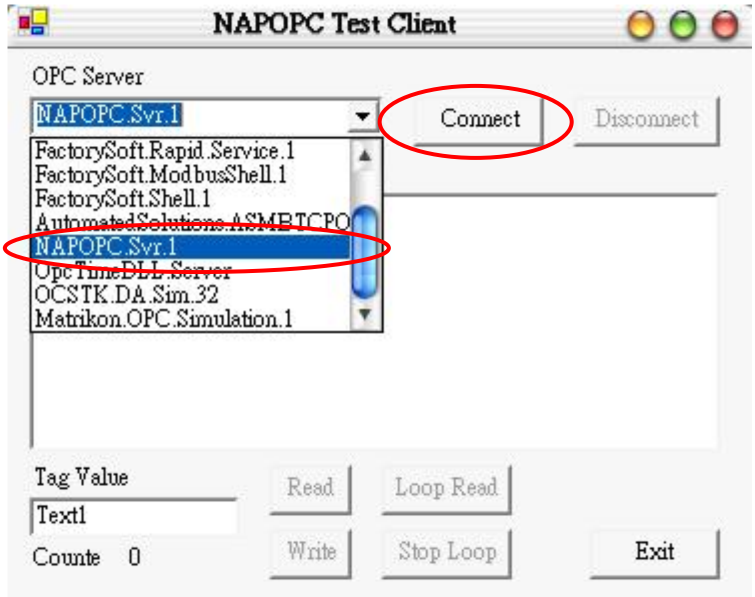
We provide three OPC client demo programs for Visual Basic 5.0, Visual Basic .Net and Visual C# .Net. It is placed under the "\\ICPDAS \NAPOPC_XPE \Client\" folder after installation of our NAPOPC_XPE server.

Note: The .Net demo programs could compatibility for Visual Studio .Net 2003 or later.

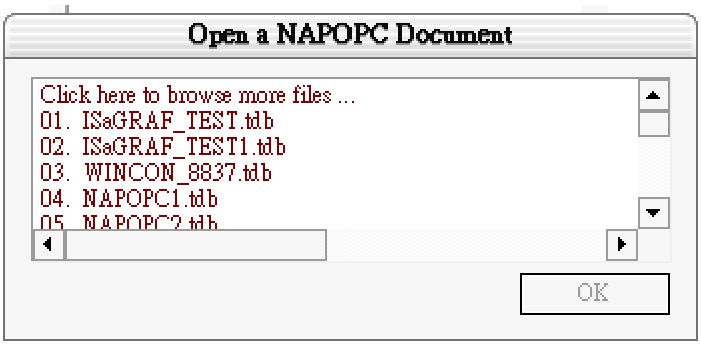
Step 1: Launch the client demo program.
(The client program will search the system registry to find OPC servers. The new servers will be added to the list.)

Step 2: Select the "NAPOPC.Svr.1" OPC Server.

Step 3: Click on the "Connect" button.



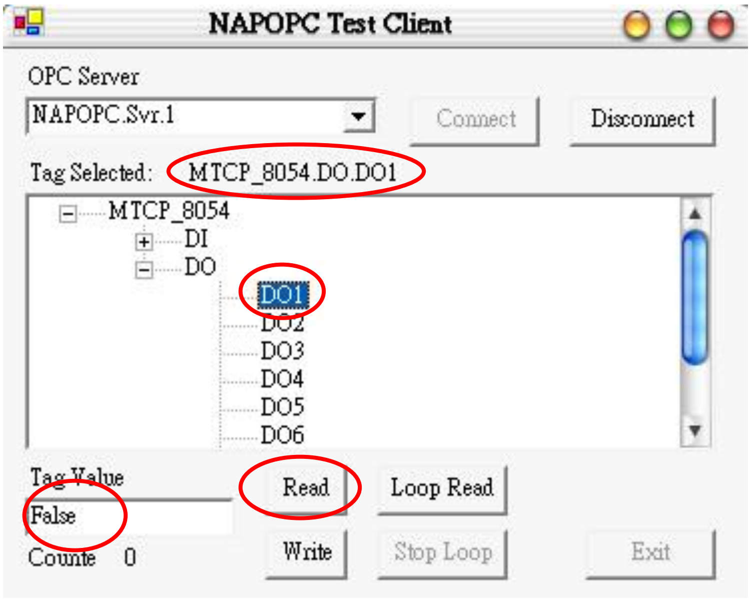
Step 4: Select a file which you want to use and click on the "OK" button.



The following steps 5 ~ 6 are read operation of Modbus TCP controllers.

Step 5: Select a tag matching on your configuration.
(For example: Select the "i-8437" controller, "DO" group and "DO1" tag in the tree-view window.)

Step 6: Click on the "Read" button to read the ""MTCP_8054.DO.DO1" value.



OPC Server: NAPOPC.Svr.1

Tag Selected: MTCP_8054.DO.DO1

Tag Value: False

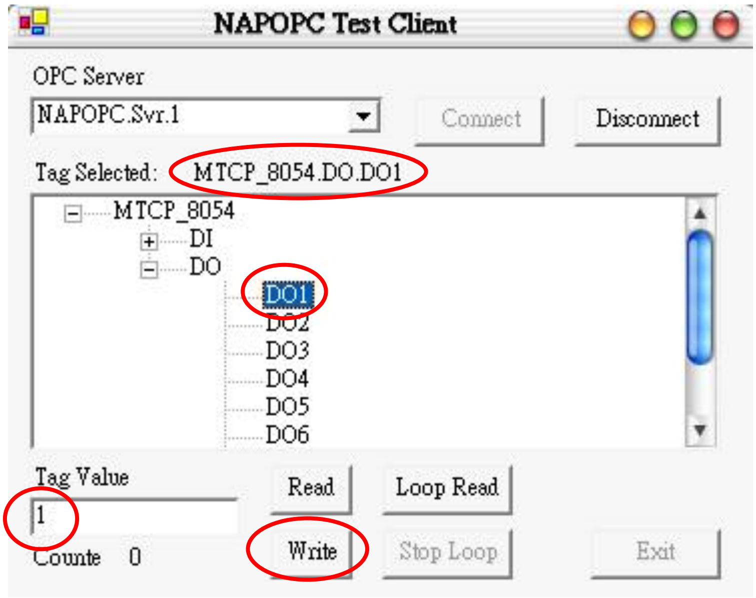
Counte 0

The following steps 7 ~ 9 are write operation of Modbus TCP controllers.

Step 7: Select a tag matching on your configuration.
(For example: Select the "i-8437" controller, "DO" group and "DO1" tag in the tree-view window.)

Step 8: Fill in the "Tag-Value" field with 1.

Step 9: Click the "Write" button to write the "MTCP_8054.DO.DO1" value.



OPC Server: NAPOPC.Svr.1

Tag Selected: MTCP_8054.DO.DO1

Tag Value: 1

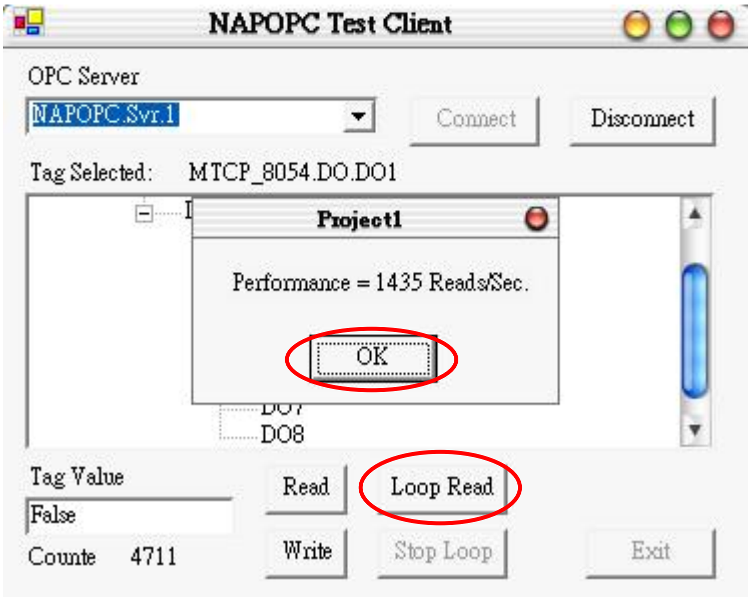
Counte 0

The following steps 10 ~ 12 are read operation, which loops unless you stop it.

Step 10: Click on the "Loop Read" to continuously read data.

Step 11: After about 5 seconds (or more), click on the "Stop Loop" button to stop reading.

Step 12: A window pops up to show the performance.
Close it by clicking the "OK" button.



OPC Server: NAPOPC.Svr.1

Tag Selected: MTCP_8054.DO.DO1

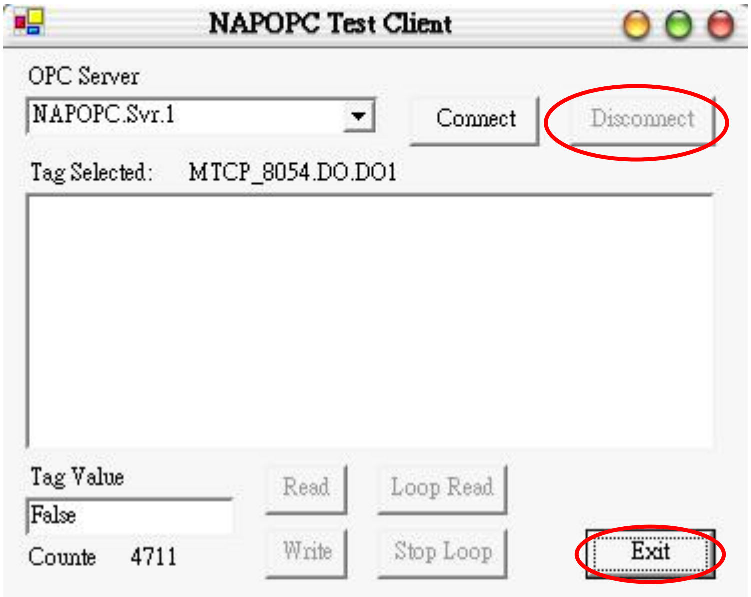
Performance = 1435 Reads/Sec.

Tag Value: False

Counte 4711

Step 13: Click on the "Disconnect" button to disconnect from the OPC server.

Step 14: Click on the "Exit" button to end the client demo program.



OPC Server: NAPOPC.Svr.1

Tag Selected: MTCP_8054.DO.DO1

Tag Value: False

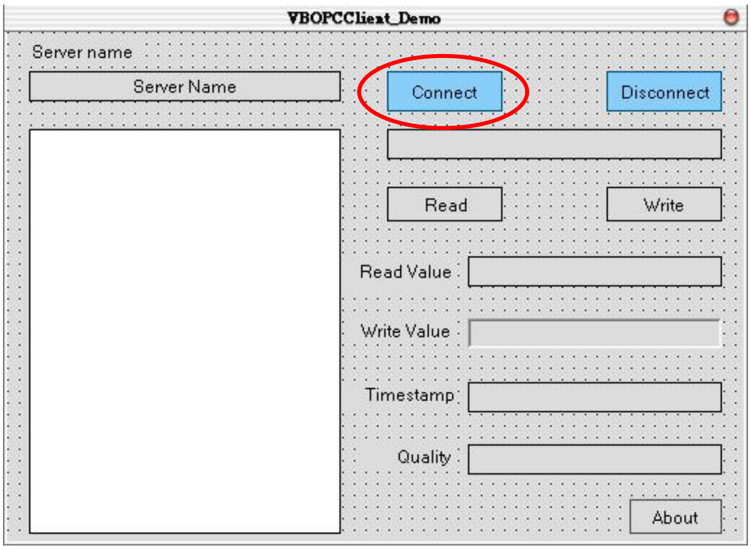
Counte 4711

3.3 .Net Client Demo Program

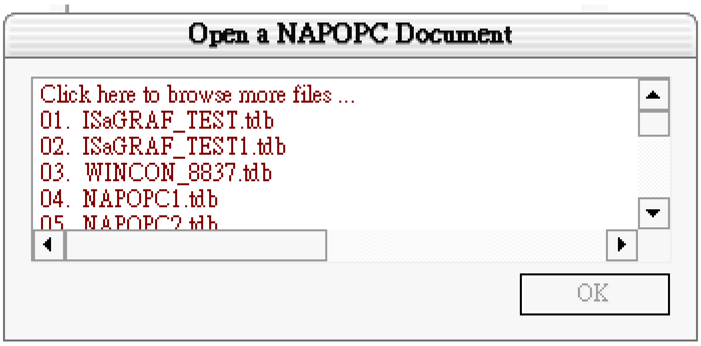
We provide another two OPC client demo programs for Visual Basic .Net 2003 and Visual C# .Net 2003. It's placed under the following folder "\\ICPDAS\NAPOPC_XPE\Client\OPC_NetClientDemo\VBOPCClient_Demo" and "\\ICPDAS\NAPOPC_XPE \Client\OPC_NetClientDemo\VCSOPCClient_Demo"after installation of our NAPOPC_XPE server.

Step 1: Launch the client demo program "VBOPCClient_Demo.exe" or "VCSOPCClient_Demo.exe". (The client program set "NAPOPC.Svr.1" as default OPC Server)

Step 2: Click on the "Connect" button.



Step 3: Select a file which you want to use and click on the "OK" button.

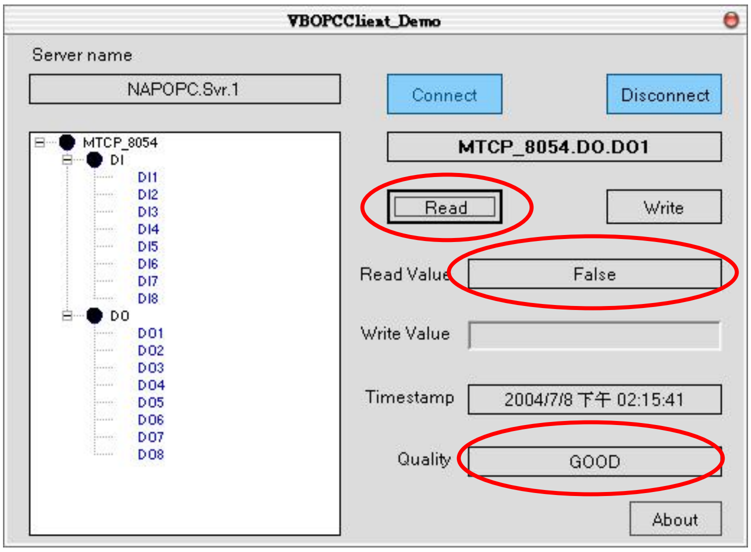

Open a NAPOPC Document

Click here to browse more files ...
01. ISaGRAF_TEST.tdb
02. ISaGRAF_TEST1.tdb
03. WINCON_8837.tdb
04. NAPOPC1.tdb
05. NAPOPC2.tdb

The following steps 4 ~ 5 are read operation of Modbus TCP controllers.

Step 4: Select a tag matching on your configuration. (For example: Select the "i-8437" controller, "DO" group and "DO1" tag in the tree-view window.)

Step 5: Click on the "Read" button to read the "MTCP_8054.DO.DO1" value.



VBOPCClient_Demo

Server name: NAPOPC.Svr.1

MTCP_8054.DO.DO1

Read Value: False

Write Value:

Timestamp: 2004/7/8 下午 02:15:41

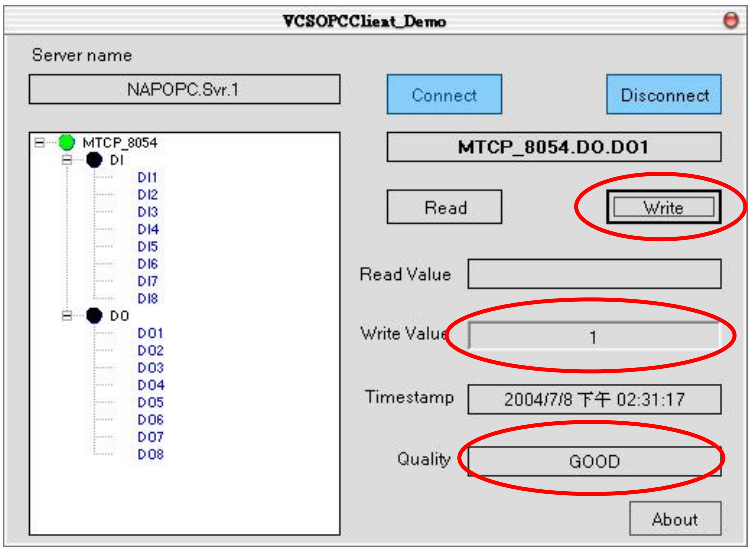
Quality: GOOD

The following steps 6 ~ 8 are write operation of Modbus TCP controllers.

Step 6: Select a tag matching on your configuration.
(For example: Select the "i-8437" controller, "DO" group and "DO1" tag in the tree-view window.)

Step 7: Fill in the "Write Value" field with 1.

Step 8: Click the "Write" button to write the "MTCP_8054.DO.DO1" value.



VCSOPCClient_Demo

Server name: NAPOPC.Svr.1

MTCP_8054.DO.DO1

Read Value:

Write Value: 1

Timestamp: 2004/7/8 下午 02:31:17

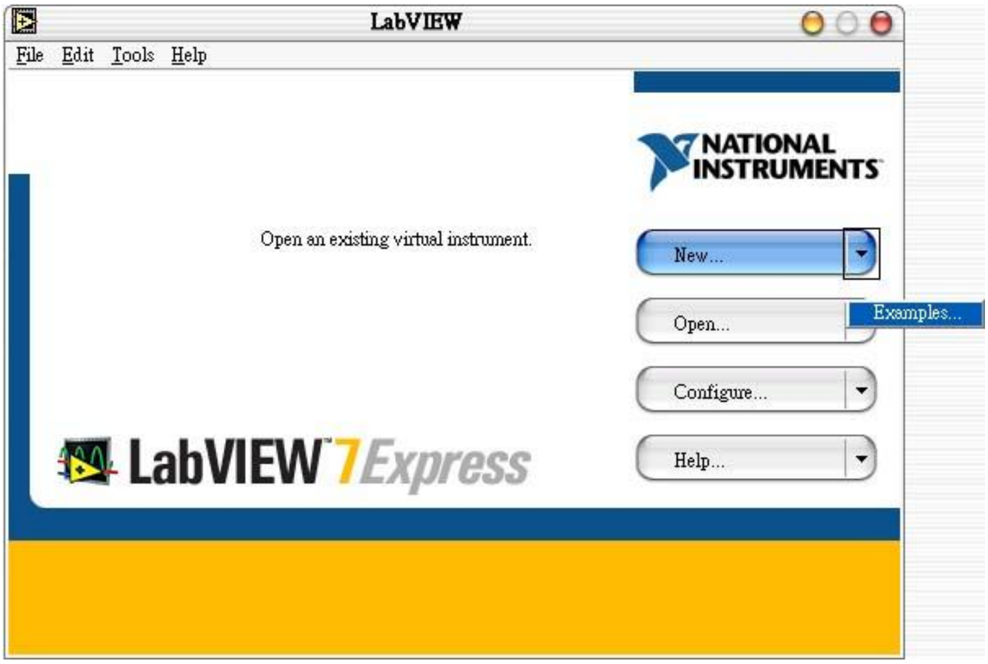
Quality: GOOD

Step 13: Click on the "Disconnect" button to disconnect from the OPC server.

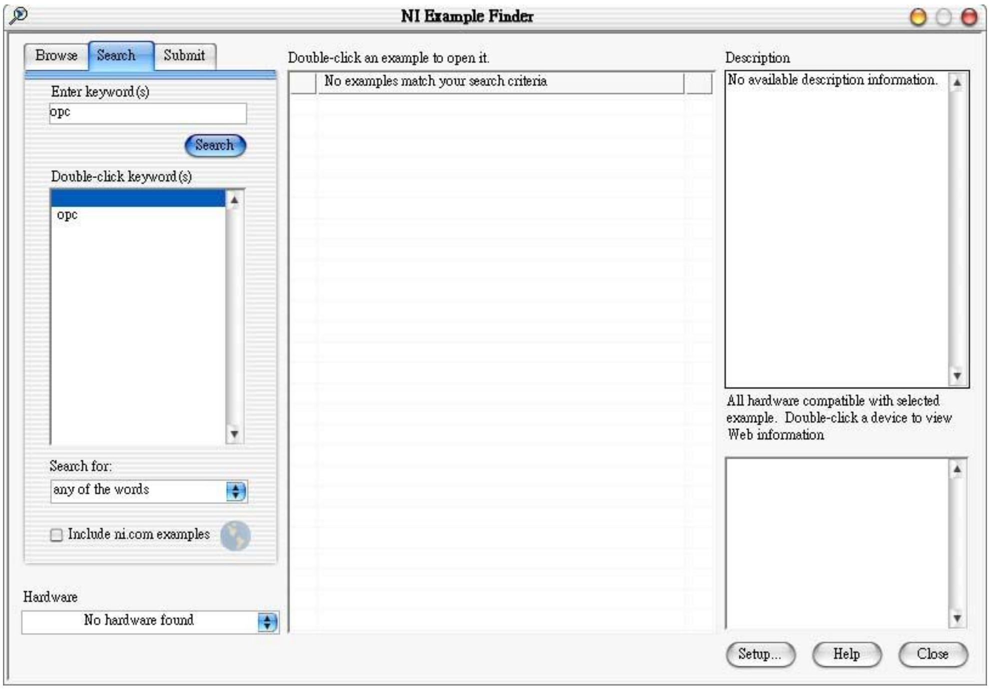
3.4 LabVIEW

Note:The following steps are on PC not on XPAC. If you have LabVIEW SCADA based on Windows XP Embedded, you can refer to the following steps to try it.

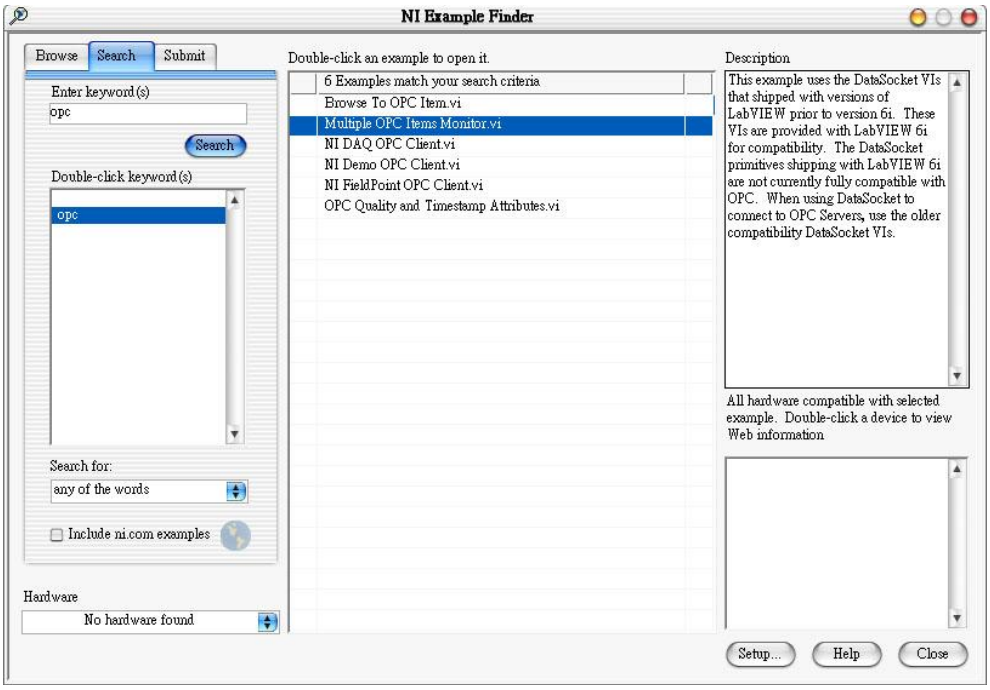
Step 1: Run the LabVIEW program and select "Open…" -> Example



Step 2: Click on the "Search Examples" button to get information on using OPC.


NI Example Finder
Browse
Search
Submit
Enter keyword(s)
opc
Search
Double-click keyword(s)
opc
Search for:
any of the words
□ Include ni.com examples
Hardware
No hardware found
Double-click an example to open it.
No examples match your search criteria
Description
No available description information.
All hardware compatible with selected example. Double-click a device to view Web information
Setup...
Help
Close

Step 3: Double-click on the "Multiple OPC Items Monitor.vi" item in the middle window of NI Example Finder dialog.

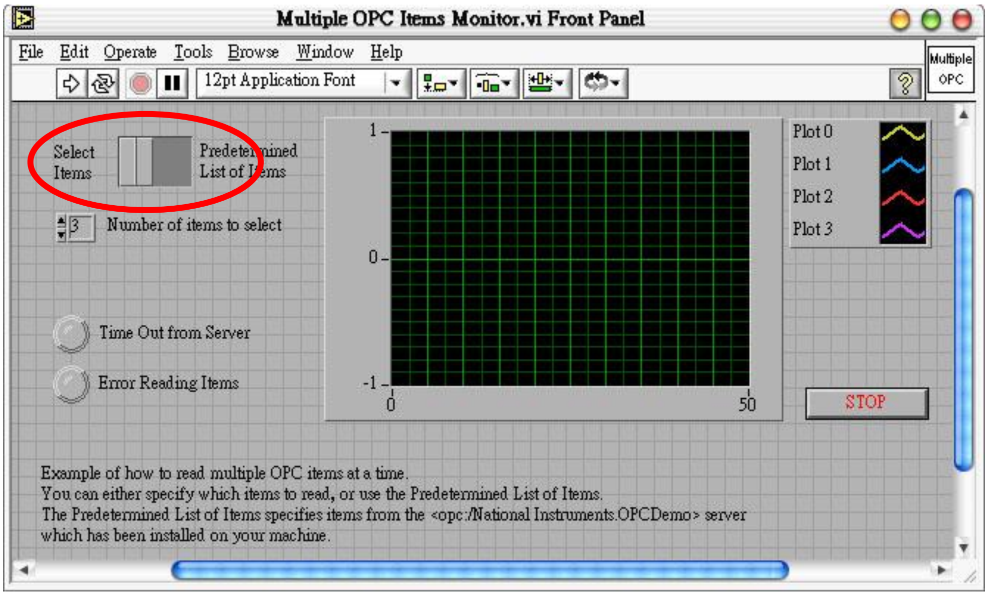


6 Examples match your search criteria
Browse To OPC Item.vi
Multiple OPC Items Monitor.vi
NI DAQ OPC Client.vi
NI Demo OPC Client.vi
NI FieldPoint OPC Client.vi
OPC Quality and Timestamp Attributes.vi

This example uses the DataSocket VIs that shipped with versions of LabVIEW prior to version 6i. These VIs are provided with LabVIEW 6i for compatibility. The DataSocket primitives shipping with LabVIEW 6i are not currently fully compatible with OPC. When using DataSocket to connect to OPC Servers, use the older compatibility DataSocket VIs.

All hardware compatible with selected example. Double-click a device to view Web information

Step 4: Click on the "Select Items" item in the "Multiple OPC Items Monitor.vi" demo.



Example of how to read multiple OPC items at a time.
You can either specify which items to read, or use the Predetermined List of Items.
The Predetermined List of Items specifies items from the <opc:/National Instruments.OPCDemo> server which has been installed on your machine.

Step 5: Run this demo.

Step 6: Click on a machine name in the "Network" tree-view.

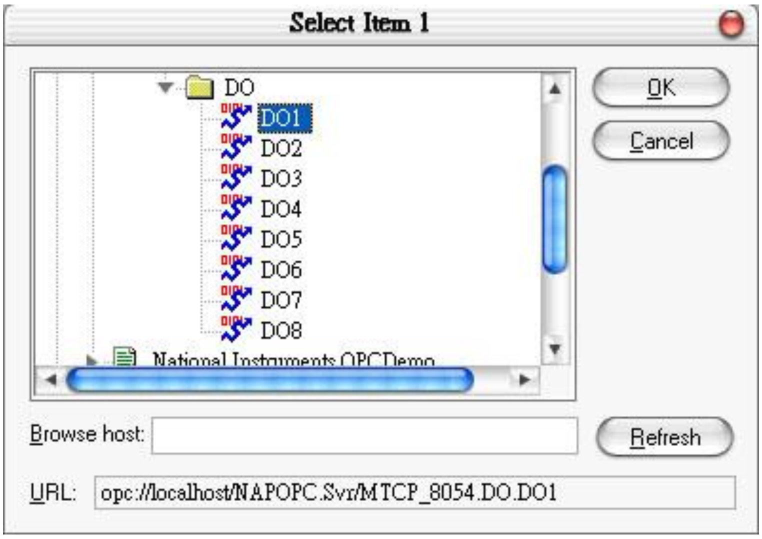
Step 7: Select the "NAPOPC.Svr" OPC server.

Step 8: Click on the "OK" button to close it.

Step 9: Select a file which you want to use and click on the "OK" button.

Step 10: Select an item (tag) in the tree-view.

Step 11: Click on the "OK" button to add this one.



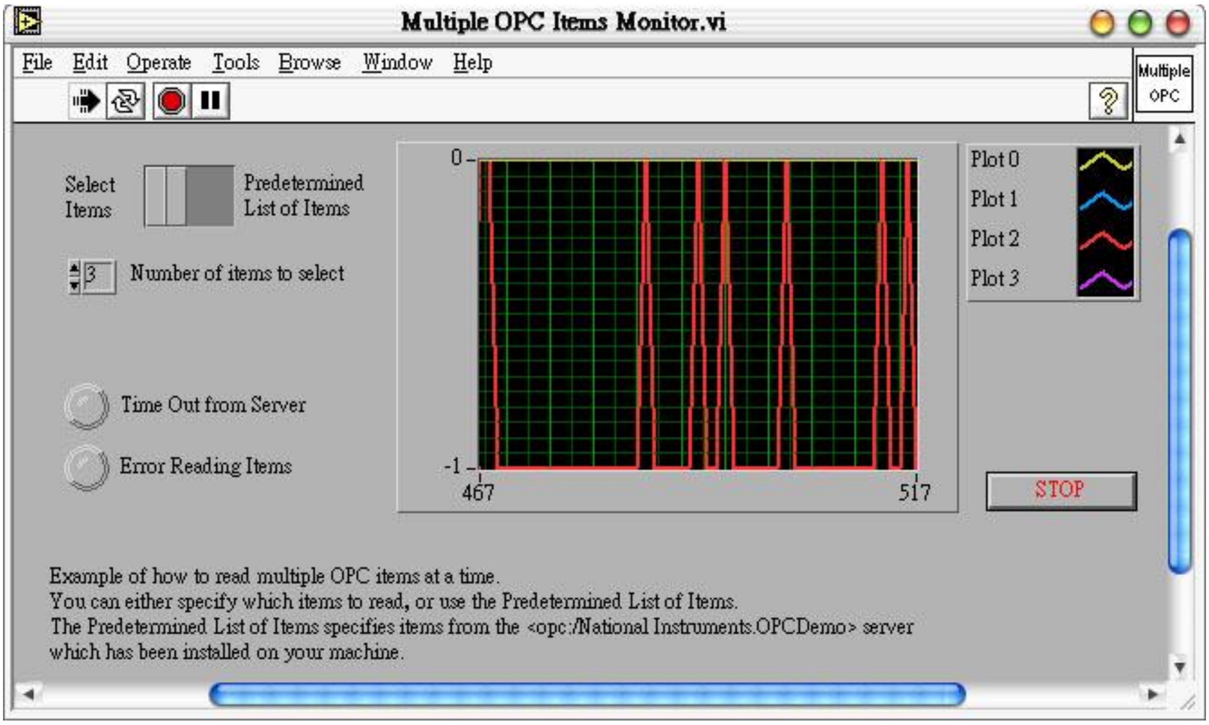
Browse host:

URL: opc://localhost/NAPOPC.Svr/MTCP_8054.DO.DO1

Step 12: Repeat the steps 6 ~ 11 to add more items(tags).

Step 13: Click on the "Cancel" button to finish adding items(tags).

Step 14: The grid window graphs a line(s) to show changes of each item (tag).

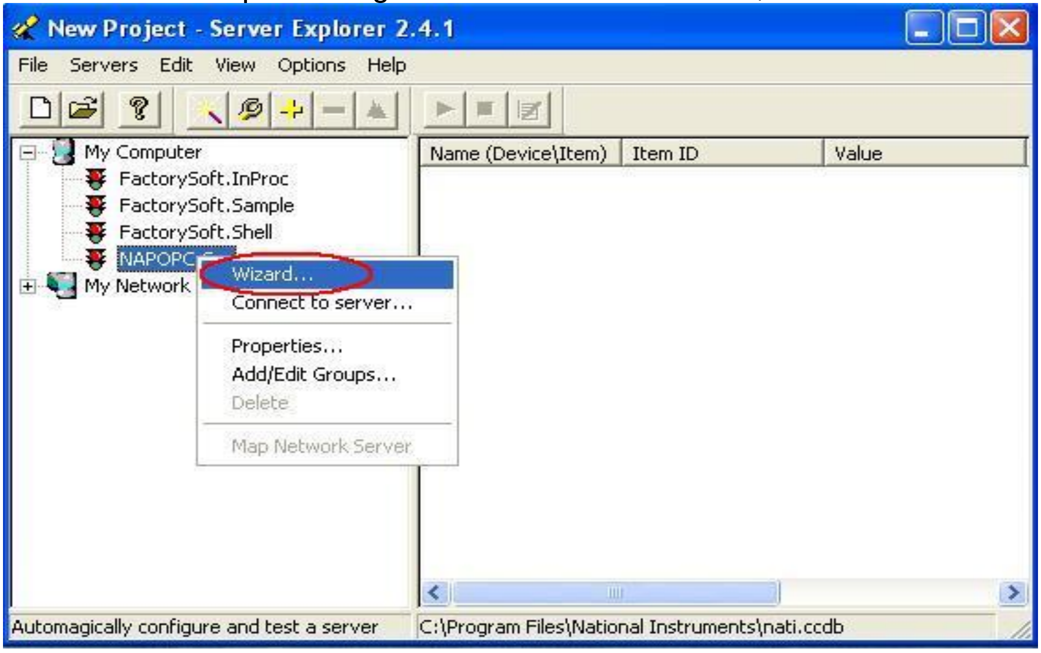

Multiple OPC Items Monitor.vi
File Edit Operate Tools Browse Window Help
Multiple OPC
Select Items
Predetermined List of Items
3
Number of items to select
Time Out from Server
Error Reading Items
0
-1
467
517
Plot 0
Plot 1
Plot 2
Plot 3
STOP

Example of how to read multiple OPC items at a time.
You can either specify which items to read, or use the Predetermined List of Items.
The Predetermined List of Items specifies items from the <opc:/National Instruments.OPCDemo> server which has been installed on your machine.

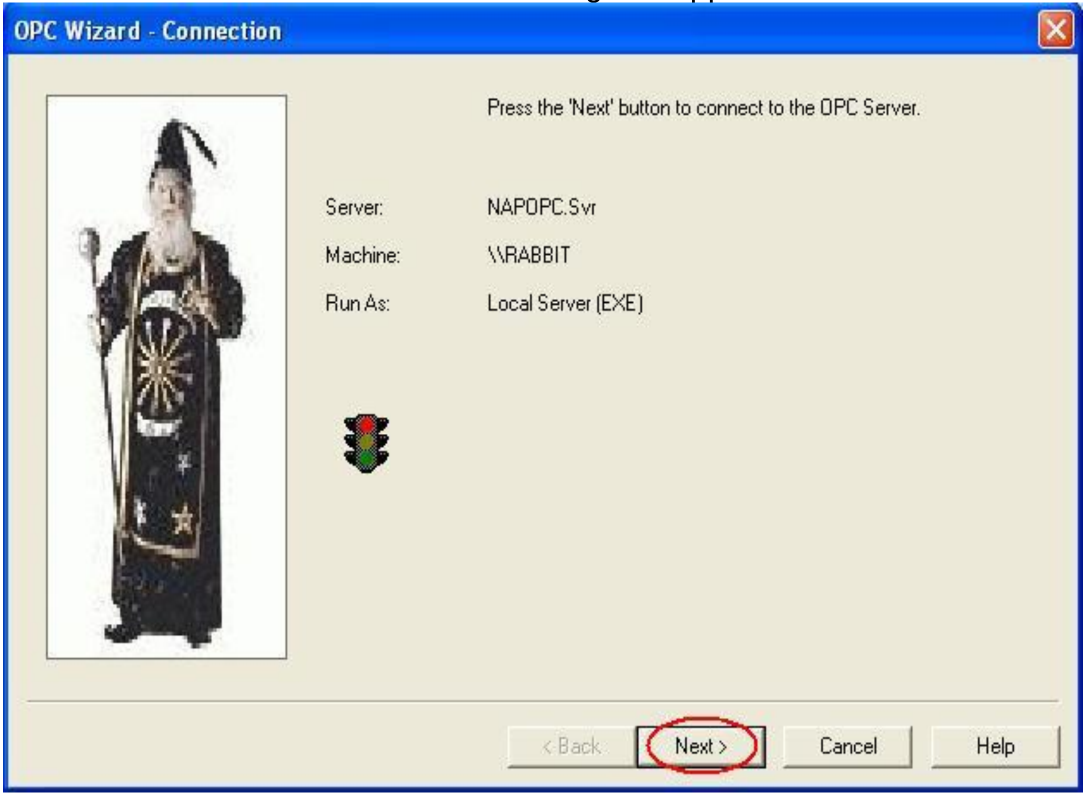
3.5 NATIONAL INSTRUMENTS

National Instruments is a comprehensive industrial automation company by providing the software, hardware, and technologies necessary to transform personal computers into powerful computer-based and networked measurement and automation systems. The ServerExplorer is one of their products for connecting to OPC Server. For more information, please visit http://www.ni.com

Step 1: Start ServerExplorer. Right-click on “NAPOPC.Svr”, then select “Wizard”.



Step 2: The **OPC Wizard - Connection** dialog box appears. Then click on “Next>”



Press the 'Next' button to connect to the OPC Server.

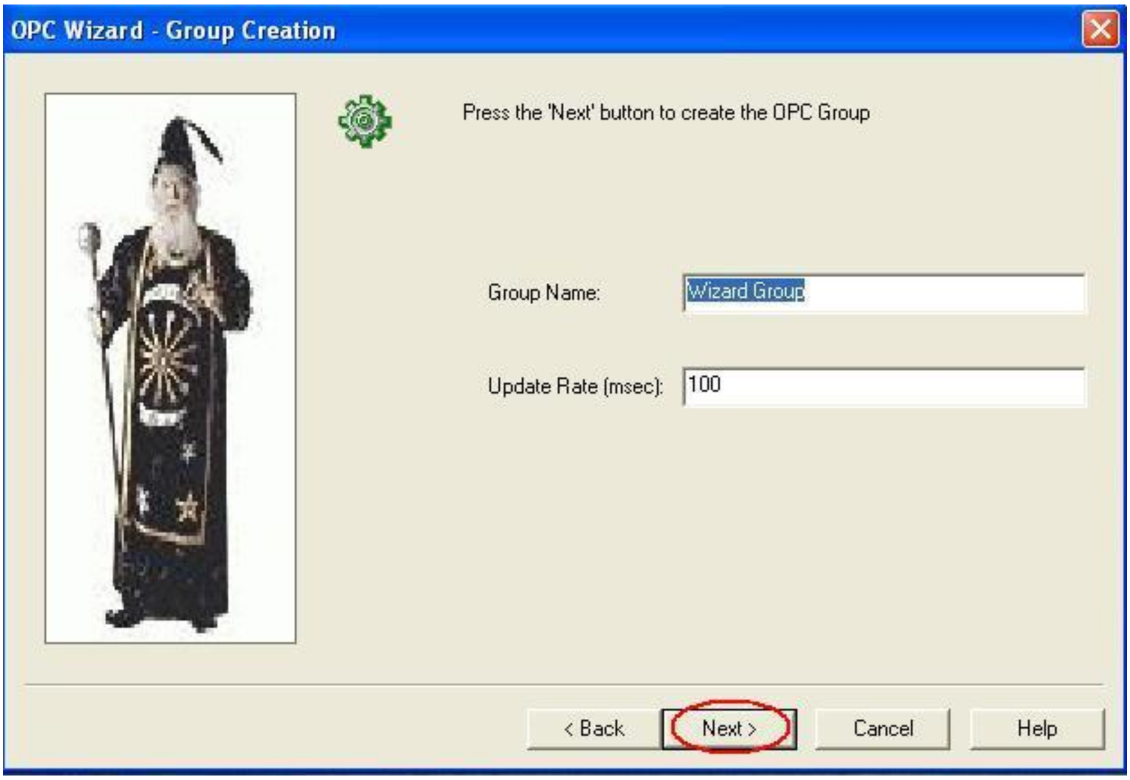
Server: NAPOPC.Svr

Machine: \\RABBIT

Run As: Local Server (EXE)

Step 3: Select a file which you want to use and click on the "OK" button.

Step 4: Enter the **Group Name** and **Update Rate**. The name can be any name you want. Click “Next>” to continue.



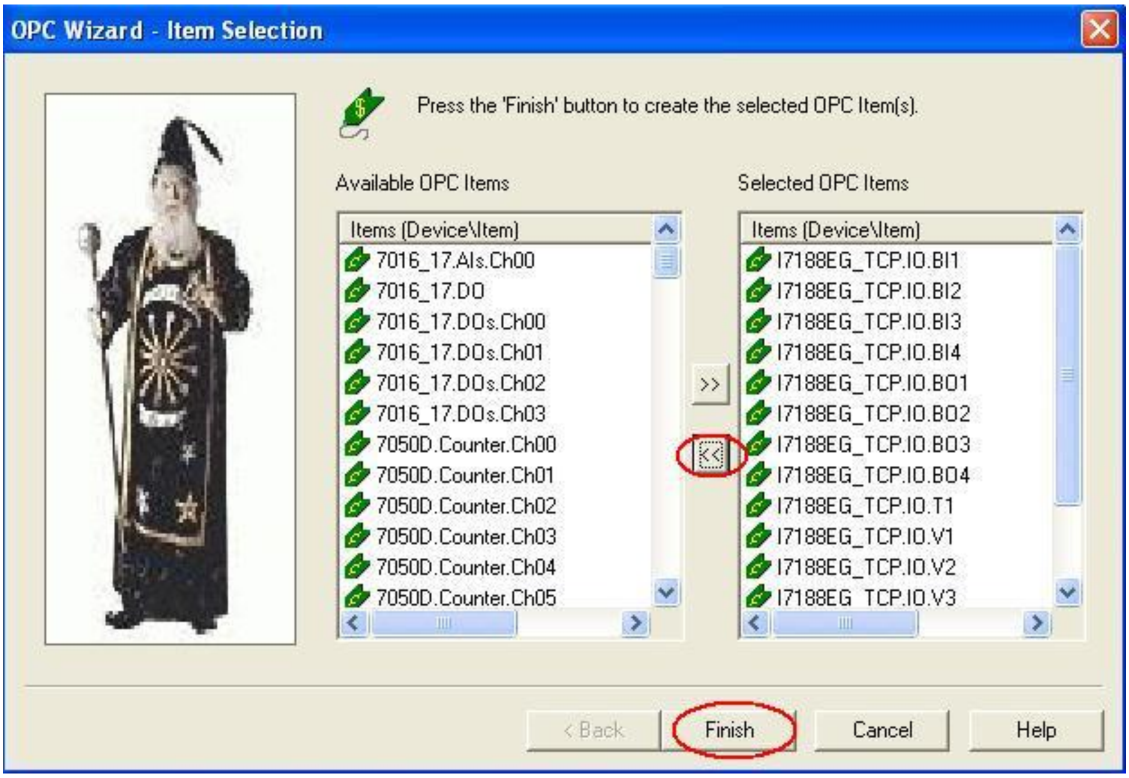
OPC Wizard - Group Creation

Press the 'Next' button to create the OPC Group

Group Name: Wizard Group

Update Rate (msec): 100

Step 5: Select all the items that you want to view from the **Available OPC Items** list. Then click "Finish".



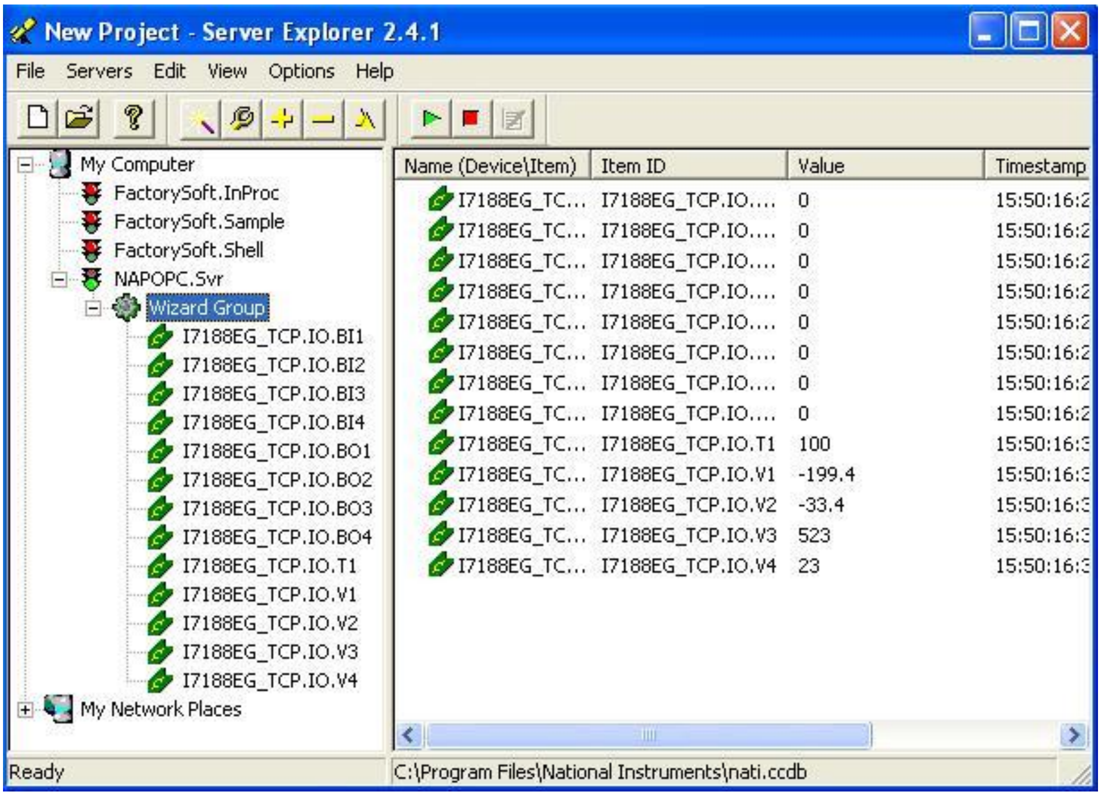
OPC Wizard - Item Selection

Press the 'Finish' button to create the selected OPC Item(s).

Available OPC Items
Items (Device\Item)
7016_17.AIs.Ch00
7016_17.DO
7016_17.DOs.Ch00
7016_17.DOs.Ch01
7016_17.DOs.Ch02
7016_17.DOs.Ch03
7050D.Counter.Ch00
7050D.Counter.Ch01
7050D.Counter.Ch02
7050D.Counter.Ch03
7050D.Counter.Ch04
7050D.Counter.Ch05

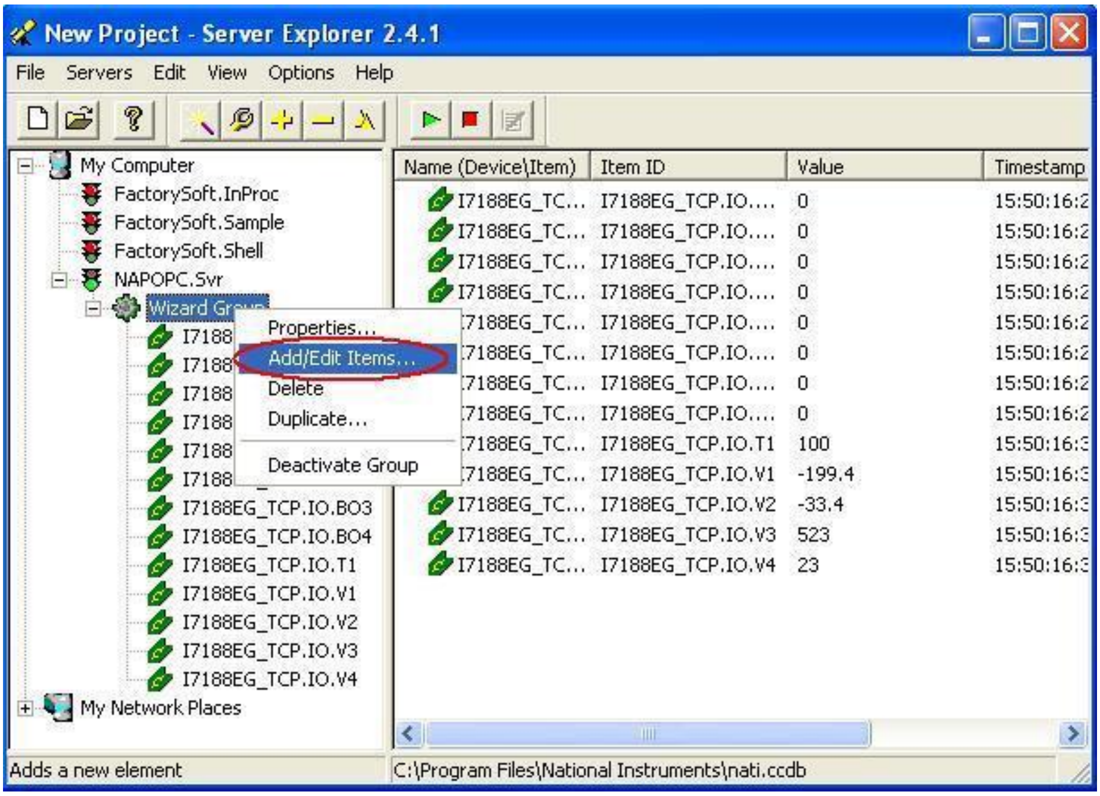
Selected OPC Items
Items (Device\Item)
I7188EG_TCP.IO.BI1
I7188EG_TCP.IO.BI2
I7188EG_TCP.IO.BI3
I7188EG_TCP.IO.BI4
I7188EG_TCP.IO.BO1
I7188EG_TCP.IO.BO2
I7188EG_TCP.IO.BO3
I7188EG_TCP.IO.BO4
I7188EG_TCP.IO.T1
I7188EG_TCP.IO.V1
I7188EG_TCP.IO.V2
I7188EG_TCP.IO.V3

Step 6: Now you should be able to view the OPC connection that you just created.



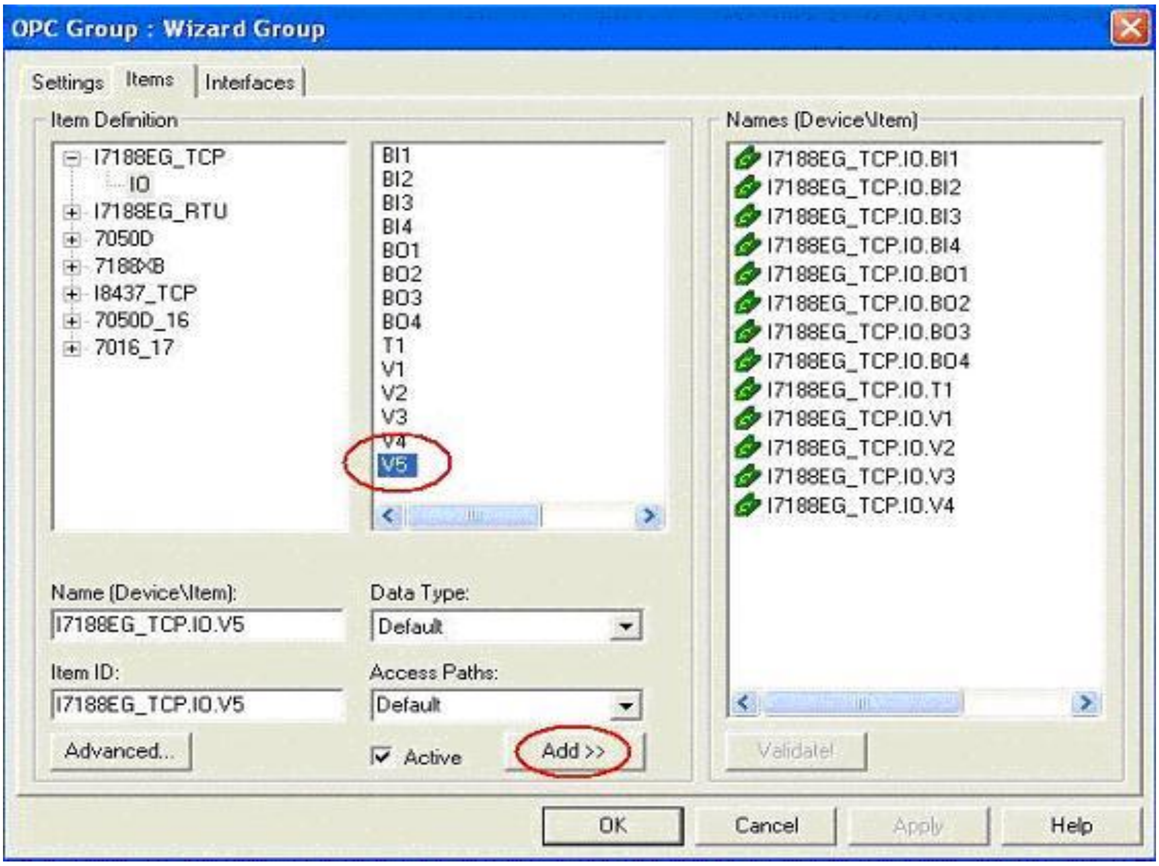
Name (Device\Item)	Item ID	Value	Timestamp
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO.T1	100	15:50:16:3
I7188EG_TC...	I7188EG_TCP.IO.V1	-199.4	15:50:16:3
I7188EG_TC...	I7188EG_TCP.IO.V2	-33.4	15:50:16:3
I7188EG_TC...	I7188EG_TCP.IO.V3	523	15:50:16:3
I7188EG_TC...	I7188EG_TCP.IO.V4	23	15:50:16:3

Step 7: To add new items, right-click on the group name then select "Add/Edit Items".



Name (Device\Item)	Item ID	Value	Timestamp
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
I7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
7188EG_TC...	I7188EG_TCP.IO....	0	15:50:16:2
7188EG_TC...	I7188EG_TCP.IO.T1	100	15:50:16:3
7188EG_TC...	I7188EG_TCP.IO.V1	-199.4	15:50:16:3
I7188EG_TC...	I7188EG_TCP.IO.V2	-33.4	15:50:16:3
I7188EG_TC...	I7188EG_TCP.IO.V3	523	15:50:16:3
I7188EG_TC...	I7188EG_TCP.IO.V4	23	15:50:16:3

Step 8: Make sure the **Item ID** textbox at the bottom has the correct object and item name. Then click "Add>>" to add the item to the list on the right. In this example, we add the "I7188EG_TCP.IO.V5". Click "OK" when you are done.



- I7188EG_TCP
  - IO
- I7188EG_RTU
- 7050D
- 7188XB
- I8437_TCP
- 7050D_16
- 7016_17

BI1
BI2
BI3
BI4
BO1
BO2
BO3
BO4
T1
V1
V2
V3
V4
V5

Name (Device\Item): I7188EG_TCP.IO.V5

Item ID: I7188EG_TCP.IO.V5

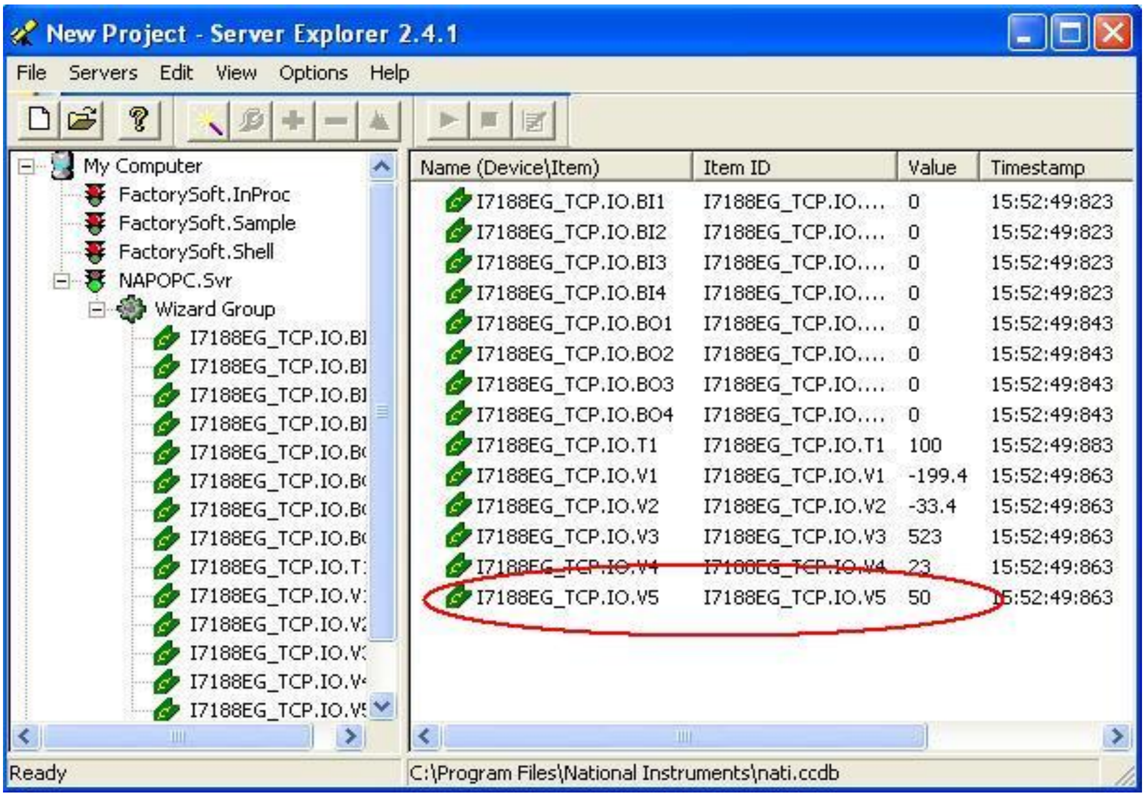
Data Type: Default

Access Paths: Default

☑ Active

Names (Device\Item):
I7188EG_TCP.IO.BI1
I7188EG_TCP.IO.BI2
I7188EG_TCP.IO.BI3
I7188EG_TCP.IO.BI4
I7188EG_TCP.IO.BO1
I7188EG_TCP.IO.BO2
I7188EG_TCP.IO.BO3
I7188EG_TCP.IO.BO4
I7188EG_TCP.IO.T1
I7188EG_TCP.IO.V1
I7188EG_TCP.IO.V2
I7188EG_TCP.IO.V3
I7188EG_TCP.IO.V4

Step 9: Now you should be able to read all the items that you added in the main window.



- My Computer
  - FactorySoft.InProc
  - FactorySoft.Sample
  - FactorySoft.Shell
  - NAPOPC.Svr
    - Wizard Group
      - I7188EG_TCP.IO.BI
      - I7188EG_TCP.IO.BI
      - I7188EG_TCP.IO.BI
      - I7188EG_TCP.IO.BI
      - I7188EG_TCP.IO.B(
      - I7188EG_TCP.IO.B(
      - I7188EG_TCP.IO.B(
      - I7188EG_TCP.IO.B(
      - I7188EG_TCP.IO.T:
      - I7188EG_TCP.IO.V:
      - I7188EG_TCP.IO.V:
      - I7188EG_TCP.IO.V:
      - I7188EG_TCP.IO.V·
      - I7188EG_TCP.IO.V!

Name (Device\Item)	Item ID	Value	Timestamp
I7188EG_TCP.IO.BI1	I7188EG_TCP.IO....	0	15:52:49:823
I7188EG_TCP.IO.BI2	I7188EG_TCP.IO....	0	15:52:49:823
I7188EG_TCP.IO.BI3	I7188EG_TCP.IO....	0	15:52:49:823
I7188EG_TCP.IO.BI4	I7188EG_TCP.IO....	0	15:52:49:823
I7188EG_TCP.IO.BO1	I7188EG_TCP.IO....	0	15:52:49:843
I7188EG_TCP.IO.BO2	I7188EG_TCP.IO....	0	15:52:49:843
I7188EG_TCP.IO.BO3	I7188EG_TCP.IO....	0	15:52:49:843
I7188EG_TCP.IO.BO4	I7188EG_TCP.IO....	0	15:52:49:843
I7188EG_TCP.IO.T1	I7188EG_TCP.IO.T1	100	15:52:49:883
I7188EG_TCP.IO.V1	I7188EG_TCP.IO.V1	-199.4	15:52:49:863
I7188EG_TCP.IO.V2	I7188EG_TCP.IO.V2	-33.4	15:52:49:863
I7188EG_TCP.IO.V3	I7188EG_TCP.IO.V3	523	15:52:49:863
I7188EG_TCP.IO.V4	I7188EG_TCP.IO.V4	23	15:52:49:863
I7188EG_TCP.IO.V5	I7188EG_TCP.IO.V5	50	15:52:49:863

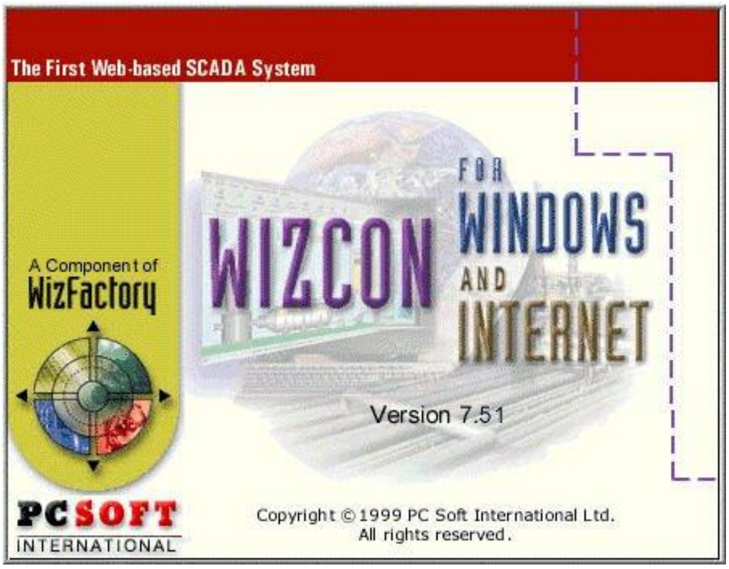
C:\Program Files\National Instruments\nati.ccdb

3.6 WIZCON

Wizcon for Windows and Internet is a powerful HMI/SCADA software package that delivers real-time and historical information from the plant floor to the boardroom and beyond. For more information, please visit http://www.emation.com

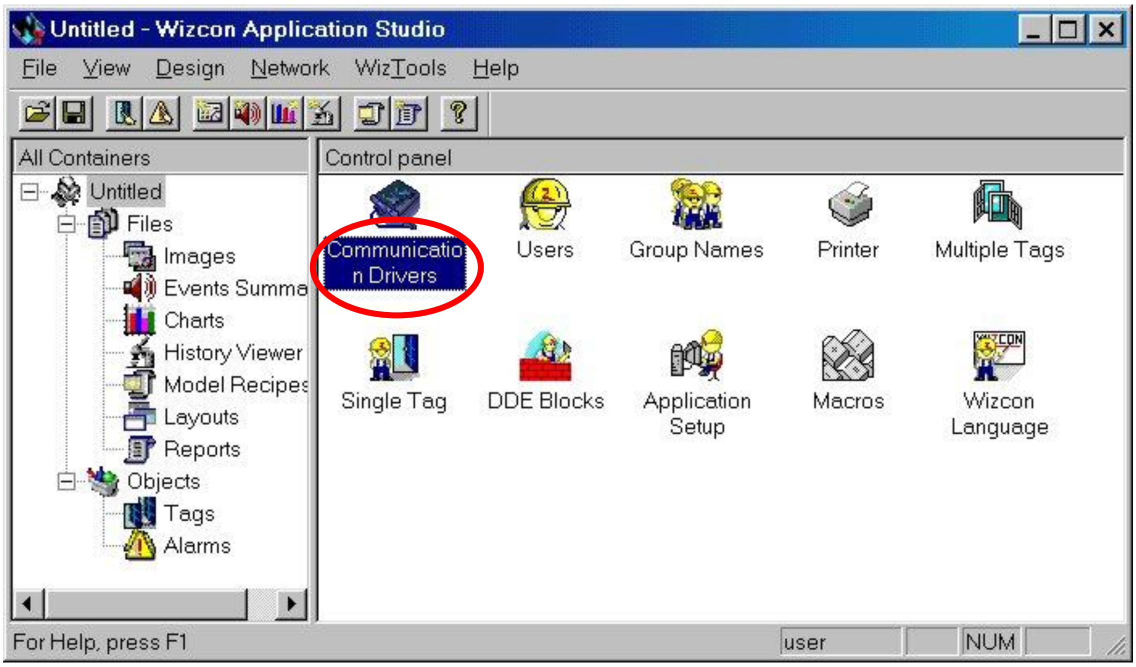
Note:The following steps are on PC not on XPAC. If you have WIZCON SCADA based on Windows XP Embedded, you can refer to the following steps to try it.

Step 1: Run WIZCON (Version 7.51 or newer) program.

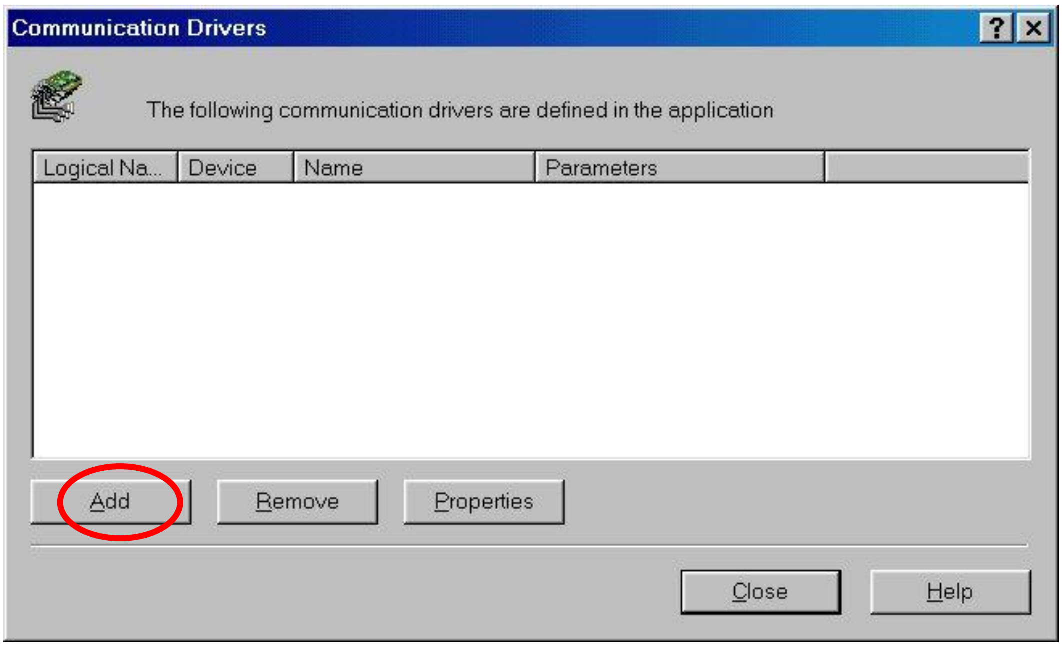


Step 2: Create a new project.

Step 3: Click on the "Communication Drivers" icon in the right hand window.

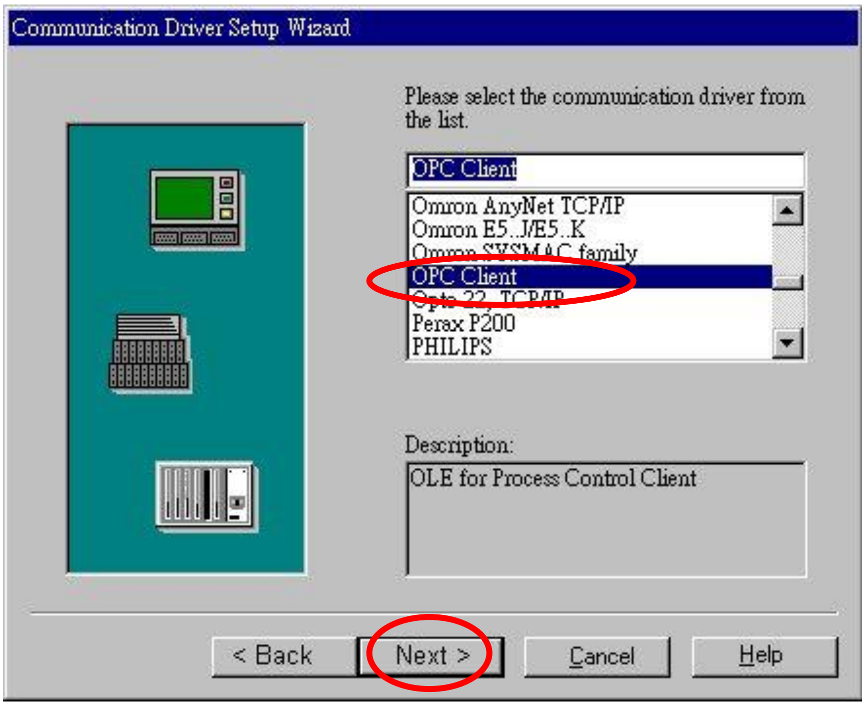


Step 4: Click on the "Add" button to add new drivers.



Step 5: Select the "OPC Client" item.

Step 6: Click on the "Next >" button.

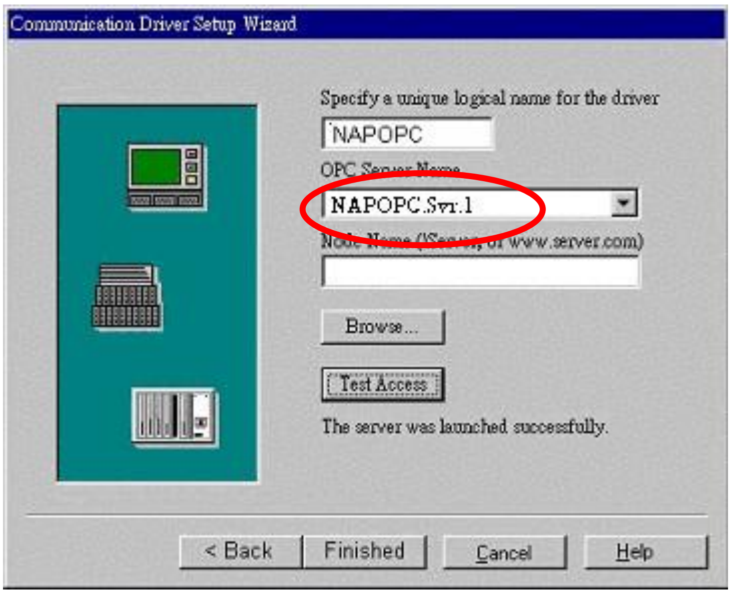


Step 7: Enter the driver name (for example: "NAPOPC_XPE").

Step 8: Select the OPC Server Name as "NAPOPC.Svr.1".

Step 9: Click on the "Test Access" button to see if the OPC server can be accessed.

Step 10: Click on the "Finished" button.



Communication Driver Setup Wizard

Specify a unique logical name for the driver

NAPOPC

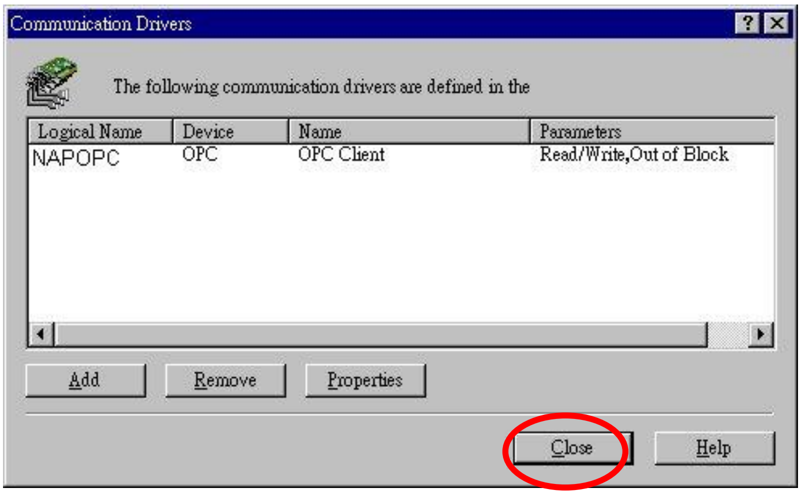
OPC Server Name

NAPOPC.Svr.1

Node Name (\\Server, or www.server.com)

The server was launched successfully.

Step 11: Click on the "Close" button.

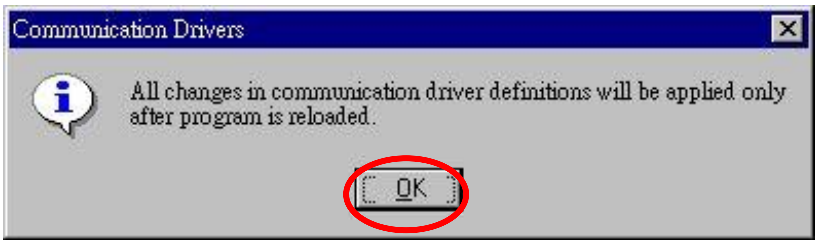


Communication Drivers

The following communication drivers are defined in the

Logical Name	Device	Name	Parameters
NAPOPC	OPC	OPC Client	Read/Write,Out of Block

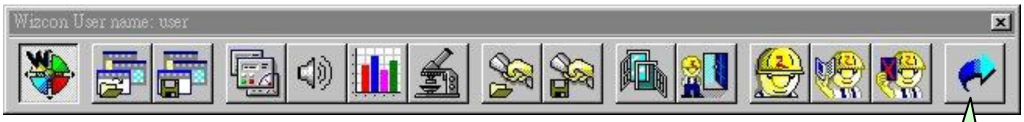
Step 12: Click on the "OK" button to close the window.



Communication Drivers

All changes in communication driver definitions will be applied only after program is reloaded.

Step 13: Click on the far-right icon (the arrow) to close the WIZCON.

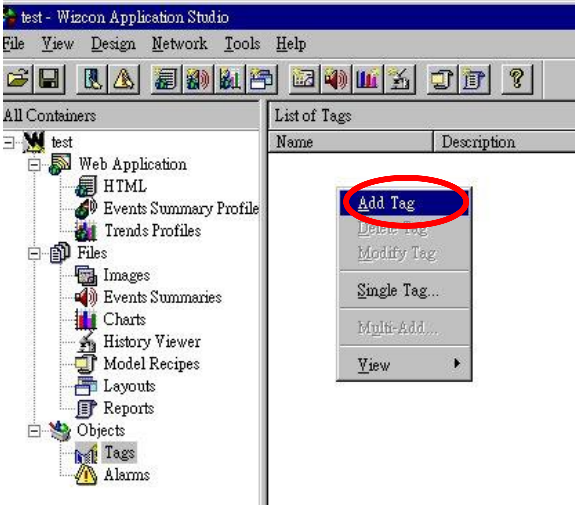


Close

Step 14: Restart the WIZCON.

Step 15: Select the "Tags" item from the left-hand window.

Step 16: Right click the mouse button and select the "Add Tag" option to add tag(s).



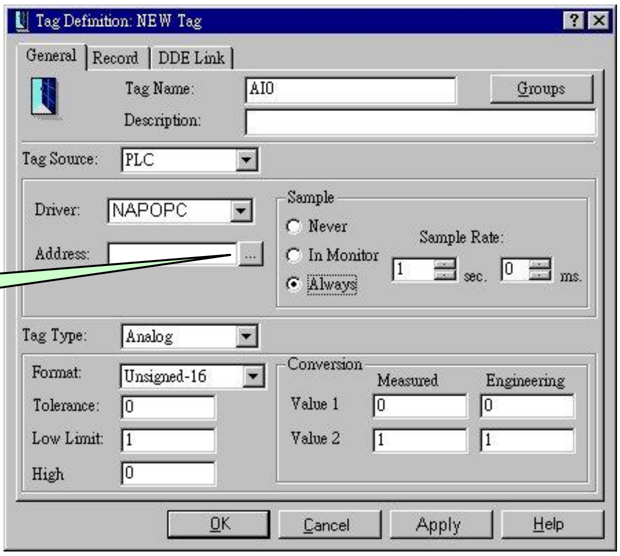
Step 17: Enter a tag name in the "Tag Name" field.

Step 18: Select "PLC" in the "Tag Source" field.

Step 19: Select "NAPOPC_XPE " in the "Driver" field.

Step 20: Select "Always" in the "Sample" field.

Step 21: Click on the "…" button to set the "Address" field.



Tag Definition: NEW Tag

General | Record | DDE Link

Tag Name: AI0 Groups

Description:

Tag Source: PLC

Driver: NAPOPC

Address: ...

Sample: ○ Never ○ In Monitor ● Always

Sample Rate: 1 sec. 0 ms.

Tag Type: Analog

Format: Unsigned-16

Tolerance: 0

Low Limit: 1

High 0

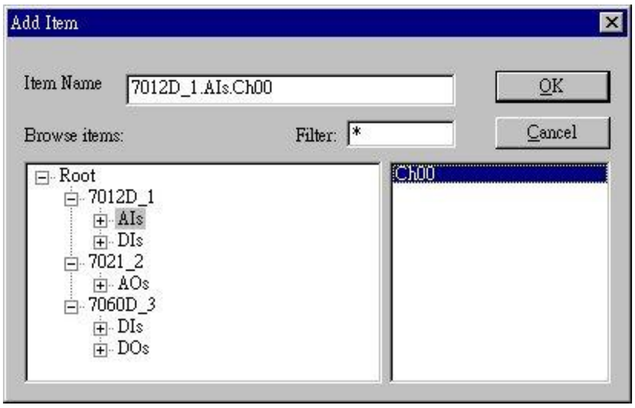
Conversion

	Measured	Engineering
Value 1	0	0
Value 2	1	1

OK Cancel Apply Help

Set Address

Step 22: Select a tag and click on the "OK" button.

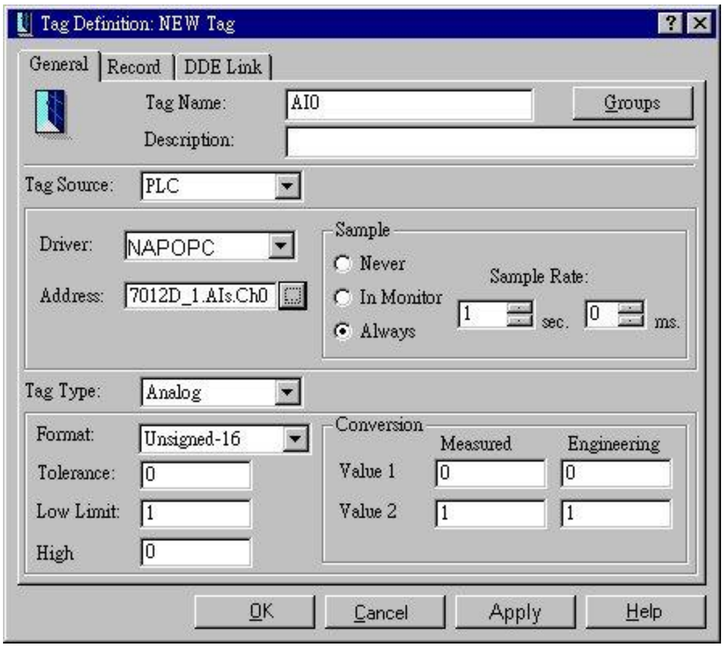


Item Name: 7012D_1.AIs.Ch00

Filter: *

Ch00

Step 23: Click on the "OK" button to close it.



Tag Name: AI0

Description:

Tag Source: PLC

Driver: NAPOPC

Address: 7012D_1.AIs.Ch0

Sample: ○ Never ○ In Monitor ● Always

Sample Rate: 1 sec. 0 ms.

Tag Type: Analog

Format: Unsigned-16

Tolerance: 0

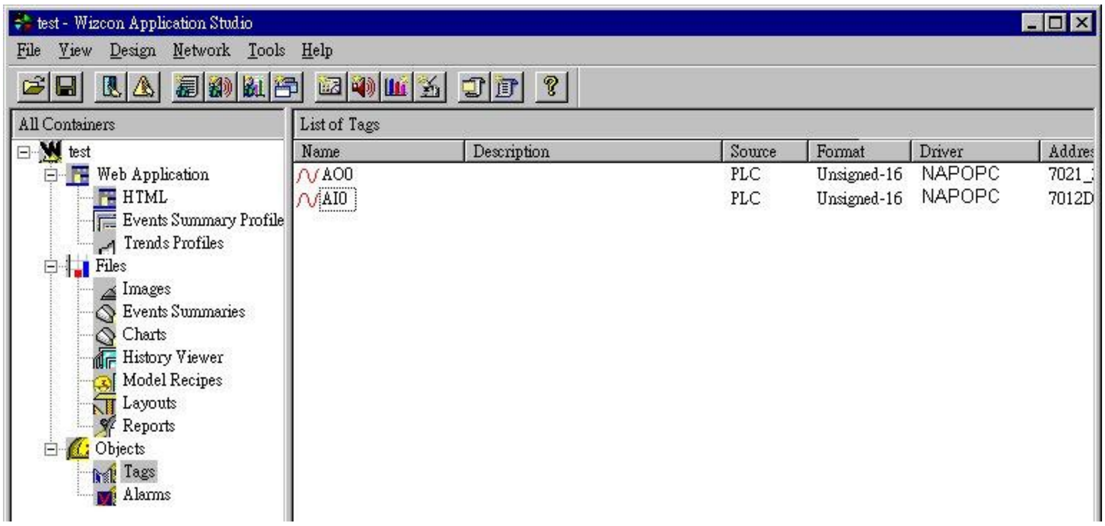
Low Limit: 1

High: 0

Conversion

	Measured	Engineering
Value 1	0	0
Value 2	1	1

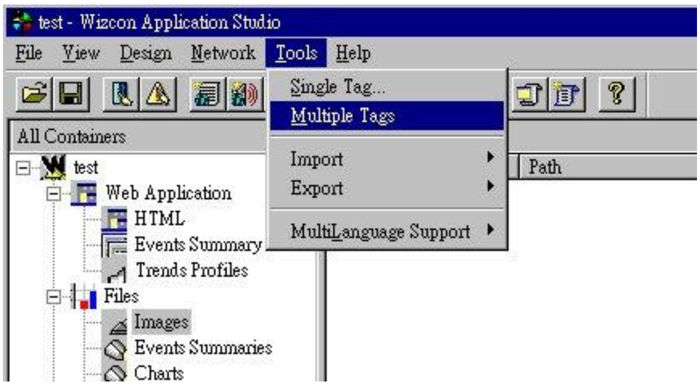
Step 24: The right hand window shows the tag(s) that were previously added.



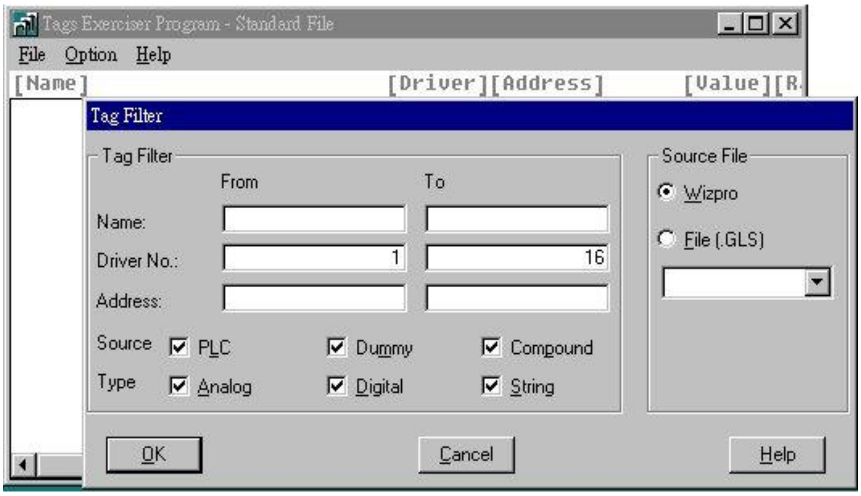
List of Tags

Name	Description	Source	Format	Driver	Addres
AO0		PLC	Unsigned-16	NAPOPC	7021_
AI0		PLC	Unsigned-16	NAPOPC	7012D

Step 25: Click on the "Tools/ Multiple Tags" menu item.



Step 26: Click on the "OK" button to close the "Tag Filter" window.


Tags Exerciser Program - Standard File
File Option Help
[Name] [Driver][Address] [Value][R
Tag Filter
OK
Cancel
Help

Tag Filter

From To

Name:

Driver No.: 1 16

Address:

Source ☑ PLC ☑ Dummy ☑ Compound

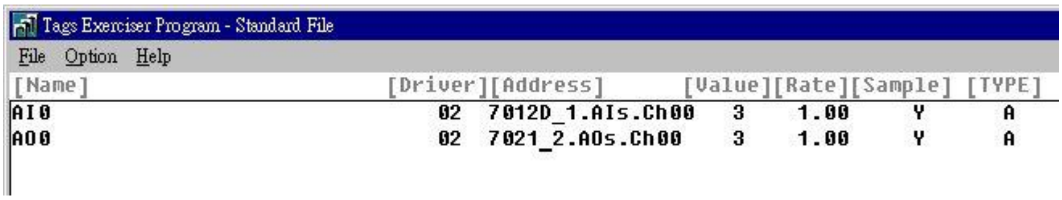
Type ☑ Analog ☑ Digital ☑ String

Source File

● Wizpro

○ File (.GLS)

Step 27: The "Tags Exerciser Program" window shows tag(s) and value(s).



```
[Name]                      [Driver][Address]        [Value][Rate][Sample] [TYPE]
AI0                              02   7012D_1.AIs.Ch00   3     1.00      Y      A
AO0                              02   7021_2.AOs.Ch00    3     1.00      Y      A
```

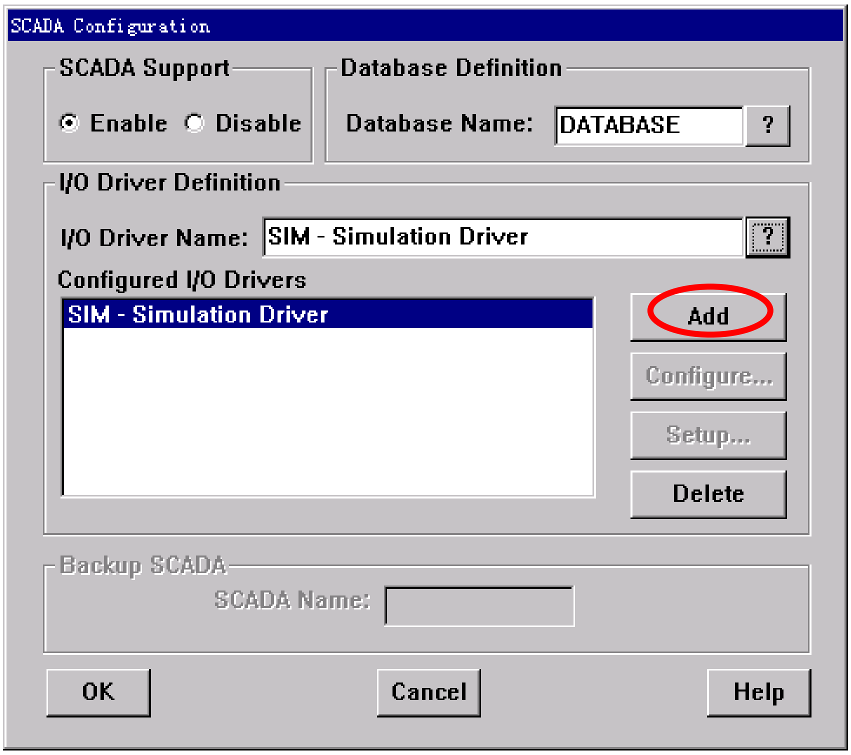
3.7 iFix

iFIX is a powerful HMI/SCADA system that features full process visualization, data collection and management, and supervisory control. iFIX, the HMI/SCADA component of the Installation Dynamics family of automation software, is a Windows NT-based industrial automation solution for monitoring and controlling manufacturing operations. For more information, please visit http://www.intellution.com.

Note:The following steps are on PC not on XPAC. If you have iFix based on Windows XP Embedded, you can refer to the following steps to try it.

Step 1: Run iFix 2.1 and start system configuration.

Step 2: Click on the "Add" button to add I/O drivers.



● Enable ○ Disable

Database Name: DATABASE

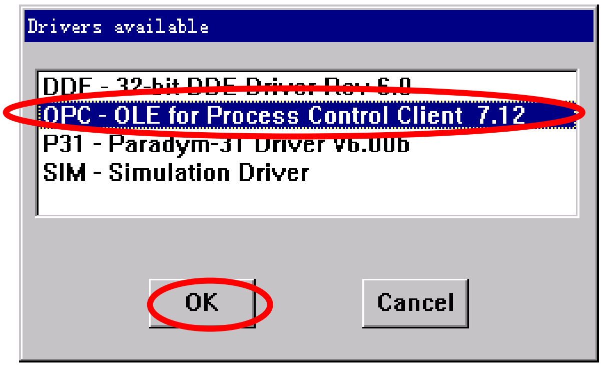
I/O Driver Name: SIM - Simulation Driver

Configured I/O Drivers

SIM - Simulation Driver

Step 3: Select the "OPC - OLE for Process Control Client 7.12" driver.

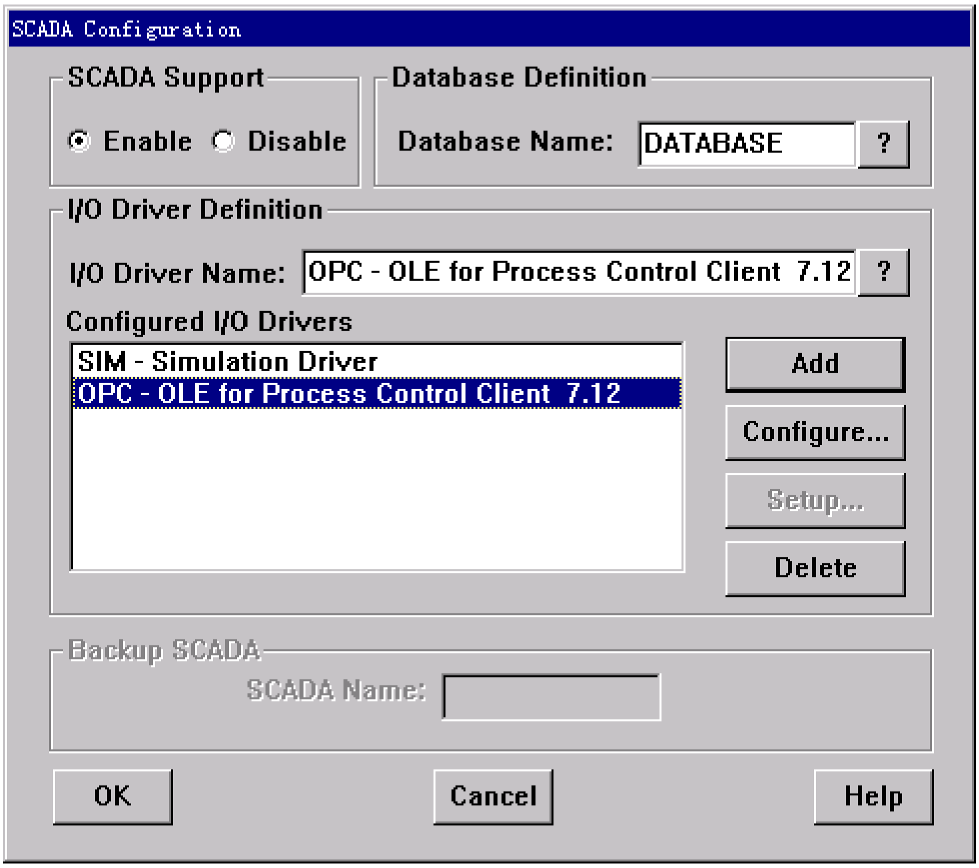
Step 4: Click on the "OK" button.


Drivers available

DDE - 32-bit DDE Driver Rev 6.0
OPC - OLE for Process Control Client 7.12
P31 - Paradym-31 Driver V6.006
SIM - Simulation Driver

Step 5: Select the "OPC - OLE for Process Control Client 7.12" driver.

Step 6: Click on the "Configure…" button to configure the I/O driver.


SCADA Configuration

SCADA Support: ● Enable ○ Disable

Database Definition

Database Name: DATABASE

I/O Driver Definition

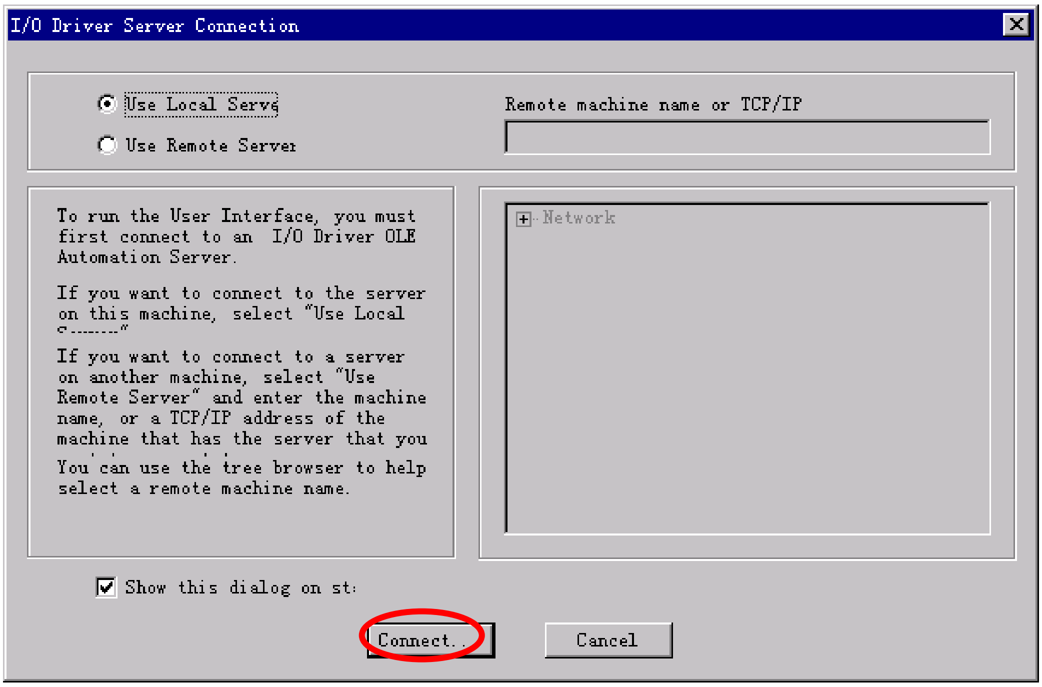
I/O Driver Name: OPC - OLE for Process Control Client 7.12

Configured I/O Drivers:
SIM - Simulation Driver
OPC - OLE for Process Control Client 7.12

Backup SCADA

SCADA Name:

Step 7: Click on the "Connect…" button.



● Use Local Serve

○ Use Remote Server

To run the User Interface, you must first connect to an I/O Driver OLE Automation Server.

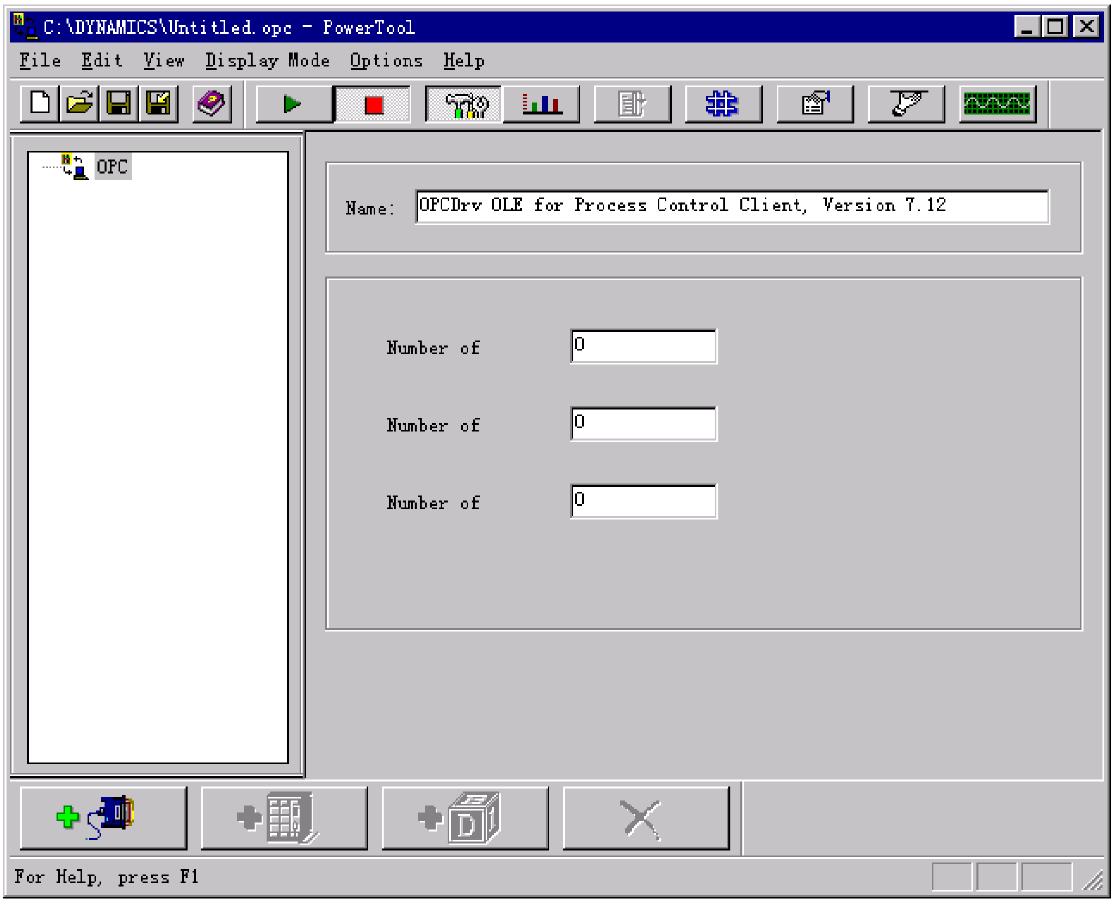
If you want to connect to the server on this machine, select “Use Local

If you want to connect to a server on another machine, select “Use Remote Server” and enter the machine name, or a TCP/IP address of the machine that has the server that you

You can use the tree browser to help select a remote machine name.

☑ Show this dialog on st:

Step 8: To configure the OPC server.



C:\DYNAMICS\Untitled.opc - PowerTool

Name: OPCDrv OLE for Process Control Client, Version 7.12

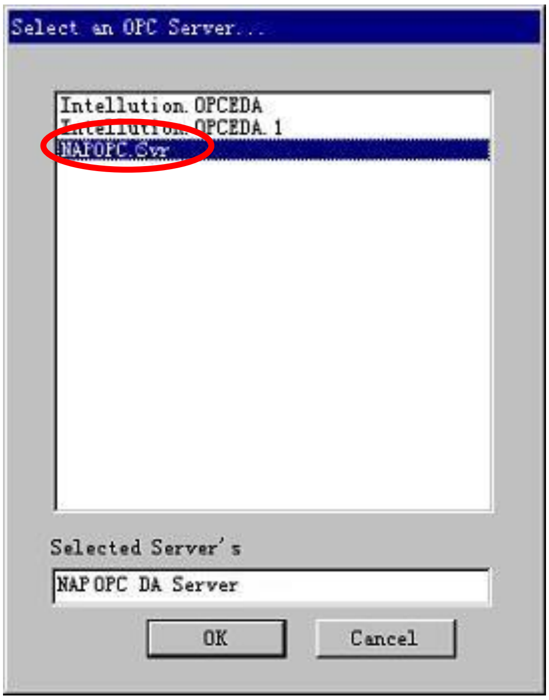
Number of 0

Number of 0

Number of 0

For Help, press F1

Step 9: Select the "NAPOPC.Svr" and click on the "OK" button.

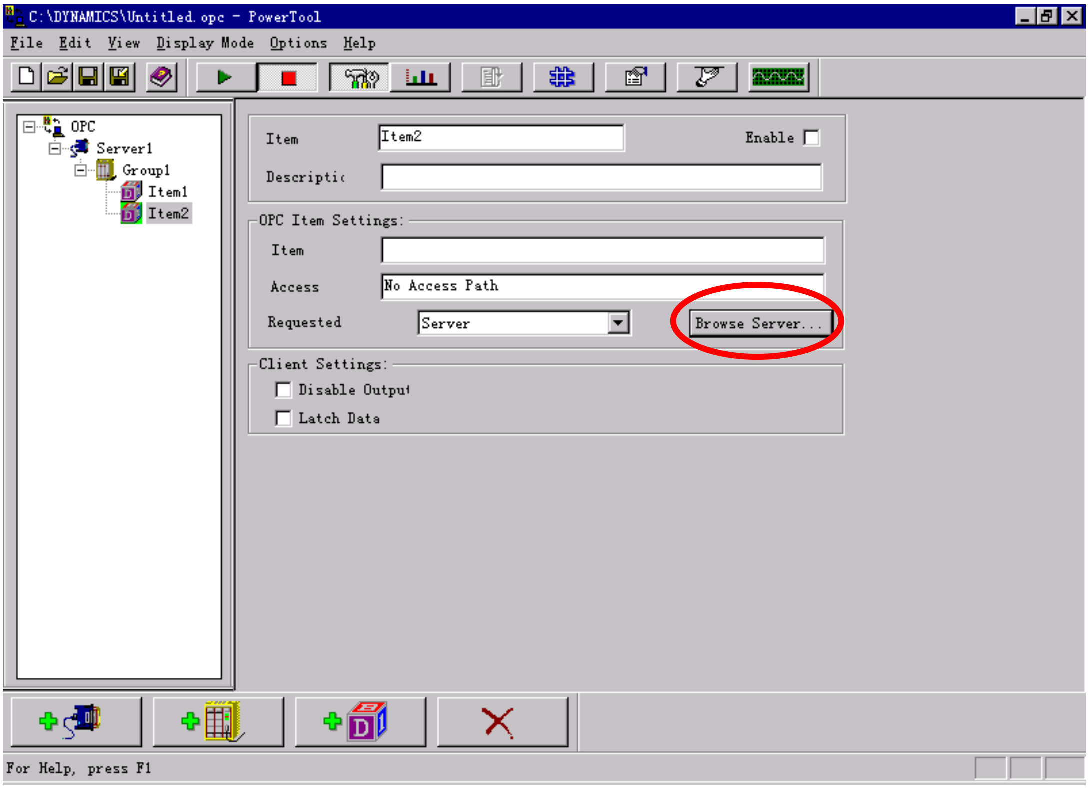


Select an OPC Server...

Intellution.OPCEDA
Intellution.OPCEDA.1
NAPOPC.Svr

Selected Server's
NAP OPC DA Server

Step 10: Add server, group and items. Fill propherties by clicking on the "Browse Server…" button.


C:\DYNAMICS\Untitled.opc - PowerTool
File Edit View Display Mode Options Help
Browse Server...
For Help, press F1

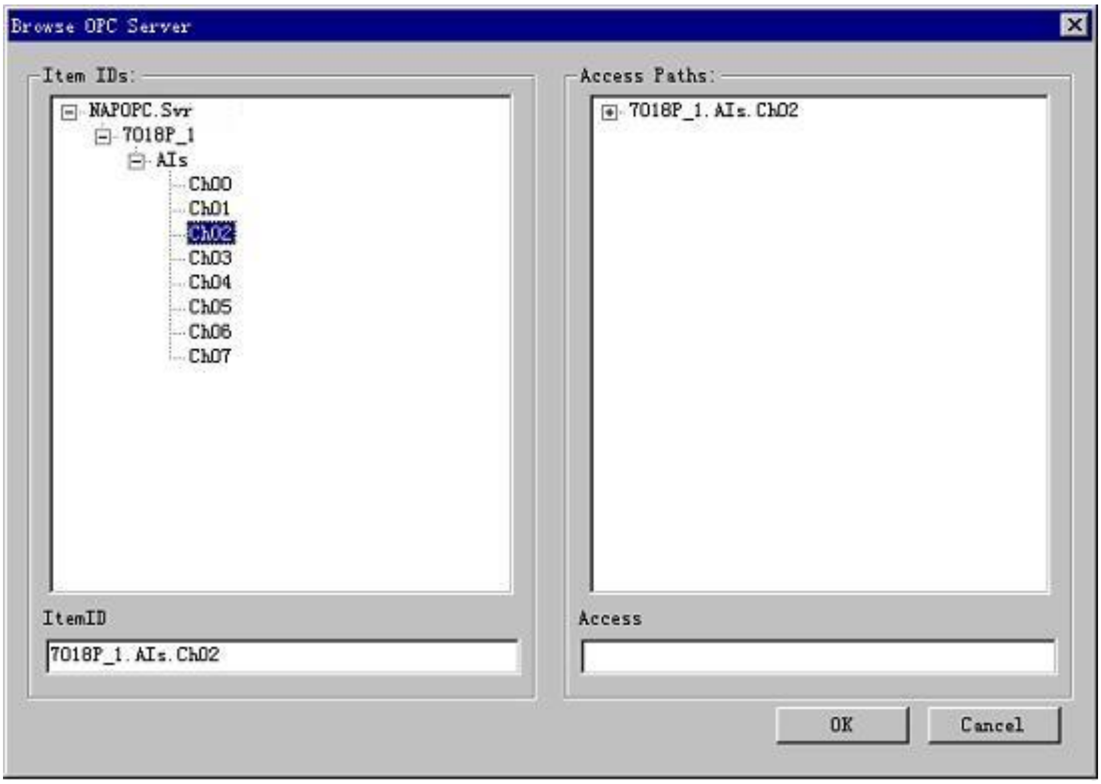
OPC
Server1
Group1
Item1
Item2

Item: Item2
Enable □
Descriptic:

OPC Item Settings:
Item:
Access: No Access Path
Requested: Server

Client Settings:
□ Disable Output
□ Latch Data

Step 11: Select the item which you need. Click on the "OK" button.



Item IDs:
NAPOPC.Svr
7018P_1
AIs
Ch00
Ch01
Ch02
Ch03
Ch04
Ch05
Ch06
Ch07

Access Paths:
7018P_1.AIs.Ch02

ItemID: 7018P_1.AIs.Ch02

Access:

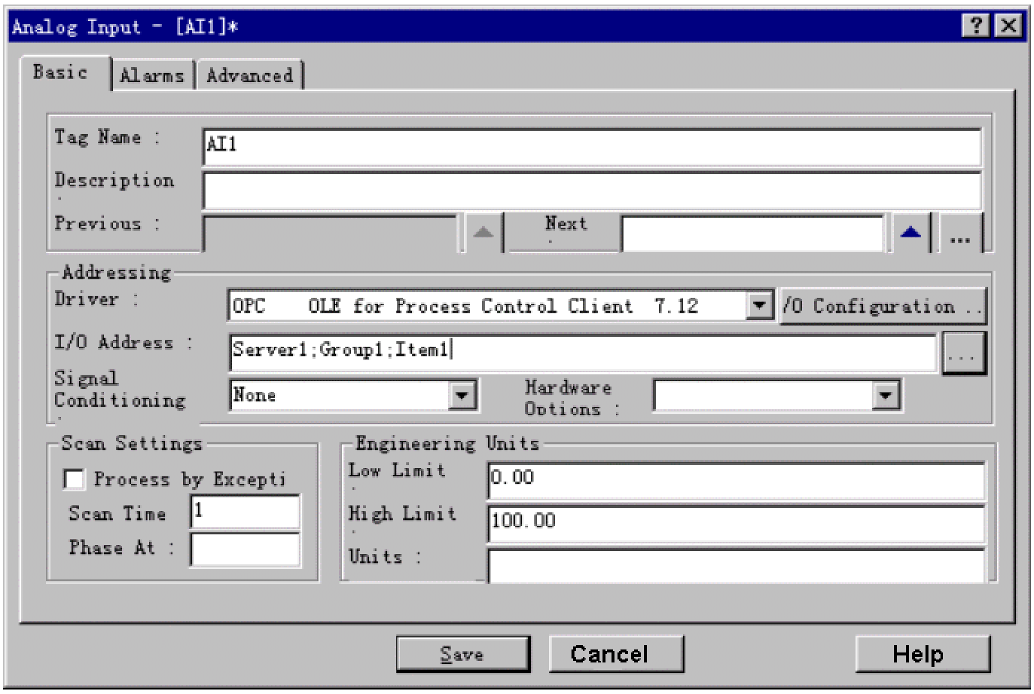
Step 12: Enter database manager of iFix 2.1.

Step 13: Add relative data units.

For example: AI.

Driver: "OPC OLE for Process Control Client 7.12".

I/O Address: "Server1;Group1:Item1".



Analog Input - [AI1]*

Tag Name : AI1

Description :

Previous :

Next :

Addressing

Driver : OPC OLE for Process Control Client 7.12

I/O Address : Server1;Group1;Item1

Signal Conditioning : None

Hardware Options :

Scan Settings

□ Process by Excepti

Scan Time 1

Phase At :

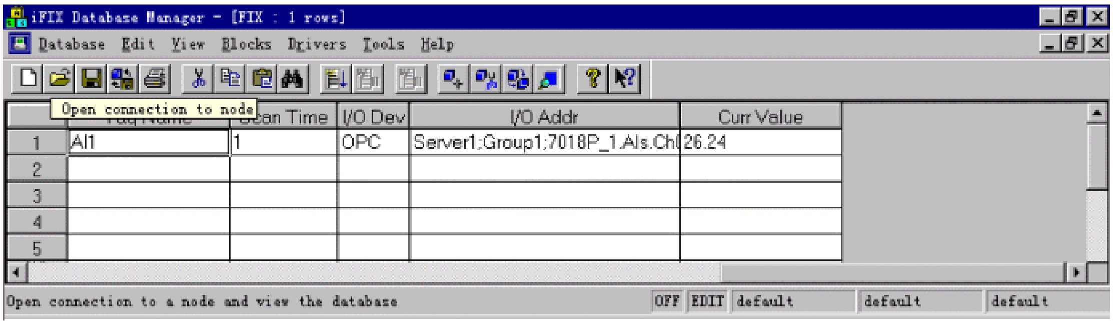
Engineering Units

Low Limit : 0.00

High Limit : 100.00

Units :

Step 14: The window displays the current value of the AI unit.



iFIX Database Manager - [FIX : 1 rows]

Open connection to node

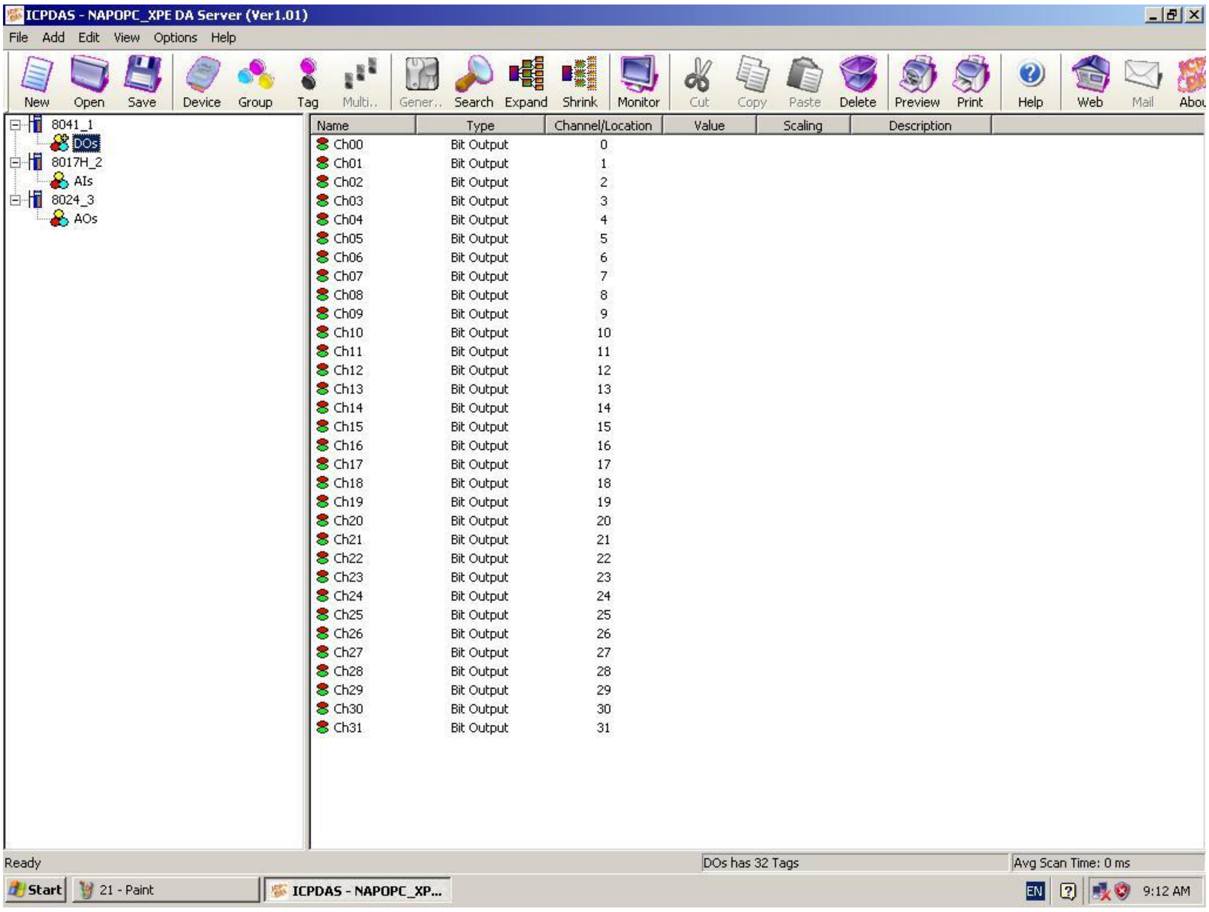
	Tag Name	Scan Time	I/O Dev	I/O Addr	Curr Value
1	AI1	1	OPC	Server1;Group1;7018P_1.AIs.Ch(	26.24
2					
3					
4					
5					

Open connection to a node and view the database

3.8 InduSoft

InduSoft Web Studio is a powerful, integrated collection of automation tools that includes all the building blocks needed to develop human machine interfaces (HMIs), supervisory control and data acquisition (SCADA) systems, and embedded instrumentation and control applications. Web Studio runs in native Windows NT, 2000, XP and CE 3.0 environments and conforms to industry standards such as Microsoft DNA, OPC, DDE, ODBC, XML, SOAP and ActiveX. For more information please visit: http://www.indusoft.com/

Step 1: Before using the InduSoft OPC Client module, you need to install and configure the OPC server in the machines you will run it.


ICPDAS - NAPOPC_XPE DA Server (Ver1.01)
File Add Edit View Options Help
New Open Save Device Group Tag Multi.. Gener.. Search Expand Shrink Monitor Cut Copy Paste Delete Preview Print Help Web Mail Abou
8041_1
DOs
8017H_2
AIs
8024_3
AOs
Start
21 - Paint
ICPDAS - NAPOPC_XP...
EN
9:12 AM

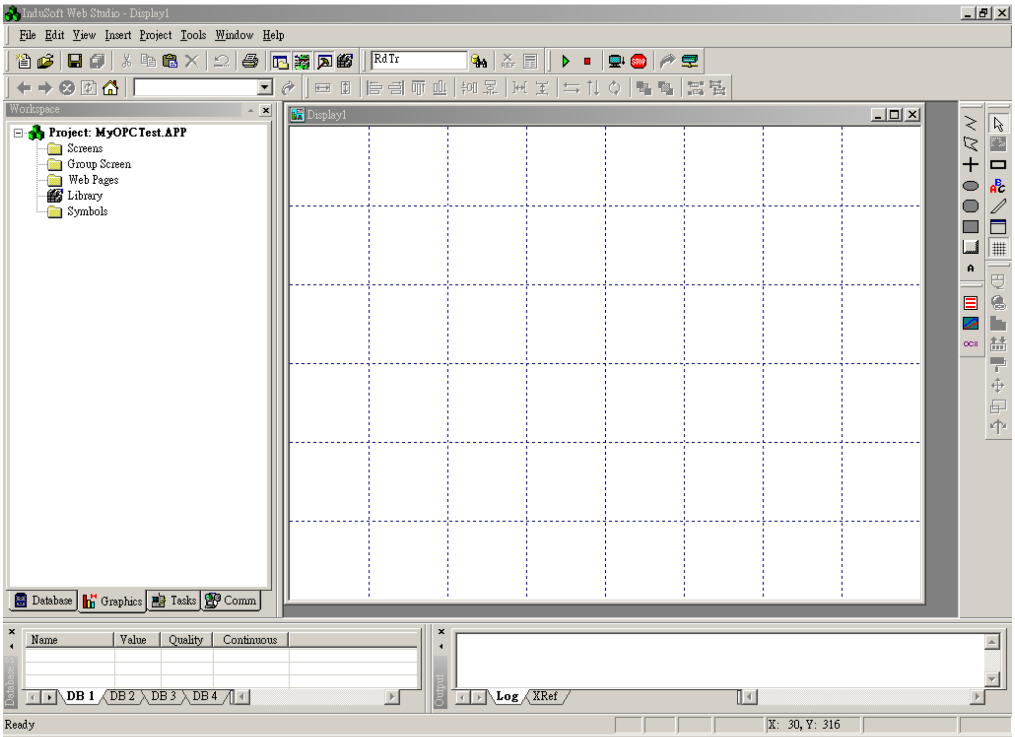
Name	Type	Channel/Location	Value	Scaling	Description
Ch00	Bit Output	0			
Ch01	Bit Output	1			
Ch02	Bit Output	2			
Ch03	Bit Output	3			
Ch04	Bit Output	4			
Ch05	Bit Output	5			
Ch06	Bit Output	6			
Ch07	Bit Output	7			
Ch08	Bit Output	8			
Ch09	Bit Output	9			
Ch10	Bit Output	10			
Ch11	Bit Output	11			
Ch12	Bit Output	12			
Ch13	Bit Output	13			
Ch14	Bit Output	14			
Ch15	Bit Output	15			
Ch16	Bit Output	16			
Ch17	Bit Output	17			
Ch18	Bit Output	18			
Ch19	Bit Output	19			
Ch20	Bit Output	20			
Ch21	Bit Output	21			
Ch22	Bit Output	22			
Ch23	Bit Output	23			
Ch24	Bit Output	24			
Ch25	Bit Output	25			
Ch26	Bit Output	26			
Ch27	Bit Output	27			
Ch28	Bit Output	28			
Ch29	Bit Output	29			
Ch30	Bit Output	30			
Ch31	Bit Output	31			

Ready | DOs has 32 Tags | Avg Scan Time: 0 ms

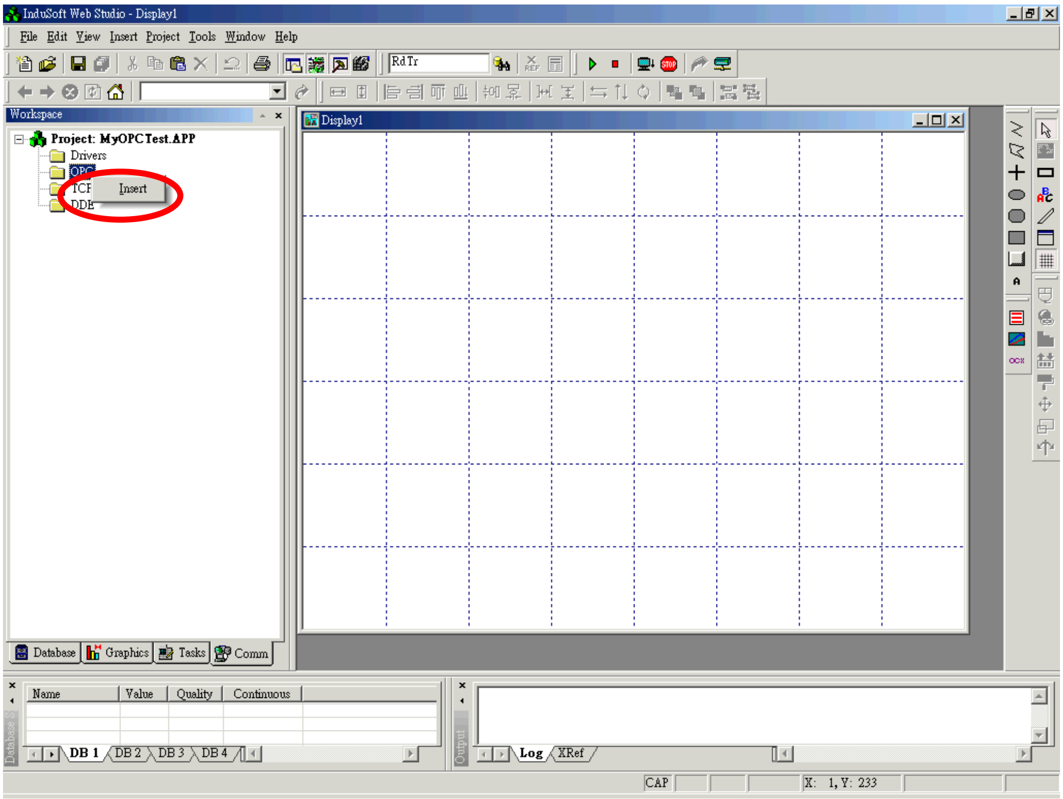
Step 2: Run the InduSoft (Version 4.1 or newer)

Web-Ready
InduSoft
Tools for Automation
Web and Wireless Automation the easy way
Copyright ©1997-2004 InduSoft Ltd.
This program is protected by International laws.

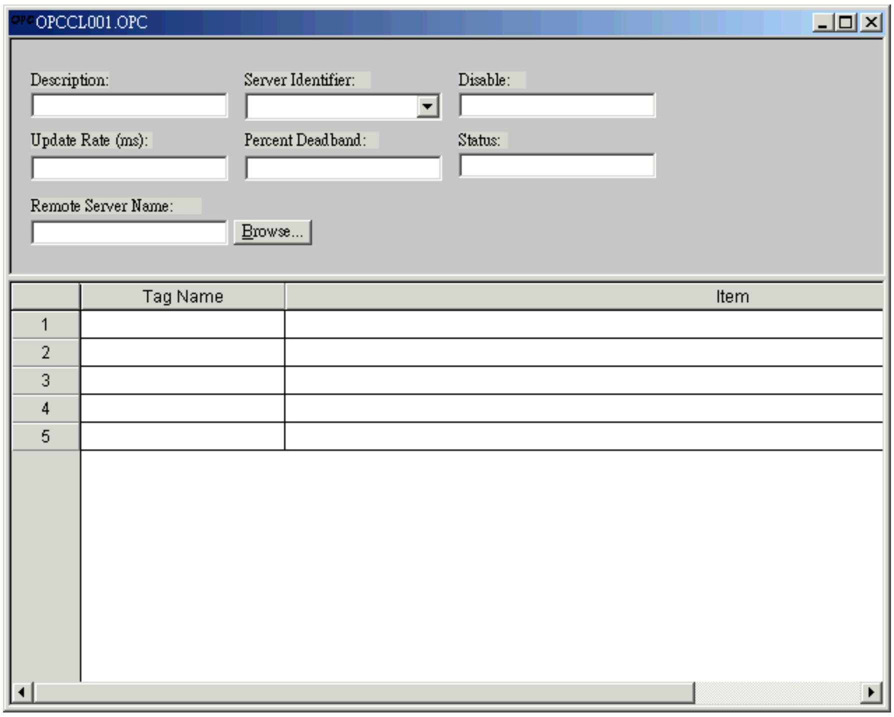
Step 3: Create the new project.



Step 4: In the Studio Workspace window, click the OPC tab, right-click the OPC folder, and click "Insert":



Step 5: OPC Attributes window pops up.



Description:

Server Identifier:

Disable:

Update Rate (ms):

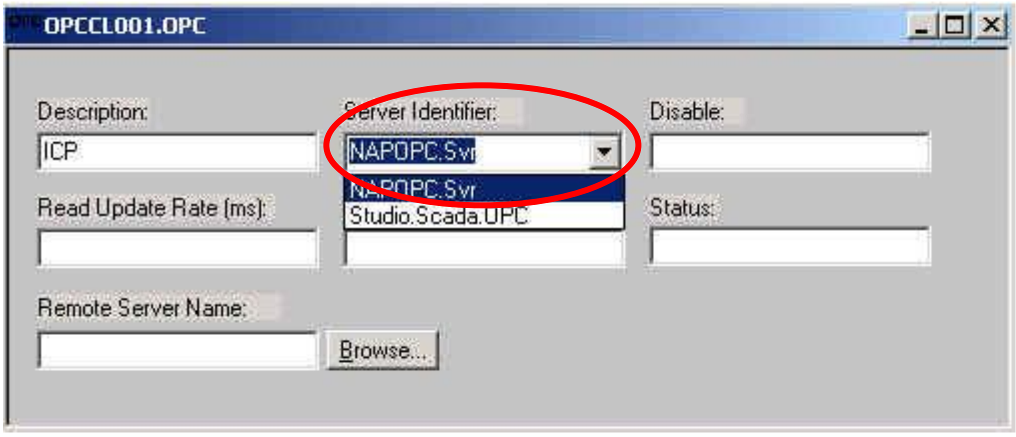
Percent Deadband:

Status:

Remote Server Name:

	Tag Name	Item
1		
2		
3		
4		
5		

Step 6: Click on the Server Identifier: drop-down menu and select the "NAPOPC.Svr".

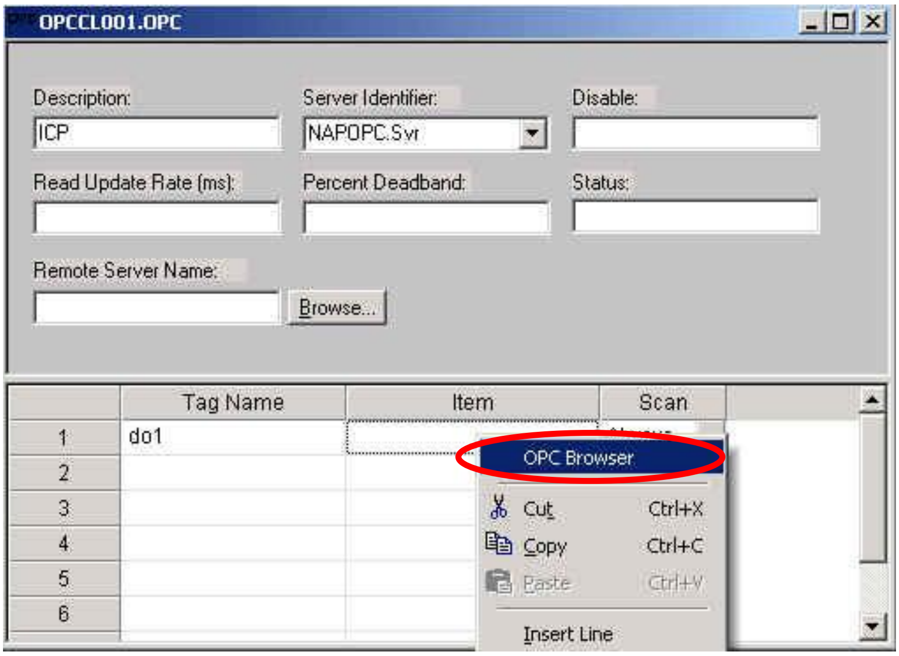


The configuration table for OPC has the following entries:

- Description: this field is used for documentation only. The OPC Client module ignores it.
- Server Identifier: this field should contain the name of the server you want to connect. If the server is installed in the computer, its name can be selected through the list box.
- Disable: this field should contain a tag or a constant. If its value is different of zero, the communication with the OPC server is disabled.
- Update Rate: this field indicates how often the server will update this group in milliseconds. If it is zero indicates the server should use the fastest practical rate.
- Percent Deadband: this field indicates the percent change in an item value that will cause a notification by the server. It's only valid for analog items.
- Tag Name: these fields should contain the tags linked to the server items.
- Item: these fields should contain the name of the server's items

Step 7: In the first cell of the Tag Name column type the tag name created in database.

Step 8: In the first cell of the item you can right-click it to get a menu.



OPCCL001.OPC

Description: ICP

Server Identifier: NAPOPC.Svr

Disable:

Read Update Rate (ms):

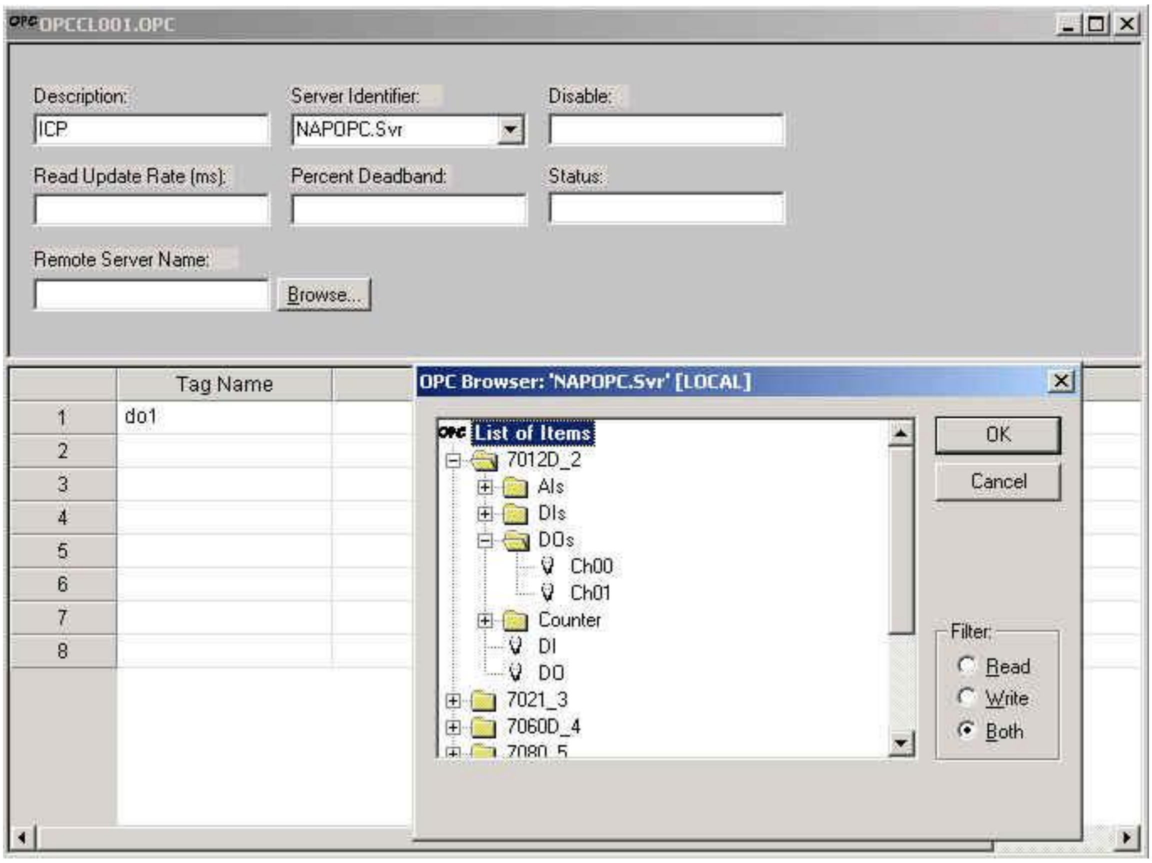
Percent Deadband:

Status:

Remote Server Name:

	Tag Name	Item	Scan
1	do1		
2			
3			
4			
5			
6			

Step 9: Click the OPC Browser to appear the OPC Browser window.



OPCCL001.OPC

Description: ICP

Server Identifier: NAPOPC.Svr

Disable:

Read Update Rate (ms):

Percent Deadband:

Status:

Remote Server Name:

	Tag Name
1	do1
2	
3	
4	
5	
6	
7	
8	

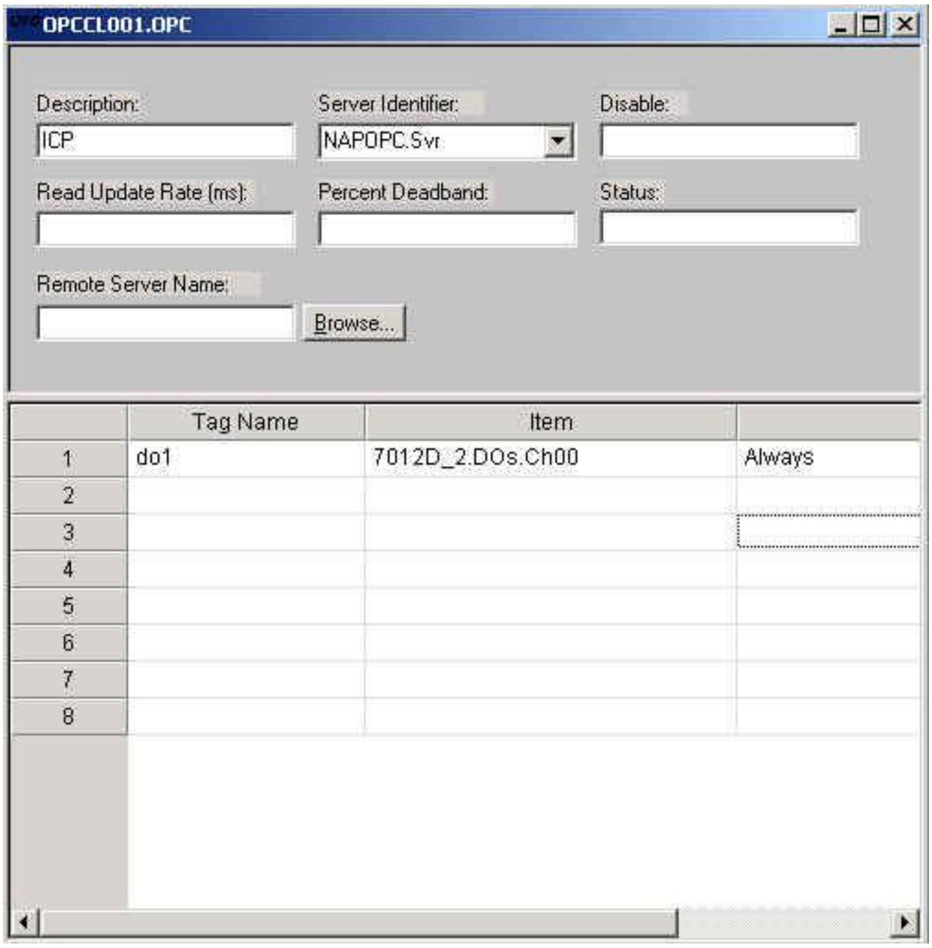
OPC Browser: 'NAPOPC.Svr' [LOCAL]

```
List of Items
⊟ 7012D_2
    ⊞ AIs
    ⊞ DIs
    ⊟ DOs
        Ch00
        Ch01
    ⊞ Counter
      DI
      DO
⊞ 7021_3
⊞ 7060D_4
⊞ 7080_5
```

Filter: ○ Read ○ Write ● Both

Step 10: Select an item(tag) in the tree-view.

Step 11: Click the "OK" button to add this one.



OPCCL001.OPC

Description: ICP

Server Identifier: NAPOPC.Svr

Disable:

Read Update Rate (ms):

Percent Deadband:

Status:

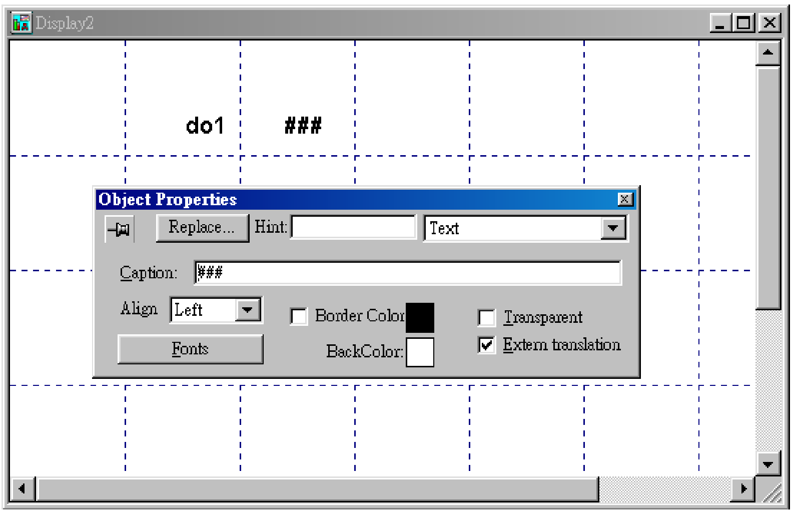
Remote Server Name:

	Tag Name	Item	
1	do1	7012D_2.DOs.Ch00	Always
2			
3			
4			
5			
6			
7			
8			

Step 12: Repeat the step between 7 to 11 to add more tags.

Step 13: Creating a Text String for the Input/Output Dynamic. Click the Text icon on the Object Editing toolbar. Position the crosshairs in the Display2.scr. Press the"#" key three times to display "###" in the gray square.

Step 14: Click the Text Input/Output property icon on the Object Editing toolbar. *Text I/O* appears in the drop-down menu of the Object Properties window. In the Tag/Expression field type the tag name you want to link.



Display2

do1 ###

Object Properties

Hint:

Text

Caption: ###

Align: Left

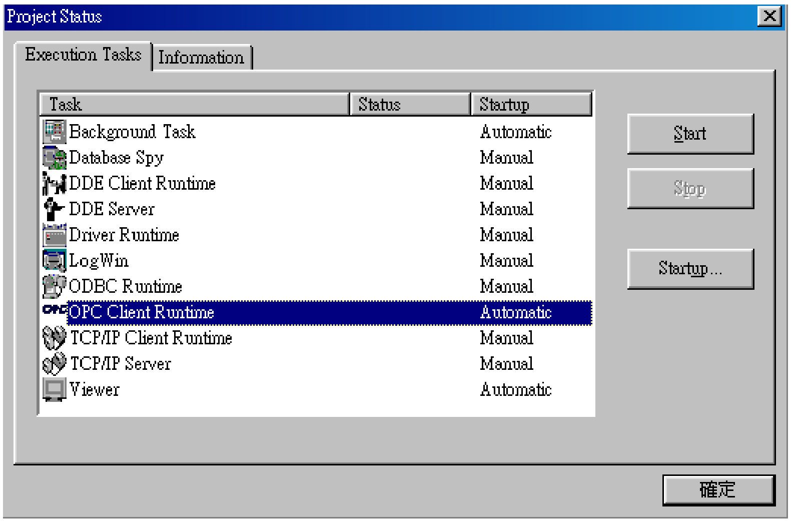
□ Border Color

BackColor:

□ Transparent

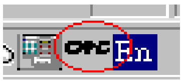
☑ Extern translation

Step 15: From the Project select status. Then select the OPC Client Runtime in “Execution Tasks” tab. Click on the Startup button to setup the Startup as Automatic.

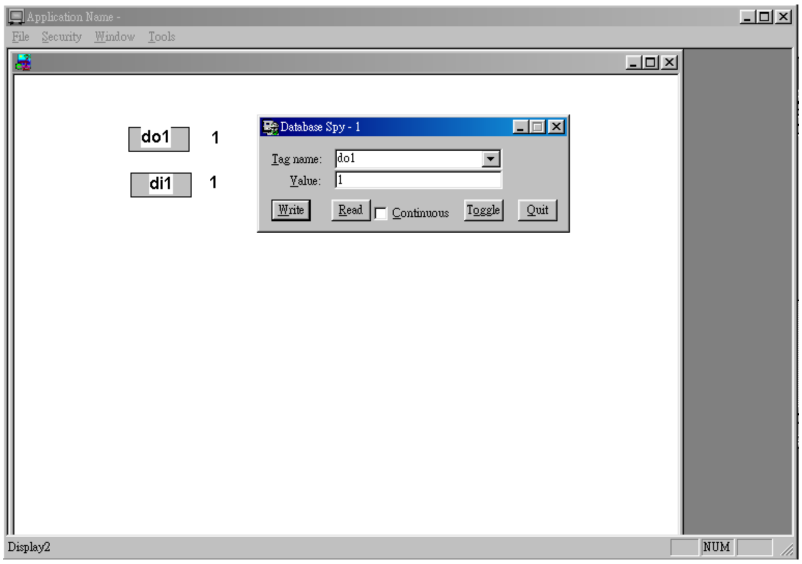


Task	Status	Startup
Background Task		Automatic
Database Spy		Manual
DDE Client Runtime		Manual
DDE Server		Manual
Driver Runtime		Manual
LogWin		Manual
ODBC Runtime		Manual
OPC Client Runtime		Automatic
TCP/IP Client Runtime		Manual
TCP/IP Server		Manual
Viewer		Automatic

Step 16: Run the program InduSoft OPC Client Runtime module automatically or by the menu "Project->Status". After running this program, a small icon will appear in your system tray. To close the InduSoft OPC Client module, right-click its icon in the system tray, and select “Exit”.



Step 17: Database Spy allows you to monitor and forces application tags, reading and writing to the database. You can find it in Tools menu.



do1 1

di1 1

Tag name: do1

Value: 1

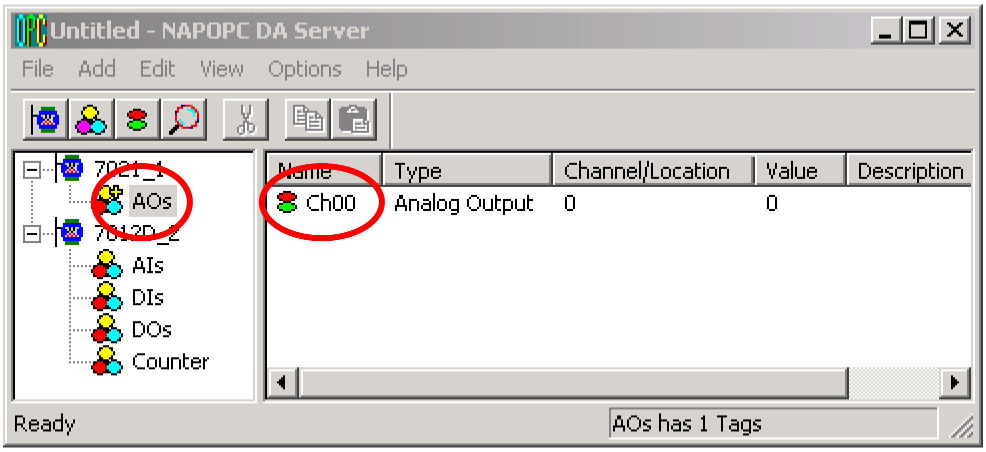
□ Continuous

3.9 Citect SCADA

CitectSCADA is a reliable, scaleable and high performance SCADA system that includes over 100 drivers and free development software. Used in a wide range of industries, CitectSCADA enables users to reduce costs by optimizing process operations. Furthermore, it not only reduce risk with built-in redundancy for servers, networks and communications, but open data connectivity via OPC client & server, OLE DB, ODBC, DDE and API as well as over a hundred native drivers. CitectSCADA can implement in Windows 98, NT and 2000. Visit http://www.citect.com for more information about CitectSCADA

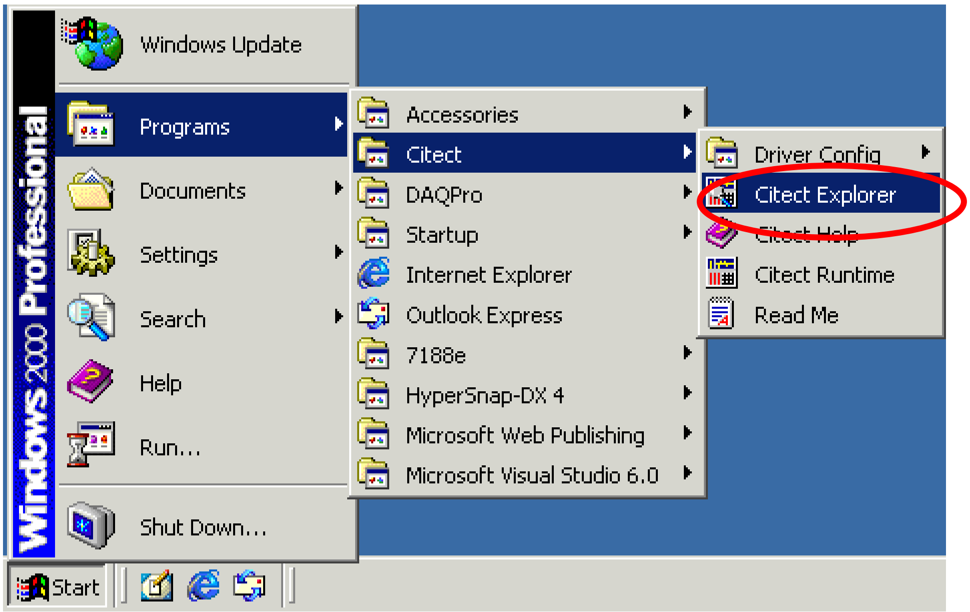
Note:The following steps are on PC not on XPAC. If you have Citect SCADA based on Windows XP Embedded, you can refer to the following steps to try it.

Step 1: Before using the CitectSCADA, you need to install and configure the OPC server in the machines you will run it (see Chapter 1).

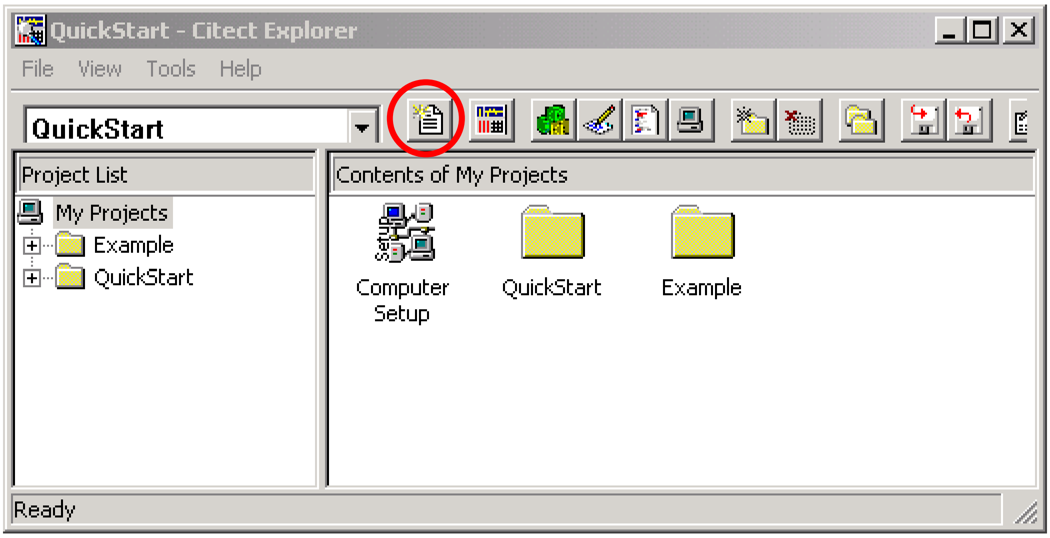


Name	Type	Channel/Location	Value	Description
Ch00	Analog Output	0	0	

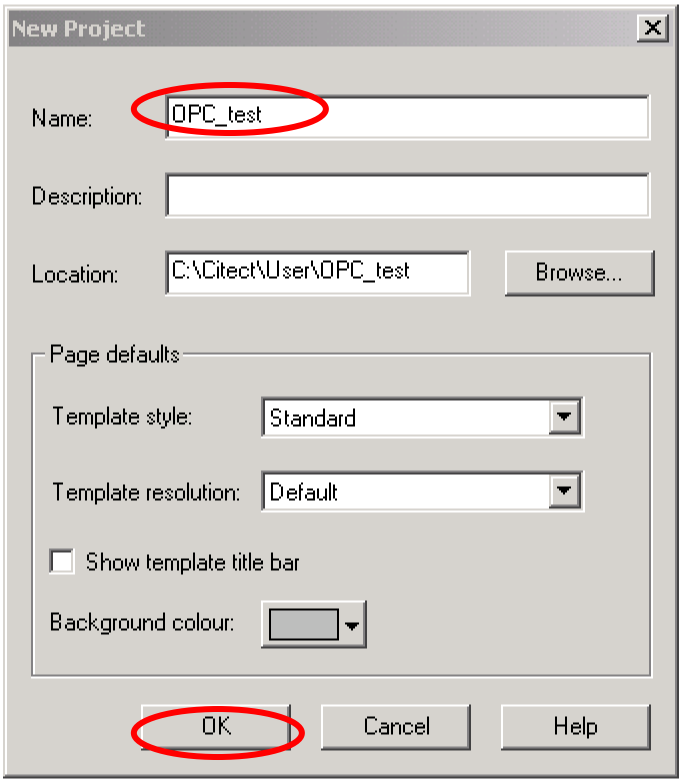
Step2: Start up the CitectSCADA with version 5.40.



Step3: Left click the "page-marked button" or select "File / New Project…" from the CitectSCADA window menu to build a new project in the CitectSCADA.



Step4: Fill a name of new project in the blank and then click "OK button" to finish this process.



Name: OPC_test

Description:

Location: C:\Citect\User\OPC_test

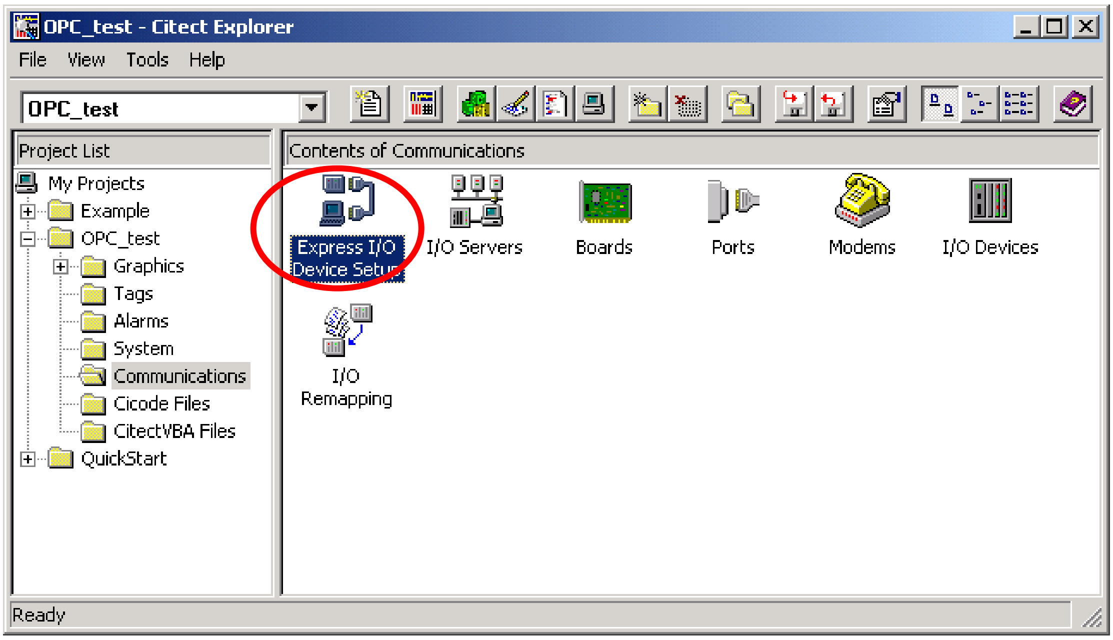
Template style: Standard

Template resolution: Default

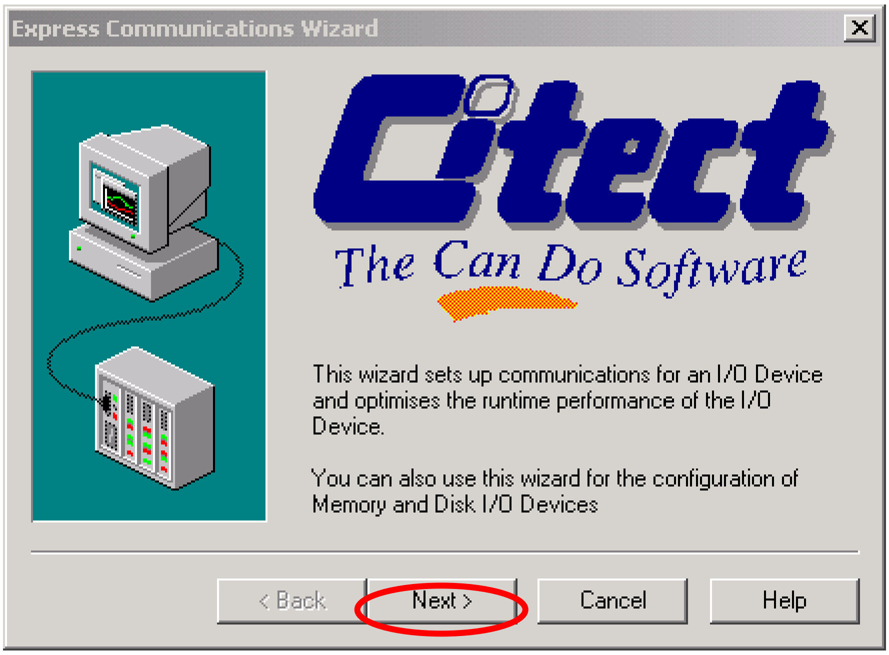
□ Show template title bar

Background colour:

Step5: Click the "Express I/O Device Setup" icon to set all communication parameters.



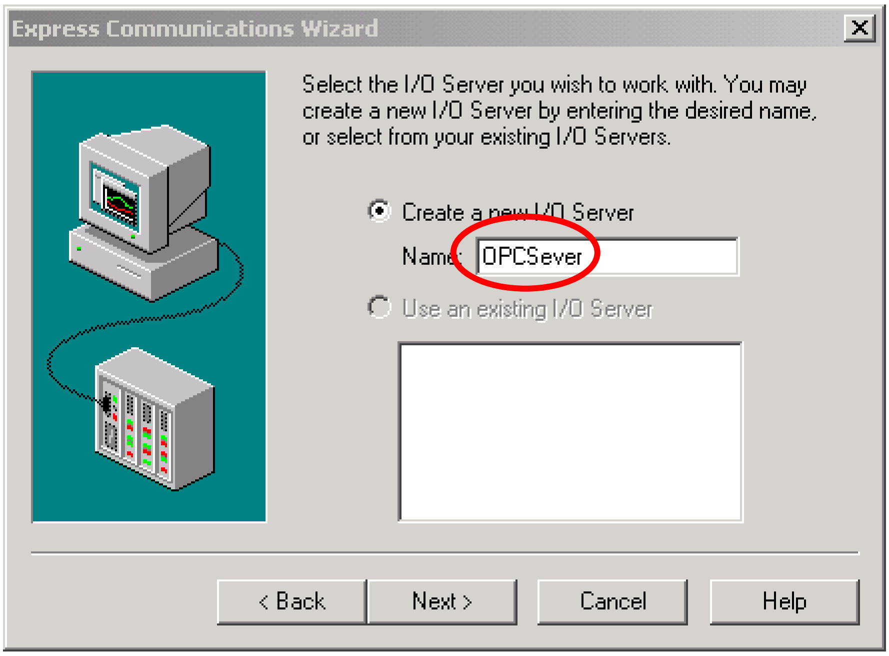
Step6: Start up the "Express Communications Wizard Dialog".



This wizard sets up communications for an I/O Device and optimises the runtime performance of the I/O Device.

You can also use this wizard for the configuration of Memory and Disk I/O Devices

Step7: Create a new I/O Server and define a name called "OPCServer" for that one.



Express Communications Wizard

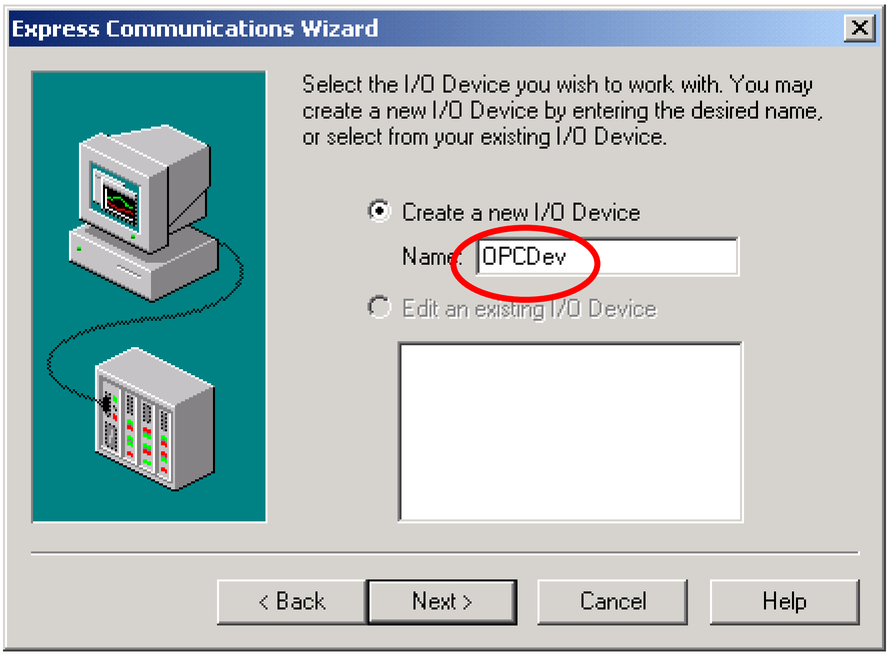
Select the I/O Server you wish to work with. You may create a new I/O Server by entering the desired name, or select from your existing I/O Servers.

● Create a new I/O Server

Name: OPCSever

○ Use an existing I/O Server

Step8: Create a new I/O Device under the I/O Server that created previously and define a name called OPCDev for that one.



Express Communications Wizard

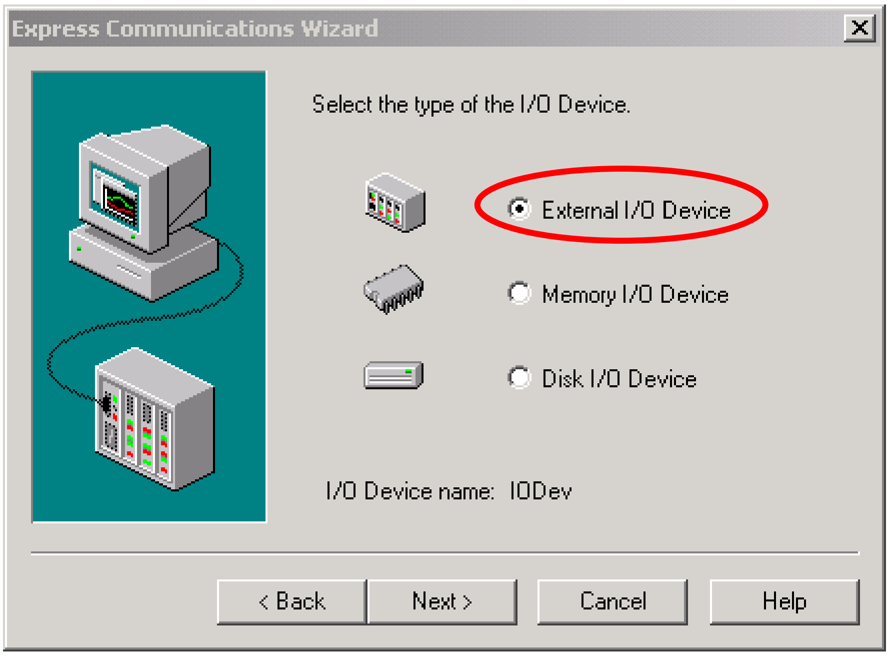
Select the I/O Device you wish to work with. You may create a new I/O Device by entering the desired name, or select from your existing I/O Device.

● Create a new I/O Device

Name: OPCDev

○ Edit an existing I/O Device

Step9: Select "External I/O Device" to be the type of OPCDev I/O Device.



Express Communications Wizard

Select the type of the I/O Device.

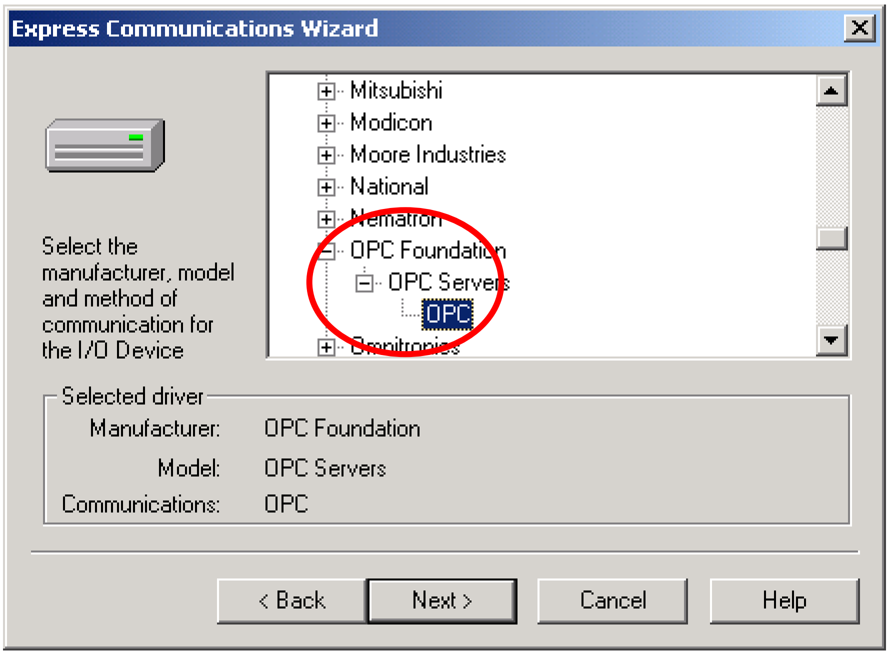
● External I/O Device

○ Memory I/O Device

○ Disk I/O Device

I/O Device name: IODev

Step10: Set OPC to be the method of communication for OPCDev I/O Device.



Express Communications Wizard

Select the manufacturer, model and method of communication for the I/O Device

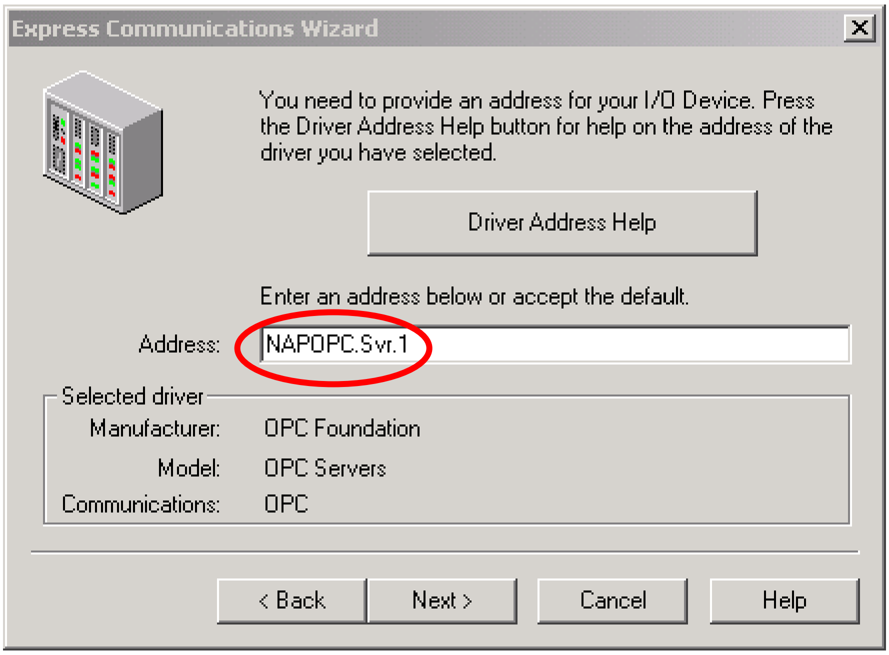
Selected driver

Manufacturer: OPC Foundation

Model: OPC Servers

Communications: OPC

Step11: Set Address to be "NAPOPC.Svr.1" for OPCDev I/O Device.



Express Communications Wizard

You need to provide an address for your I/O Device. Press the Driver Address Help button for help on the address of the driver you have selected.

Enter an address below or accept the default.

Address: NAPOPC.Svr.1

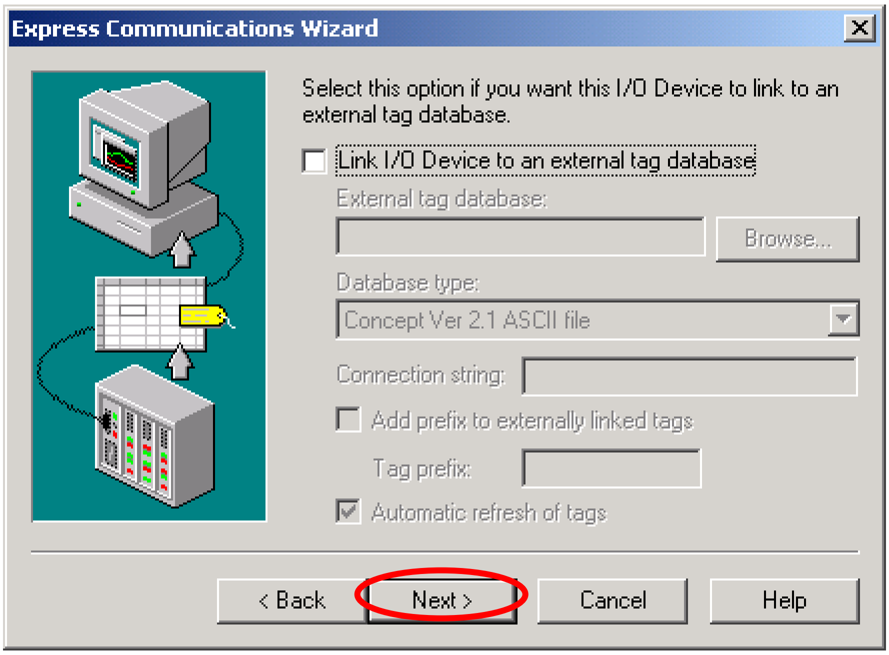
Selected driver

Manufacturer: OPC Foundation

Model: OPC Servers

Communications: OPC

Step12: Do not set any parameter in this step.



Express Communications Wizard

Select this option if you want this I/O Device to link to an external tag database.

□ Link I/O Device to an external tag database

External tag database:

Database type: Concept Ver 2.1 ASCII file

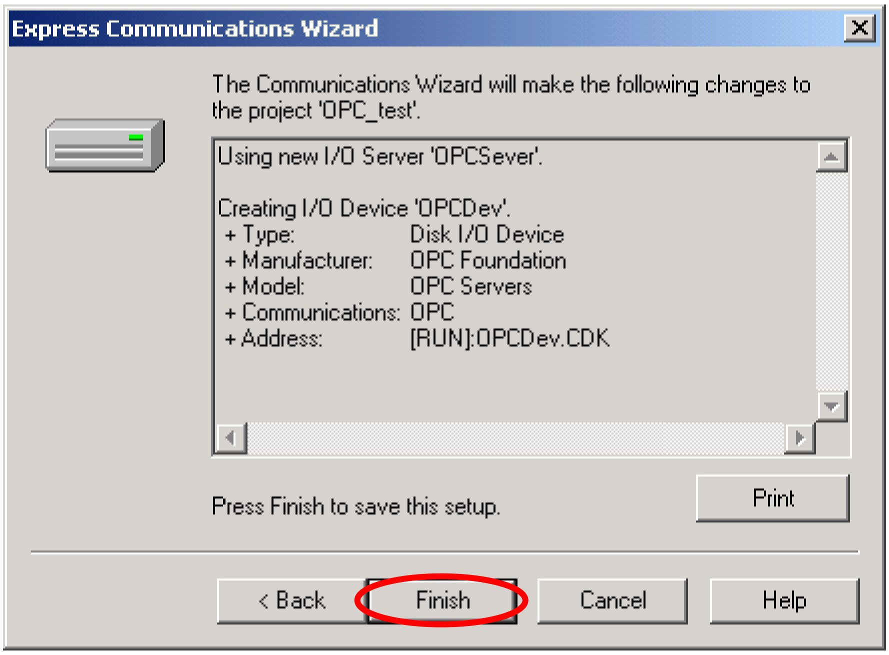
Connection string:

□ Add prefix to externally linked tags

Tag prefix:

☑ Automatic refresh of tags

Step13: Click the "OK button" to finish the setting of communication parameters.



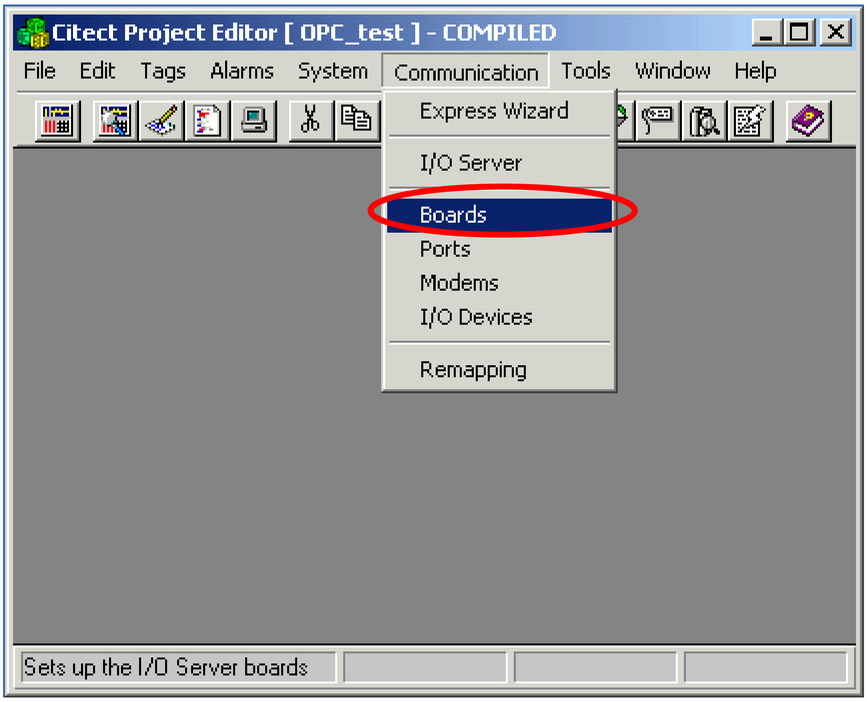
The Communications Wizard will make the following changes to the project 'OPC_test'.

```
Using new I/O Server 'OPCSever'.

Creating I/O Device 'OPCDev'.
 + Type:             Disk I/O Device
 + Manufacturer:     OPC Foundation
 + Model:            OPC Servers
 + Communications:  OPC
 + Address:          [RUN]:OPCDev.CDK
```

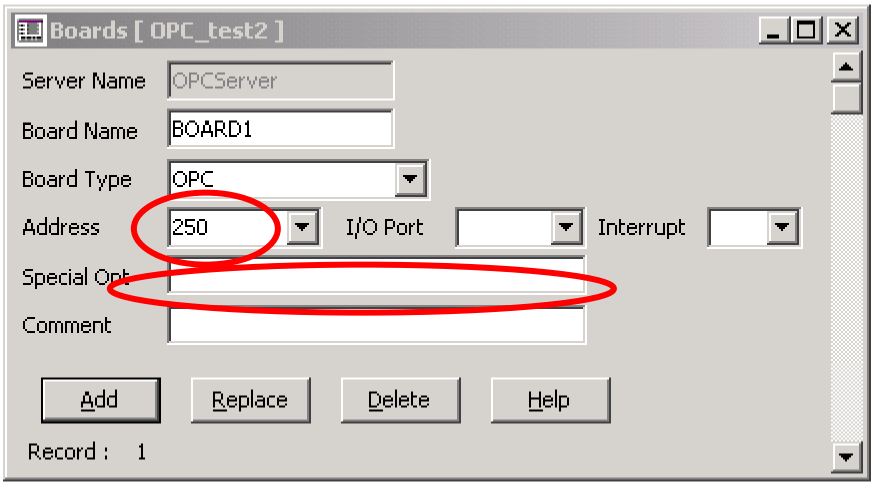
Press Finish to save this setup.

Step14: Open the "Citect Project Editor window" to edit Boards parameters.



Sets up the I/O Server boards

Step15: Edit Boards parameters. The Address (scanning period) is set to be "250ms", and the Special Opt is set to be blank.



Boards [OPC_test2]

Server Name: OPCServer

Board Name: BOARD1

Board Type: OPC

Address: 250

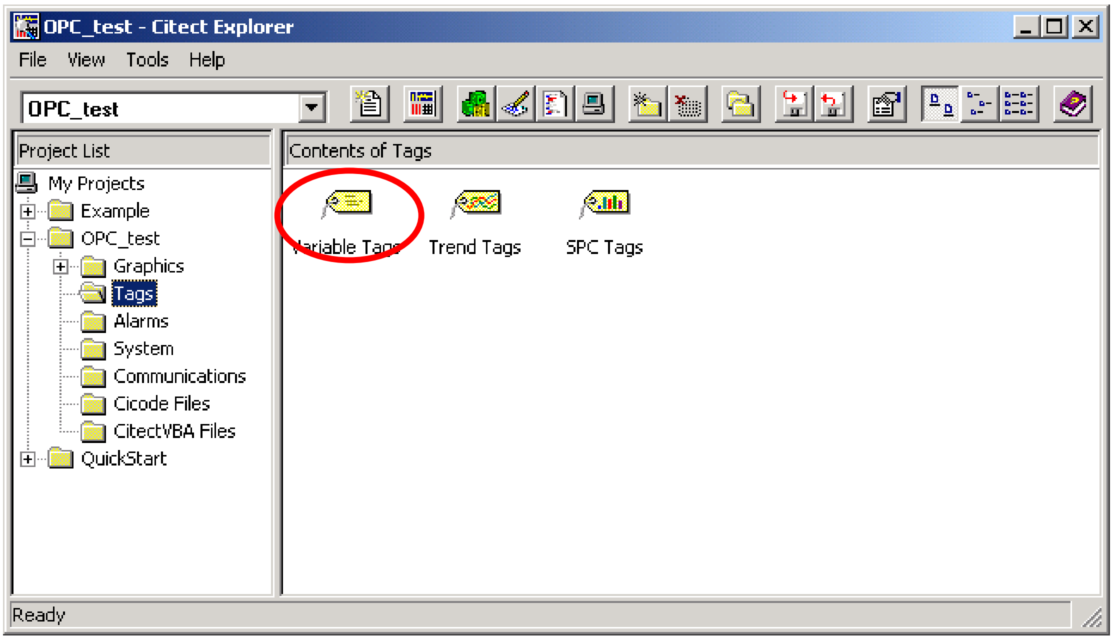
I/O Port:

Interrupt:

Special Opt:

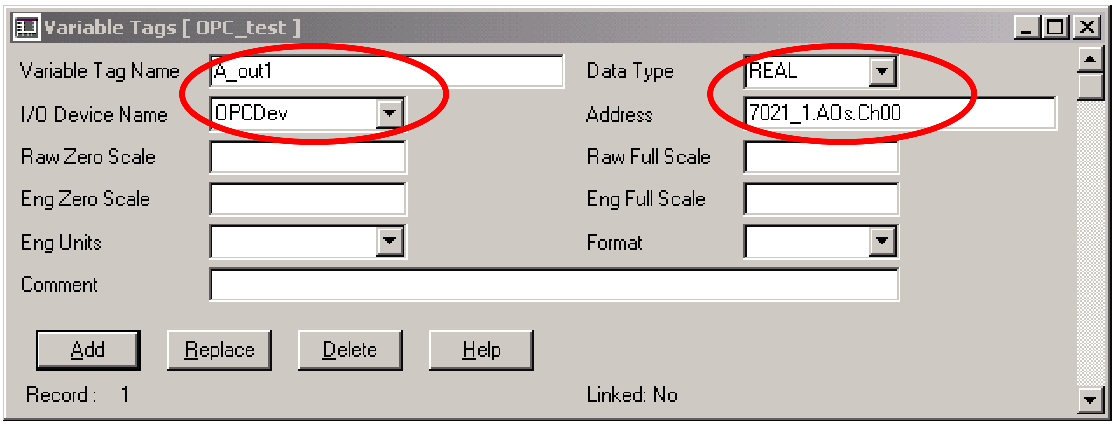
Comment:

Step16: Define two Variable Tags.



OPC_test - Citect Explorer

Step17: Define a variable tag for analog output of the modules 7021. The Variable Tag Name is "A_out1", the Data type is "REAL", the I/O Device Name is selected to be "OPCDev", and the Address is "7021_1.AOs.Ch00".(see Step 6)



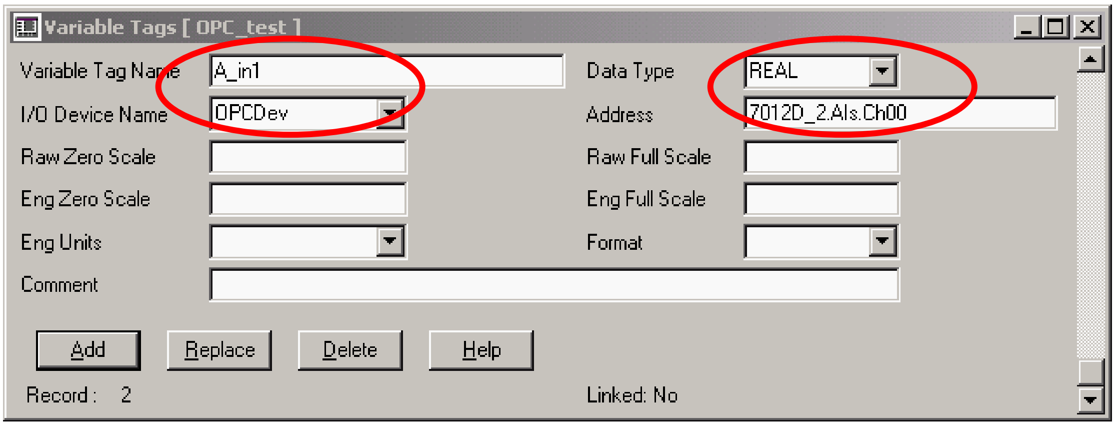
Variable Tags [OPC_test]

Variable Tag Name	A_out1	Data Type	REAL
I/O Device Name	OPCDev	Address	7021_1.AOs.Ch00
Raw Zero Scale		Raw Full Scale	
Eng Zero Scale		Eng Full Scale	
Eng Units		Format	
Comment			

Record : 1

Linked: No

Step18: Define another variable tag for analog input of the modules 7012D. The Variable Tag Name is "A_in1", the Data type is "REAL", the I/O Device Name is selected to be "OPCDev", and the Address is "7012D_2.AIs.Ch00". (See Step 6)



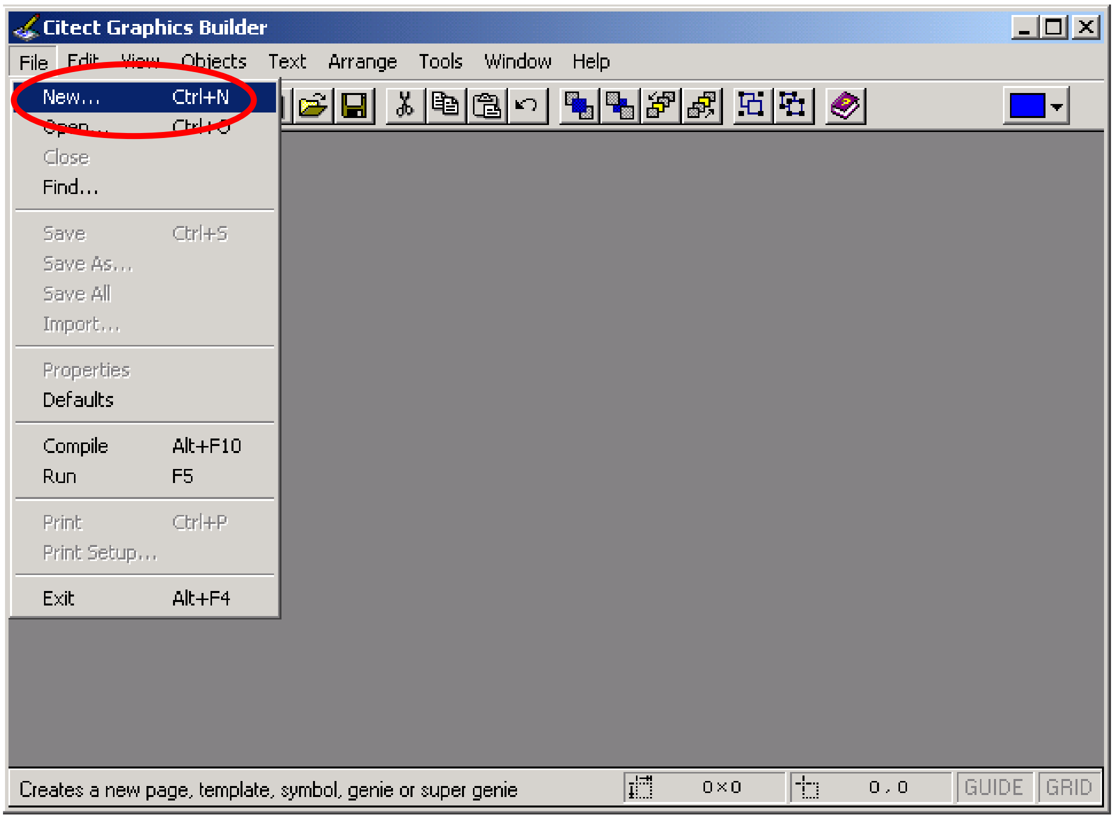
Variable Tags [OPC_test]

Variable Tag Name	A_in1	Data Type	REAL
I/O Device Name	OPCDev	Address	7012D_2.AIs.Ch00
Raw Zero Scale		Raw Full Scale	
Eng Zero Scale		Eng Full Scale	
Eng Units		Format	
Comment			

Record : 2

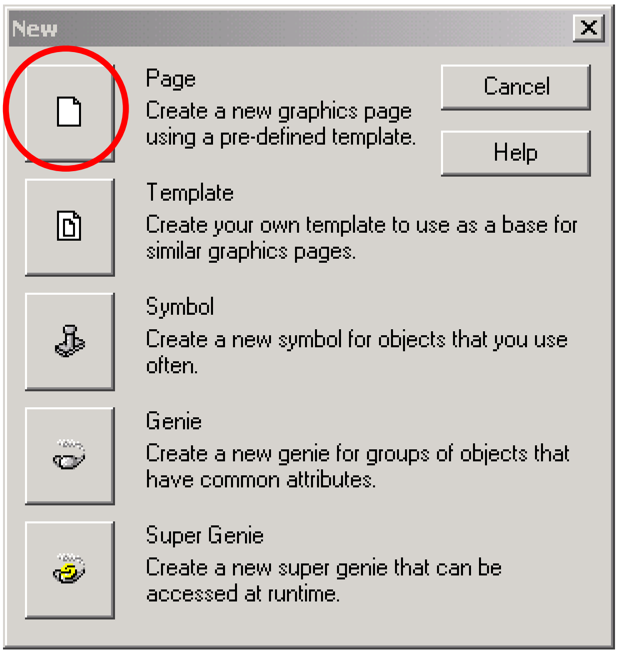
Linked: No

Step19: Create a new page in the "Citect Graphics Builder window".



Creates a new page, template, symbol, genie or super genie

Step20: Click the "page-marked button" to create a new page.



Page
Create a new graphics page using a pre-defined template.

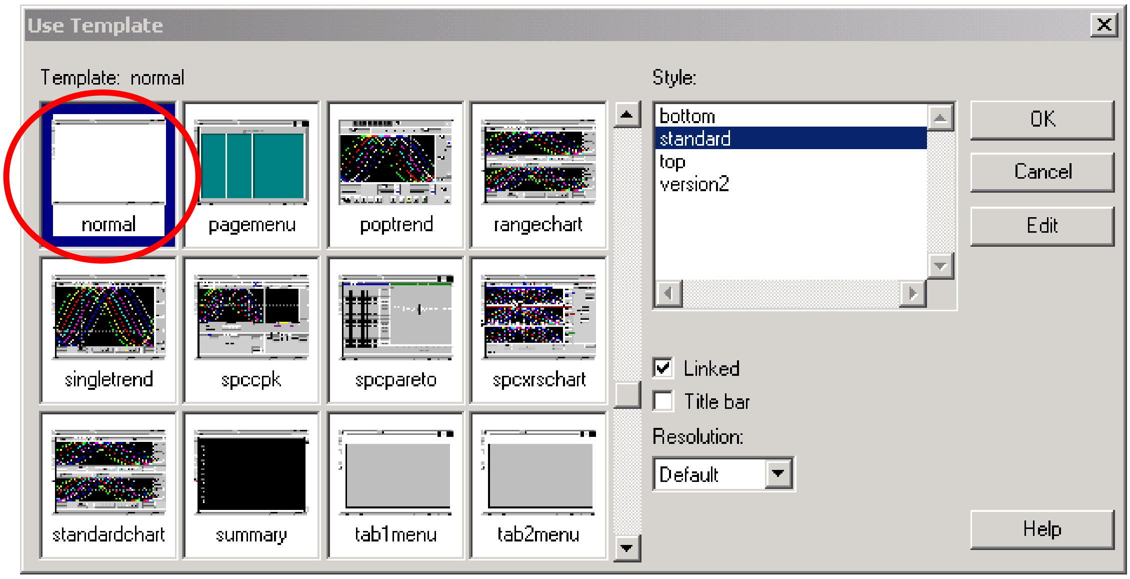
Template
Create your own template to use as a base for similar graphics pages.

Symbol
Create a new symbol for objects that you use often.

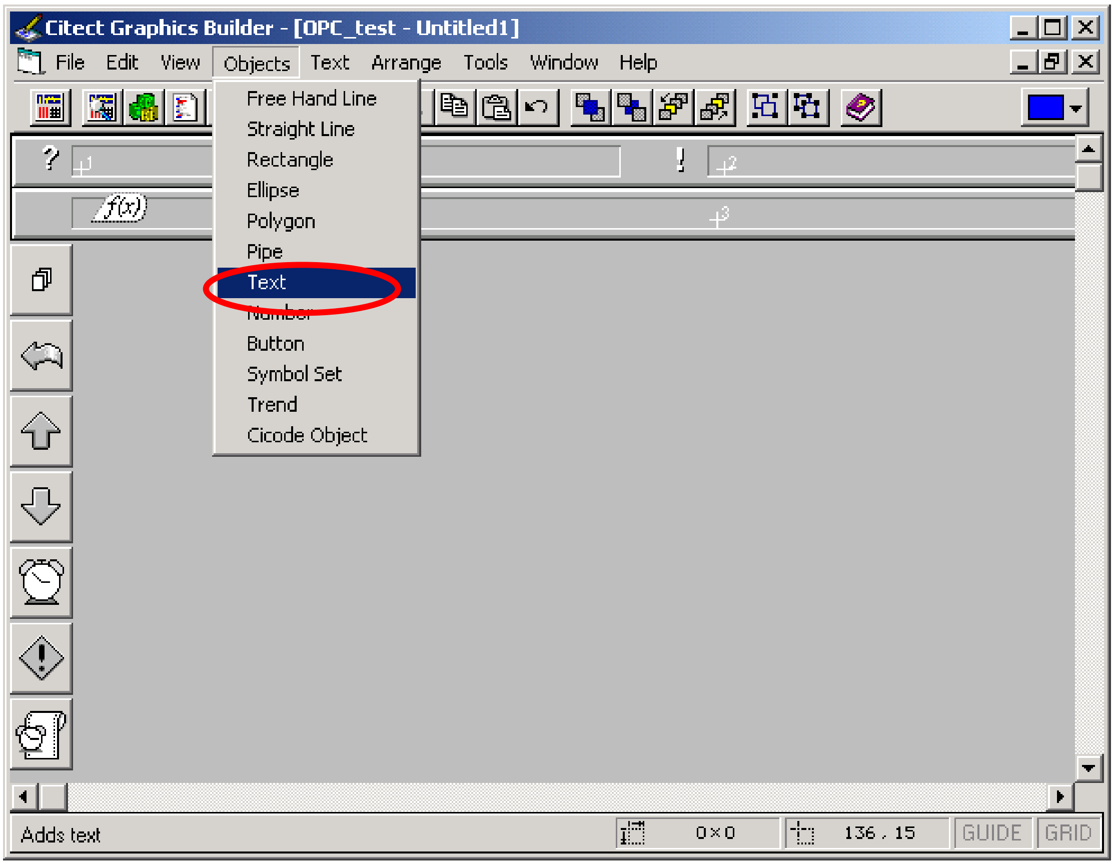
Genie
Create a new genie for groups of objects that have common attributes.

Super Genie
Create a new super genie that can be accessed at runtime.

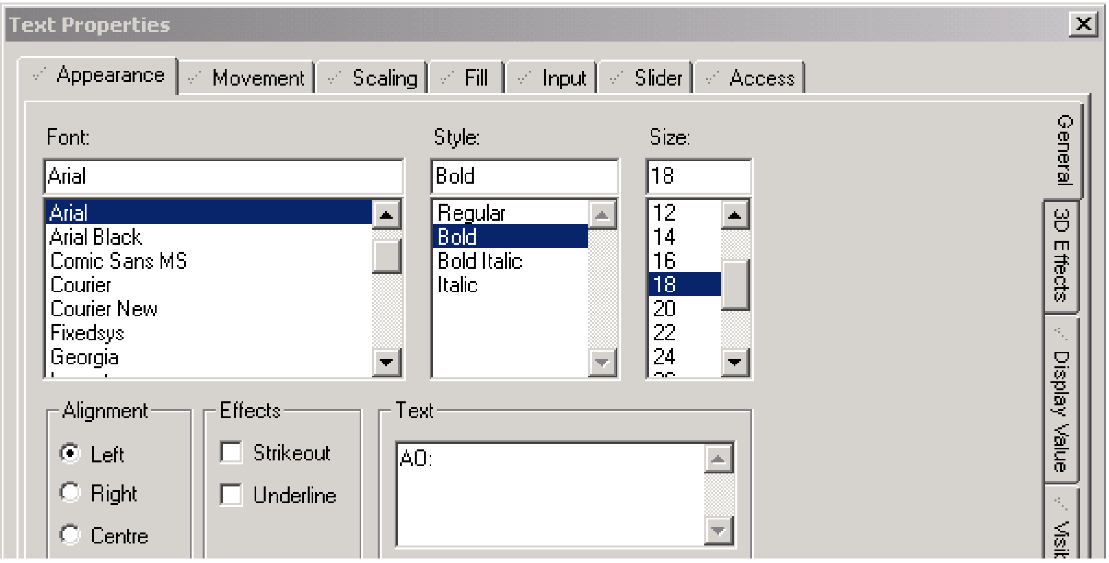
Step21: Select normal template to be the background and function of this page.



Step22: Select "Objects /Text " from "Citect Graphics Builder window menu" to insert a "Text Object" on the page.



Step23: Key-in the words "AO:" in the Text object. Then, left click to put the Text object on the page and set "Appearance parameters" of Text object.



Font: Arial

Arial
Arial Black
Comic Sans MS
Courier
Courier New
Fixedsys
Georgia

Style: Bold

Regular
Bold
Bold Italic
Italic

Size: 18

12
14
16
18
20
22
24

Alignment

● Left
○ Right
○ Centre

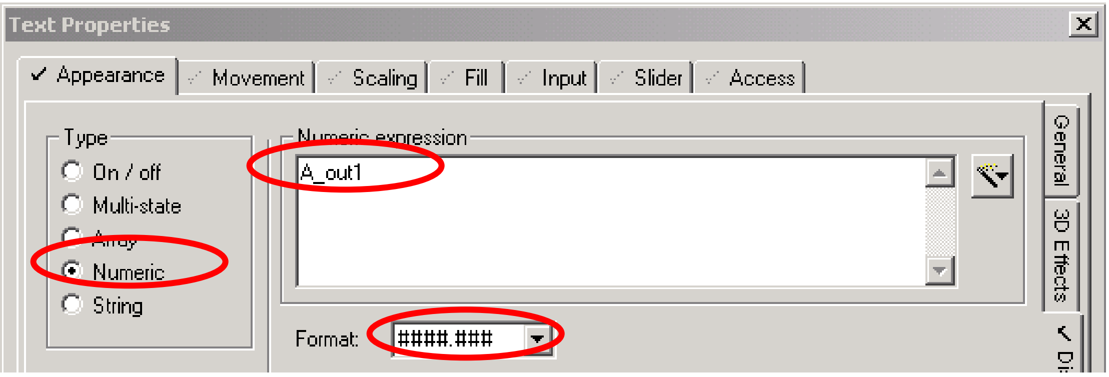
Effects

□ Strikeout
□ Underline

Text

AO:

Step24: Select "Objects/Number" from "Citect Graphics Builder window menu" to insert a "Number Object" on the page. Left Click to put Number object on the page. Then, set "Appearance attributes" and "Input attributes" for this Number object.



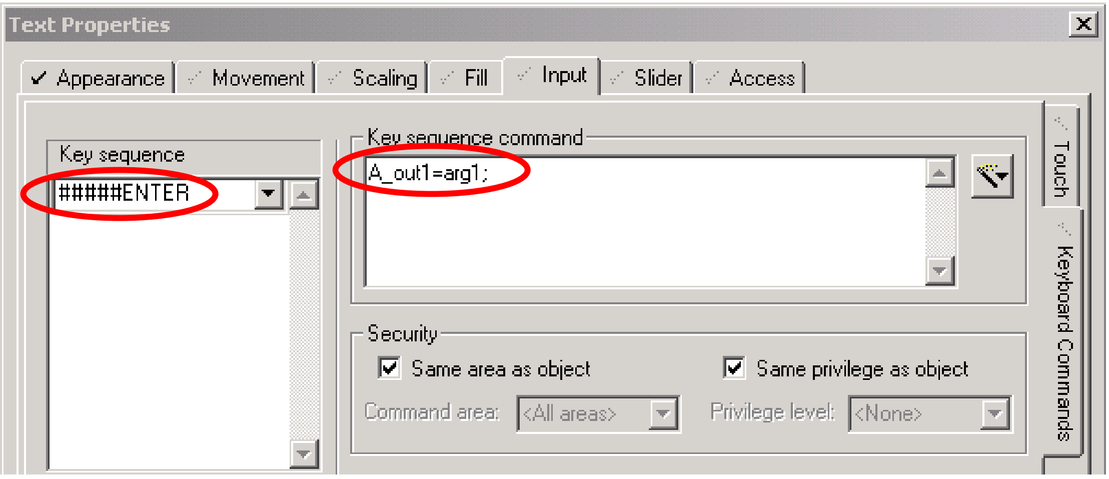
Type

○ On / off
○ Multi-state
○ Array
● Numeric
○ String

Numeric expression

A_out1

Format: ####.###



Key sequence

#####ENTER

Key sequence command

A_out1=arg1;

Security

☑ Same area as object

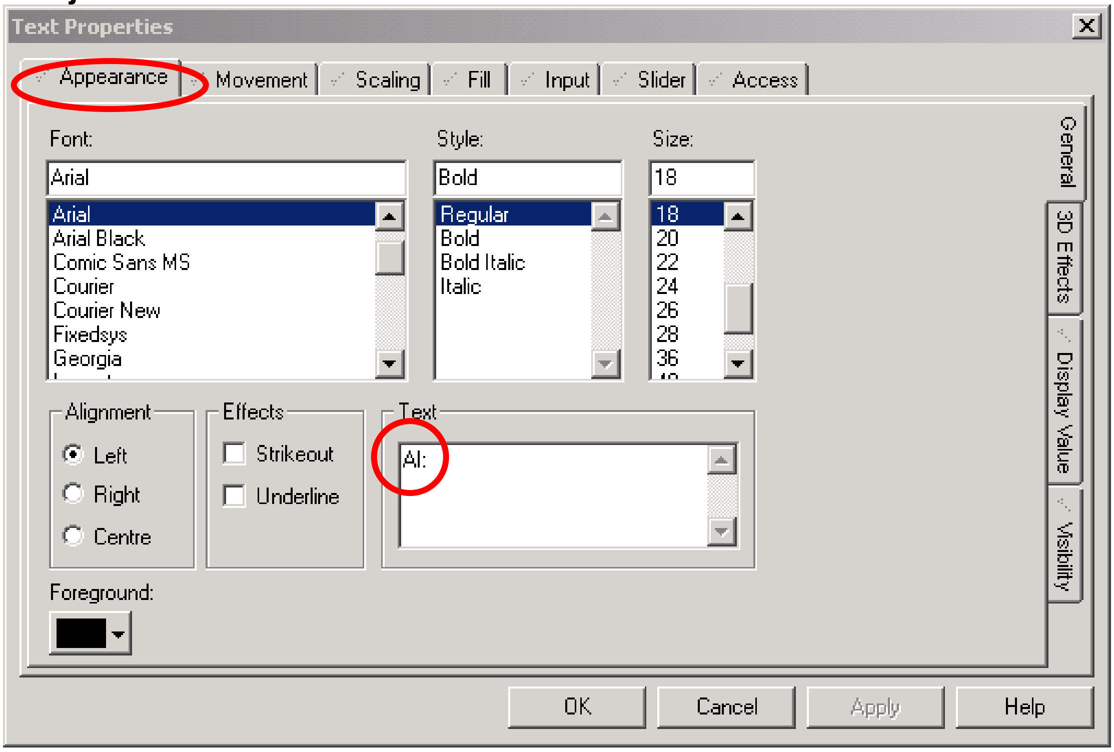
☑ Same privilege as object

Command area: <All areas>

Privilege level: <None>

Step25: Set another Text object and Number object by the same way, and the "Appearance attributes" of Text and Number object are showed below.

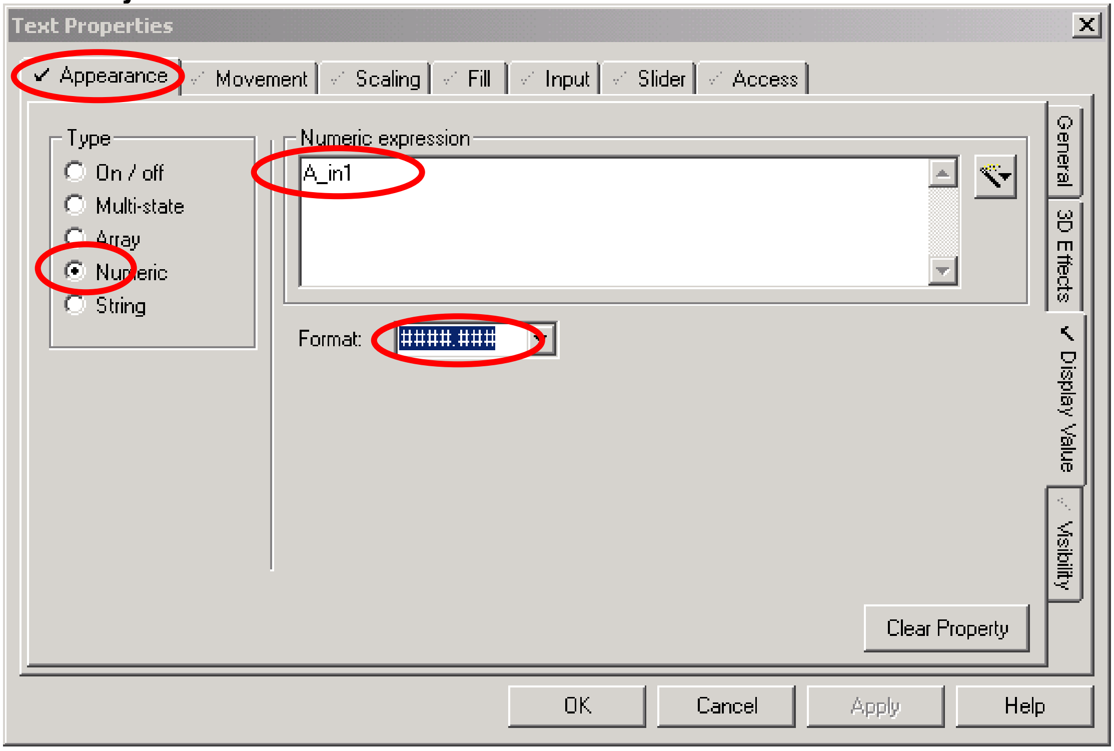
Text object:



Text Properties

Appearance tab: Font: Arial; Style: Bold; Size: 18; Alignment: ● Left ○ Right ○ Centre; Effects: □ Strikeout □ Underline; Text: AI:; Foreground: ■

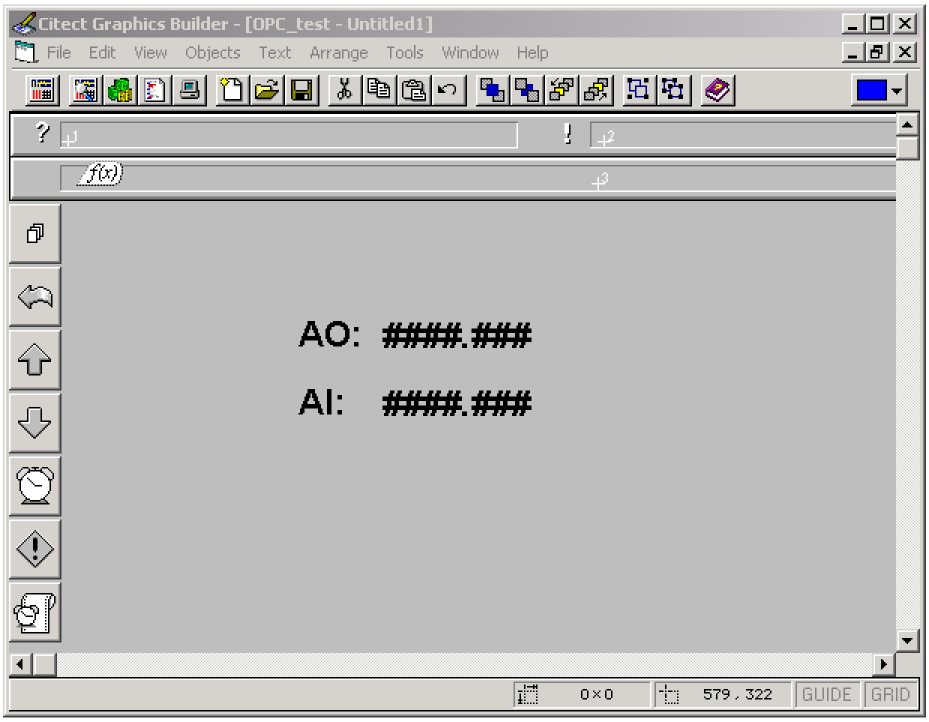
Number object:



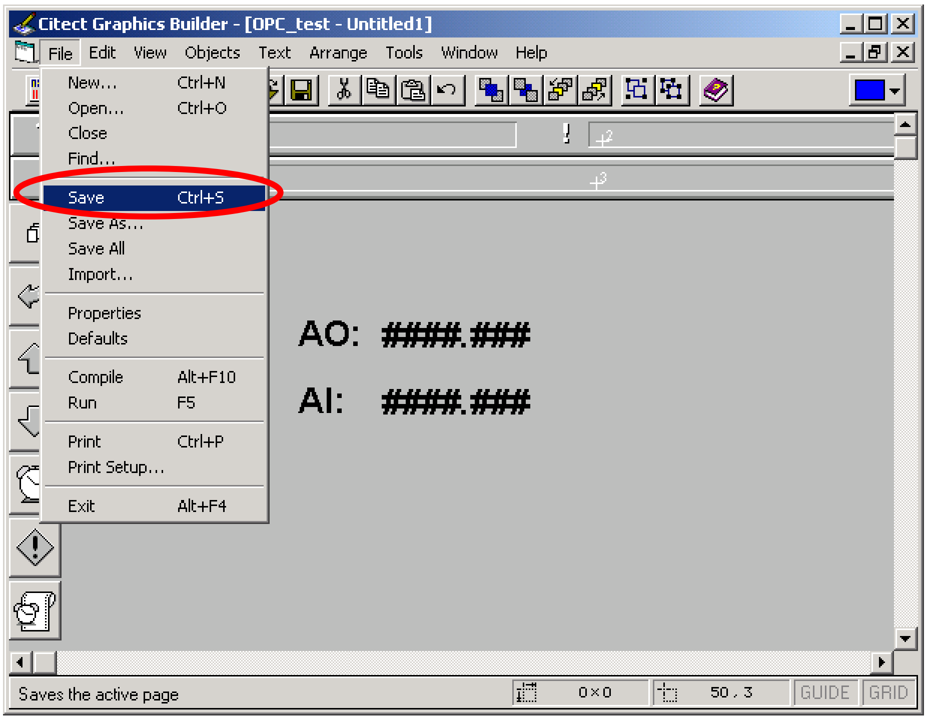
Text Properties

Appearance tab: Type: ○ On / off ○ Multi-state ○ Array ● Numeric ○ String; Numeric expression: A_in1; Format: #####.###

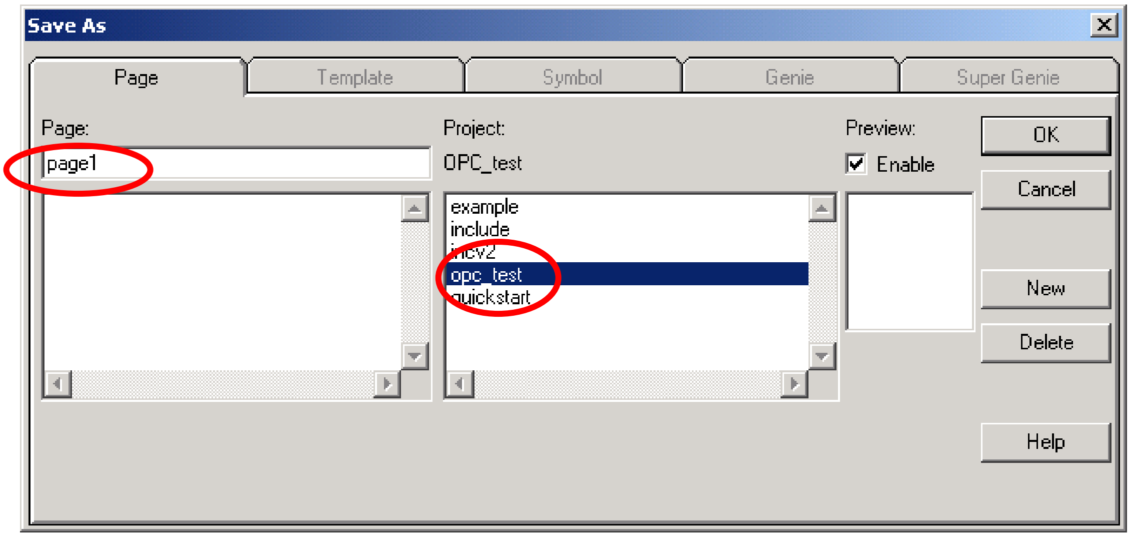
Step26: When finish the all object and attribute setting, the page is looked like as one, which shows below.


Citect Graphics Builder - [OPC_test - Untitled1]
File Edit View Objects Text Arrange Tools Window Help
AO: ####.###
AI: ####.###
0×0
579, 322
GUIDE
GRID

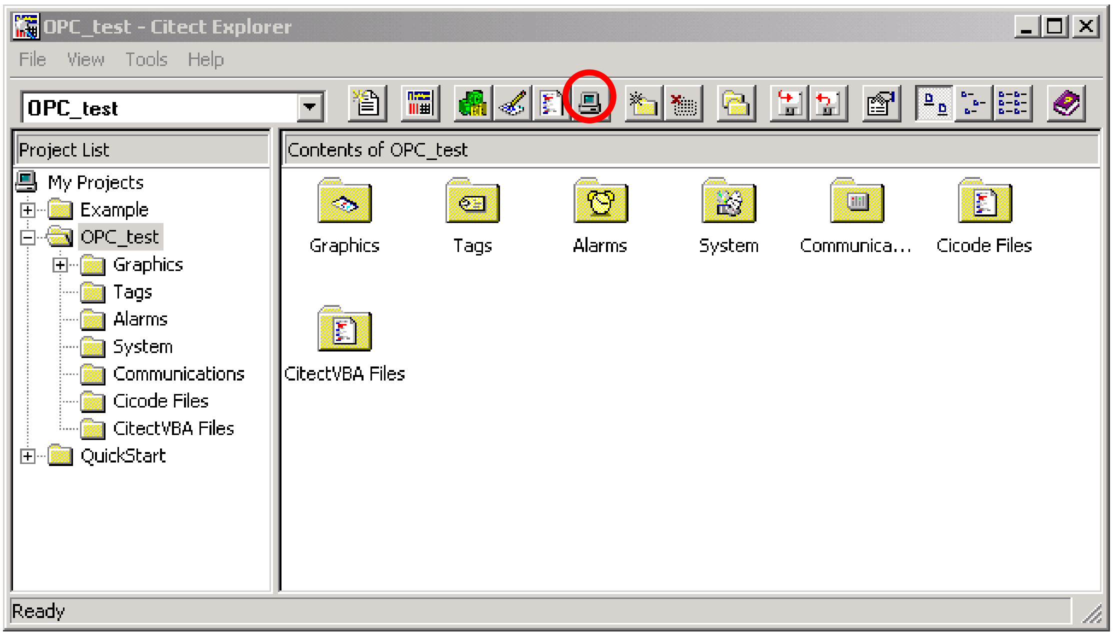
Step27: Select "File/Save " from "Citect Graphics Builder window menu" to save this page.



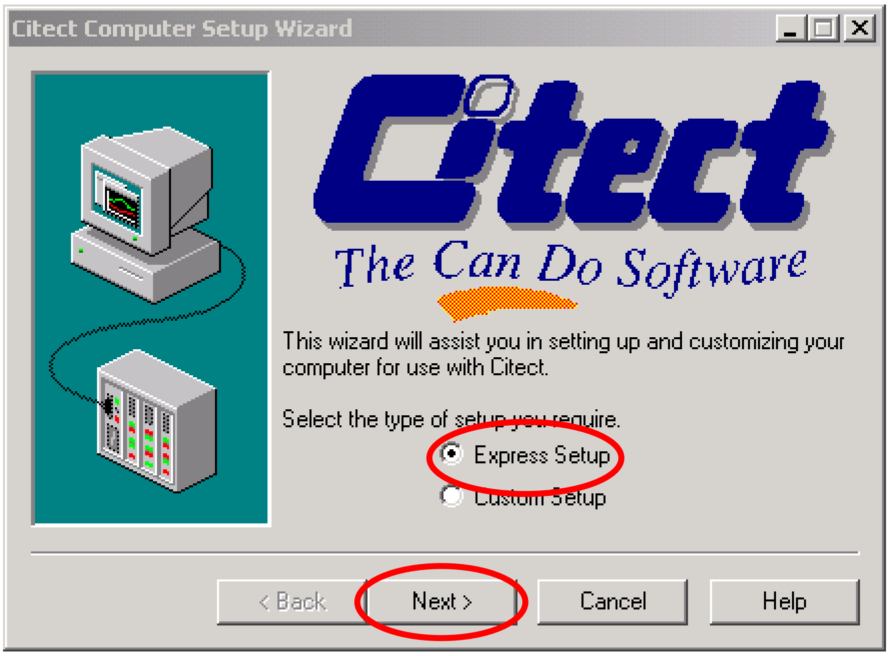
Step28: Fill the name of this page and save it under OPC_test project.



Step29: Left click the "computer-marked button" to define the role of this computer.



Step30: Start up the Citect Computer Setup Wizard.



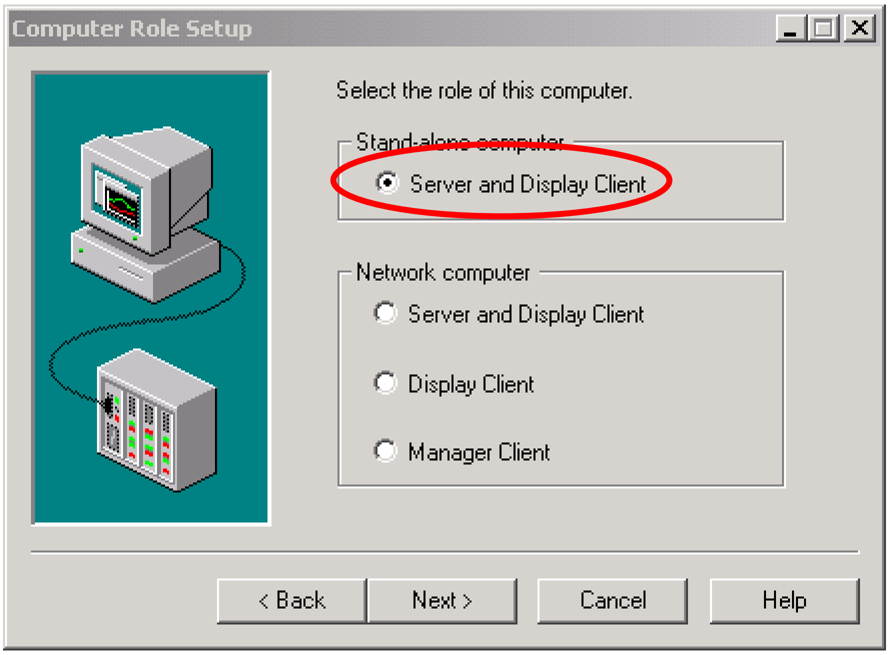
This wizard will assist you in setting up and customizing your computer for use with Citect.

Select the type of setup you require.

● Express Setup

○ Custom Setup

Step31: Select the "Stand-alone computer" item to be the role of this computer.



Select the role of this computer.

Stand-alone computer

● Server and Display Client

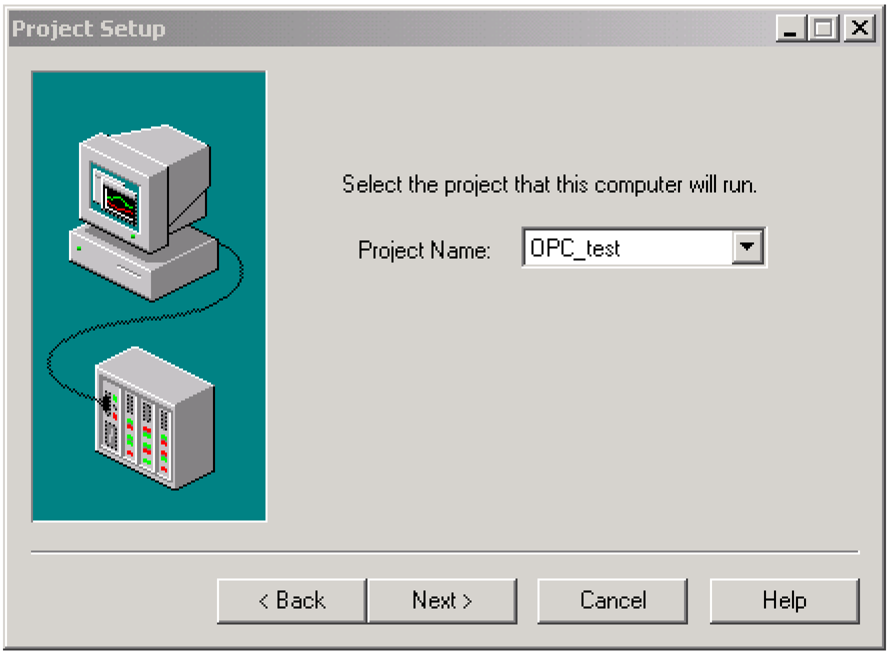
Network computer

○ Server and Display Client

○ Display Client

○ Manager Client

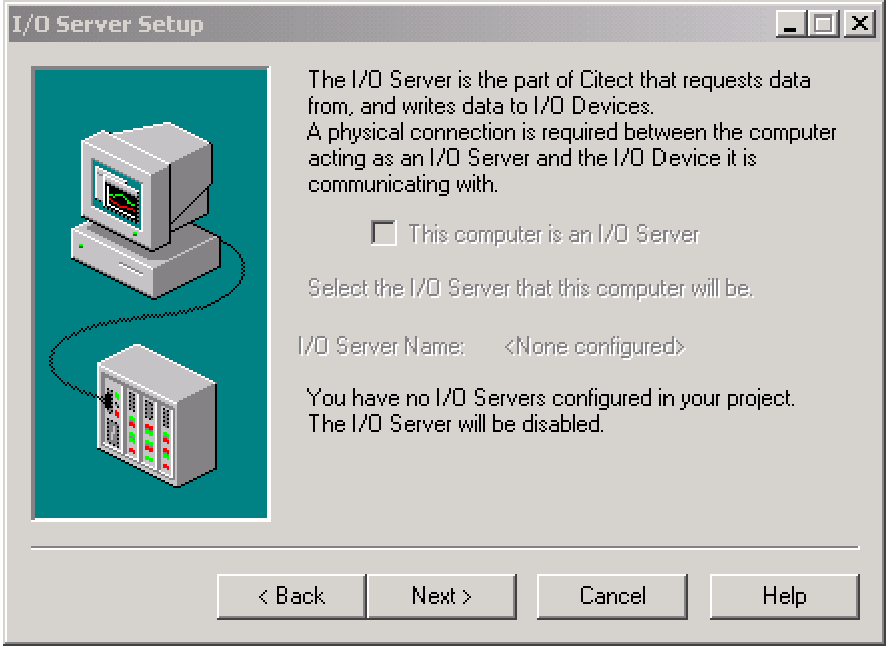
Step32: Select the project "OPC_test" to be the project that this computer will run.



Select the project that this computer will run.

Project Name: OPC_test

Step33: Click the "Next button" to next step.



The I/O Server is the part of Citect that requests data from, and writes data to I/O Devices.
A physical connection is required between the computer acting as an I/O Server and the I/O Device it is communicating with.

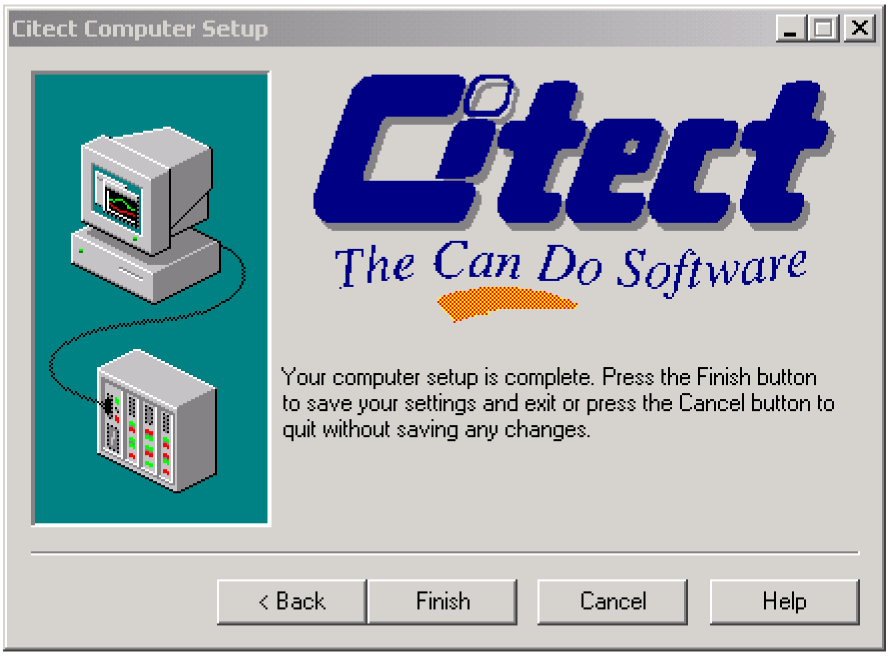
□ This computer is an I/O Server

Select the I/O Server that this computer will be.

I/O Server Name: <None configured>

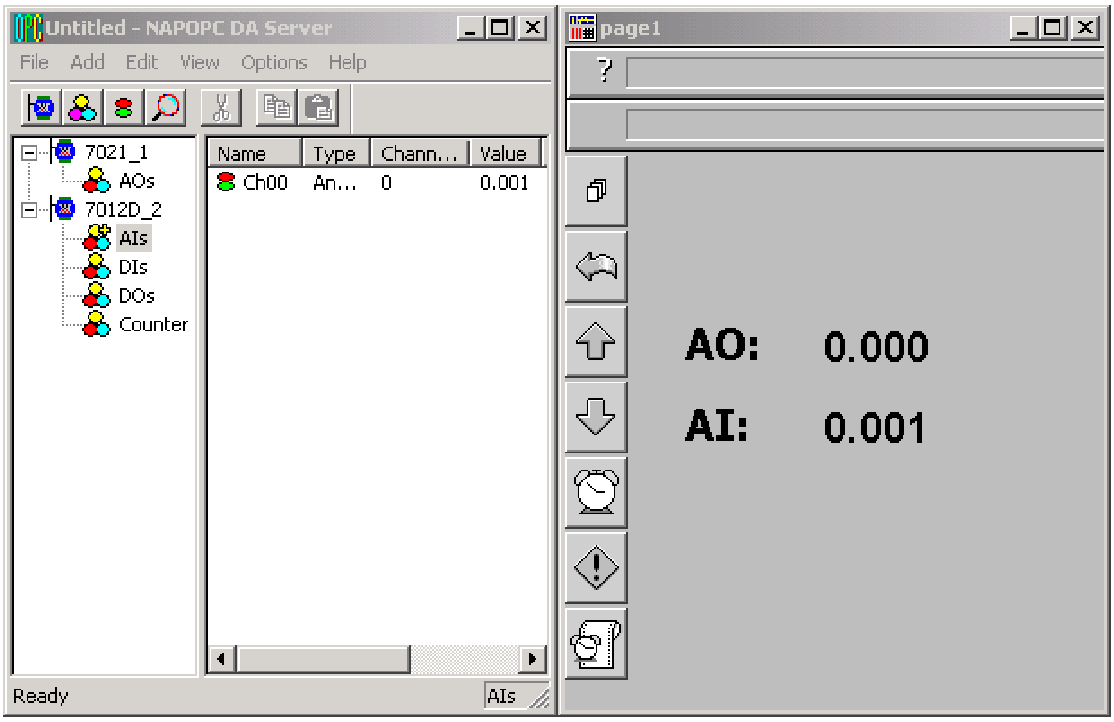
You have no I/O Servers configured in your project.
The I/O Server will be disabled.

Step34: Left click the "Finish button" to finish the computer setup.



Your computer setup is complete. Press the Finish button to save your settings and exit or press the Cancel button to quit without saving any changes.

Step35: Select OPC_test project and press "F5" to run this project. Compare the NAPOPC_XPE Server monitor and CitectSCADA runtime window.

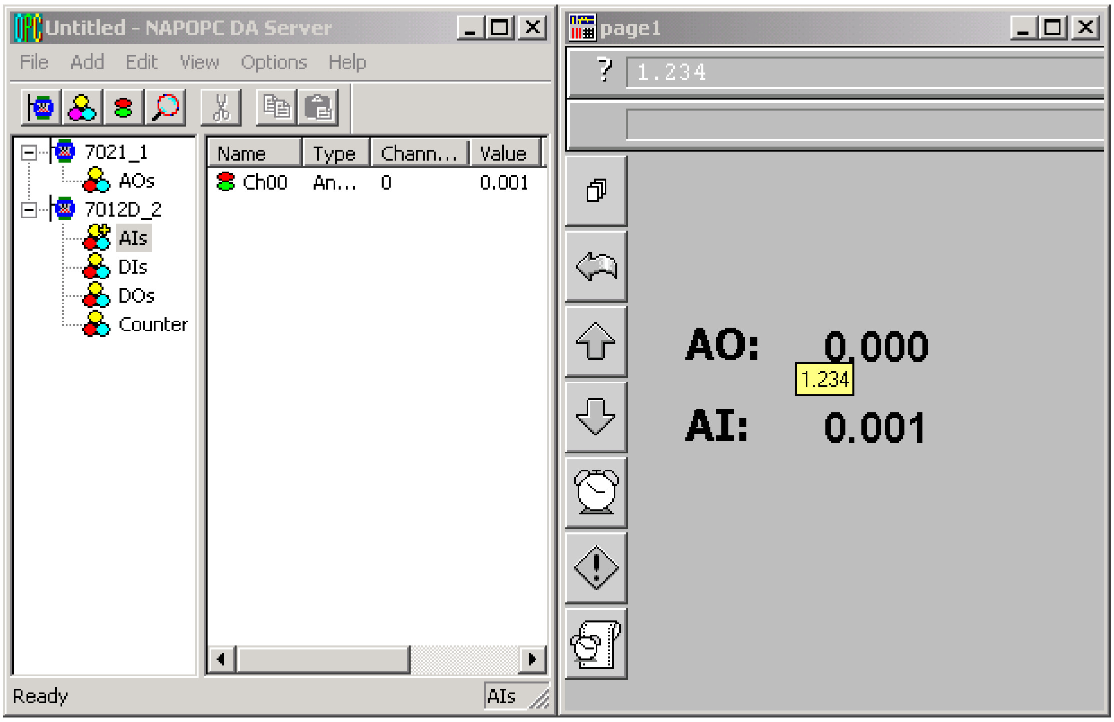


Name	Type	Chann...	Value
Ch00	An...	0	0.001

AO: 0.000

AI: 0.001

Step36: Key-in the value "1.234" to output a voltage via the module 7021.



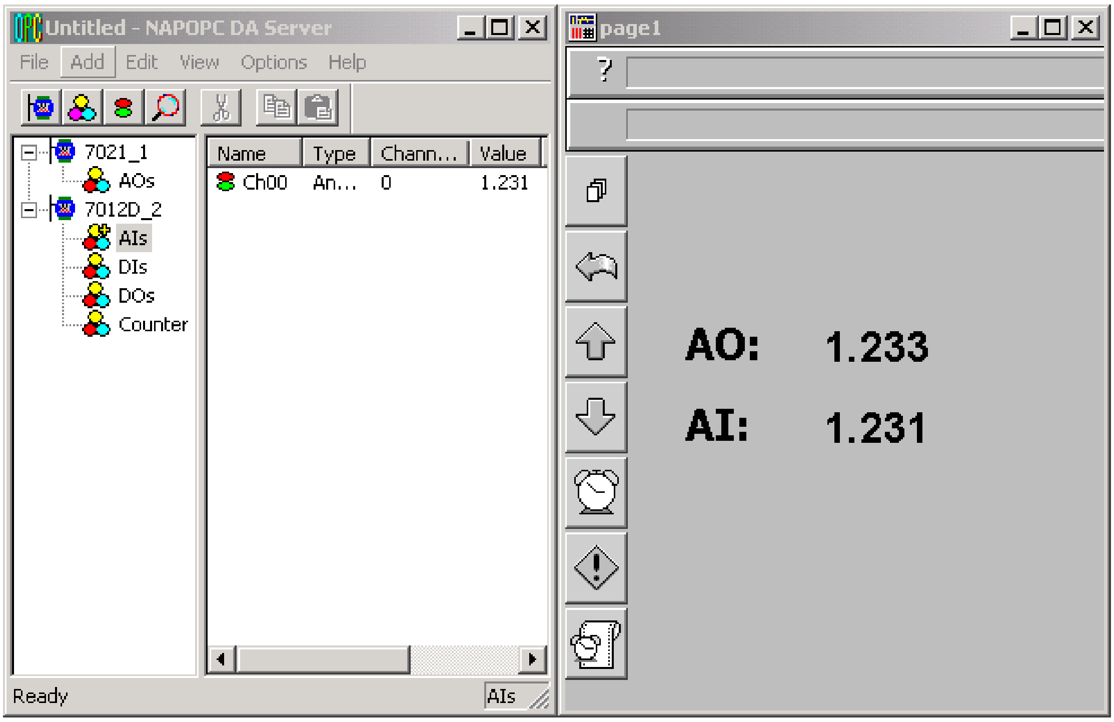
Name	Type	Chann...	Value
Ch00	An...	0	0.001

1.234

AO: 0.000

AI: 0.001

Step37: Both NAPOPC_XPE Server monitor and CitectSCADA runtime window show the analog input of the modules 7012D is 1.231V.

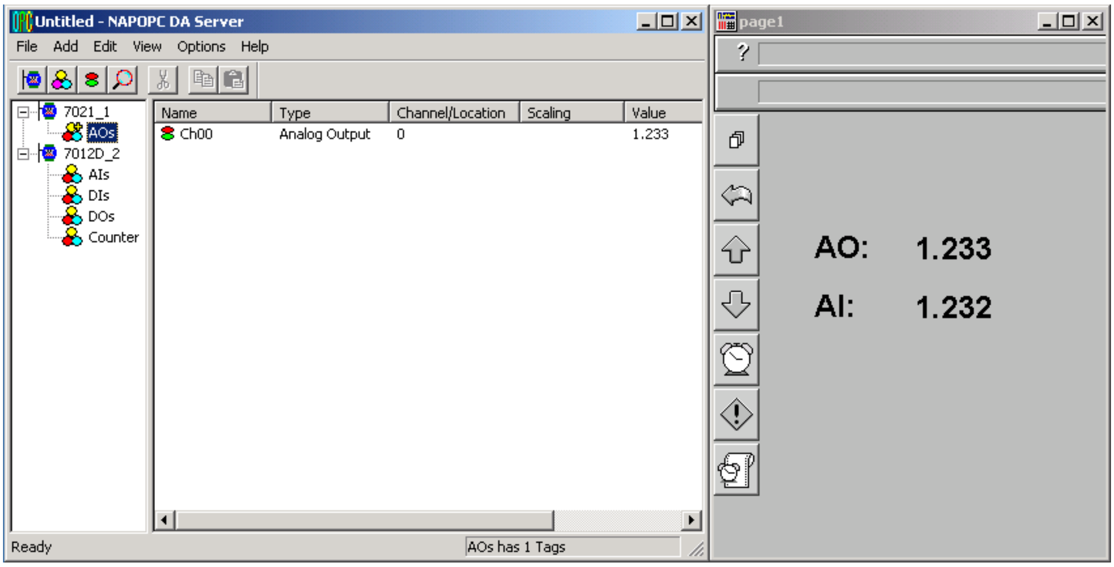


Name	Type	Chann...	Value
Ch00	An...	0	1.231

AO: 1.233

AI: 1.231

Step38: Check if the analog output value of the modules 7021 that showed in the NAPOPC_XPE Server monitor is 1.234V.



Name	Type	Channel/Location	Scaling	Value
Ch00	Analog Output	0		1.233

AO: 1.233

AI: 1.232

4 Remote Accessing

OPC Client has two ways to access the OPC Server. One is called "Local Accessing", and the other is called "Remote Accessing". If the OPC Client and the OPC Server are at the same computer, we said this kind of architecture is "Local Accessing". In other words, if the OPC Client should access OPC Server through a network, we said this kind of architecture is "Remote Accessing".

The following figure shows the integrated architecture including "Local Accessing" and "Remote Accessing". At the real Process Industry, the two ways are often used at the same time. At the Process Management Layer, we often use "Local Accessing" architecture to monitor and control manufacturing processes. At the Business Management Layer, we just set up the OPC Client to collect the process information from the Process Management Layer. If you just want to construct the "Local Accessing" architecture, you do not need to read this chapter. If you want to construct the "Remote Accessing" architecture, you have to know how to set up the DCOM between OPC Client and OPC Server.

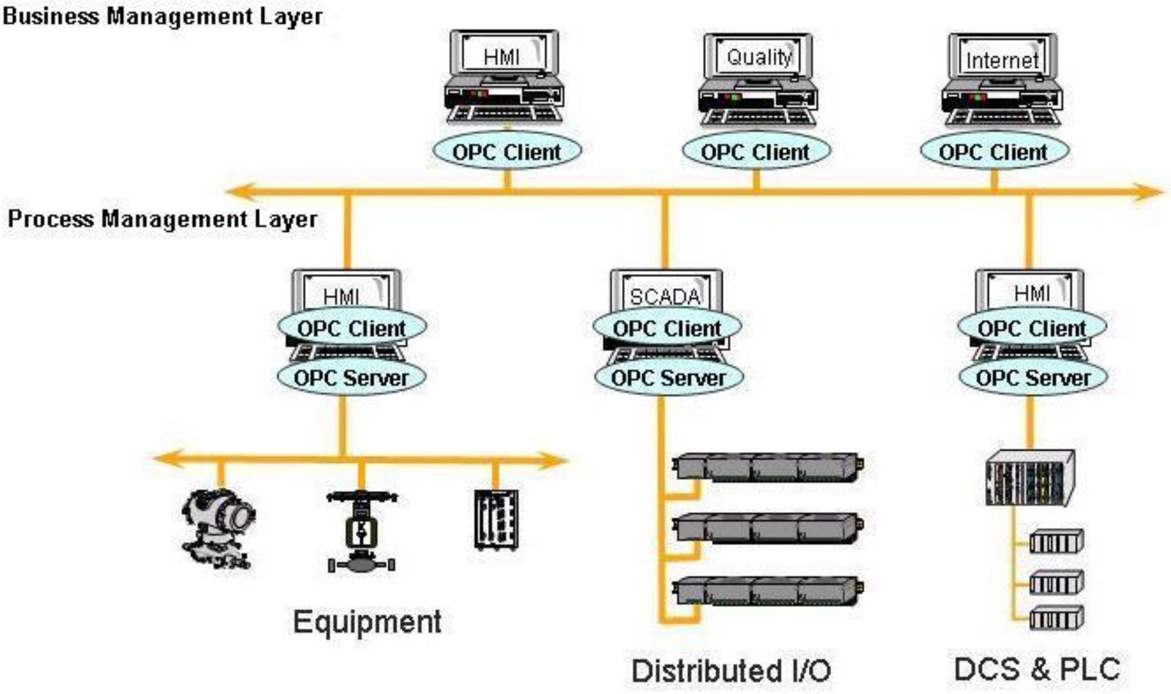


Figure 4-0-1 Local access and Remote access architecture.

4.1 System Requirement

To access a remote OPC server over a network, it is required to enable the DCOM mechanism on both stations, where the client and server are resided.

It is not possible to launch a secure process on a Windows 95 computer from a client computer. All processes in Windows 95 run in the security context of the currently logged-on user; therefore, DCOM on Windows 95 does not support remote activation. A server application on a Windows 95 computer will have to be launched manually or by some other mechanism to be accessed by a client application on another computer. Consequently, the "DefaultLaunchPermissions" and "LaunchPermissions" registry values have no affect on Windows 95.

Platform	Does the platform support the DCOM?
Windows 95	No. Users need to download and install the DCOM95.EXE and DCM95CFG.EXE from Microsoft's web site to enable the remote access.
Windows 98	Yes. Windows 98 supports the DCOM mechanism. It is recommended to upgrade to the newest version of DCOM98. The newest DCOM98 is also available at Microsoft's web site.
Windows NT 4.0	Yes. Windows NT 4.0 supports the DCOM mechanism. It is recommended to upgrade to the newest Service Pack for Windows NT 4.0 (Service Pack 3 or newer one).
Windows 2000	Yes. Windows 2000 supports the DCOM mechanism.
Windows XP	Yes. Windows XP supports the DCOM mechanism.

4.2 Configuring DCOM

Before making changes, register the server application in the registry of both the client and server computers. This may involve either running the server applications setup program or running the server application, then shutting it down on both computers. The server application does not need to reside on the client computer.

If the server uses custom interfaces, the marshaling code must be installed on the client and server computers. Automation servers that support "vtbl-binding" must install their type libraries on the client and server computers. Automation servers that do not support "vtbl-binding" do not need to install their type libraries on the client computer.

After changing the registry, run the client application on the client computer. The DCOM looks at the server application registry entries on the client computer and determines the name of the server computer. It will then connect to the server computer, use the server computer registry to determine the location of the server application, and start the server application on that computer.

You can change the registry with the DCOMCnfg.exe tool, the OLE Viewer tool, or manually. For more information on using OLE Viewer or manual changes, please refer to the “Q158582, HOWTO: Configure a Non-DCOM Server and Client to Use DCOM” article on Microsoft’s web site. For more information on using DCOMCnfg.exe to configure the DCOM, please refer to “Inside Distributed COM", written by Guy Eddon and Henry Eddon in 1998 for Microsoft Press.

This section shows you how to configure the DCOM status with DCOMCnfg.exe graphic-driven utility (can be found in the Windows NT system32 folder or in the Windows95/98 system folder) on the client and server computer.

The following table shows four combinations of DCOM settings related to XPAC. You can see XPAC and PC can be client site and server site with each other, but WinPAC only can be server site against XPAC. The limitation is due to DCOM security. We only choose Windows XP for example to set up DCOM because there are too many kinds of OS on PC. You can use other Microsoft desktop operation system on our PC.

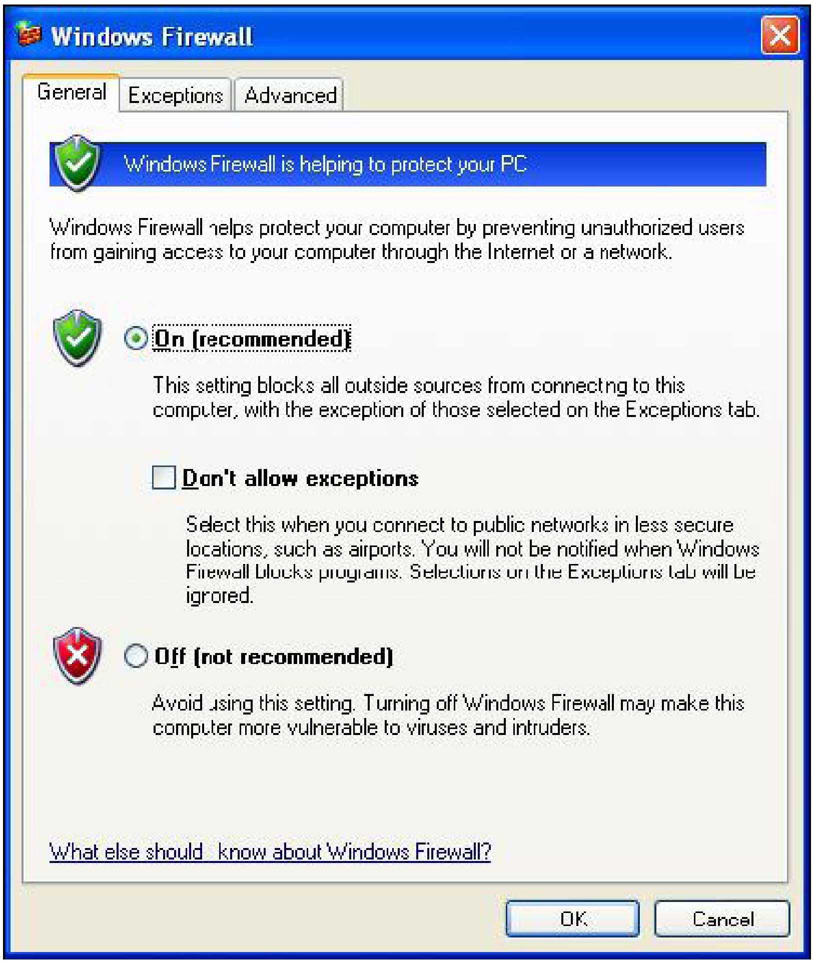
Client Site	Server Site
PC(NAPOPC_ST Server)	XPAC(NAPOPC_XPE Server)
XPAC(NAPOPC_XPE Server)	XPAC(NAPOPC_XPE Server)
XPAC(NAPOPC_XPE Server)	PC(NAPOPC_ST Server)
XPAC(NAPOPC_XPE Server)	WinPAC(NAPOPC_CE5 Server)

4.2.1 Configuring On the Server Site (XPAC)

Configuring the Firewall

Step1: By default the windows firewall is set to "On". This setting is recommended by Microsoft and by OPC to give your machine the highest possible protection. For trouble shooting, you may wish to temporarily turn off the firewall to prove or disprove that the firewall configuration is the source of any communication failure.

Note: It may be appropriate to permanently turn off the firewall if the machine is sufficiently protected behind a corporate firewall. When turned off, the individual firewall settings outlined here need not be performed to allow OPC communication.



Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

● On (recommended)

This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.

□ Don't allow exceptions

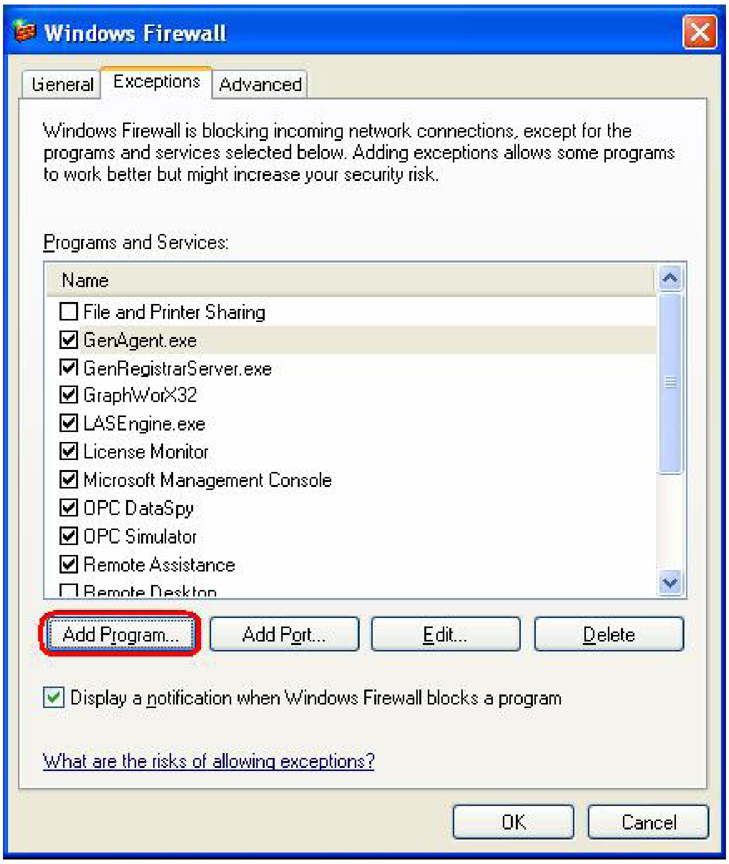
Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.

○ Off (not recommended)

Avoid using this setting. Turning off Windows Firewall may make this computer more vulnerable to viruses and intruders.

What else should I know about Windows Firewall?

Step 2: Select the .Exceptions tab and add all OPC Clients and Servers to the exception list. Also add Microsoft Management Console (used by the DCOM configuration utility in the next section) and the OPC utility OPCEnum.exe found in the Windows\System32 directory.



Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

Programs and Services:

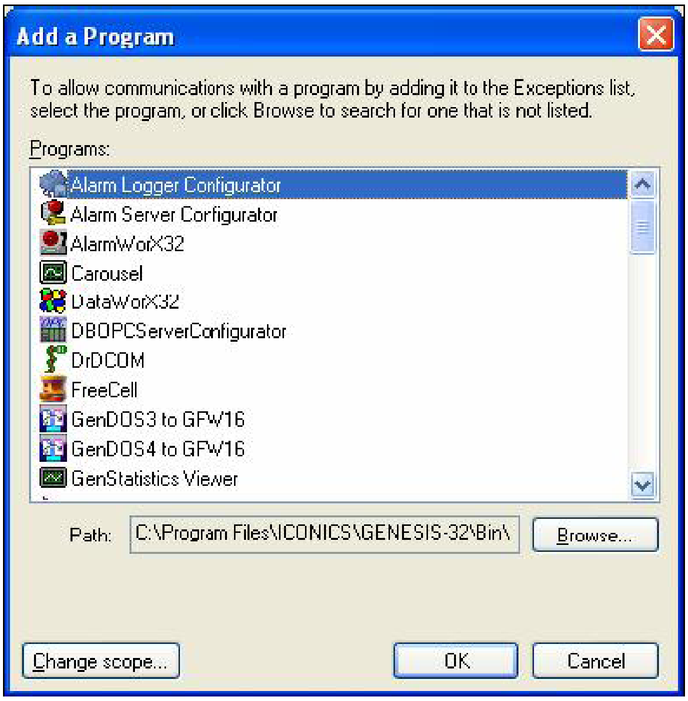
Name

- □ File and Printer Sharing
- ☑ GenAgent.exe
- ☑ GenRegistrarServer.exe
- ☑ GraphWorX32
- ☑ LASEngine.exe
- ☑ License Monitor
- ☑ Microsoft Management Console
- ☑ OPC DataSpy
- ☑ OPC Simulator
- ☑ Remote Assistance
- □ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

In the Add a Program dialog, there is a listing of most applications on the machine, but note that not all of them show up on this list. Use the "Browse" button to find other executables installed on the computer.

Note: Only EXE files are added to the exceptions list. For in-process OPC Servers and Clients (DLLs and OCXs) you will need to add the EXE applications that call them to the list instead.



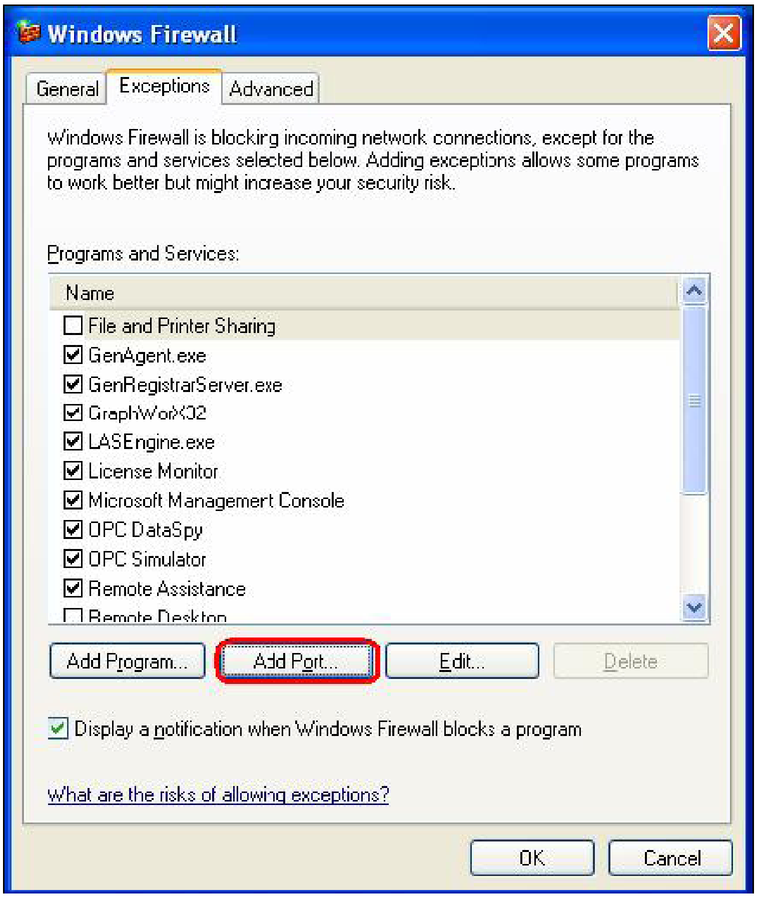
To allow communications with a program by adding it to the Exceptions list, select the program, or click Browse to search for one that is not listed.

Programs:

- Alarm Logger Configurator
- Alarm Server Configurator
- AlarmWorX32
- Carousel
- DataWorX32
- DBOPCServerConfigurator
- DrDCOM
- FreeCell
- GenDOS3 to GFW16
- GenDOS4 to GFW16
- GenStatistics Viewer

Path: C:\Program Files\ICONICS\GENESIS-32\Bin\

Step 3: Add TCP port 135 as it is needed to initiate DCOM communications, and allow for incoming echo requests. In the Exceptions tab of the Windows Firewall, click on Add Port.



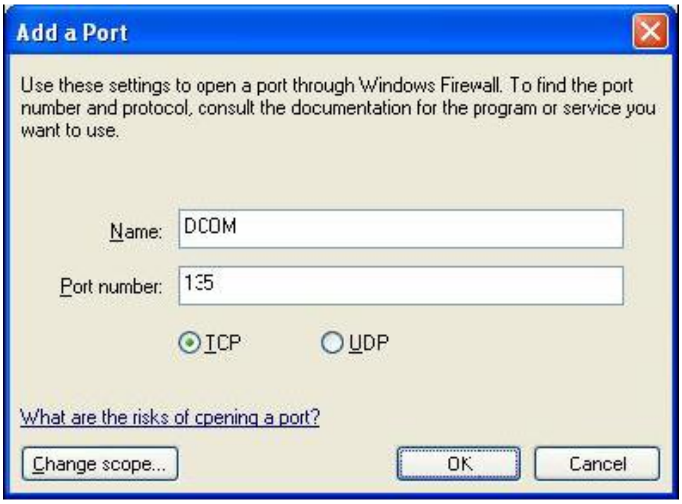
Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

Programs and Services:

Name
- ☐ File and Printer Sharing
- ☑ GenAgent.exe
- ☑ GenRegistrarServer.exe
- ☑ GraphWorX32
- ☑ LASEngine.exe
- ☑ License Monitor
- ☑ Microsoft Management Console
- ☑ OPC DataSpy
- ☑ OPC Simulator
- ☑ Remote Assistance
- ☐ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

In the Add a Port dialog, fill out the fields as follows:
Name: DCOM
Port number: 135
Choose the TCP radio button



Use these settings to open a port through Windows Firewall. To find the port number and protocol, consult the documentation for the program or service you want to use.

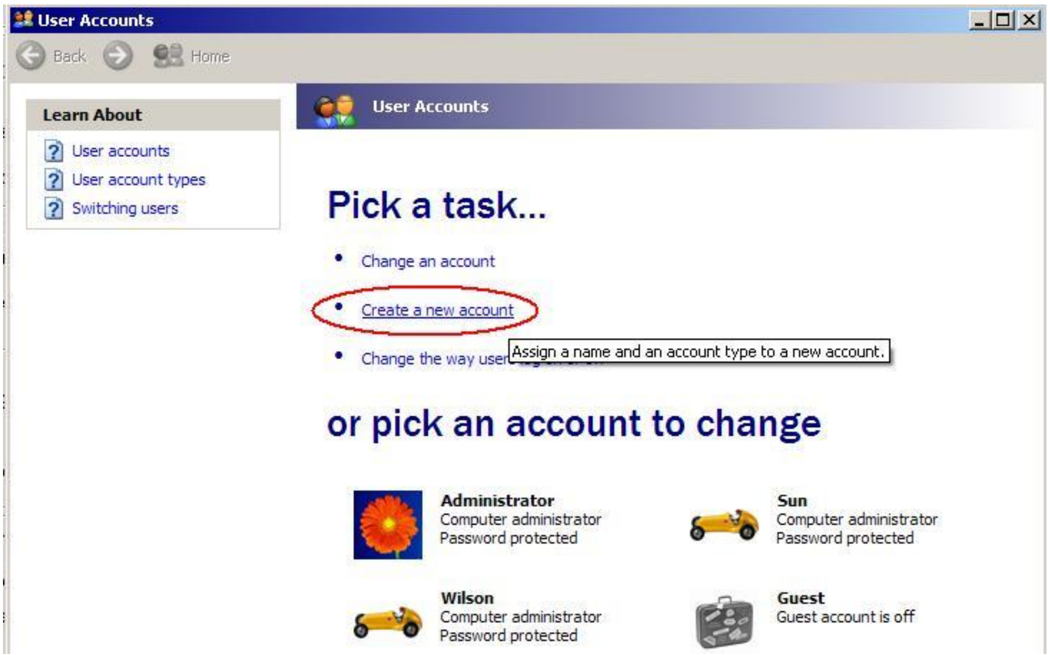
Name: DCOM

Port number: 135

● TCP ○ UDP

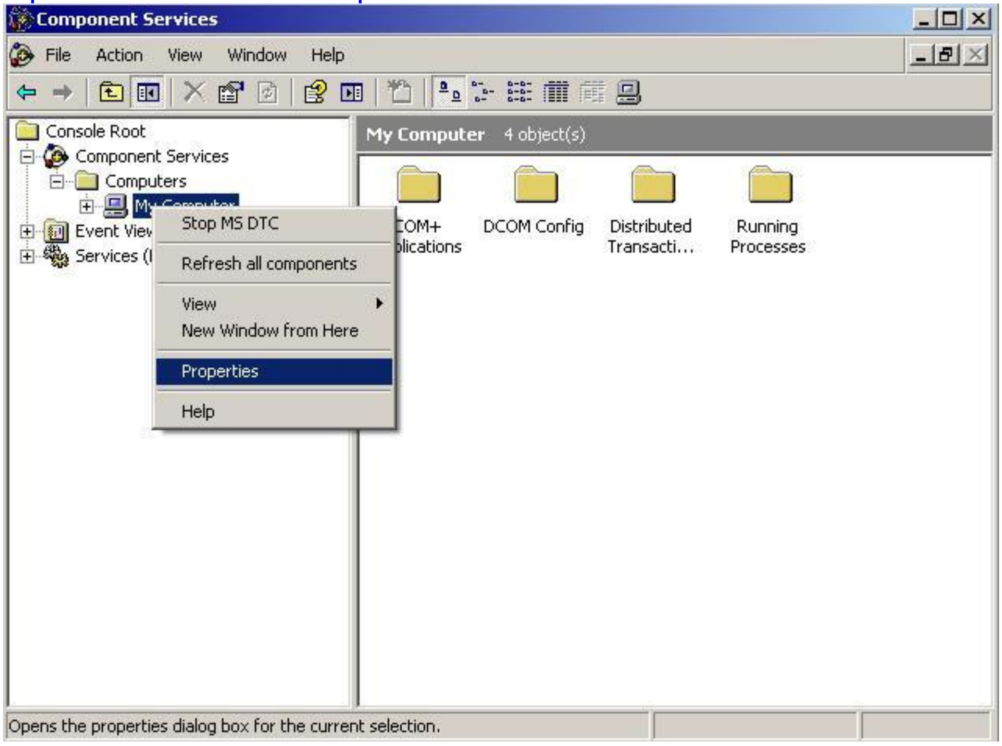
Creating the Account

Step 1: Create a account which must be the same with the account of client site.



Configuring DCOM

Step 1: Run the dcomcnfg.exe program to launch component services. Right clieck "My Computer" and choose "Properties".

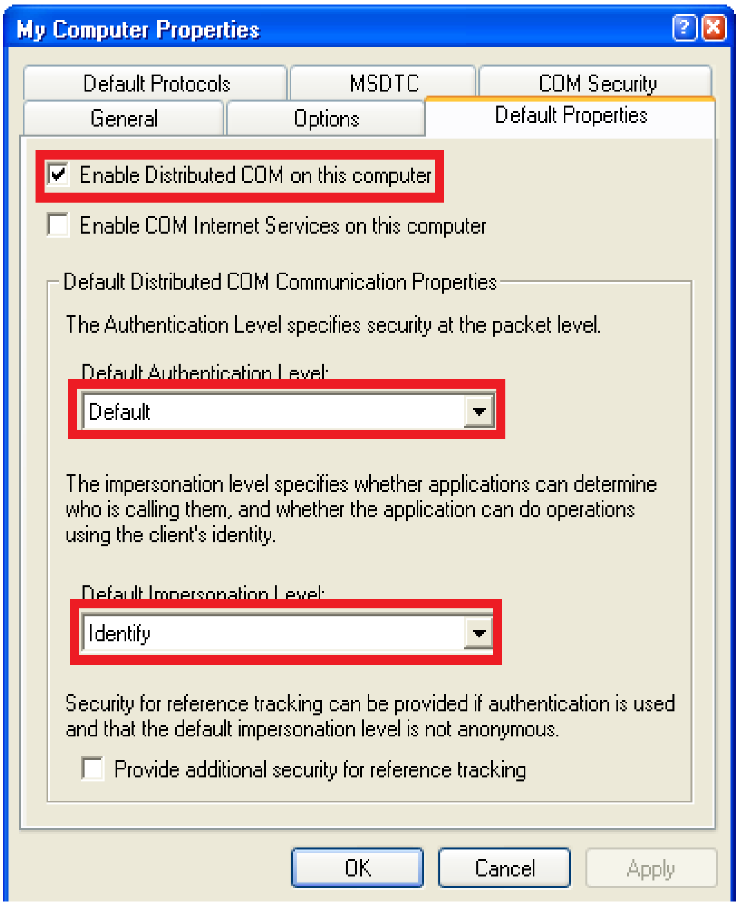


Opens the properties dialog box for the current selection.

Step 2: Select the "Default Properties" tab page.

Step 3: Use the following settings:

Field Name	Set to
Enable Distributed COM on this computer	Checked
Default Authentication Level:	Default
Default Impersonation Level:	Identify



☑ Enable Distributed COM on this computer

□ Enable COM Internet Services on this computer

Default Distributed COM Communication Properties

The Authentication Level specifies security at the packet level.

Default Authentication Level: Default

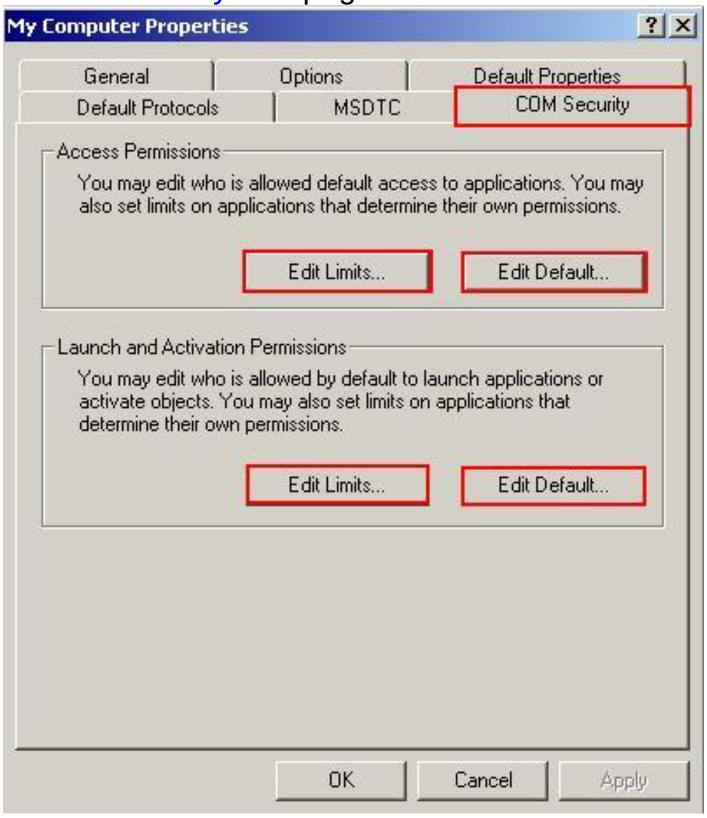
The impersonation level specifies whether applications can determine who is calling them, and whether the application can do operations using the client's identity.

Default Impersonation Level: Identify

Security for reference tracking can be provided if authentication is used and that the default impersonation level is not anonymous.

□ Provide additional security for reference tracking

Step 4: Select the "COM Security" tab page.



My Computer Properties

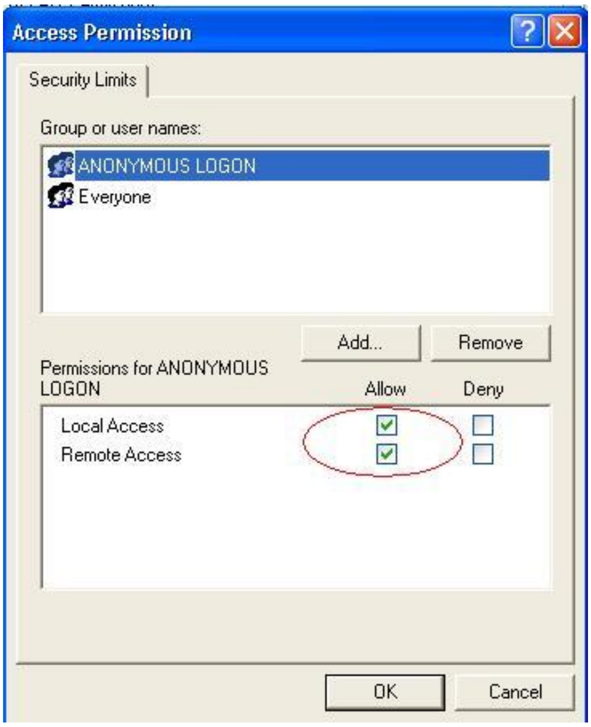
Access Permissions

You may edit who is allowed default access to applications. You may also set limits on applications that determine their own permissions.

Launch and Activation Permissions

You may edit who is allowed by default to launch applications or activate objects. You may also set limits on applications that determine their own permissions.

Step 5: Click on the “Edit Limits…” of “Access Permissions” button to set.

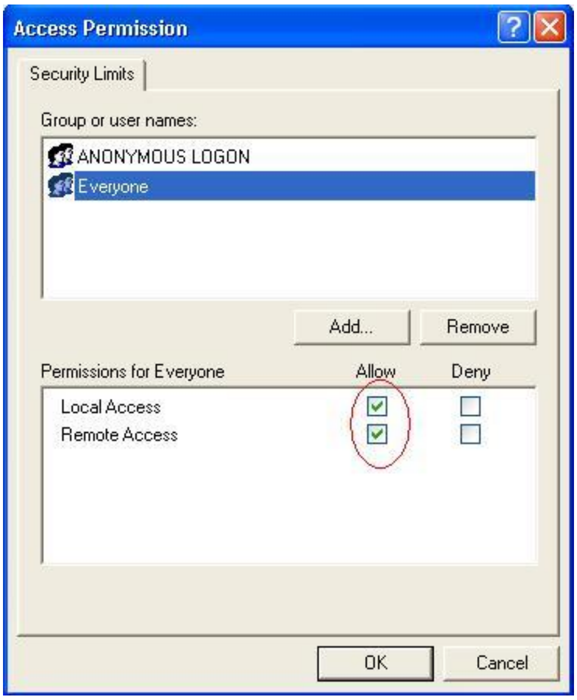


Access Permission

Group or user names:

- ANONYMOUS LOGON
- Everyone

Permissions for ANONYMOUS LOGON	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

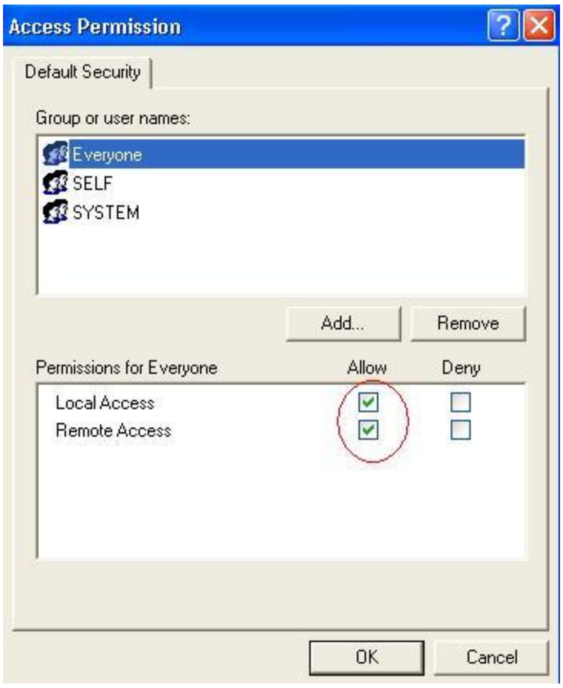


Group or user names:

ANONYMOUS LOGON
Everyone

Permissions for Everyone	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

Step 6: Click on the “Edit Default…” of “Access Permissions” button to set.

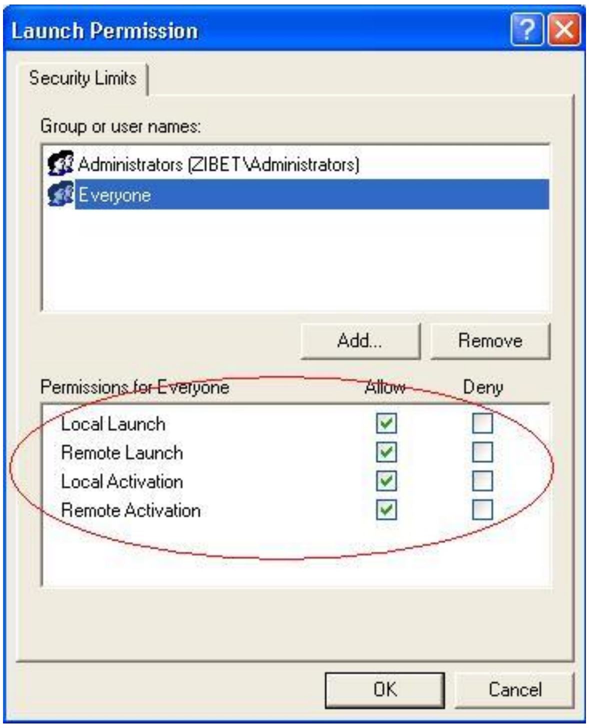


Group or user names:

Everyone
SELF
SYSTEM

Permissions for Everyone	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

Step 7: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.

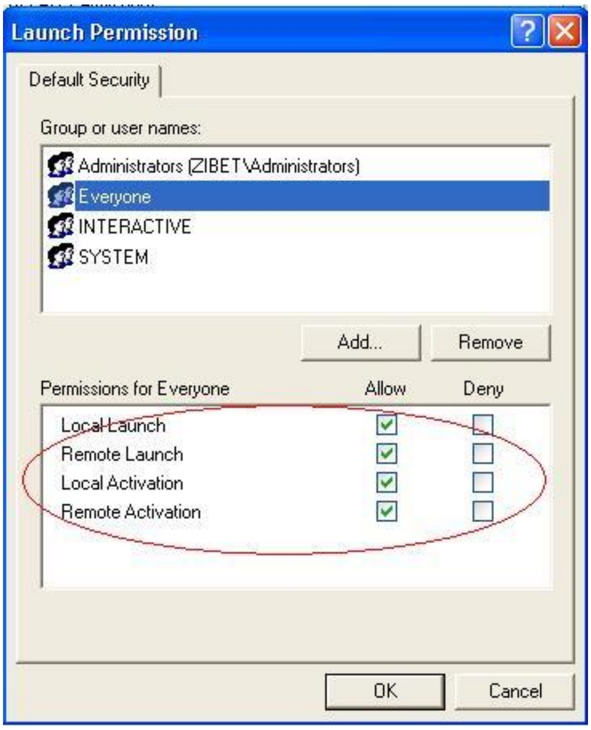


Group or user names:

- Administrators (ZIBET\Administrators)
- Everyone

Permissions for Everyone	Allow	Deny
Local Launch	☑	□
Remote Launch	☑	□
Local Activation	☑	□
Remote Activation	☑	□

Step 8: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.

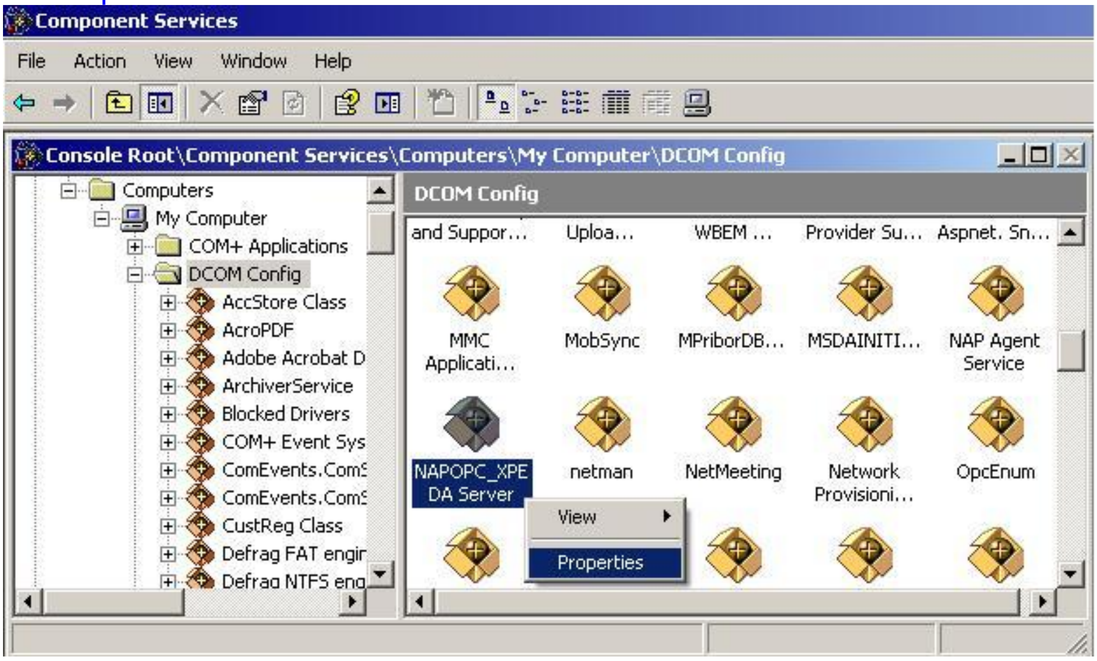


Group or user names:

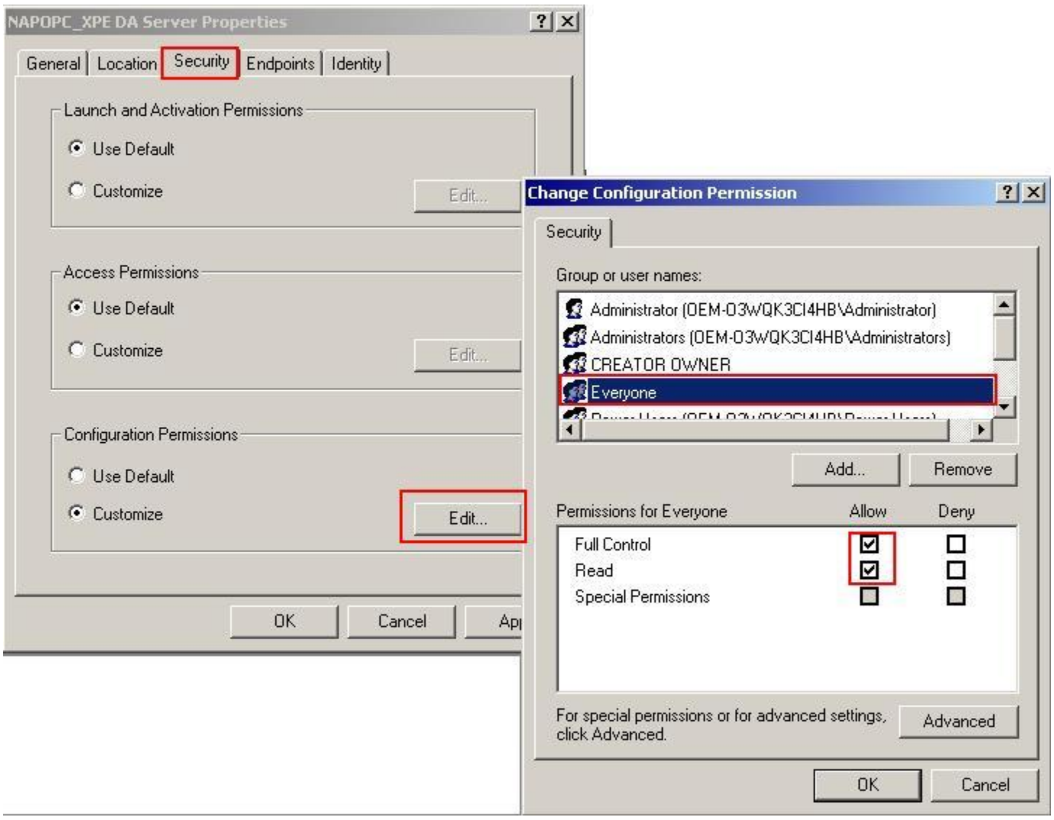
- Administrators (ZIBET\Administrators)
- Everyone
- INTERACTIVE
- SYSTEM

Permissions for Everyone	Allow	Deny
Local Launch	☑	□
Remote Launch	☑	□
Local Activation	☑	□
Remote Activation	☑	□

Step 9: Right click on the “NAPOPC_XPE DA Server” of “DCOM Config” button and select “Properties”.



Step 10: Select the "Security" tab page and click “Edit…” of “Configuration Permissions”. To make sure there is “Everyone” in “Group or user names” and allow “Full Control” and “Read”


NAPOPC_XPE DA Server Properties

General | Location | Security | Endpoints | Identity

Launch and Activation Permissions
● Use Default
○ Customize

Access Permissions
● Use Default
○ Customize

Configuration Permissions
○ Use Default
● Customize

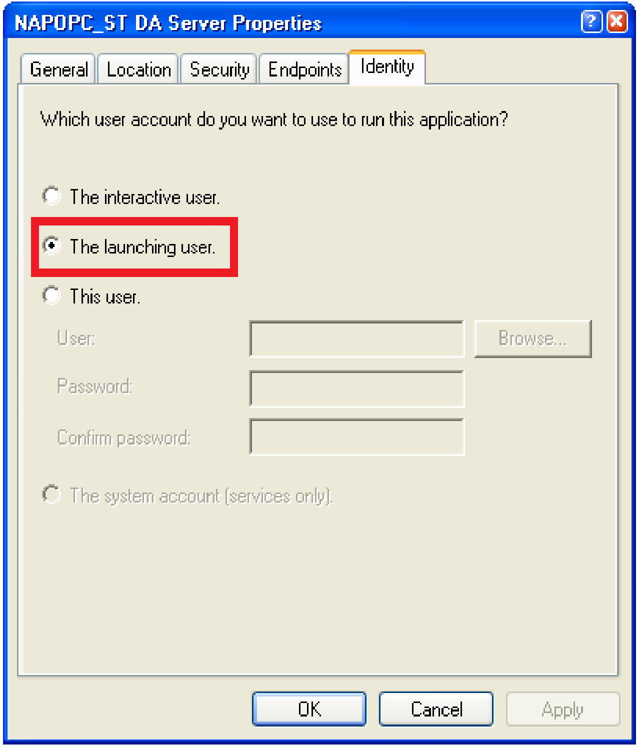
Change Configuration Permission

Group or user names:
- Administrator (OEM-03WQK3CI4HB\Administrator)
- Administrators (OEM-03WQK3CI4HB\Administrators)
- CREATOR OWNER
- Everyone

Permissions for Everyone	Allow	Deny
Full Control	☑	□
Read	☑	□
Special Permissions	□	□

For special permissions or for advanced settings, click Advanced.

Step 11: Select the "Identity" tab page and check "The launching user"



NAPOPC_ST DA Server Properties

Which user account do you want to use to run this application?

○ The interactive user.

● The launching user.

○ This user.

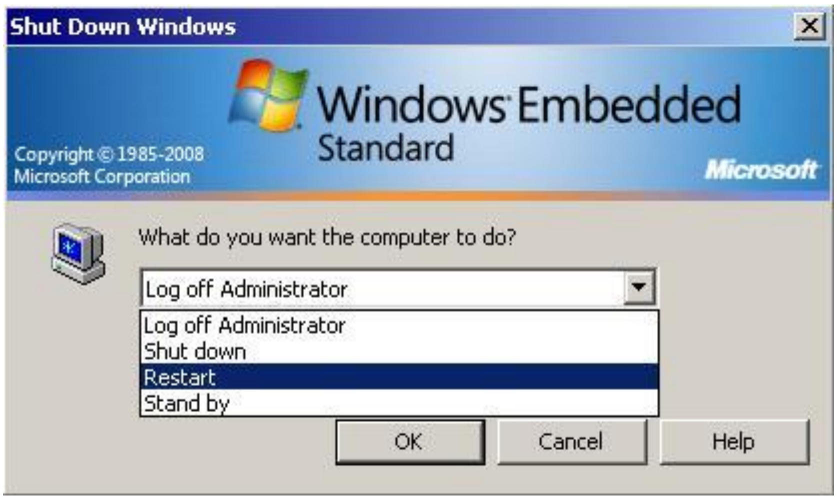
User:

Password:

Confirm password:

○ The system account (services only).

Step 12: Restart XPAC



Shut Down Windows

Copyright © 1985-2008 Microsoft Corporation

What do you want the computer to do?

Log off Administrator

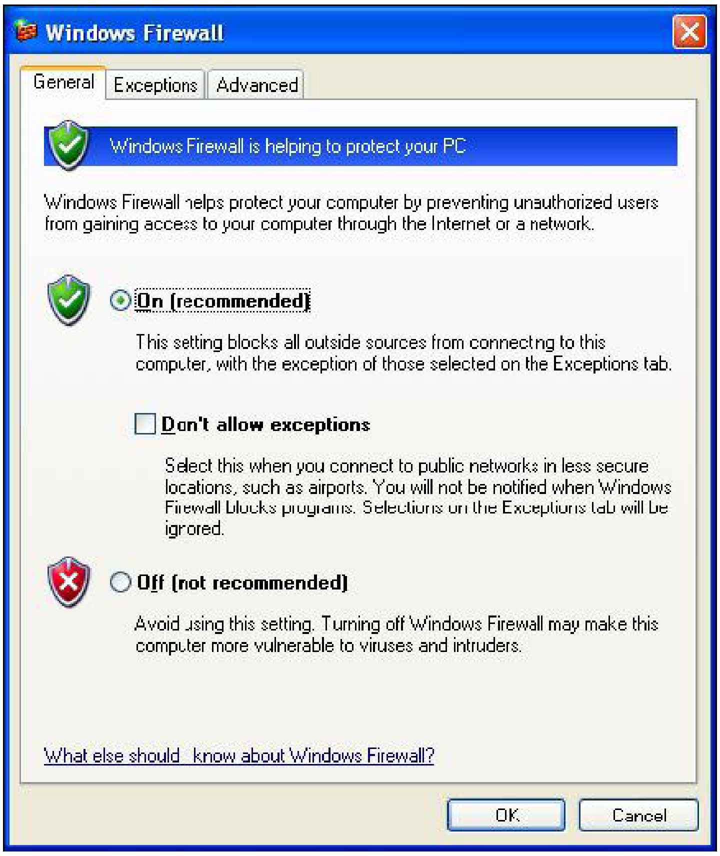
Log off Administrator
Shut down
Restart
Stand by

4.2.2 Configuring On the Server Site (PC)

Configuring the Firewall

Step1: By default the windows firewall is set to "On". This setting is recommended by Microsoft and by OPC to give your machine the highest possible protection. For trouble shooting, you may wish to temporarily turn off the firewall to prove or disprove that the firewall configuration is the source of any communication failure.

Note: It may be appropriate to permanently turn off the firewall if the machine is sufficiently protected behind a corporate firewall. When turned off, the individual firewall settings outlined here need not be performed to allow OPC communication.



Windows Firewall

General | Exceptions | Advanced

Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

● **On (recommended)**

This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.

□ **Don't allow exceptions**

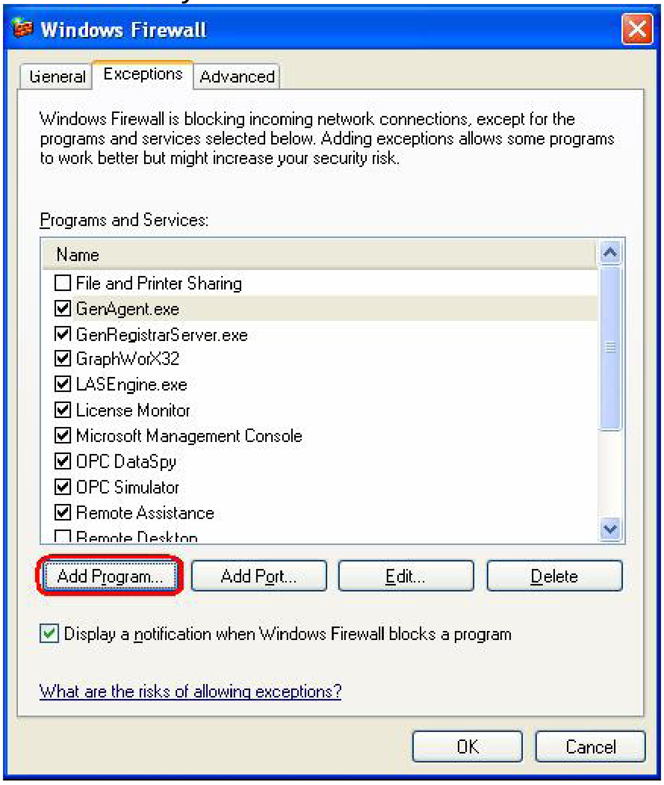
Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.

○ **Off (not recommended)**

Avoid using this setting. Turning off Windows Firewall may make this computer more vulnerable to viruses and intruders.

What else should I know about Windows Firewall?

Step 2: Select the .Exceptions tab and add all OPC Clients and Servers to the exception list. Also add Microsoft Management Console (used by the DCOM configuration utility in the next section) and the OPC utility OPCEnum.exe found in the Windows\System32 directory.



Windows Firewall

General | Exceptions | Advanced

Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

Programs and Services:

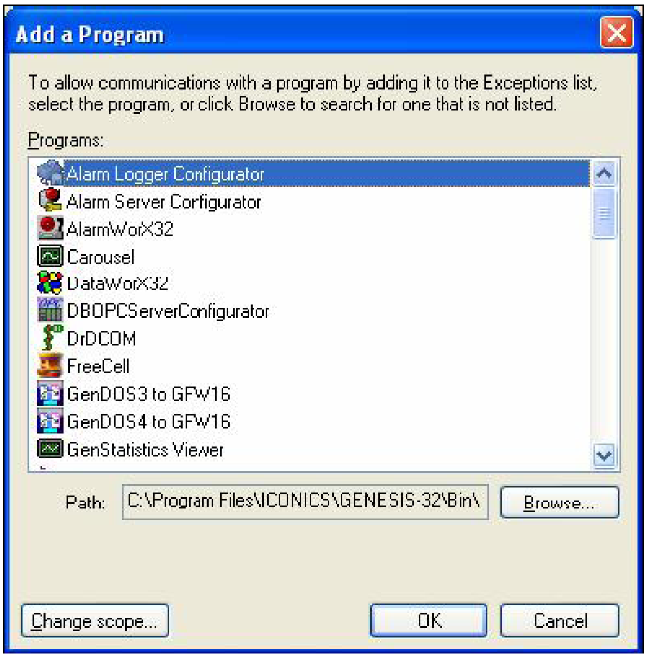
Name
- □ File and Printer Sharing
- ☑ GenAgent.exe
- ☑ GenRegistrarServer.exe
- ☑ GraphWorX32
- ☑ LASEngine.exe
- ☑ License Monitor
- ☑ Microsoft Management Console
- ☑ OPC DataSpy
- ☑ OPC Simulator
- ☑ Remote Assistance
- □ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

What are the risks of allowing exceptions?

In the Add a Program dialog, there is a listing of most applications on the machine, but note that not all of them show up on this list. Use the "Browse" button to find other executables installed on the computer.

Note: Only EXE files are added to the exceptions list. For in-process OPC Servers and Clients (DLLs and OCXs) you will need to add the EXE applications that call them to the list instead.



Add a Program

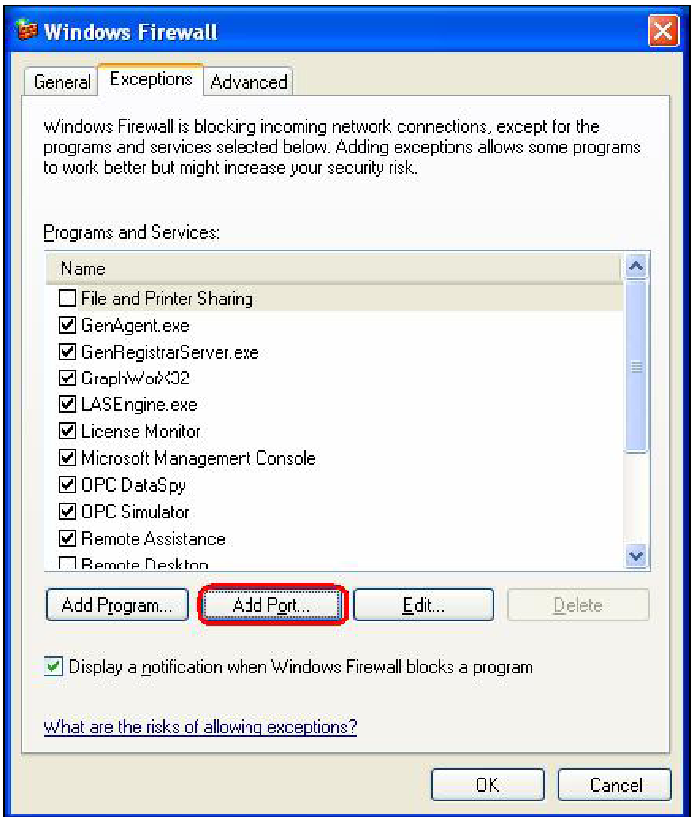
To allow communications with a program by adding it to the Exceptions list, select the program, or click Browse to search for one that is not listed.

Programs:

- Alarm Logger Configurator
- Alarm Server Configurator
- AlarmWorX32
- Carousel
- DataWorX32
- DBOPCServerConfigurator
- DrDCOM
- FreeCell
- GenDOS3 to GFW16
- GenDOS4 to GFW16
- GenStatistics Viewer

Path: C:\Program Files\ICONICS\GENESIS-32\Bin\

Step 3: Add TCP port 135 as it is needed to initiate DCOM communications, and allow for incoming echo requests. In the Exceptions tab of the Windows Firewall, click on Add Port.



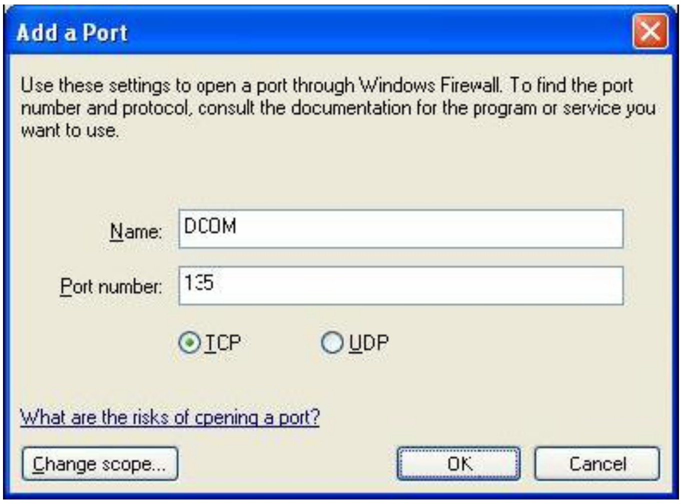
Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

Programs and Services:

Name
☐ File and Printer Sharing
☑ GenAgent.exe
☑ GenRegistrarServer.exe
☑ GraphWorX32
☑ LASEngine.exe
☑ License Monitor
☑ Microsoft Management Console
☑ OPC DataSpy
☑ OPC Simulator
☑ Remote Assistance
☐ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

In the Add a Port dialog, fill out the fields as follows:
Name: DCOM
Port number: 135
Choose the TCP radio button



Add a Port

Use these settings to open a port through Windows Firewall. To find the port number and protocol, consult the documentation for the program or service you want to use.

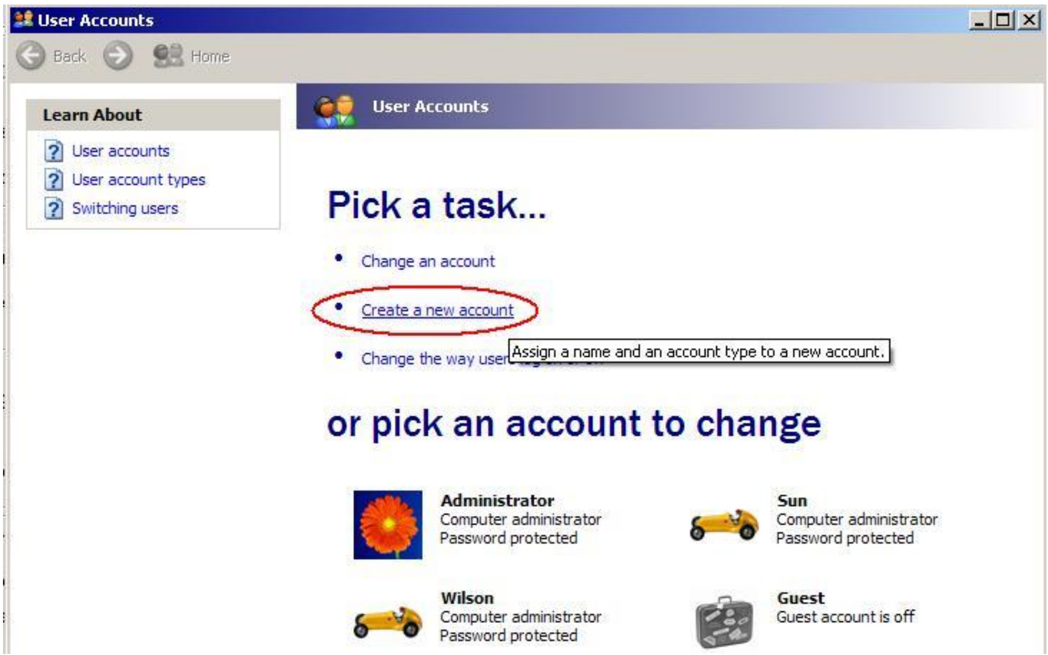
Name: DCOM

Port number: 135

● TCP ○ UDP

Creating the Account

Step 1: Create a account which must be the same with the account of client site.



Pick a task...

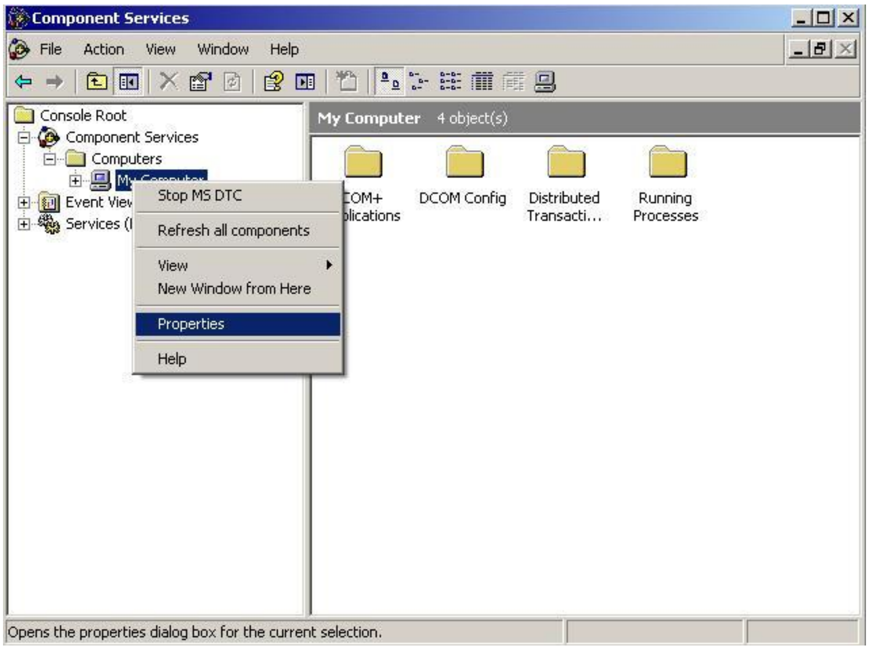
- Change an account
- Create a new account
- Change the way user

Assign a name and an account type to a new account.

or pick an account to change

Configuring DCOM

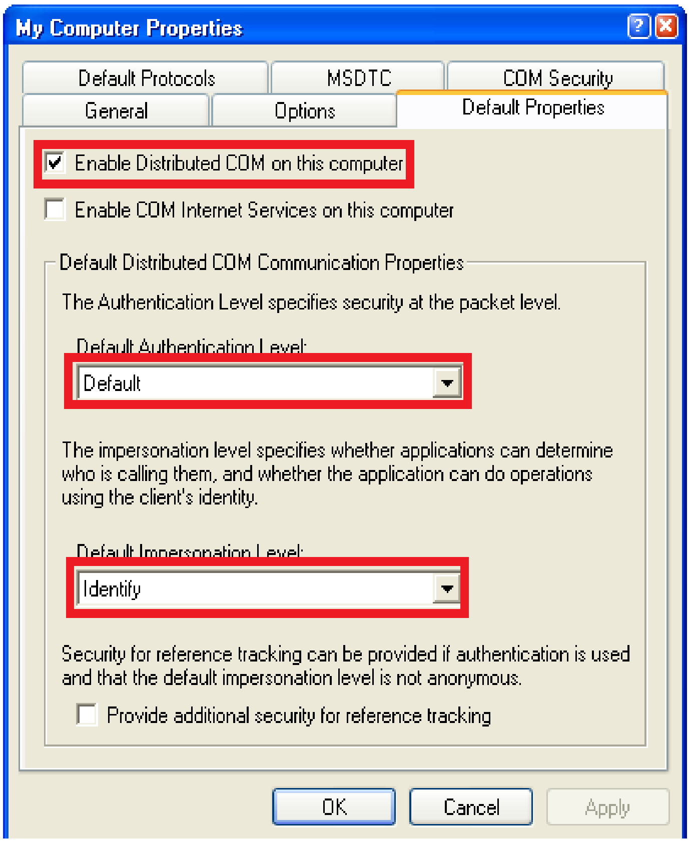
Step 1: Run the dcomcnfg.exe program to launch component services. Right clieck "My Computer" and choose "Properties".



Step 2: Select the "Default Properties" tab page.

Step 3: Use the following settings:

Field Name	Set to
Enable Distributed COM on this computer	Checked
Default Authentication Level:	Default
Default Impersonation Level:	Identify



☑ Enable Distributed COM on this computer

□ Enable COM Internet Services on this computer

Default Distributed COM Communication Properties

The Authentication Level specifies security at the packet level.

Default Authentication Level: Default

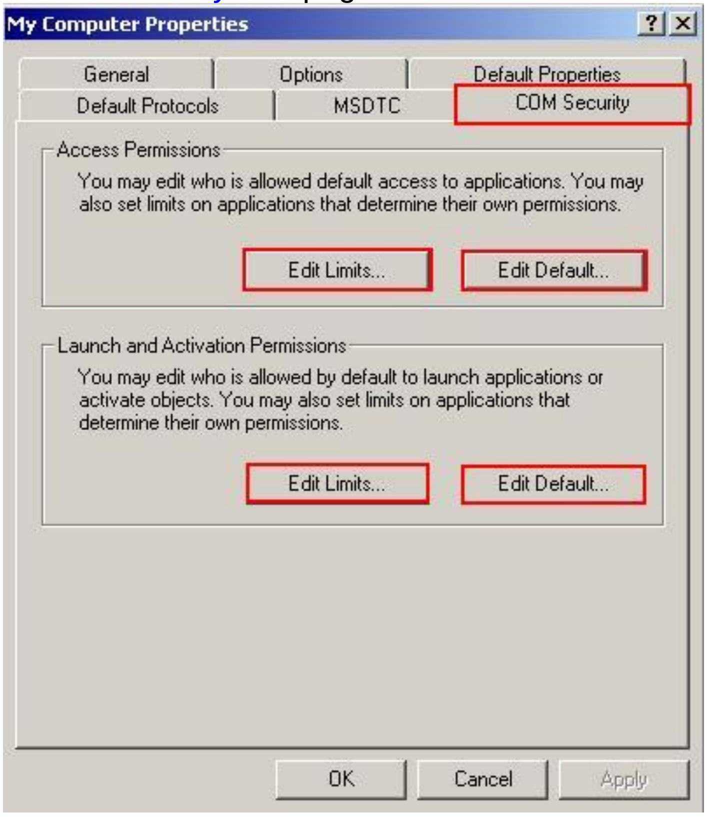
The impersonation level specifies whether applications can determine who is calling them, and whether the application can do operations using the client's identity.

Default Impersonation Level: Identify

Security for reference tracking can be provided if authentication is used and that the default impersonation level is not anonymous.

□ Provide additional security for reference tracking

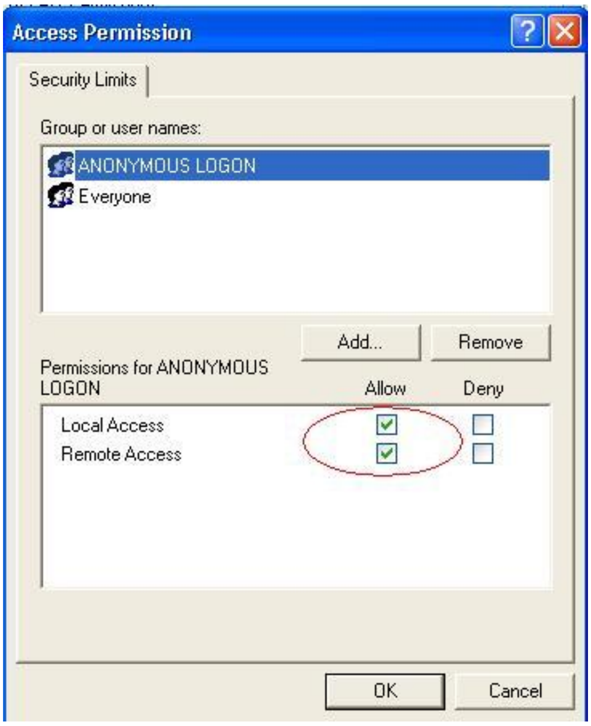
Step 4: Select the "COM Security" tab page.



You may edit who is allowed default access to applications. You may also set limits on applications that determine their own permissions.

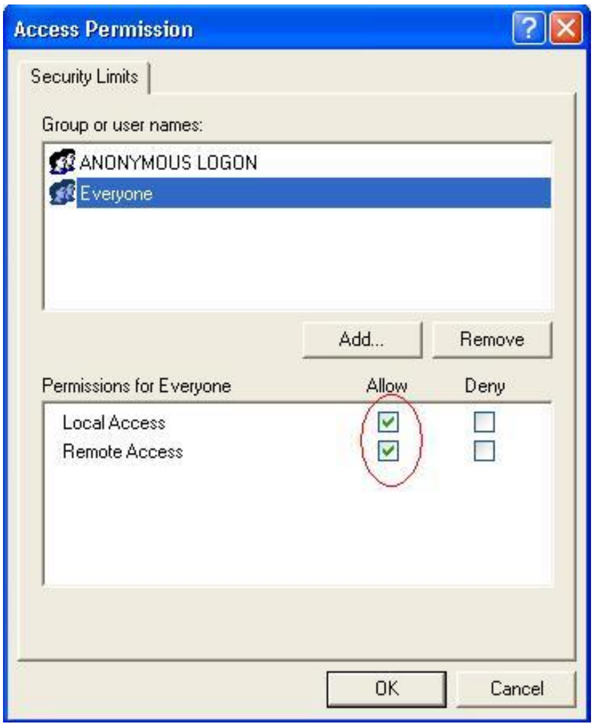
You may edit who is allowed by default to launch applications or activate objects. You may also set limits on applications that determine their own permissions.

Step 5: Click on the “Edit Limits…” of “Access Permissions” button to set.



Permissions for ANONYMOUS LOGON

	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

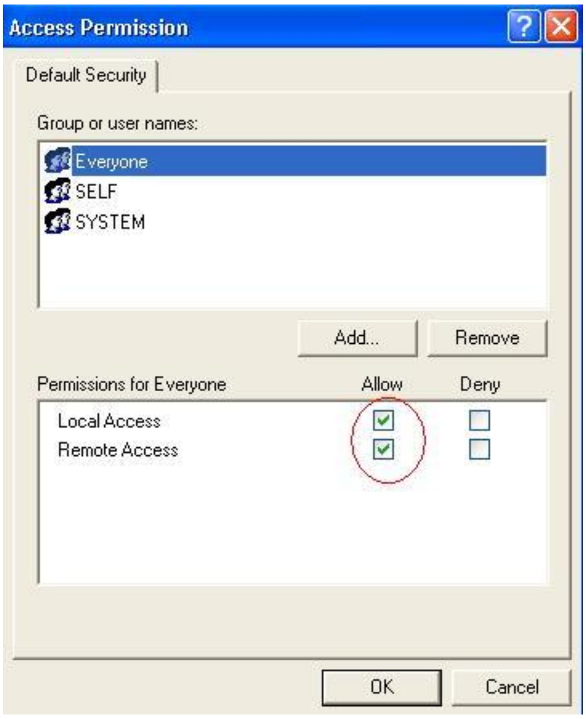


Group or user names:

- ANONYMOUS LOGON
- Everyone

Permissions for Everyone	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

Step 6: Click on the "Edit Default…" of "Access Permissions" button to set.

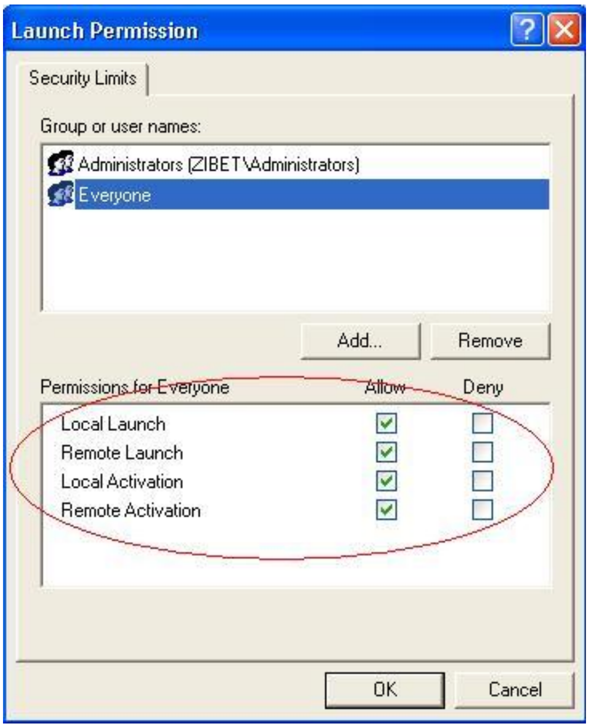


Group or user names:

- Everyone
- SELF
- SYSTEM

Permissions for Everyone	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

Step 7: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.



Launch Permission

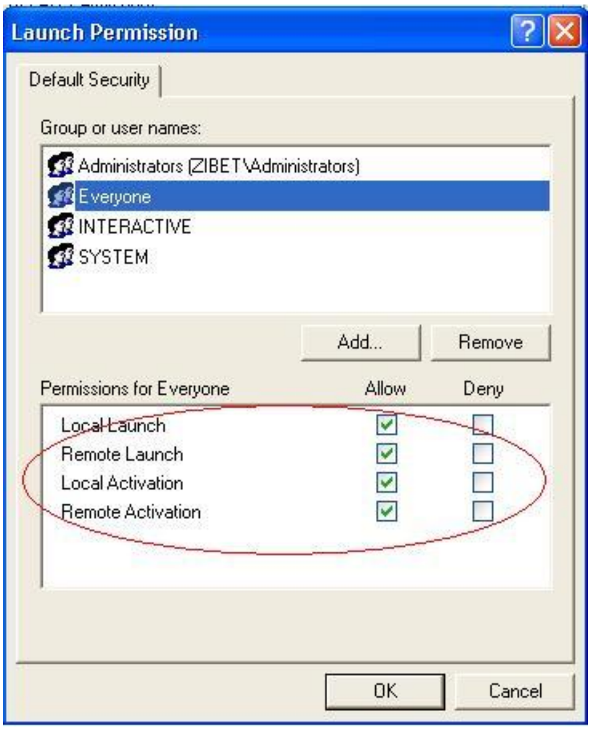
Security Limits

Group or user names:

- Administrators (ZIBET\Administrators)
- Everyone

Permissions for Everyone	Allow	Deny
Local Launch	☑	□
Remote Launch	☑	□
Local Activation	☑	□
Remote Activation	☑	□

Step 8: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.



Launch Permission

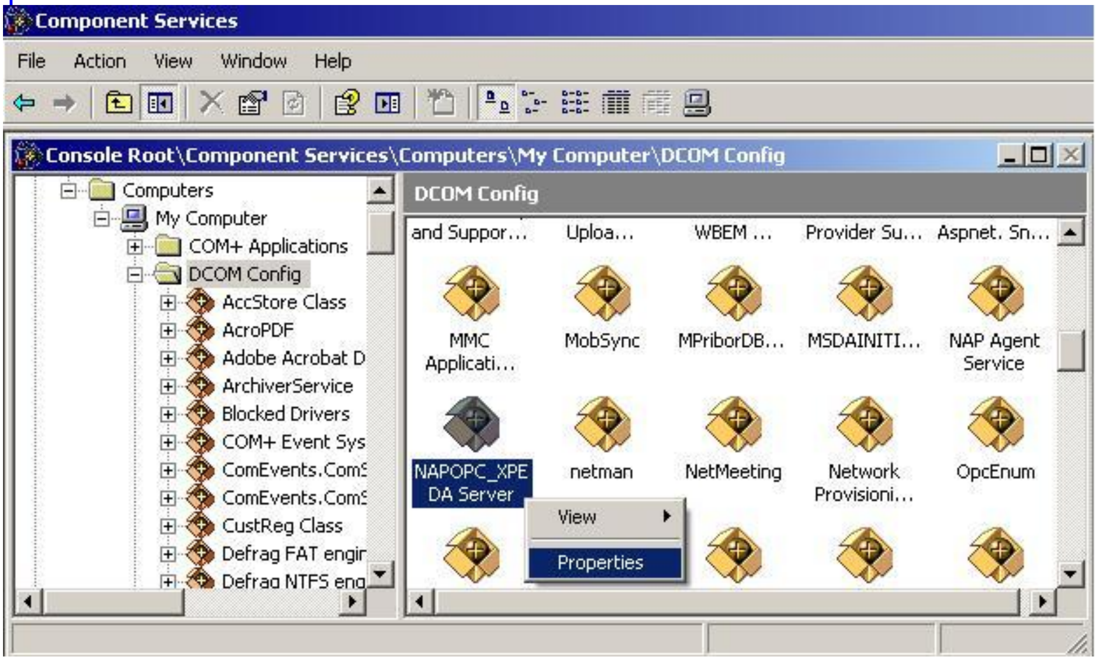
Default Security

Group or user names:

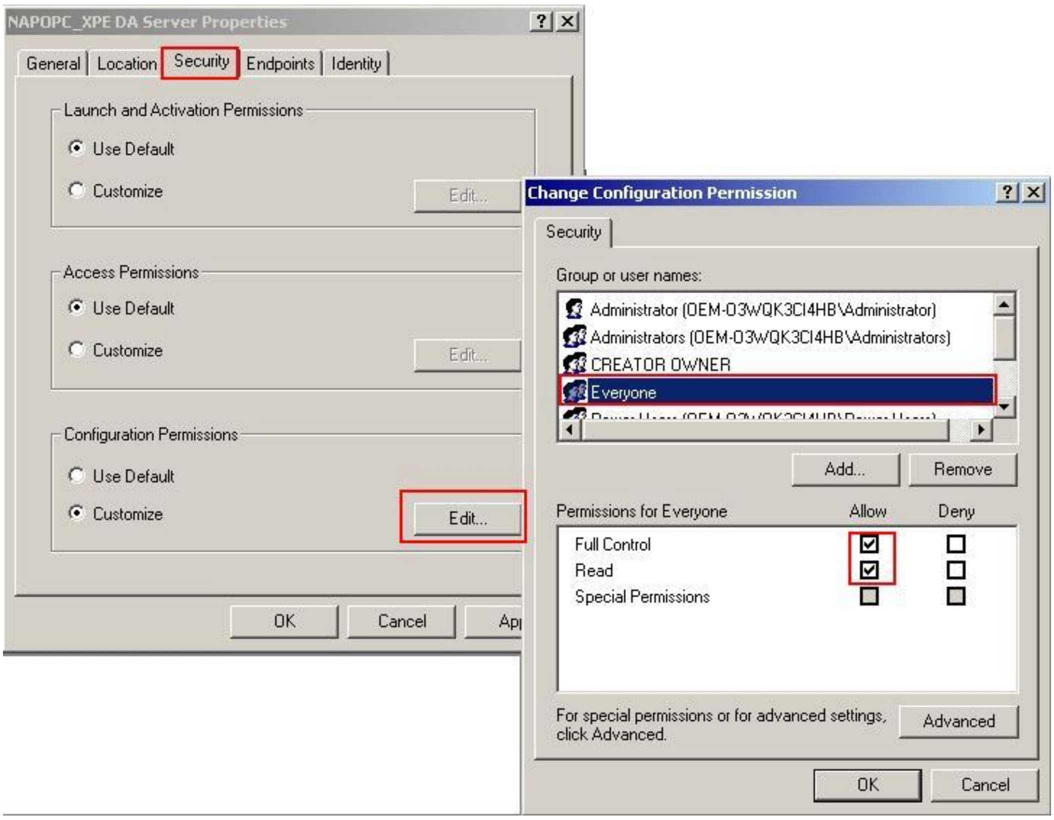
- Administrators (ZIBET\Administrators)
- Everyone
- INTERACTIVE
- SYSTEM

Permissions for Everyone	Allow	Deny
Local Launch	☑	□
Remote Launch	☑	□
Local Activation	☑	□
Remote Activation	☑	□

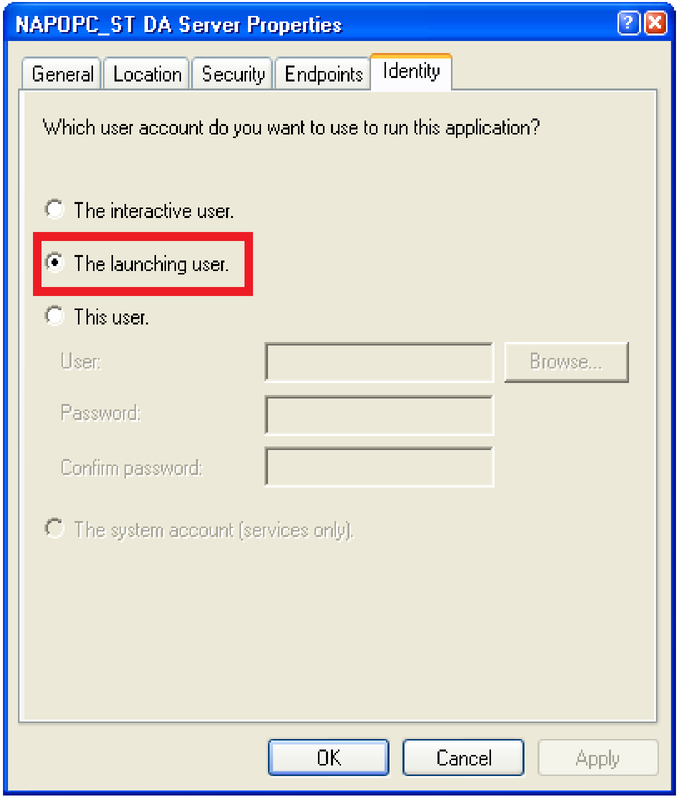
Step 9: Right click on the "NAPOPC DA Server" of "DCOM Config" button and select "Properties".



Step 10: Select the "Security" tab page and click "Edit…" of "Configuration Permissions". To make sure there is "Everyone" in "Group or user names" and allow "Full Control" and "Read"


NAPOPC_XPE DA Server Properties
General Location Security Endpoints Identity
Launch and Activation Permissions
● Use Default
○ Customize
Edit...
Access Permissions
● Use Default
○ Customize
Edit...
Configuration Permissions
○ Use Default
● Customize
Edit...
OK Cancel Ap
Change Configuration Permission
Security
Group or user names:
Administrator (OEM-03WQK3CI4HB\Administrator)
Administrators (OEM-03WQK3CI4HB\Administrators)
CREATOR OWNER
Everyone
Add... Remove
Permissions for Everyone Allow Deny
Full Control ☑ □
Read ☑ □
Special Permissions □ □
For special permissions or for advanced settings, click Advanced.
Advanced
OK Cancel

Step 11: Select the "Identity" tab page and check "The launching user"



NAPOPC_ST DA Server Properties

Which user account do you want to use to run this application?

○ The interactive user.

● The launching user.

○ This user.

User:

Password:

Confirm password:

○ The system account (services only).

Step 12: Restart PC

4.2.3 Configuring On the Server Site (WinPAC)

System Requirement

OS version:

WinPAC OS 1.3.04 or later

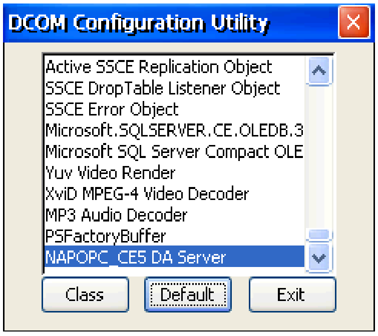
Program:

NAPOPC_CE5
DCOMCnfg.exe
WinPAC Utility 2.0.2.1 or later

Configuring DCOM

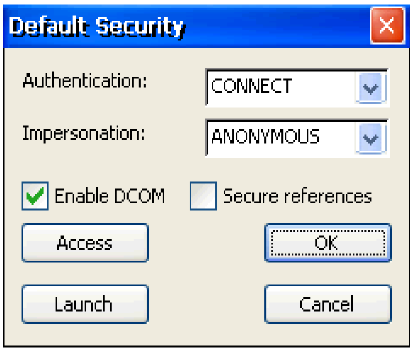
Step 1: Run the \\NAPOPC_CE5\napopc_ce5boot.exe program to register.

Step 2: Run the dcomcnfg.exe program and choose "Default".



DCOM Configuration Utility

Active SSCE Replication Object
SSCE DropTable Listener Object
SSCE Error Object
Microsoft.SQLSERVER.CE.OLEDB.3
Microsoft SQL Server Compact OLE
Yuv Video Render
XviD MPEG-4 Video Decoder
MP3 Audio Decoder
PSFactoryBuffer
NAPOPC_CE5 DA Server

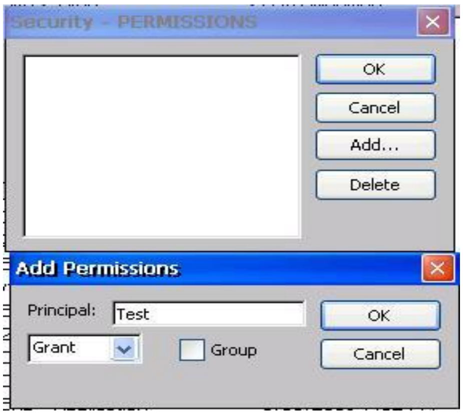


Authentication: CONNECT

Impersonation: ANONYMOUS

☑ Enable DCOM □ Secure references

Step 3: Select the "Access" button to add an account which is current connection account from client site.

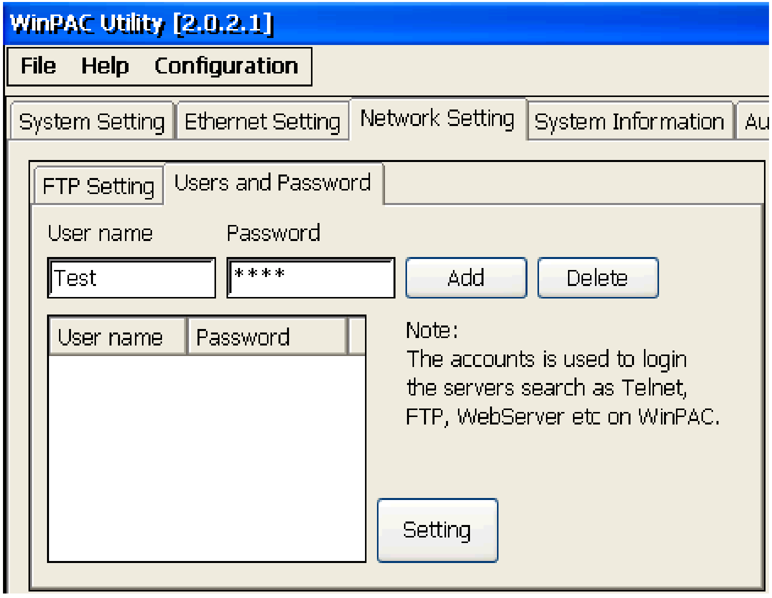


Principal: Test

Grant □ Group

Step 4: Select the "Launch" button to add an account which is current connection account from client site as above.

Step 5: Execute "WinPAC Utility->Network Setting->Users and Password"



User name: Test

Password: ****

Note:
The accounts is used to login the servers search as Telnet, FTP, WebServer etc on WinPAC.

Step 6: Fill out "User name", "Password", and press "Add". The "User name" and "Password" must be the account we set at **Step 3.** After pressing "Add", press "Setting" to finish all settings.

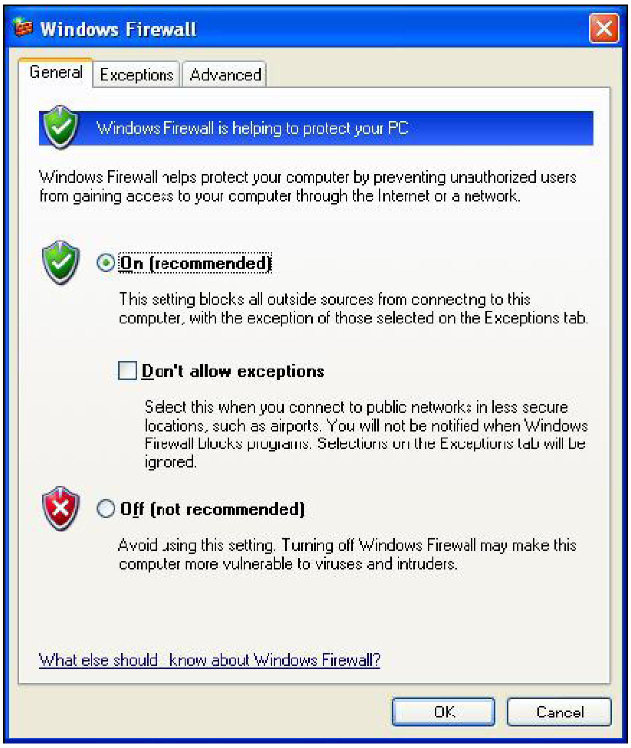
Step 7: Run WinPAC Utility to save and reboot.

4.2.4 Configuring On the Client Site (PC)

Configuring the Firewall

Step1: By default the windows firewall is set to "On". This setting is recommended by Microsoft and by OPC to give your machine the highest possible protection. For trouble shooting, you may wish to temporarily turn off the firewall to prove or disprove that the firewall configuration is the source of any communication failure.

Note: It may be appropriate to permanently turn off the firewall if the machine is sufficiently protected behind a corporate firewall. When turned off, the individual firewall settings outlined here need not be performed to allow OPC communication.



Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

● On (recommended)

This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.

□ Don't allow exceptions

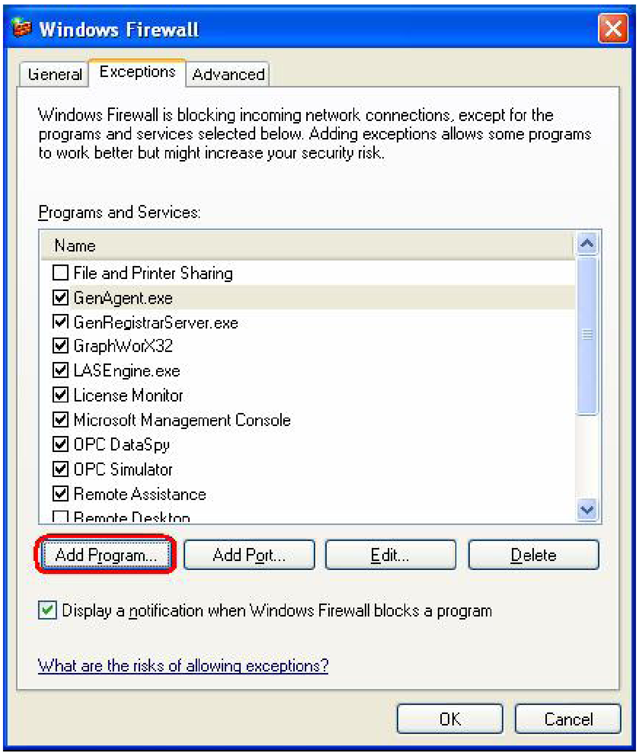
Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.

○ Off (not recommended)

Avoid using this setting. Turning off Windows Firewall may make this computer more vulnerable to viruses and intruders.

What else should I know about Windows Firewall?

Step 2: Select the .Exceptions tab and add all OPC Clients and Servers to the exception list. Also add Microsoft Management Console (used by the DCOM configuration utility in the next section) and the OPC utility OPCEnum.exe found in the Windows\System32 directory.



Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

Programs and Services:

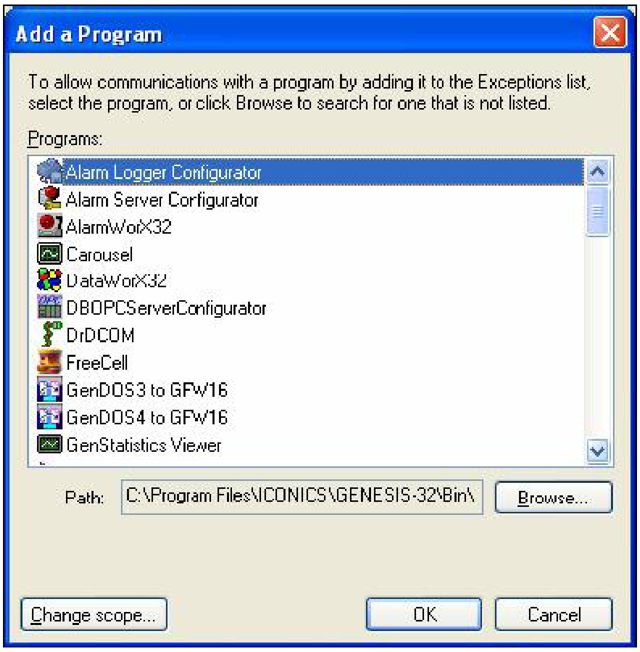
Name
- □ File and Printer Sharing
- ☑ GenAgent.exe
- ☑ GenRegistrarServer.exe
- ☑ GraphWorX32
- ☑ LASEngine.exe
- ☑ License Monitor
- ☑ Microsoft Management Console
- ☑ OPC DataSpy
- ☑ OPC Simulator
- ☑ Remote Assistance
- □ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

What are the risks of allowing exceptions?

In the Add a Program dialog, there is a listing of most applications on the machine, but note that not all of them show up on this list. Use the "Browse" button to find other executables installed on the computer.

Note: Only EXE files are added to the exceptions list. For in-process OPC Servers and Clients (DLLs and OCXs) you will need to add the EXE applications that call them to the list instead.



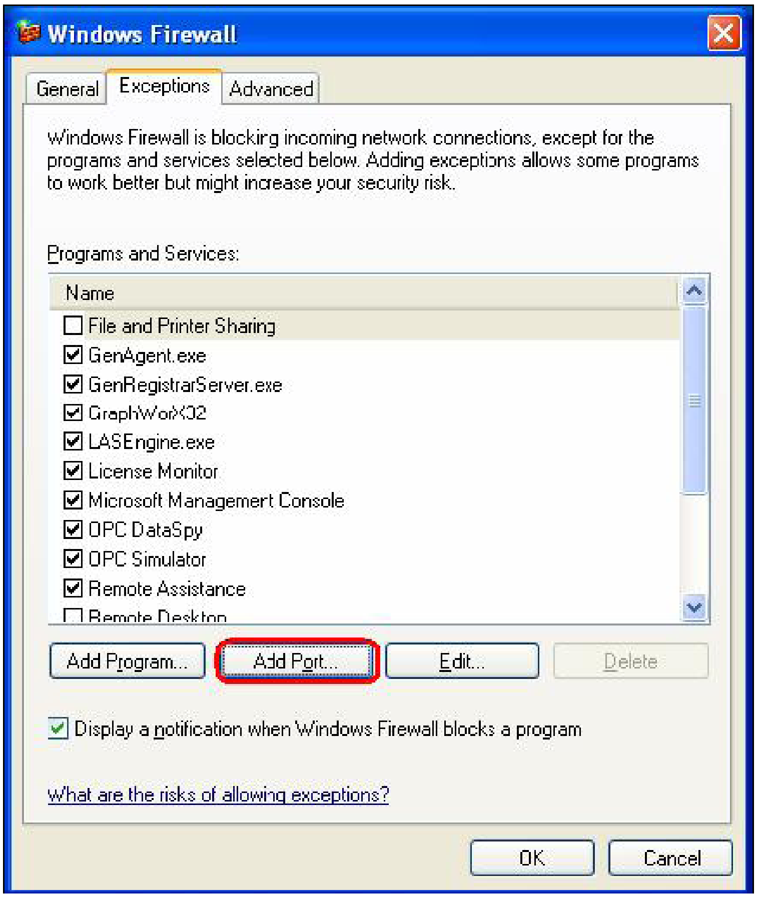
To allow communications with a program by adding it to the Exceptions list, select the program, or click Browse to search for one that is not listed.

Programs:

- Alarm Logger Configurator
- Alarm Server Configurator
- AlarmWorX32
- Carousel
- DataWorX32
- DBOPCServerConfigurator
- DrDCOM
- FreeCell
- GenDOS3 to GFW16
- GenDOS4 to GFW16
- GenStatistics Viewer

Path: C:\Program Files\ICONICS\GENESIS-32\Bin\

Step 3: Add TCP port 135 as it is needed to initiate DCOM communications, and allow for incoming echo requests. In the Exceptions tab of the Windows Firewall, click on Add Port.



Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

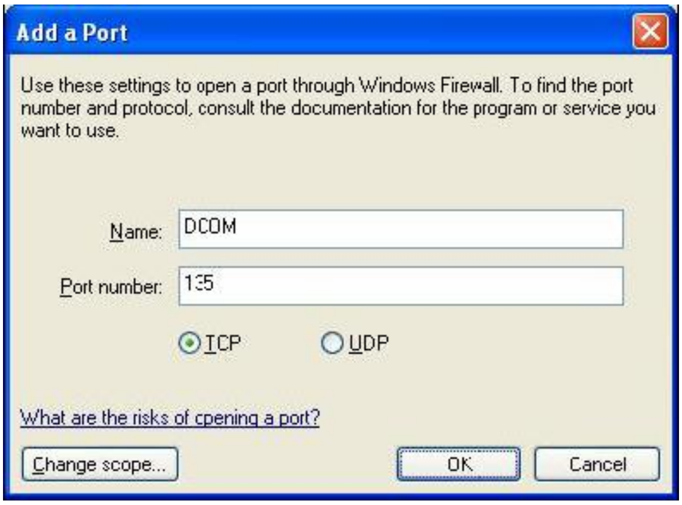
Programs and Services:

Name
- □ File and Printer Sharing
- ☑ GenAgent.exe
- ☑ GenRegistrarServer.exe
- ☑ GraphWorX32
- ☑ LASEngine.exe
- ☑ License Monitor
- ☑ Microsoft Management Console
- ☑ OPC DataSpy
- ☑ OPC Simulator
- ☑ Remote Assistance
- □ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

What are the risks of allowing exceptions?

In the Add a Port dialog, fill out the fields as follows:
Name: DCOM
Port number: 135
Choose the TCP radio button



Use these settings to open a port through Windows Firewall. To find the port number and protocol, consult the documentation for the program or service you want to use.

Name: DCOM

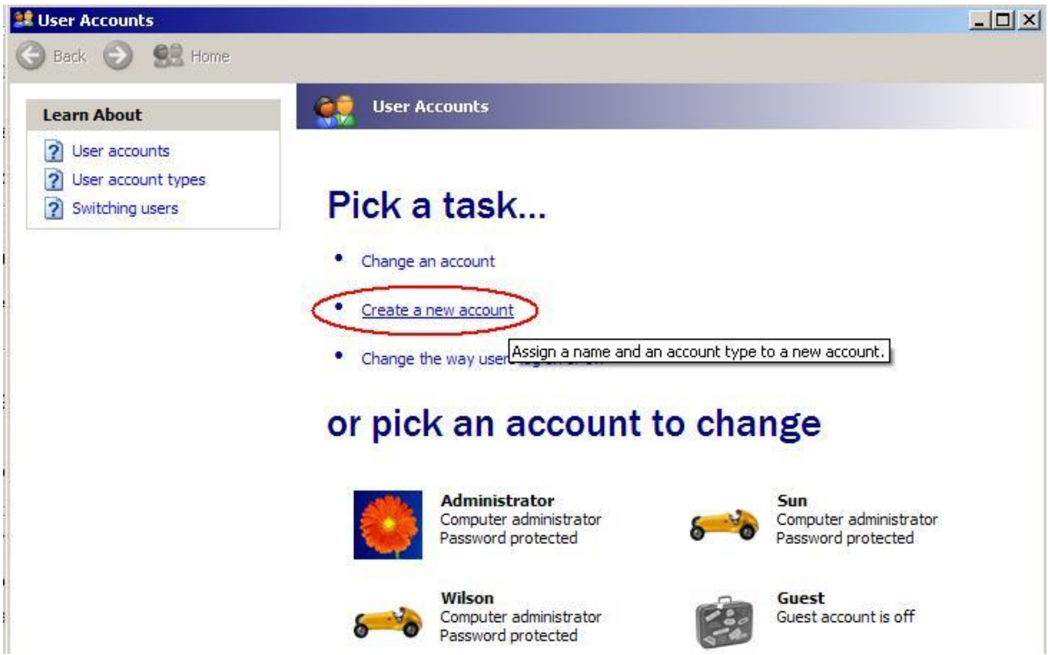
Port number: 135

● TCP ○ UDP

What are the risks of opening a port?

Creating the Account

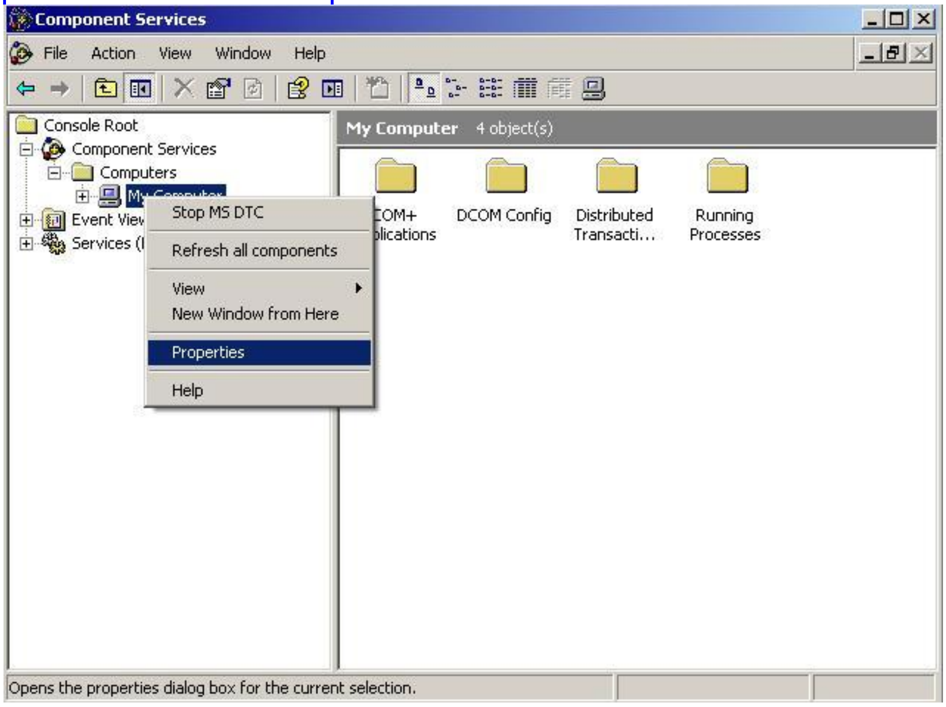
Step 1: Create a account which must be the same with the account of server site.



Assign a name and an account type to a new account.

Configuring DCOM

Step 1: Run the dcomcnfg.exe program to launch component services. Right clieck “My Computer” and choose “Properties”.

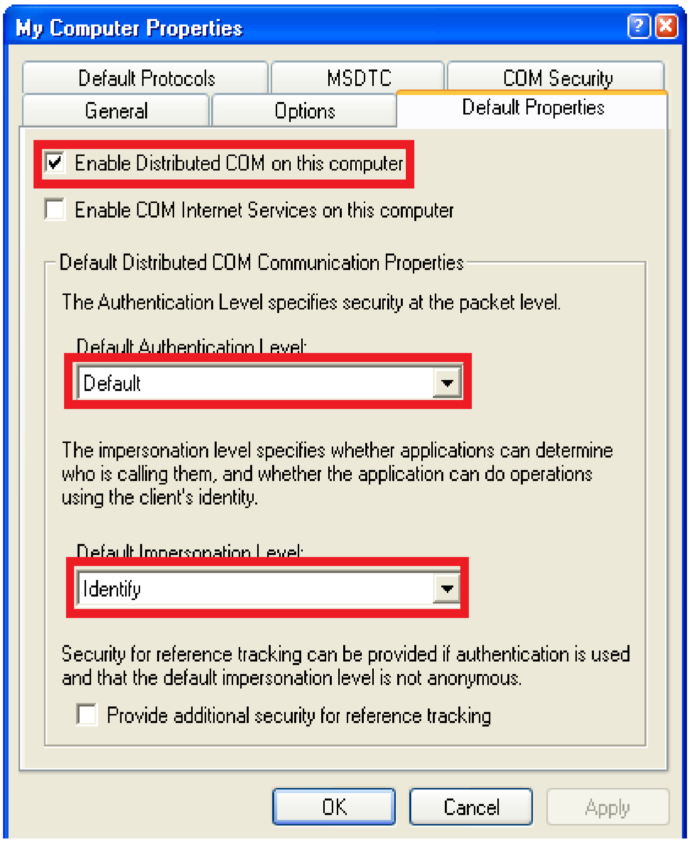


Opens the properties dialog box for the current selection.

Step 2: Select the "Default Properties" tab page.

Step 3: Use the following settings:

Field Name	Set to
Enable Distributed COM on this computer	Checked
Default Authentication Level:	Default
Default Impersonation Level:	Identify



☑ Enable Distributed COM on this computer

□ Enable COM Internet Services on this computer

Default Distributed COM Communication Properties

The Authentication Level specifies security at the packet level.

Default Authentication Level:

Default

The impersonation level specifies whether applications can determine who is calling them, and whether the application can do operations using the client's identity.

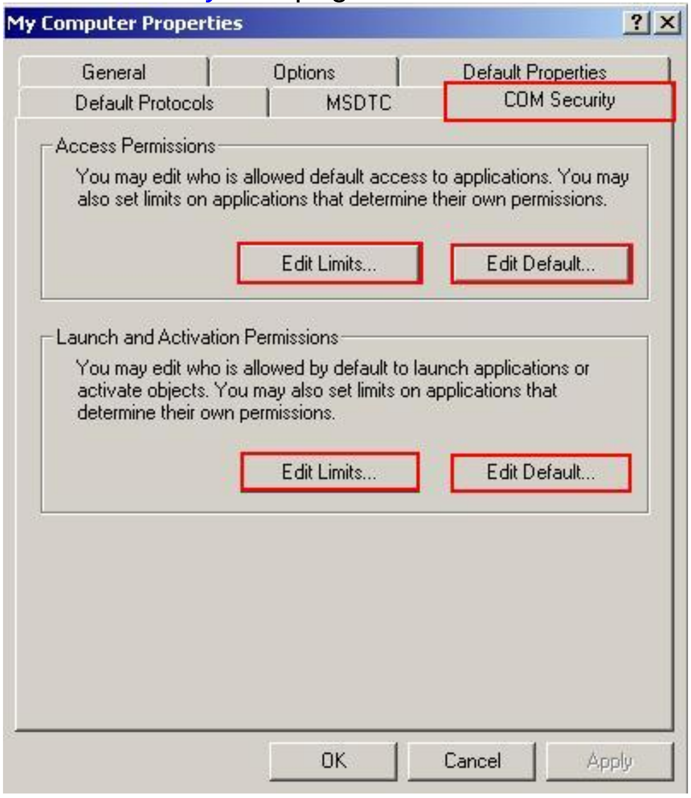
Default Impersonation Level:

Identify

Security for reference tracking can be provided if authentication is used and that the default impersonation level is not anonymous.

□ Provide additional security for reference tracking

Step 4: Select the "COM Security" tab page.



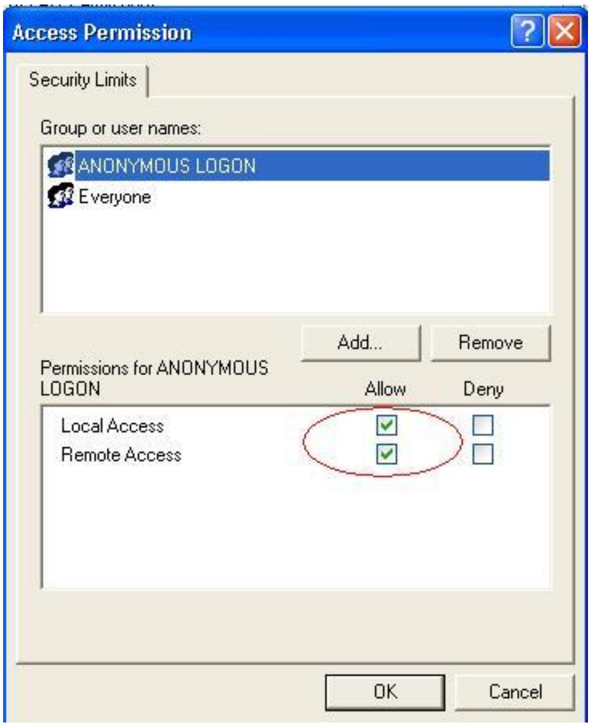
Access Permissions

You may edit who is allowed default access to applications. You may also set limits on applications that determine their own permissions.

Launch and Activation Permissions

You may edit who is allowed by default to launch applications or activate objects. You may also set limits on applications that determine their own permissions.

Step 5: Click on the "Edit Limits…" of "Access Permissions" button to set.



Access Permission

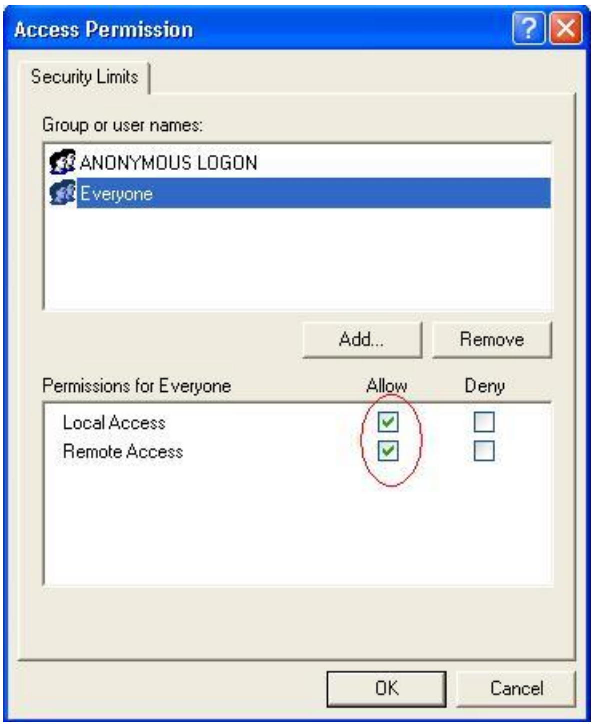
Security Limits

Group or user names:

- ANONYMOUS LOGON
- Everyone

Permissions for ANONYMOUS LOGON

	Allow	Deny
Local Access	☑	□
Remote Access	☑	□



Access Permission

Security Limits

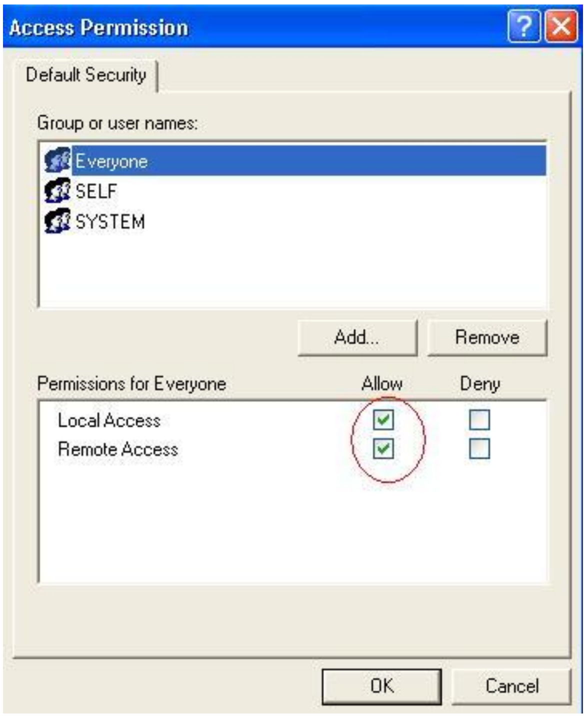
Group or user names:

- ANONYMOUS LOGON
- Everyone

Permissions for Everyone

	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

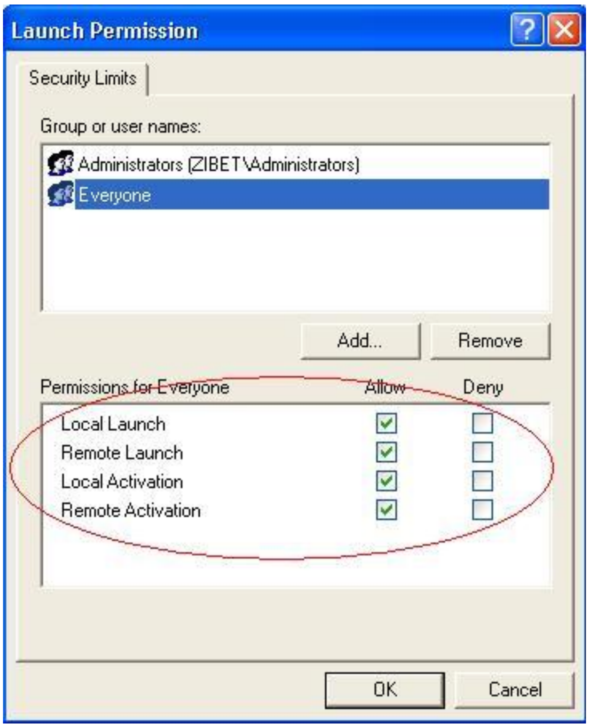
Step 6: Click on the "Edit Default…" of "Access Permissions" button to set.



Permissions for Everyone

	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

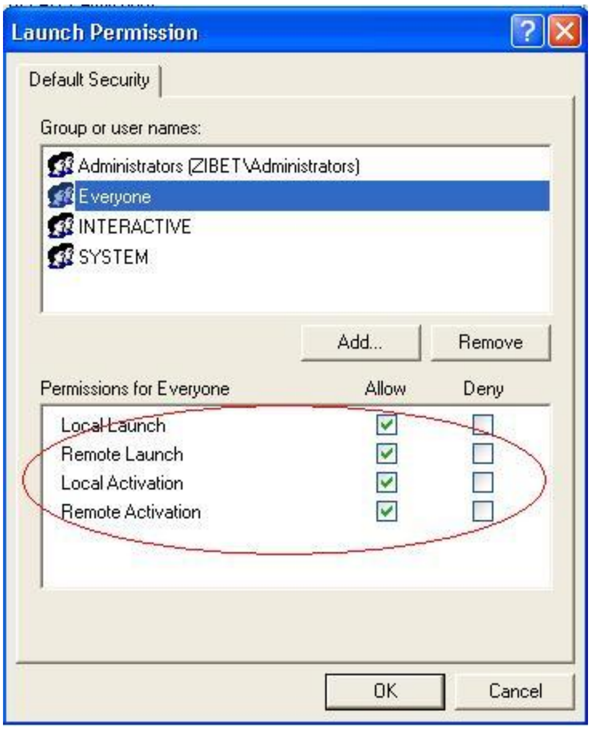
Step 7: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.



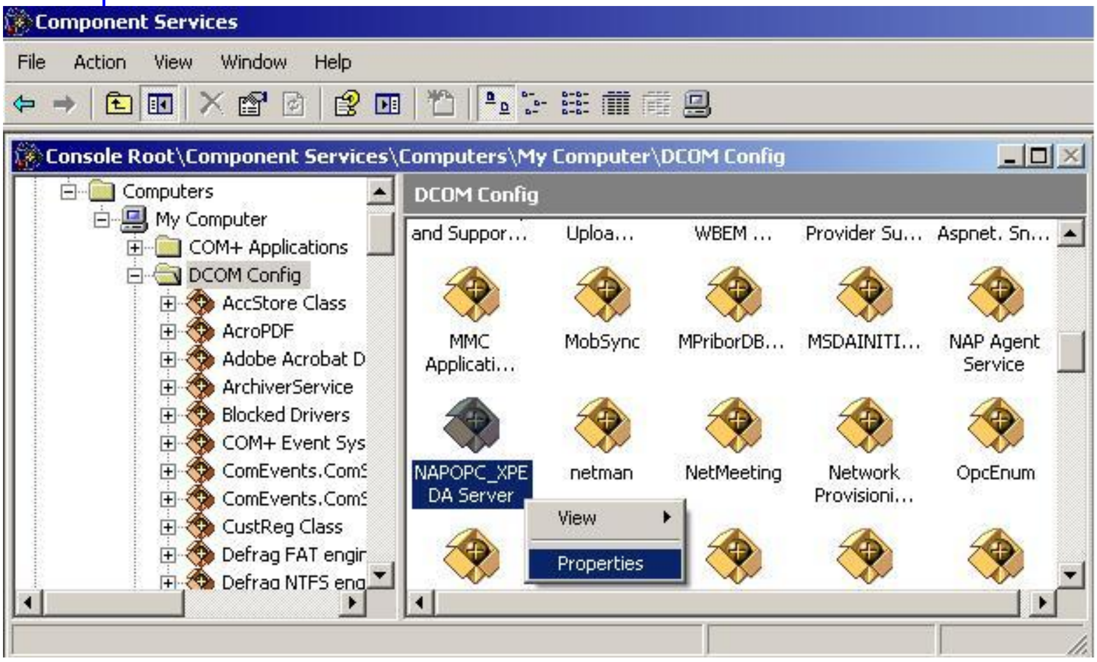
Permissions for Everyone

	Allow	Deny
Local Launch	☑	□
Remote Launch	☑	□
Local Activation	☑	□
Remote Activation	☑	□

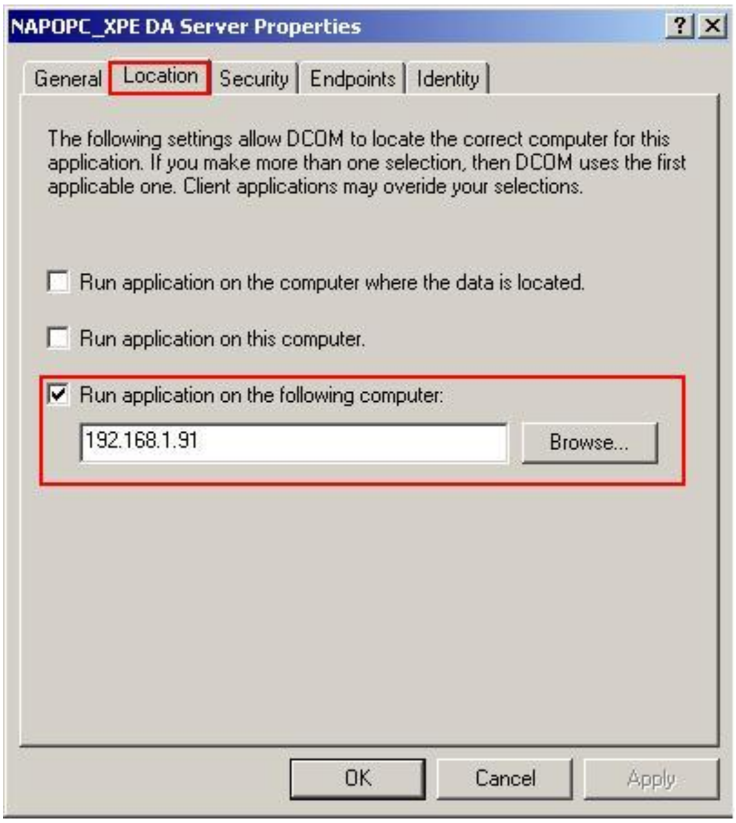
Step 8: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.



Step 9: Right click on the "NAPOPC_ST DA Server" of "DCOM Config" button and select "Properties".



Step 10: Select the "Location" tab page and check "Run application on the following computer". And enter the Server IP here.



NAPOPC_XPE DA Server Properties

The following settings allow DCOM to locate the correct computer for this application. If you make more than one selection, then DCOM uses the first applicable one. Client applications may overide your selections.

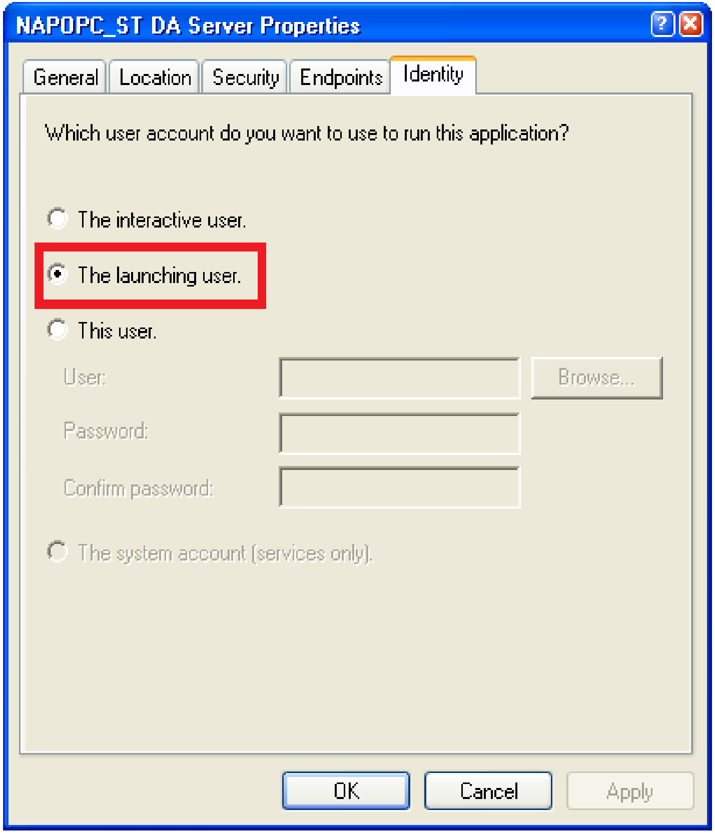
□ Run application on the computer where the data is located.

□ Run application on this computer.

☑ Run application on the following computer:

192.168.1.91

Step 11: Select the "Identity" tab page and check "The launching user"



NAPOPC_ST DA Server Properties

Which user account do you want to use to run this application?

○ The interactive user.

● The launching user.

○ This user.

User:

Password:

Confirm password:

○ The system account (services only).

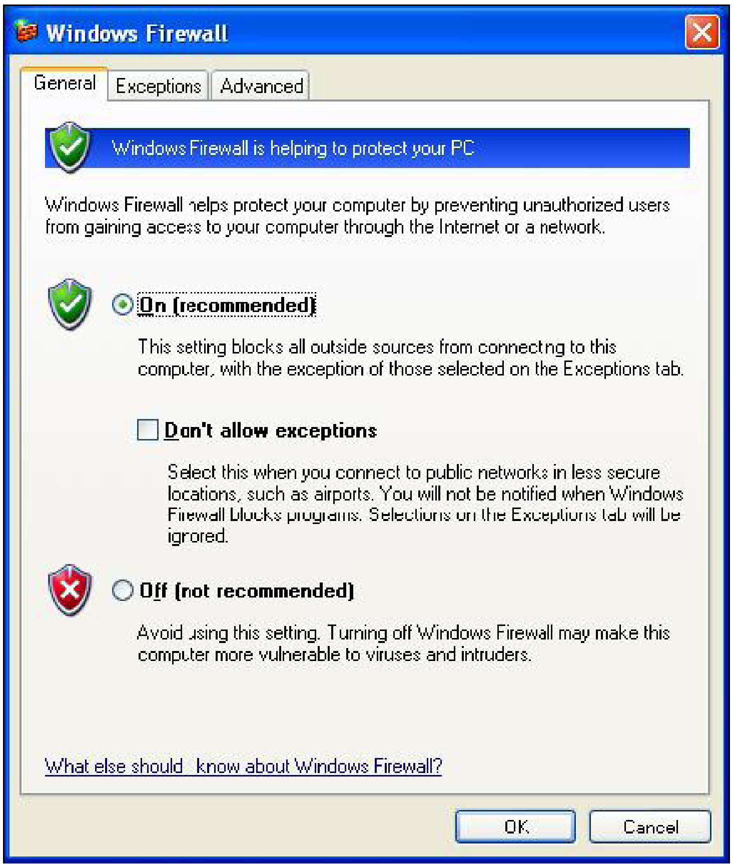
Step 12: Restart PC

4.2.5 Configuring On the Client Site (XPAC)

Configuring the Firewall

Step1: By default the windows firewall is set to "On". This setting is recommended by Microsoft and by OPC to give your machine the highest possible protection. For trouble shooting, you may wish to temporarily turn off the firewall to prove or disprove that the firewall configuration is the source of any communication failure.

Note: It may be appropriate to permanently turn off the firewall if the machine is sufficiently protected behind a corporate firewall. When turned off, the individual firewall settings outlined here need not be performed to allow OPC communication.



Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

● On (recommended)

This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.

□ Don't allow exceptions

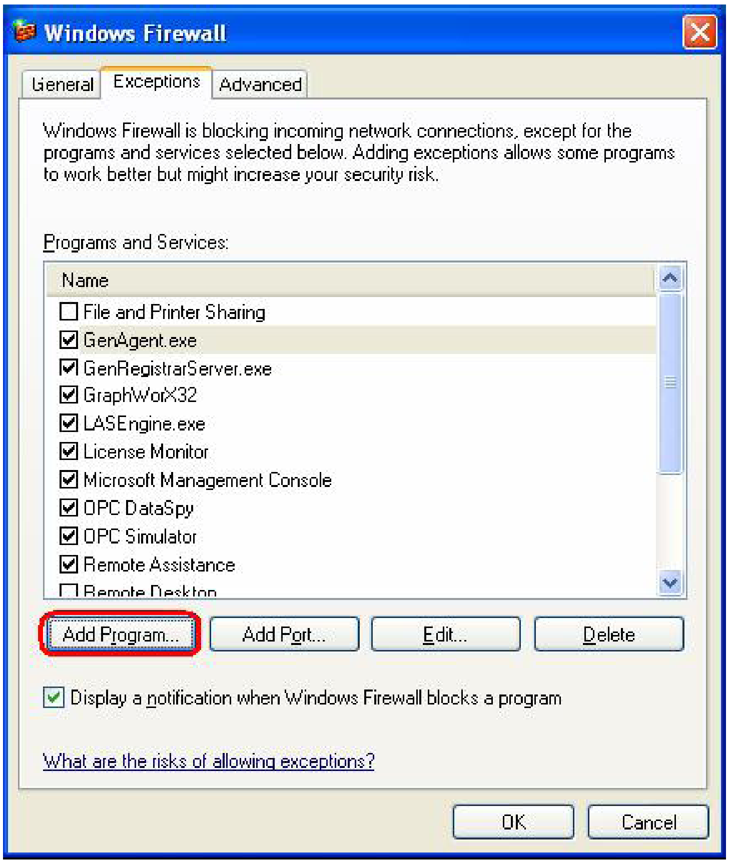
Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.

○ Off (not recommended)

Avoid using this setting. Turning off Windows Firewall may make this computer more vulnerable to viruses and intruders.

What else should I know about Windows Firewall?

Step 2: Select the .Exceptions tab and add all OPC Clients and Servers to the exception list. Also add Microsoft Management Console (used by the DCOM configuration utility in the next section) and the OPC utility OPCEnum.exe found in the Windows\System32 directory.



Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

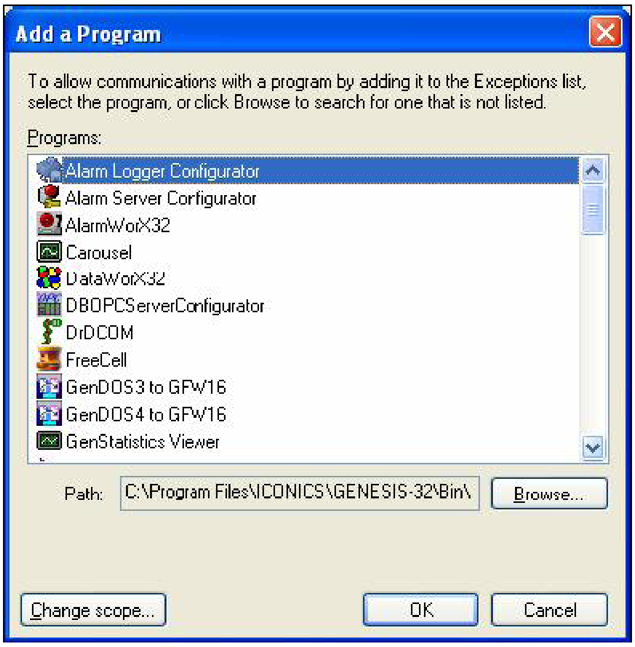
Programs and Services:

Name
□ File and Printer Sharing
☑ GenAgent.exe
☑ GenRegistrarServer.exe
☑ GraphWorX32
☑ LASEngine.exe
☑ License Monitor
☑ Microsoft Management Console
☑ OPC DataSpy
☑ OPC Simulator
☑ Remote Assistance
□ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

In the Add a Program dialog, there is a listing of most applications on the machine, but note that not all of them show up on this list. Use the "Browse" button to find other executables installed on the computer.

Note: Only EXE files are added to the exceptions list. For in-process OPC Servers and Clients (DLLs and OCXs) you will need to add the EXE applications that call them to the list instead.



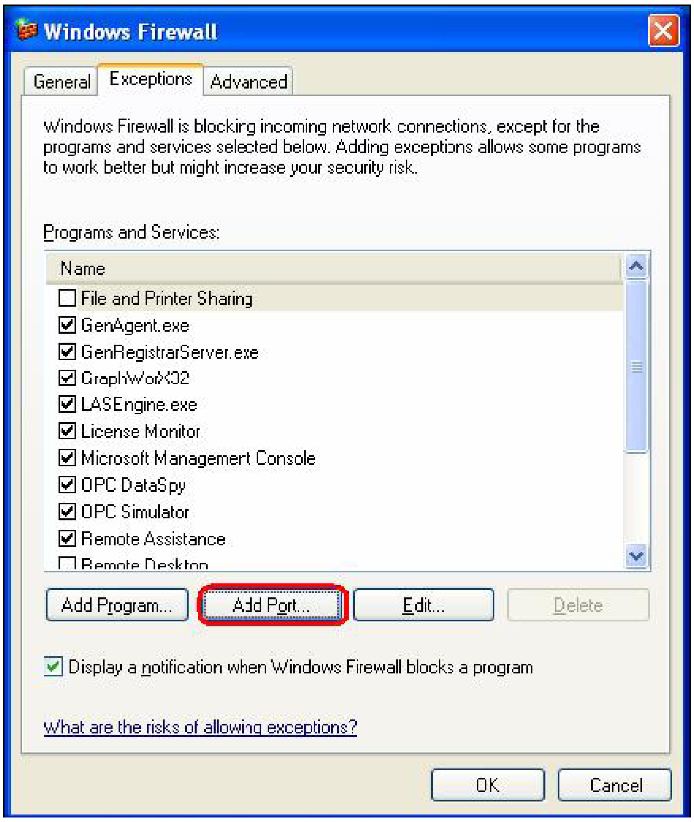
To allow communications with a program by adding it to the Exceptions list, select the program, or click Browse to search for one that is not listed.

Programs:

- Alarm Logger Configurator
- Alarm Server Configurator
- AlarmWorX32
- Carousel
- DataWorX32
- DBOPCServerConfigurator
- DrDCOM
- FreeCell
- GenDOS3 to GFW16
- GenDOS4 to GFW16
- GenStatistics Viewer

Path: C:\Program Files\ICONICS\GENESIS-32\Bin\

Step 3: Add TCP port 135 as it is needed to initiate DCOM communications, and allow for incoming echo requests. In the Exceptions tab of the Windows Firewall, click on Add Port.



Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

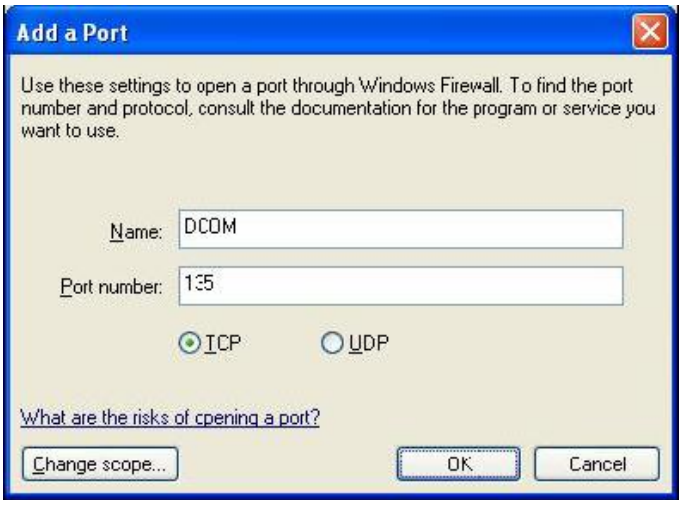
Programs and Services:

Name

☐ File and Printer Sharing
☑ GenAgent.exe
☑ GenRegistrarServer.exe
☑ GraphWorX32
☑ LASEngine.exe
☑ License Monitor
☑ Microsoft Management Console
☑ OPC DataSpy
☑ OPC Simulator
☑ Remote Assistance
☐ Remote Desktop

☑ Display a notification when Windows Firewall blocks a program

In the Add a Port dialog, fill out the fields as follows:
Name: DCOM
Port number: 135
Choose the TCP radio button



Use these settings to open a port through Windows Firewall. To find the port number and protocol, consult the documentation for the program or service you want to use.

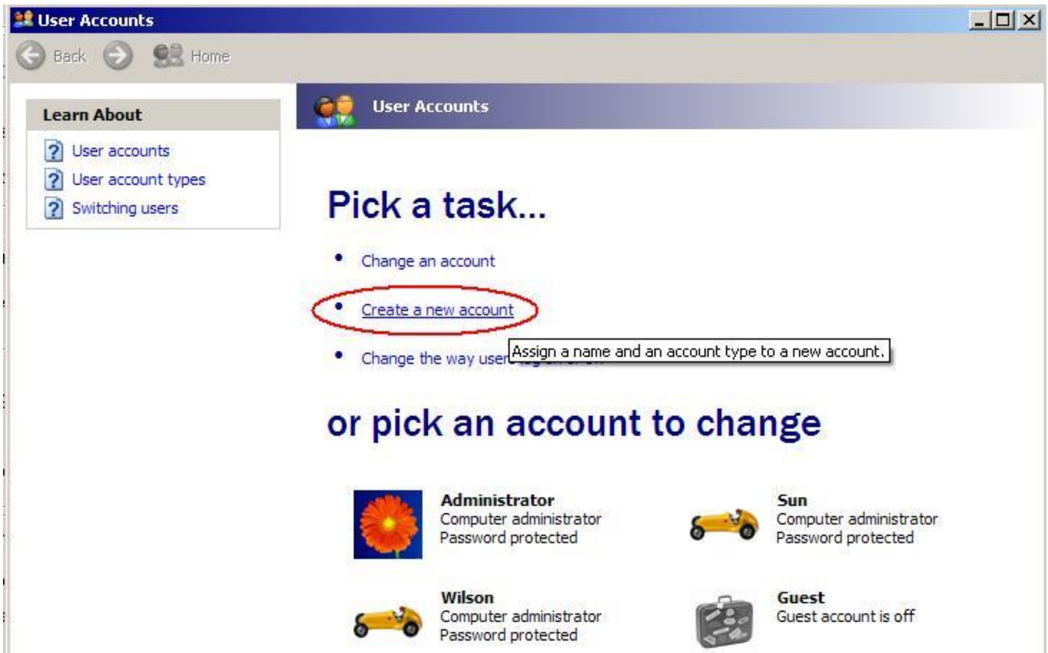
Name: DCOM

Port number: 135

● TCP ○ UDP

Creating the Account

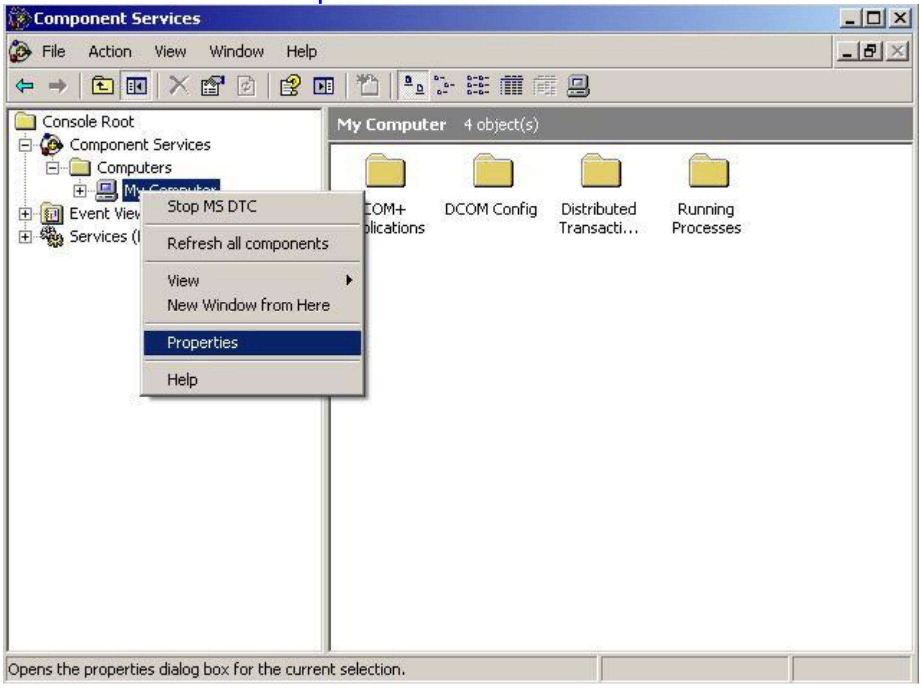
Step 1: Create a account which must be the same with the account of server site.



Assign a name and an account type to a new account.

Configuring DCOM

Step 1: Run the dcomcnfg.exe program to launch component services. Right clieck "My Computer" and choose "Properties".

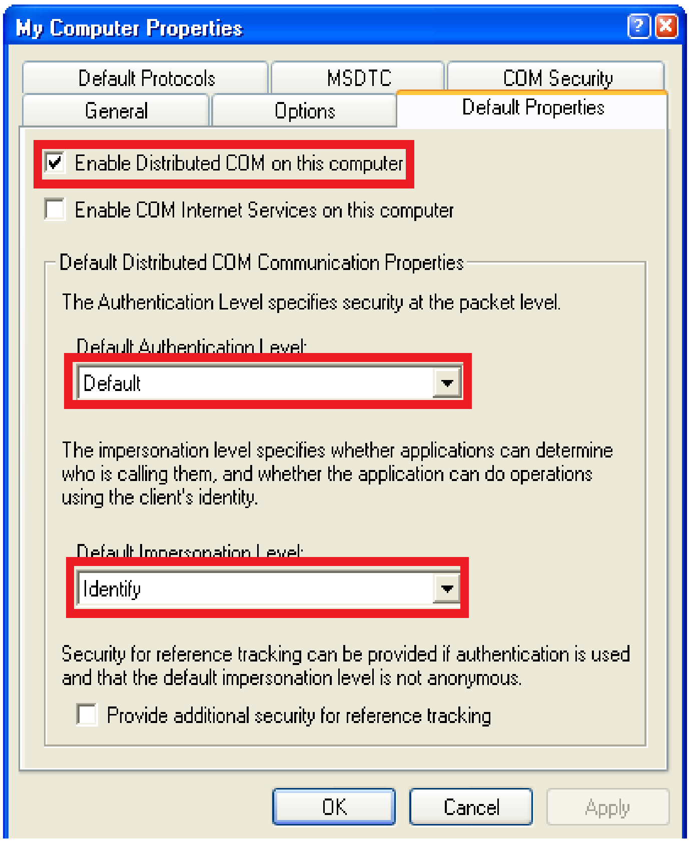


Opens the properties dialog box for the current selection.

Step 2: Select the "Default Properties" tab page.

Step 3: Use the following settings:

Field Name	Set to
Enable Distributed COM on this computer	Checked
Default Authentication Level:	Default
Default Impersonation Level:	Identify



My Computer Properties

☑ Enable Distributed COM on this computer

□ Enable COM Internet Services on this computer

Default Distributed COM Communication Properties

The Authentication Level specifies security at the packet level.

Default Authentication Level: Default

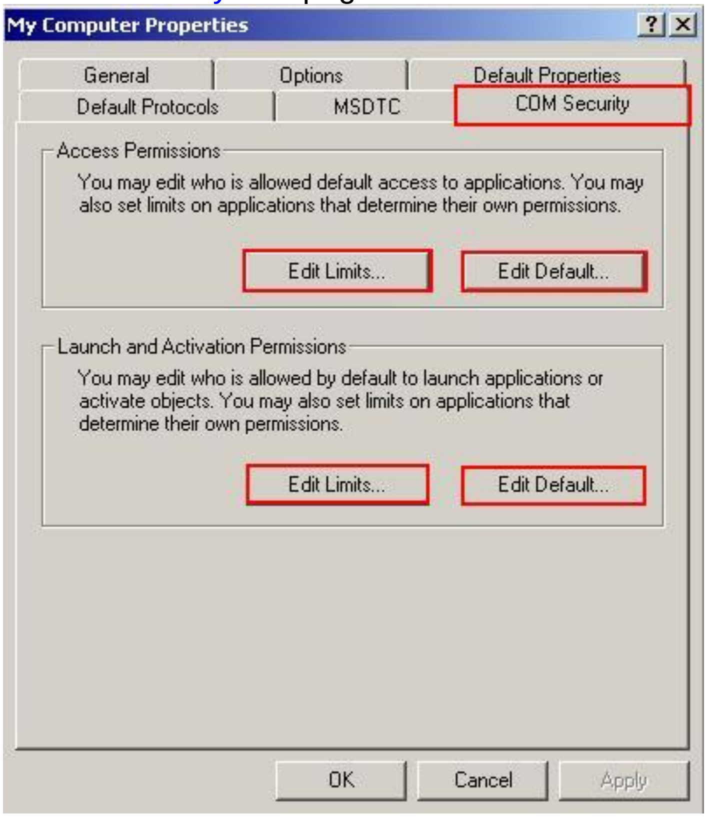
The impersonation level specifies whether applications can determine who is calling them, and whether the application can do operations using the client's identity.

Default Impersonation Level: Identify

Security for reference tracking can be provided if authentication is used and that the default impersonation level is not anonymous.

□ Provide additional security for reference tracking

Step 4: Select the "COM Security" tab page.



My Computer Properties

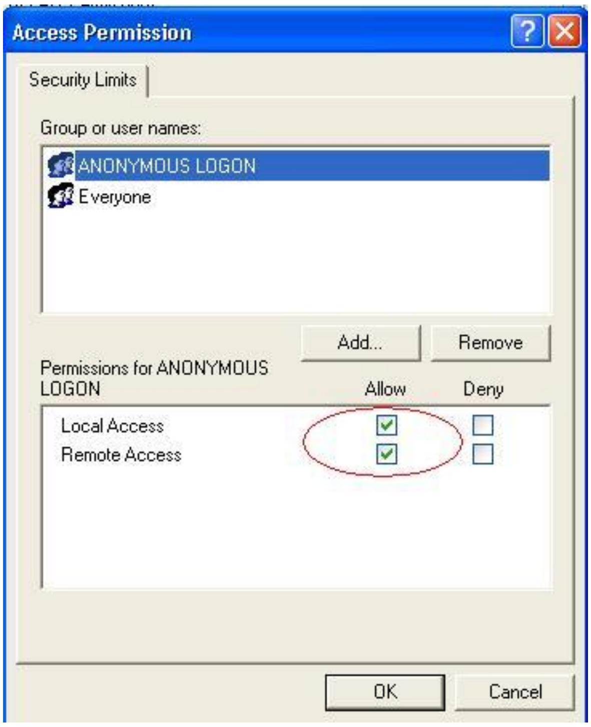
Access Permissions

You may edit who is allowed default access to applications. You may also set limits on applications that determine their own permissions.

Launch and Activation Permissions

You may edit who is allowed by default to launch applications or activate objects. You may also set limits on applications that determine their own permissions.

Step 5: Click on the “Edit Limits…” of “Access Permissions” button to set.

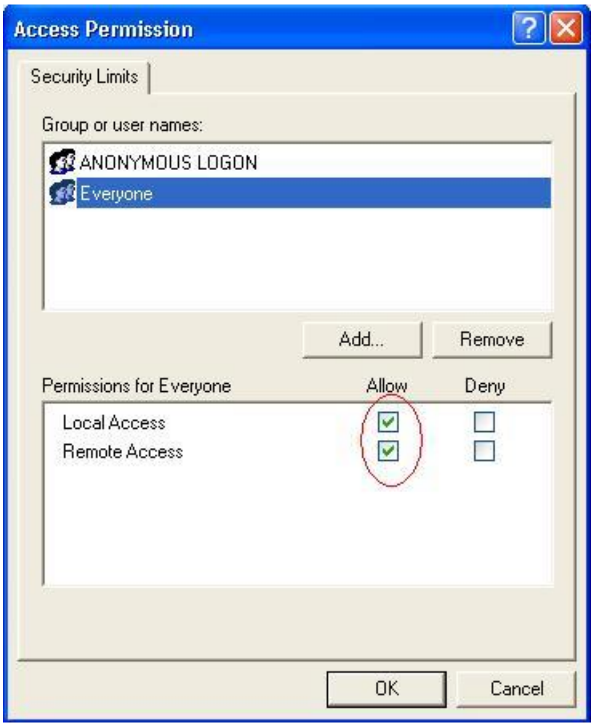


Group or user names:

ANONYMOUS LOGON

Everyone

Permissions for ANONYMOUS LOGON	Allow	Deny
Local Access	☑	□
Remote Access	☑	□



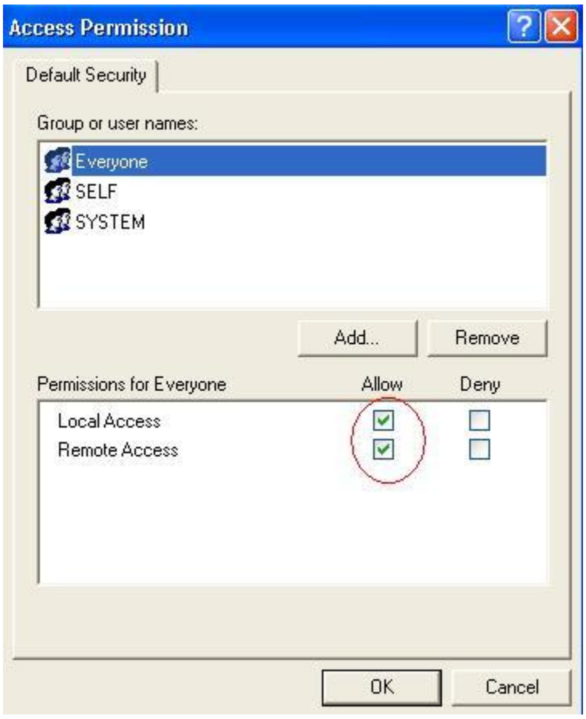
Group or user names:

ANONYMOUS LOGON

Everyone

Permissions for Everyone	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

Step 6: Click on the "Edit Default…" of "Access Permissions" button to set.

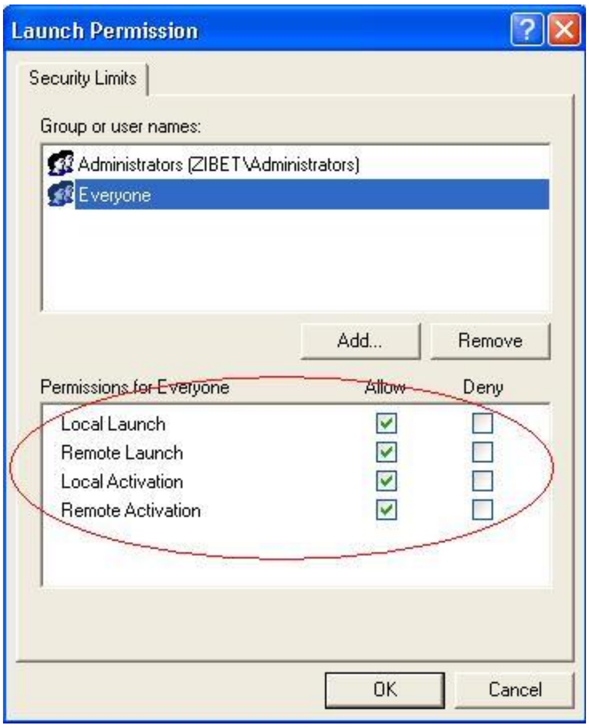


Group or user names:

- Everyone
- SELF
- SYSTEM

Permissions for Everyone	Allow	Deny
Local Access	☑	□
Remote Access	☑	□

Step 7: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.

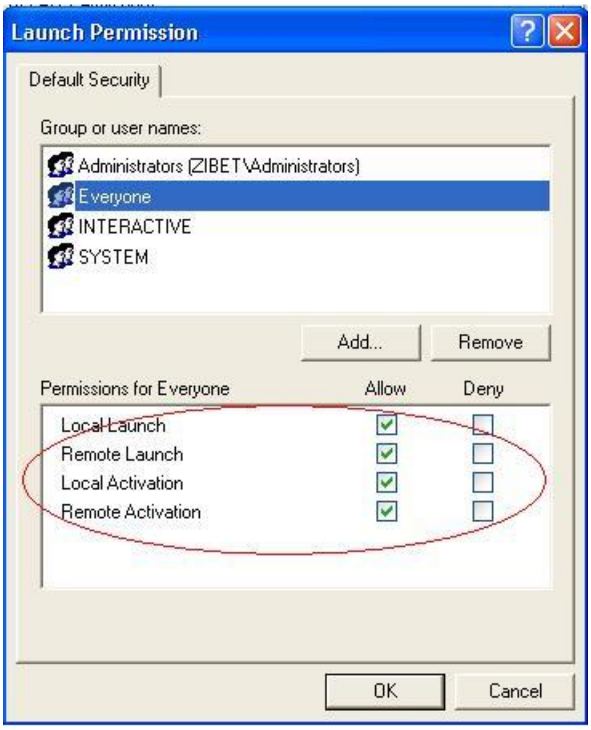


Group or user names:

- Administrators (ZIBET\Administrators)
- Everyone

Permissions for Everyone	Allow	Deny
Local Launch	☑	□
Remote Launch	☑	□
Local Activation	☑	□
Remote Activation	☑	□

Step 8: Click on the "Edit Limits…" of "Launch and Activation Permissions" button to set.



Launch Permission

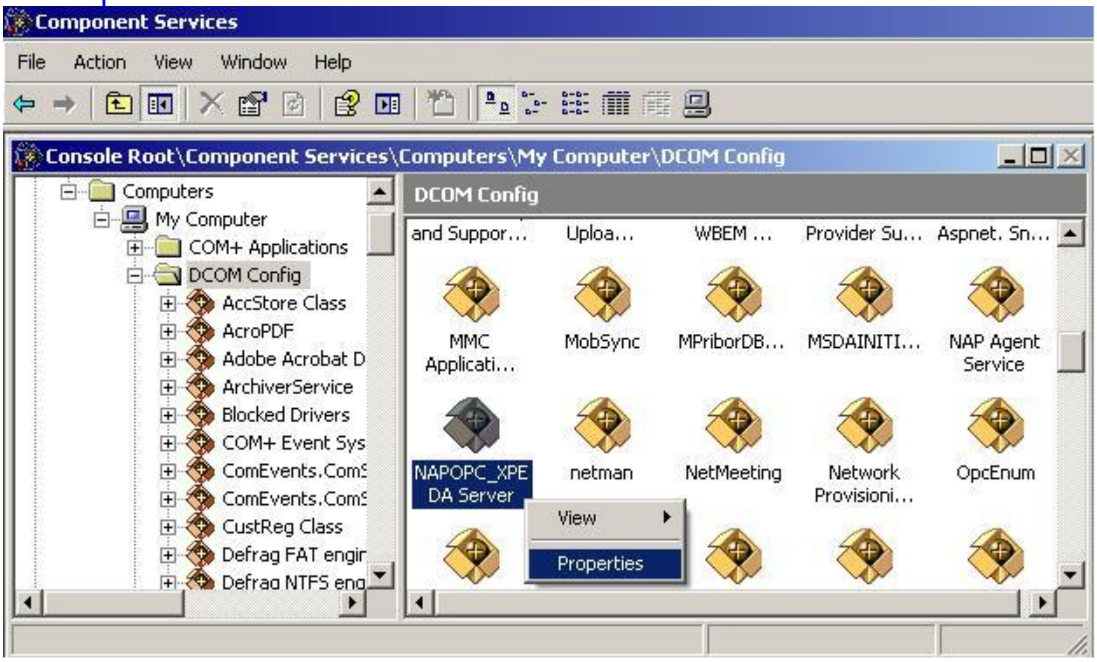
Default Security

Group or user names:

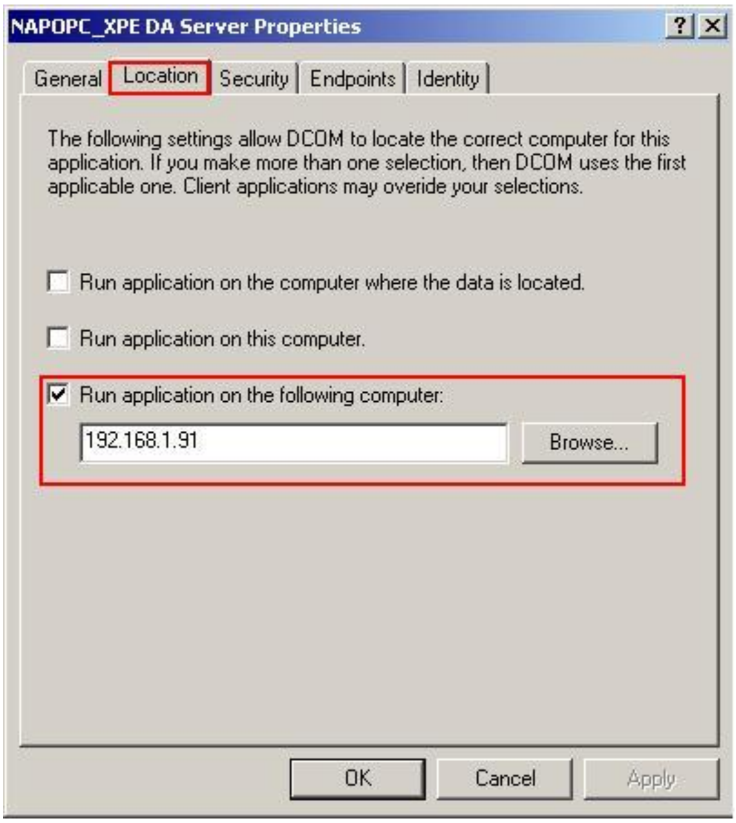
- Administrators (ZIBET\Administrators)
- Everyone
- INTERACTIVE
- SYSTEM

Permissions for Everyone	Allow	Deny
Local Launch	☑	□
Remote Launch	☑	□
Local Activation	☑	□
Remote Activation	☑	□

Step 9: Right click on the "NAPOPC_XPE DA Server" of "DCOM Config" button and select "Properties".



Step 10: Select the "Location" tab page and check "Run application on the following computer". And enter the Server IP here.



NAPOPC_XPE DA Server Properties

The following settings allow DCOM to locate the correct computer for this application. If you make more than one selection, then DCOM uses the first applicable one. Client applications may overide your selections.

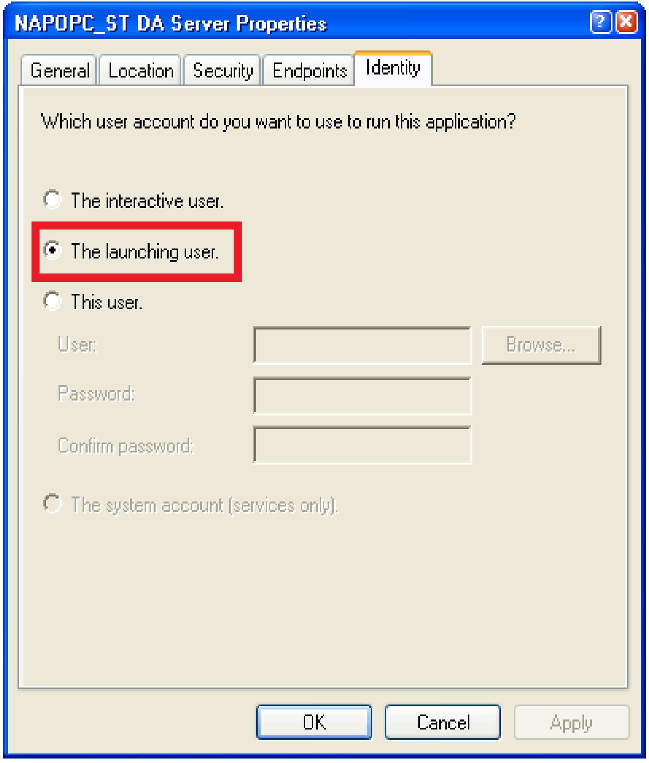
□ Run application on the computer where the data is located.

□ Run application on this computer.

☑ Run application on the following computer:

192.168.1.91

Step 11: Select the "Identity" tab page and check "The launching user"



NAPOPC_ST DA Server Properties

Which user account do you want to use to run this application?

○ The interactive user.

● The launching user.

○ This user.

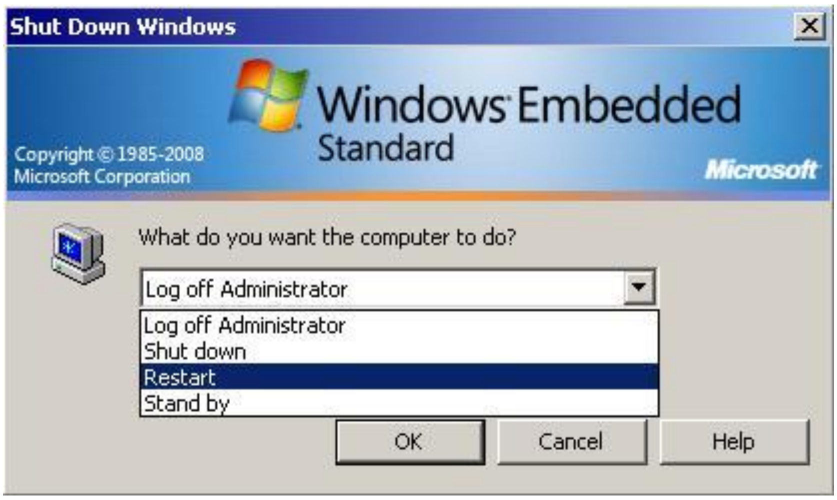
User:

Password:

Confirm password:

○ The system account (services only).

Step 12: Restart XPC



Shut Down Windows

Copyright © 1985-2008
Microsoft Corporation

What do you want the computer to do?

Log off Administrator

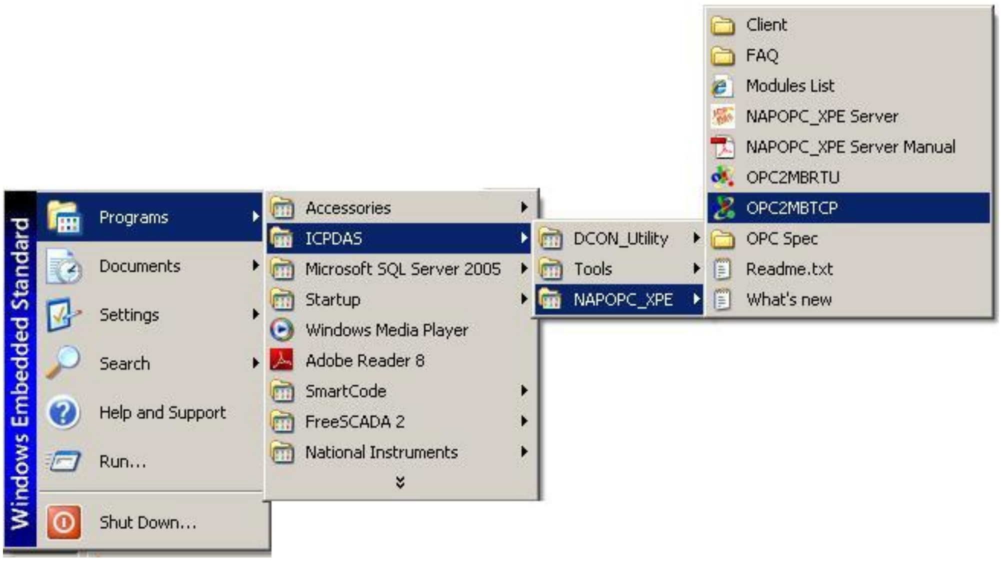
5 OPC To Modbus Slave

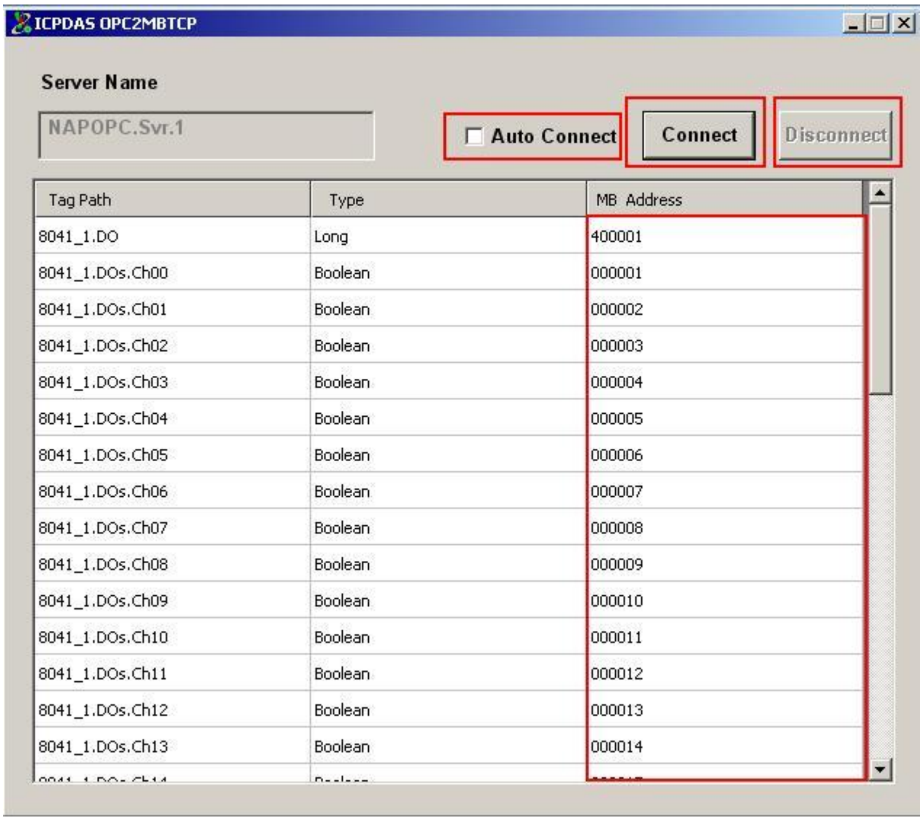
5.1 OPC To Modbus TCP Slave

Except NAPOPC_XPE DA Server, the OPC solution of XPAC provides the interface of "**OPC**" to "**Modbus TCP Slave**" to transform OPC data to Modbus TCP format. This interface allows you define OPC tag by modbus address to communicate with Modbus TCP master application. Furthermore, this interface can add in "Startup" area to execute while XPAC starting up.

5.1.1 Configuring OPC2MBTCP

OPC2MBTCP provides a clear and easy-to-use user interface to transform the OPC to Modbus TCP. You just need to press "Connect" button to connect to NAPOPC_XPE Server and generate the modbus mapping address automatically. You can use your modbus master application to read and write OPC tags via these modbus address. Or you can modify these modbus addresses by double clicking the field of "MB Address". If you want to modify modbus address, you **have to** disconnect to NAPOPC_XPE Server first by pressing "Disconnect" button. If you want OPC2MBTCP to connect to NAPOPC_XPE Server automatically while starting up, please check "Auto Connect" before closing OPC2MBTCP.





Tag Path	Type	MB Address
8041_1.DO	Long	400001
8041_1.DOs.Ch00	Boolean	000001
8041_1.DOs.Ch01	Boolean	000002
8041_1.DOs.Ch02	Boolean	000003
8041_1.DOs.Ch03	Boolean	000004
8041_1.DOs.Ch04	Boolean	000005
8041_1.DOs.Ch05	Boolean	000006
8041_1.DOs.Ch06	Boolean	000007
8041_1.DOs.Ch07	Boolean	000008
8041_1.DOs.Ch08	Boolean	000009
8041_1.DOs.Ch09	Boolean	000010
8041_1.DOs.Ch10	Boolean	000011
8041_1.DOs.Ch11	Boolean	000012
8041_1.DOs.Ch12	Boolean	000013
8041_1.DOs.Ch13	Boolean	000014

Server Name:

This name is "NAPOPC.Svr.1" and can not be modified in this program.

Auto Connect:

You can check it if you want OPC2MBTCP to connect to NAPOPC_XPE Server automatically while starting up.

Connect:

Connect to NAPOPC_XPE Server and generate the modbus address mapping of OPC tag.

Disconnect:

Disconnect to NAPOPC_XPE Server. If you want to modify modbus address, you **have to** disconnect to NAPOPC_XPE Server first.

MB Address:

Specifies the tag address. It must be the same with the the variable address of modbus master application

Modbus Type	**Range**
Output Coil	000001 - 065536
Input Coil	100001 - 165536
Input Register	300001 - 365536
Output Register	400001 - 465536

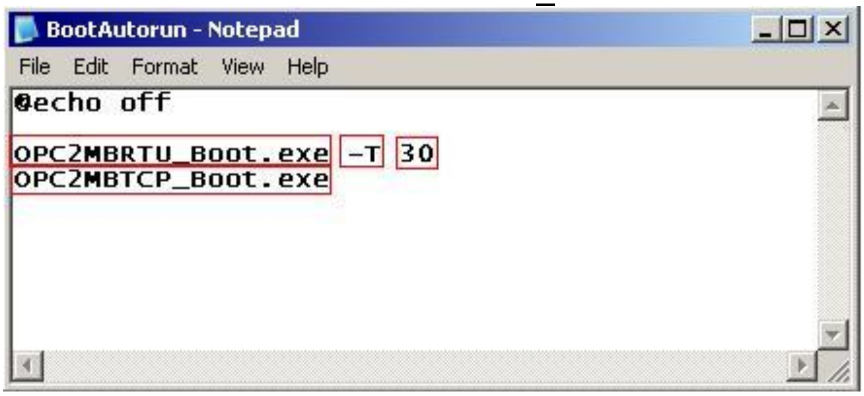
☒:

Close OPC2MBTCP.

5.1.2 Startup OPC2MBTCP

If you want to launch OPC2MBTCP, please follow these steps as below.

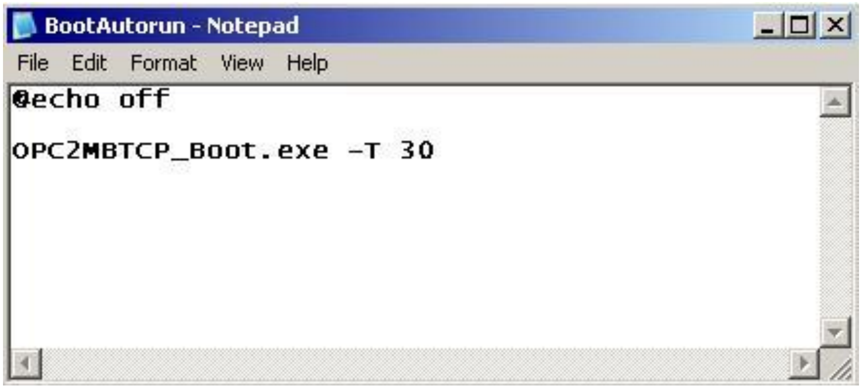
Step 1: Edit "BootAutorun.bat" in \\Root\NAPOPC_XPE\Client



```
@echo off

OPC2MBRTU_Boot.exe -T 30
OPC2MBTCP_Boot.exe
```

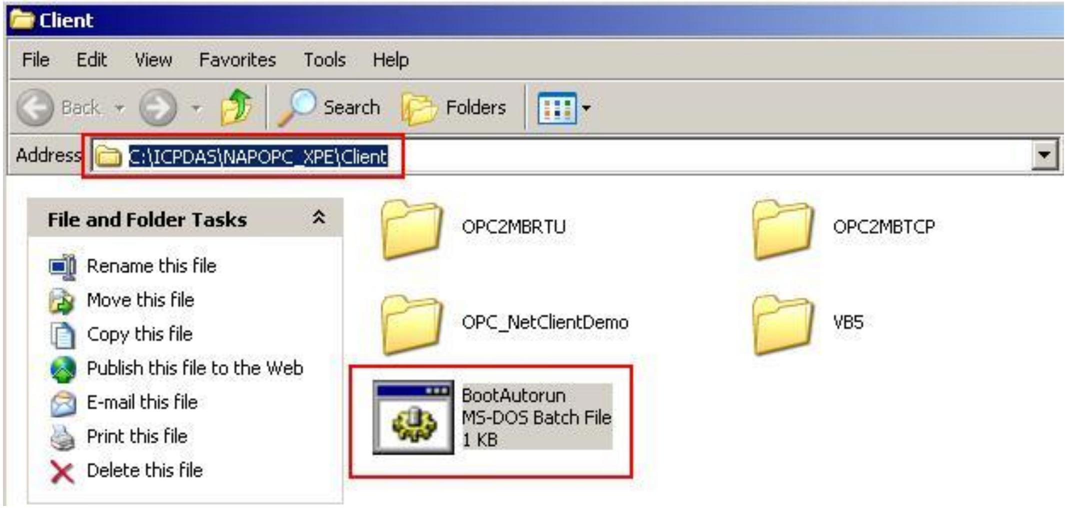
If you want to execute "OPC2MBTCP" while XPAC starting up, please write "OPC2MBTCP_Boot.exe" in this file. The option "-T" means delay time(sec.) while XPAC starting up. Because XPAC needs some time to initial some devices, OPC2MBTCP has to add delay time to delay the execution of OPC2MBTCP. Usually you have to add "-T 30" which means 30 sec. delay for the first program. For this example, the first program is "OPCMBRTU_Boot.exe". If you only want to launch "OPC2MBTCP_Boot.exe", you have to add the option behind it as below.

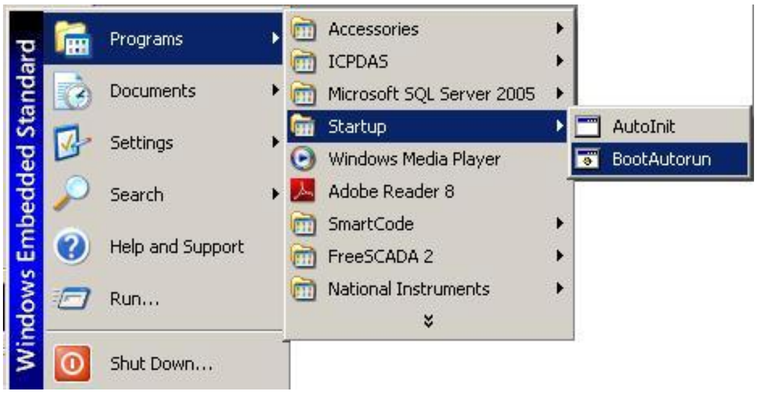


```
@echo off

OPC2MBTCP_Boot.exe -T 30
```

Step 2: Copy "BootAutorun.bat" in \\Root\NAPOPC_XPE\Client to "Startup" area.





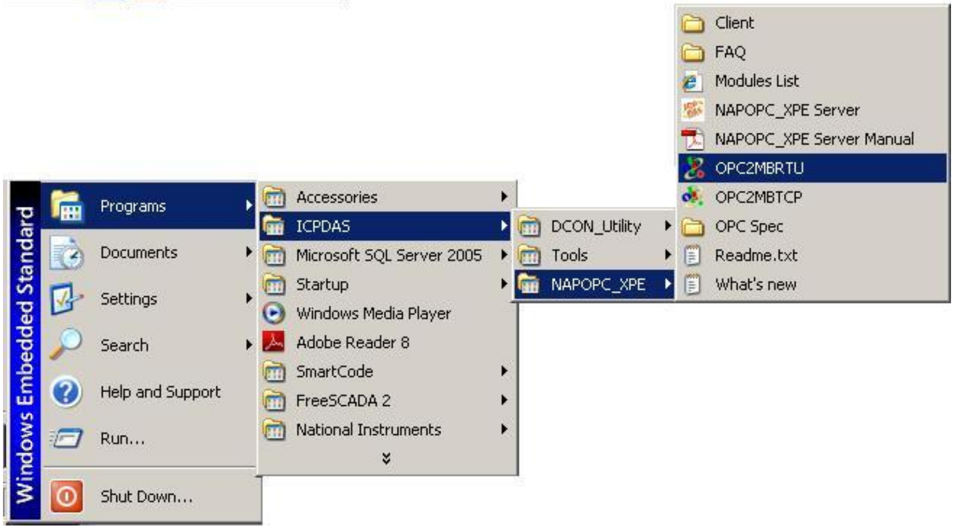
Step 3: That is all. You can test by restarting XPAC.

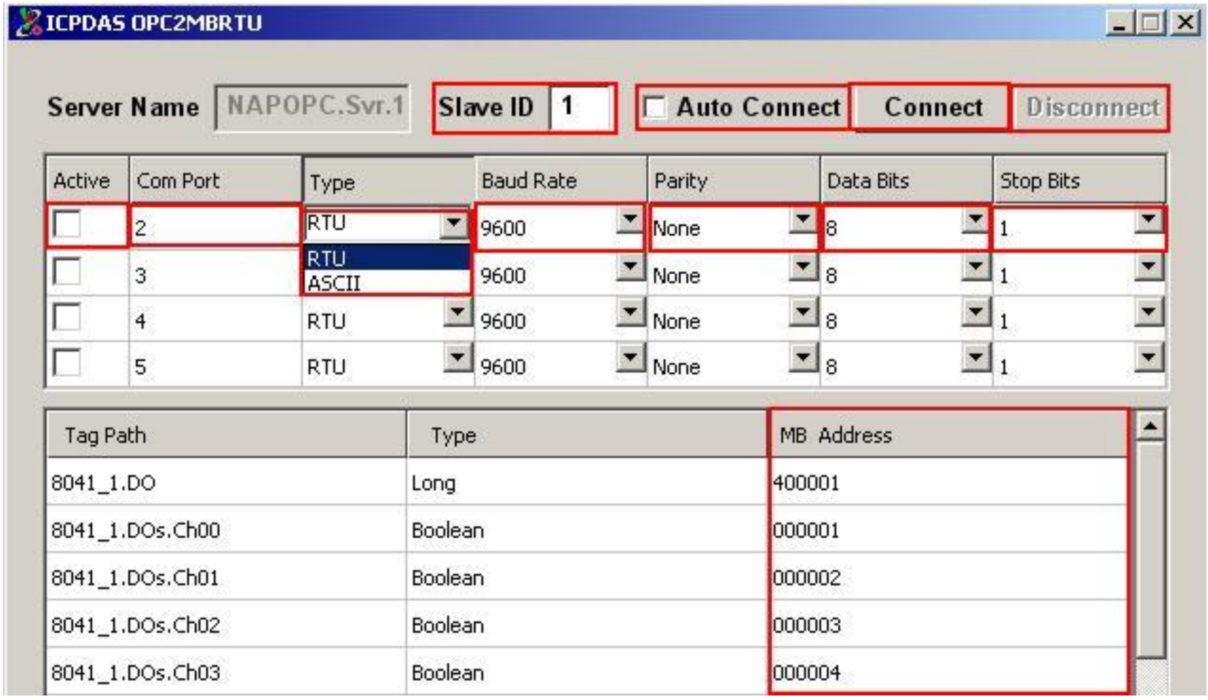
5.2 OPC To Modbus RTU/ASCII Slave

Except NAPOPC_XPE DA Server, the OPC solution of XPAC provides the interface of "**OPC**" to "**Modbus RTU/ASCII Slave**" to transform OPC data to Modbus RTU/ASCII format. This interface allows you define OPC tag by modbus address to communicate with Modbus RTU/ASCII master application. Furthermore, this interface can add in "Startup" area to execute while XPAC starting up.

5.2.1 Configuring OPC2MBRTU

OPC2MBRTU provides a clear and easy-to-use user interface to transform the OPC to Modbus RTU/ASCII. You just need to press "Connect" button to connect to NAPOPC_XPE Server and generate the modbus mapping address automatically. Naturally, you have to decide the slave ID, which COM port you want to use and which parameters of this COM port. OPC2MBRTU allows you to use four COM ports at a time. After setting up these parameters, you can use your modbus master application to read and write OPC tags via these modbus address. Or you can modify these modbus addresses by double clicking the field of "MB Address". If you want to modify modbus address, you **have to** disconnect to NAPOPC_XPE Server first by pressing "Disconnect" button. If you want OPC2MBRTU to connect to NAPOPC_XPE Server automatically while starting up, please check "Auto Connect" before closing OPC2MBRTU.





Server Name: NAPOPC.Svr.1

Slave ID: 1

□ Auto Connect

Active	Com Port	Type	Baud Rate	Parity	Data Bits	Stop Bits
□	2	RTU	9600	None	8	1
□	3		9600	None	8	1
□	4	RTU	9600	None	8	1
□	5	RTU	9600	None	8	1

Tag Path	Type	MB Address
8041_1.DO	Long	400001
8041_1.DOs.Ch00	Boolean	000001
8041_1.DOs.Ch01	Boolean	000002
8041_1.DOs.Ch02	Boolean	000003
8041_1.DOs.Ch03	Boolean	000004

Server Name:

This name is "NAPOPC.Svr.1" and can not be modified in this program.

Auto Connect:

You can check it if you want OPC2MBRTU to connect to NAPOPC_XPE Server automatically while starting up.

Connect:

Connect to NAPOPC_XPE Server and generate the modbus address mapping of OPC tag.

Disconnect:

Disconnect to NAPOPC_XPE Server. If you want to modify modbus address, you **have to** disconnect to NAPOPC_XPE Server first.

MB Address:

Specifies the tag address. It must be the same with the the variable address of modbus master application

Modbus Type	Range
Output Coil	000001 – 065536
Input Coil	100001 – 165536
Input Register	300001 – 365536
Output Register	400001 – 465536

Active:

To activate this COM port.

COM port:

Specifies the COM port to be used. Please verfiy which COM port number that the RS-232 / RS-485 network is using.

Type:

Specifies the service of Modbus RTU protocol or Modbus ASCII protocol.

Baud rate:

Specifies the baud rate to be used. Verify the module's current baud rate. A wrong setting will always cause communication error for this controller.

Parity:

Specifies the parity scheme to be used. It is one of the following values.

Value	Description
None	No parity
Even	Even
Odd	Odd

Data Bits:

Specifies the number of bits in the bytes transmitted and received.

Stop Bits:

Specifies the number of stop bits to be used. It is one of the following values.

Value	Description
1	1 stop bit
2	2 stop bits
1.5	1.5 stop bits

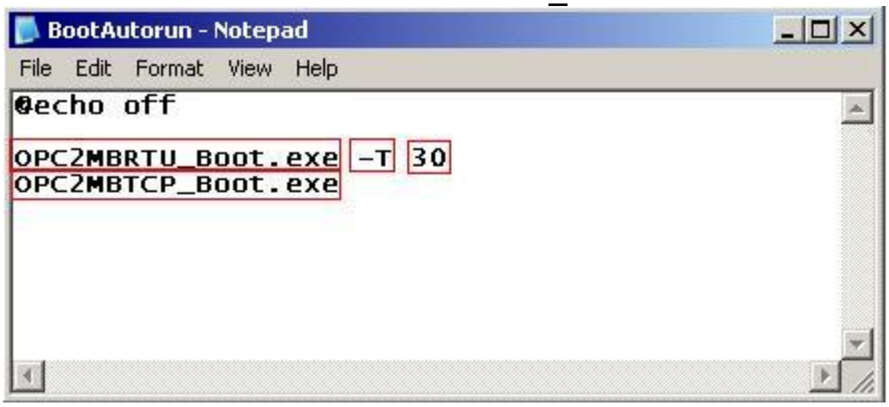
☒:

Close OPC2MBRTU.

5.2.2 Startup OPC2MBRTU

If you want to launch OPC2MBRTU, please follow these steps as below.

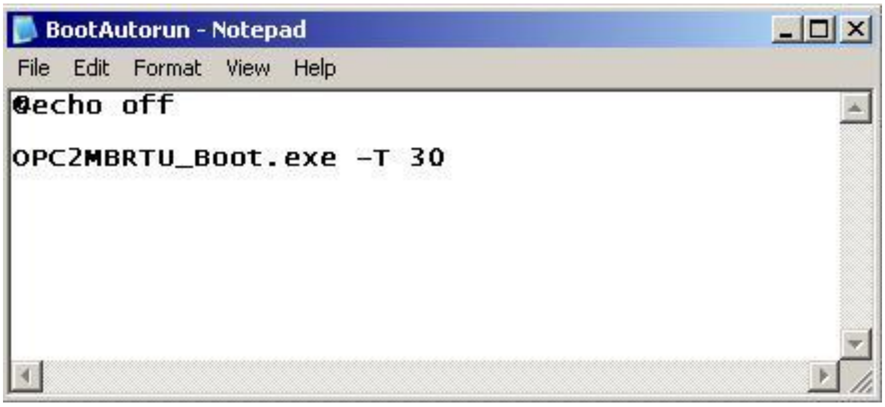
Step 1: Edit "BootAutorun.bat" in \\Root\NAPOPC_XPE\Client



```
@echo off

OPC2MBRTU_Boot.exe -T 30
OPC2MBTCP_Boot.exe
```

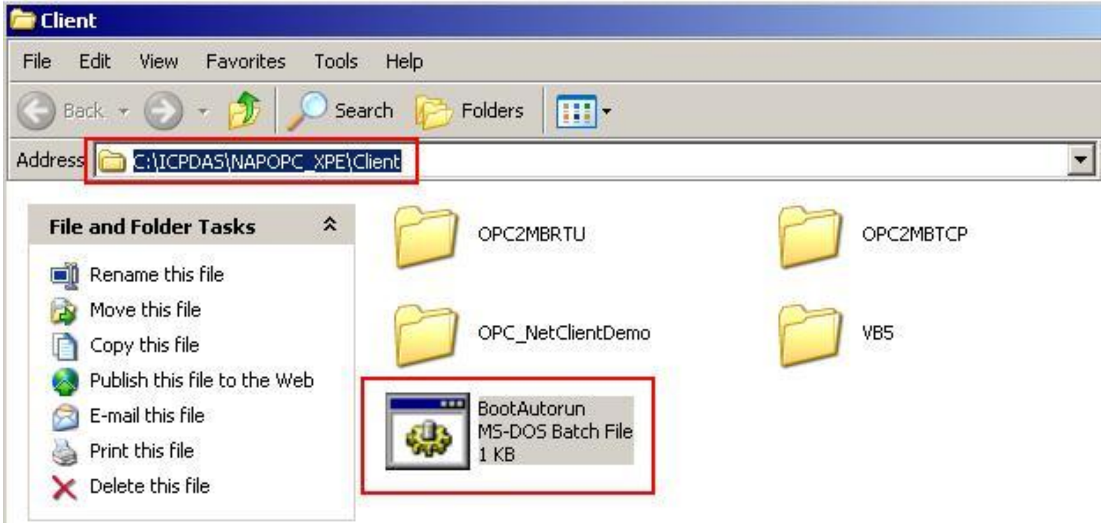
If you want to execute “OPC2MBRTU” while XPAC starting up, please write “OPC2MBRTU_Boot.exe” in this file. The option “-T” means delay time(sec.) while XPAC starting up. Because XPAC needs some time to initial some devices, OPC2MBRTU has to add delay time to delay the execution of OPC2MBRTU. Usually you have to add “-T 30” which means 30 sec. delay for the first program. For this example, the first program is "OPCMBRTU_Boot.exe". If you only want to launch "OPC2MBRTU_Boot.exe", you have to add the option behind it as below.

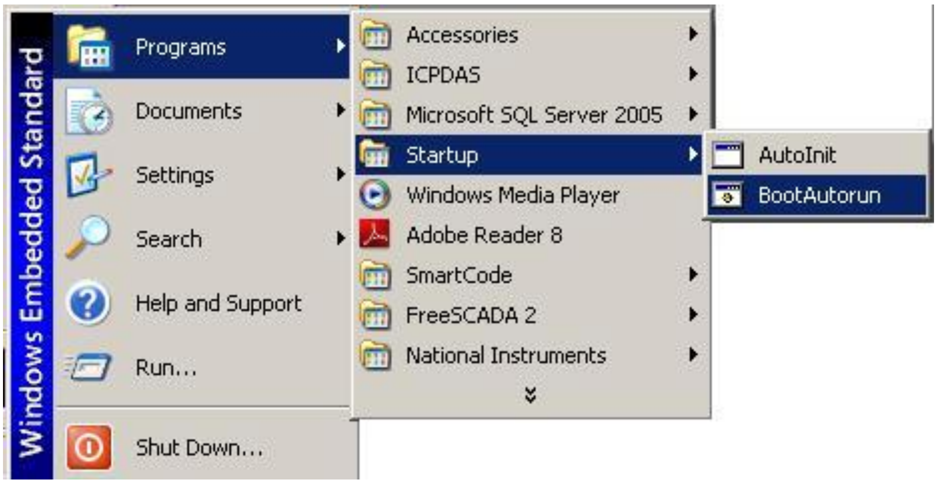


```
@echo off

OPC2MBRTU_Boot.exe -T 30
```

Step 2: Copy "BootAutorun.bat" in \\Root\NAPOPC_XPE\Client to "Startup" area.





Step 3: That is all. You can test by restarting XPAC.

6 Writing Client Program with VB

6.1 Programming with VB5

6.1.1 Overview of OPC & VB

Visual Basic language supports COM(**C**omponent **O**bject **M**odel). COM implementation from Visual Basic use what is called an "Automation" interface. The OPC Foundation supplies the source code of "Automation Wrapper" DLL which lets VB access OPC Servers and their underlying Groups and items. That's why we have to install the OPC DAC (Data Access Component.) software(see 2 Quick Start). After we install the OPC DAC software, we can use Automation Wrapper connects VB to OPC.

The following figure shows the architecture of object model for the automation wrapper. Because the OPC Server Object contains Group Objects and Items Objects by using Collection, OPC Browse Object can access the item data through the pointer of OPC Server Object.

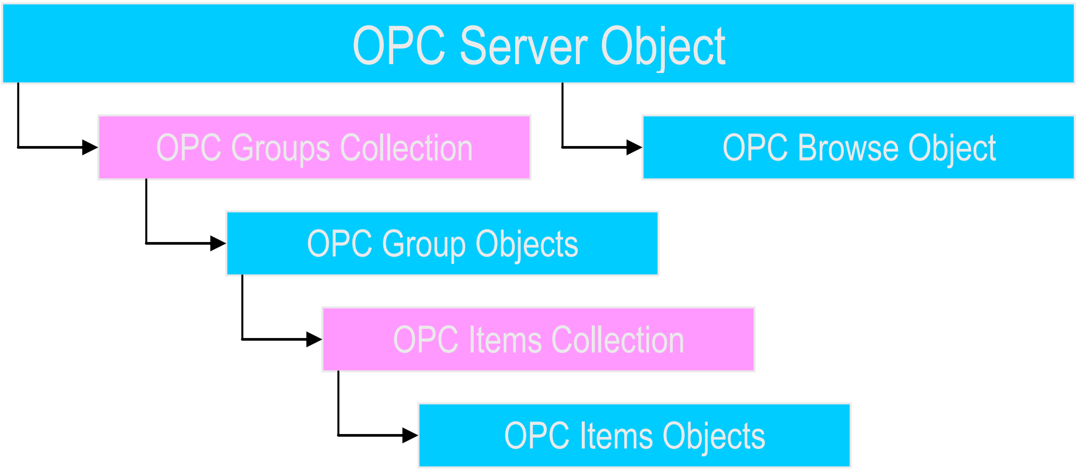


Figure 6-1-1. Object model for the Automation Wrapper

The following figure shows the architecture of OPC Server Object through the Automation Wrapper under COM/DCOM mechanism. The VB program wakes up the remote OPC Server Object through the automation wrapper object by DCOM mechanism.

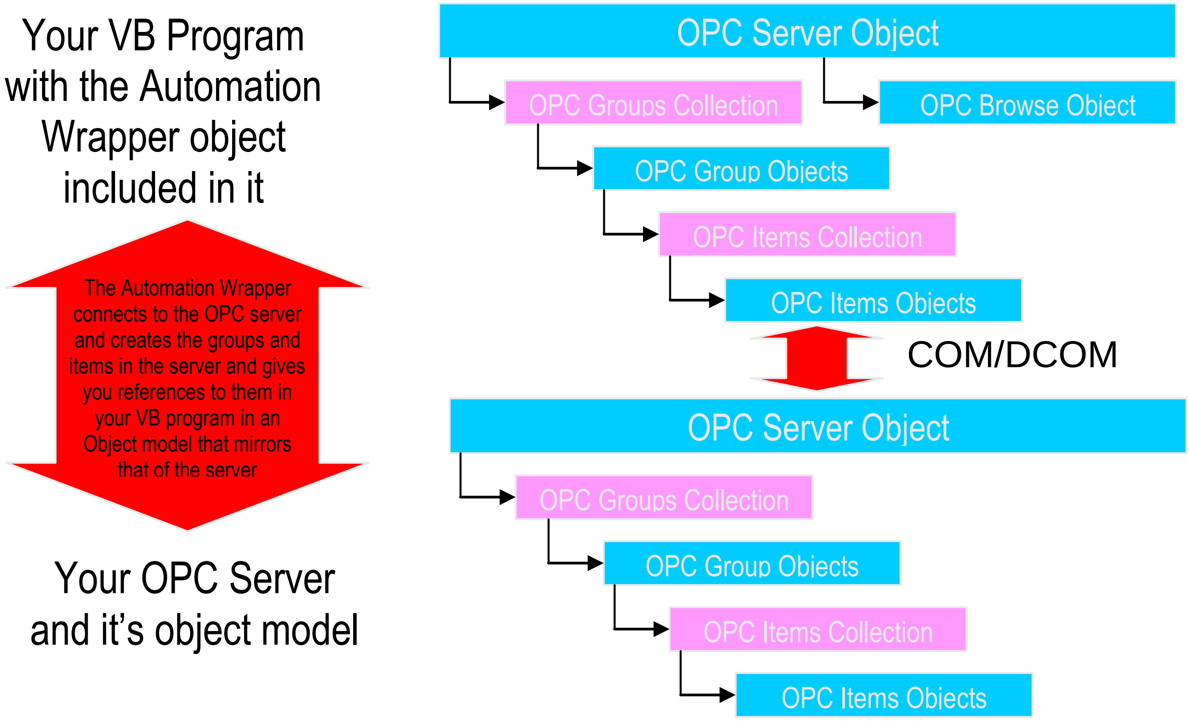

Your VB Program with the Automation Wrapper object included in it
The Automation Wrapper connects to the OPC server and creates the groups and items in the server and gives you references to them in your VB program in an Object model that mirrors that of the server
Your OPC Server and it's object model
OPC Server Object
OPC Groups Collection
OPC Browse Object
OPC Group Objects
OPC Items Collection
OPC Items Objects
COM/DCOM
OPC Server Object
OPC Groups Collection
OPC Group Objects
OPC Items Collection
OPC Items Objects

Figure 6-1-2. Architecture of OPC Server Object under COM/DCOM mechanism

6.1.2 Tools You Will Need to Build Your VB Client

If you want to build an OPC client in VB and test it, you will need the following tools.

- **Visual Basic 5 or 6** running on Windows 95/98/2000/ME/NT/XP
- **An OPC Server**
  At this manual, we use ICP DAS NAPOPC_XPE Server as the demo. You can find it at your CD:\\xp-8000\tools\ NAPOPC_XPE or you can download it from the http://opc.icpdas.com/download.htm
- **The OPC Automation Wrapper**
  You can find it from http://opc.icpdas.com/download.htm

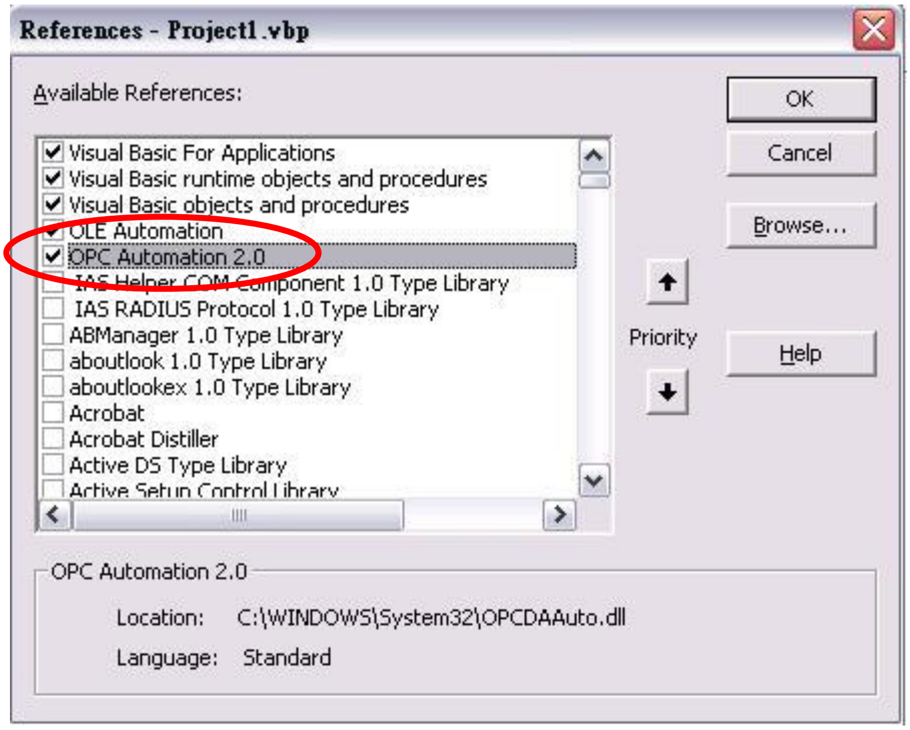
6.1.3 Building Your VB Client – Step By Step

At this section, we just focus on the key steps of building the VB client. If you want to know more information about OPC Automation 2.0 and the VB demo, please refer to the opcda20_auto.pdf in the C:\ICPDAS\NAPOPC_XPE\Manual and the VB demo source code in the C:\ICPDAS\NAPOPC_XPE\Client\VB5.

Step 1:

- Install OPCDAC on your PC
- Start a new VB project
- In VB, click on Project -> References on the VB menu bar

- The OPC Automation Wrapper appears on the dialog as "OPC Automation 2.0" – select it as shown here



Available References:

☑ Visual Basic For Applications
☑ Visual Basic runtime objects and procedures
☑ Visual Basic objects and procedures
☑ OLE Automation
☑ OPC Automation 2.0
□ IAS Helper COM Component 1.0 Type Library
□ IAS RADIUS Protocol 1.0 Type Library
□ ABManager 1.0 Type Library
□ aboutlook 1.0 Type Library
□ aboutlookex 1.0 Type Library
□ Acrobat
□ Acrobat Distiller
□ Active DS Type Library
□ Active Setup Control Library

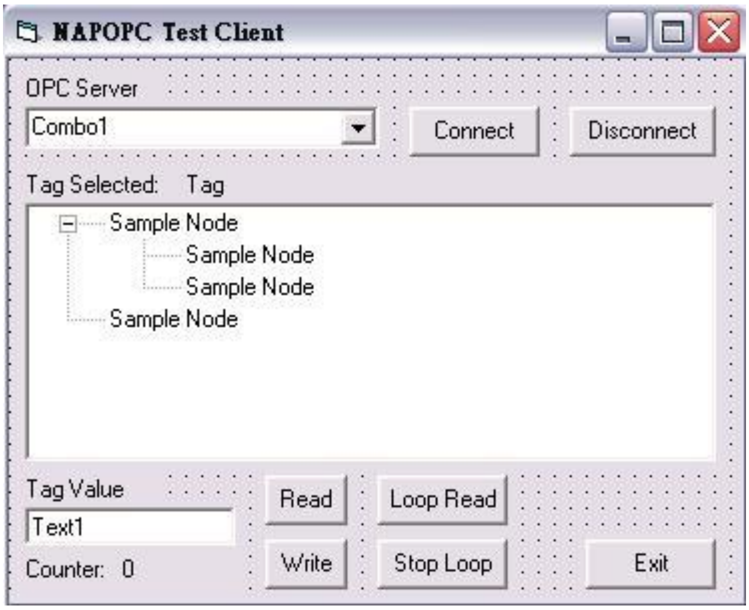
OPC Automation 2.0

Location: C:\WINDOWS\System32\OPCDAAuto.dll

Language: Standard

Step 2:

First, you have to design your UI(User Interface). You can refer to the UI demo of VB program shown as below. Next, you need to declare some variables at the General Declarations area of VB code window. The most important types of variables are OPCServer, OPCGroup, and OPCBrowser. As the declaration, we can use several functions to read/write item values through Server and Group variables.



```
'Declare a new OPC Server object
Public Server As OPCServer
'Declare a new OPC Group object
Public Group As OPCGroup
```

```
'Declare a new Browser object
Public browser As OPCBrowser
```

Step 3:

You can call GetOPCServers() to scan the OPC Servers at your PC as the following codes.

```
'Declare a Variant Variable
Dim Servers As Variant
'Create a new OPC Server object
Set Server = New OPCServer
'Call GetOPCServers to scan the OPC Servers on your PC
Servers = Server.GetOPCServers("")
'Show the servers on the Combo box
Dim lastIndex As Integer
lastIndex = 0
For I = LBound(Servers) To UBound(Servers)
   cbServerList.AddItem Servers(I)
   If Servers(I) = lastServer Then lastIndex = I - 1
Next I
cbServerList.ListIndex = lastIndex
MousePointer = vbDefault
```

Step 4:

Next, you'll go ahead and add the code rights after you get your connection to the NAPOPC_XPE Server. Please refer to the FillItems and Branch subroutine of VB demo program.

```
'Generate the tree of tags
Private Sub FillItems()
'Populate the sub branches in the browser tree
Public Sub Branch(Count As Integer, node1 As node)
```

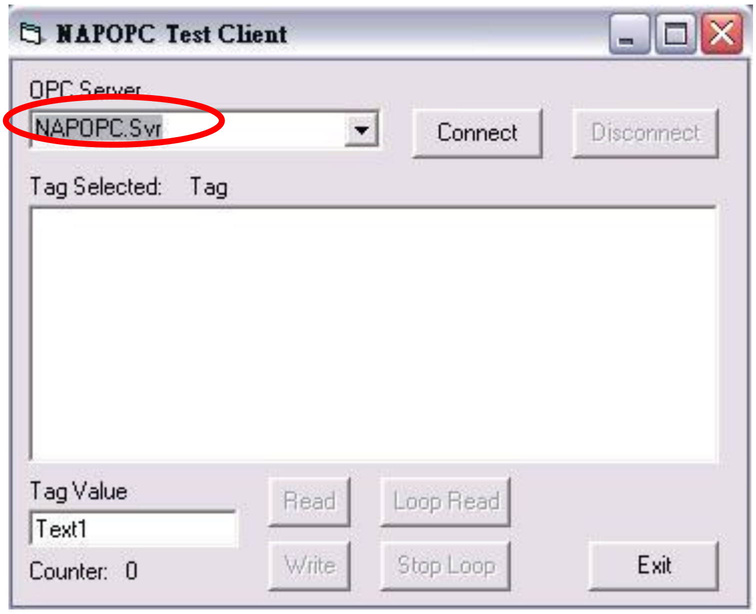
Step 5:

Now, you can add the code for the "Read" button and "Write" button. Please refer to the btnRead_Click and btnWrite_Click subroutine of VB demo program. In these two functions, the *anItem.Read* and the *anItem.Write* are two key methods.

```
'Read the OPCItem value after the read button press
Private Sub btnRead_Click()
'Write the value in the text box after the write button press
Private Sub btnWrite_Click()
```

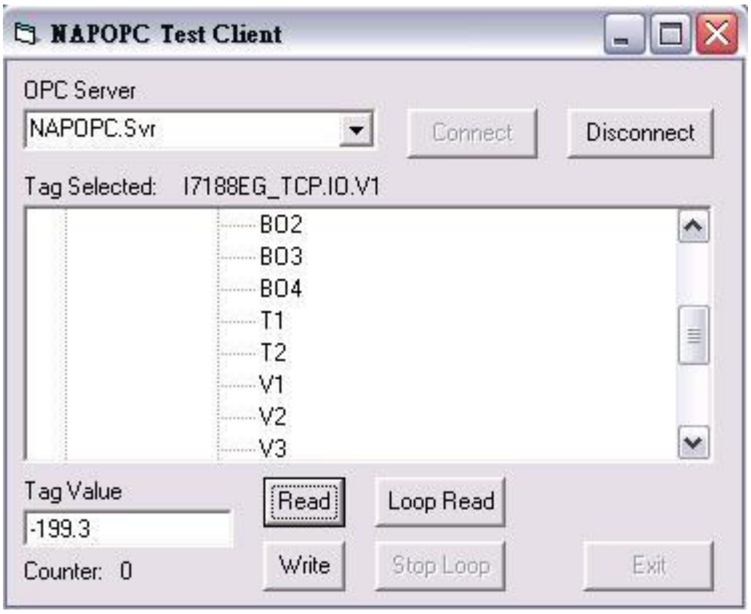
Step 6:

You can build the project and you will see the UI as below.



Step 7:

After you click on the "Connect" button, you will see the OPC Server tree list. You can choose one of them and click on the "Read" button. You will see the item value at the "Tag Value" field as below. You can also type the value you want to write in the text box and click on the "Write" button. (Refer to 4.1 Client Demo Program)



6.2 Programming with .Net

6.2.1 Limitations about .Net client programming

1. OPC DA component 2.0 must be installed. (In this case, OPC DA Component 1.0 can't be used)

2. Development was done on a Windows XP SP1 system using Microsoft Development Environment 2003 version 7.1.3091 with .Net Framework 1.1 version 1.1.4322 and any new release version of .Net will need to modify these codes.

3. Testing was done on following operation system, any others might not work
   - Windows 98 second edition 4.10.2222A
   - Windows 2000 professional 5.00.2195 service pack 4
   - Windows XP professional version 2002 service pack 1

6.2.2 Tools – You Need to Build Your .Net Client

If you want to build an OPC VB .Net or VC# client and test it, you will need following tools.

- **Visual Basic .Net 2003** or newer version running on Windows 98/2000/XP

- **OPC Automation 2.0**
  You can install NAPOPC_XPE DA Server(From CD:\\xp-8000\tools\NAPOPC_XPE or download it from http://opc.icpdas.com/download.htm). The NAPOPC_XPE DA Server installation will install OPC Automation 2.0 automatically.

- **An OPC Server**
  At this manual, we use ICPDAS NAPOPC_XPE Server as the demo. You can find it at your CD:\\xp-8000\tools\NAPOPC_XPE or download it from the http://opc.icpdas.com/download.htm

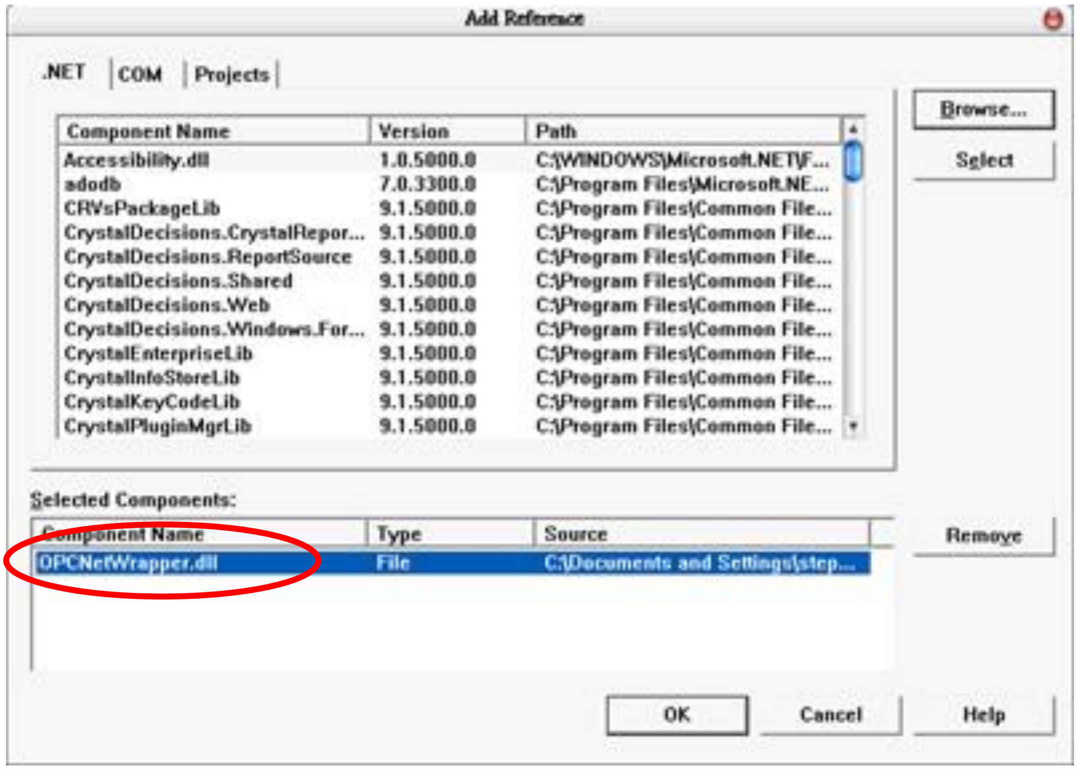
- OPC .Net wrapper named “**OPCNetWrapper.dll**”.
  After you install NAPOPC_ST DA Server (From http://opc.icpdas.com/download.htm), you can find it in Root\\ ICPDAS\ NAPOPC_XPE\Client\OPC_NetClientDemo\VBOPCClient_Demo

- **OPCNETWrapper.pdf**
  After you install NAPOPC_XPE DA Server (From http://opc.icpdas.com/download.htm), you can find it in Root\\ICPDAS\NAPOPC_XPE\Manual

6.2.3 Building Your VB.Net Client – Step By Step

Step 1:

- Install OPC DA Component 2.0 on your PC
- Start a new VB.Net project with “Windows Application Program”
- In VB.Net, click Project -> Add References on the VB.Net menu bar
- Click Browse… to select “OPCNetWrapper.dll” as shown below

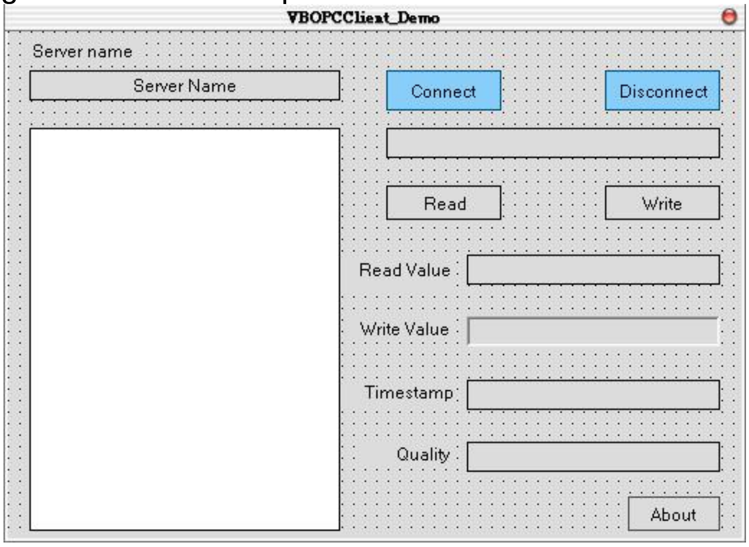


Component Name	Version	Path
Accessibility.dll	1.0.5000.0	C:\WINDOWS\Microsoft.NET\F...
adodb	7.0.3300.0	C:\Program Files\Microsoft.NE...
CRVsPackageLib	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.CrystalRepor...	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.ReportSource	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.Shared	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.Web	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.Windows.For...	9.1.5000.0	C:\Program Files\Common File...
CrystalEnterpriseLib	9.1.5000.0	C:\Program Files\Common File...
CrystalInfoStoreLib	9.1.5000.0	C:\Program Files\Common File...
CrystalKeyCodeLib	9.1.5000.0	C:\Program Files\Common File...
CrystalPluginMgrLib	9.1.5000.0	C:\Program Files\Common File...

Component Name	Type	Source
OPCNetWrapper.dll	File	C:\Documents and Settings\step...

Step 2:

First, you must Import following declarations "ICPDAS.OPC","ICPDAS.OPCDA" and "ICPDAS.OPC.NET". Second, you could design your UI (User Interface). You can refer to the UI of VB.Net program shown as below. Third, you need to declare some variables at the General Declarations area of VB.Net code window. The most important types of variables are ICPDAS_OPCServer, ICPDAS_SyncIOGroup, and ServerTreeBrowser. As the declaration, we can use several functions to read/write tag values through Server and Group variables.



```
'Declare a new ICPDAS_OPCServer object
Dim Svr As ICPDAS_OPCServer
'Declare two new ICPDAS_SyncIOGroup objects
Dim ReadWriteGroup, Grp As ICPDAS_SyncIOGroup
'Declare a new ServerTreeBrowser object
```

```
Dim TagTree As ServerTreeBrowser
'Declare two new RefreshGroup objects
Dim AsyncRefrGroup, rGrp As RefreshGroup
‘Declare a new TagDef object
Dim TagData As TagDef
‘Declare a new String variable for OPC Server ProgID
Dim ServerName As String
```

Step 3:

You can set ServerName as “NAPOPC.Svr.1” and pass it to ICPDAS_OPCServer for connecting.

```
'Set a ProgID to ServerName
ServerName = "NAPOPC.Svr.1"
'Create a new OPC Server object
Svr = New ICPDAS_OPCServerr
'Connect to NAPOPC DA Server
Svr.Connect (ServerName)
```

Step 4:

Now, you’ll go ahead and add the code rights after you get your connection to the NAPOPC Server. Please refer to following TagReadWrite and DataChangeHandler subroutine of VB demo program.

```
 ‘Handles of data change callbacks
Public Sub DataChangeHandler (ByVal sender As Object, ByVal e As
DataChangeEventArgs)

Private Sub TagReadWrite (ByVal lgrp As ICPDAS_SyncIOGroup, ByVal lrgrp As
RefreshGroup, ByVal iTree As ServerTreeBrowser, ByVal tagId As String)
```

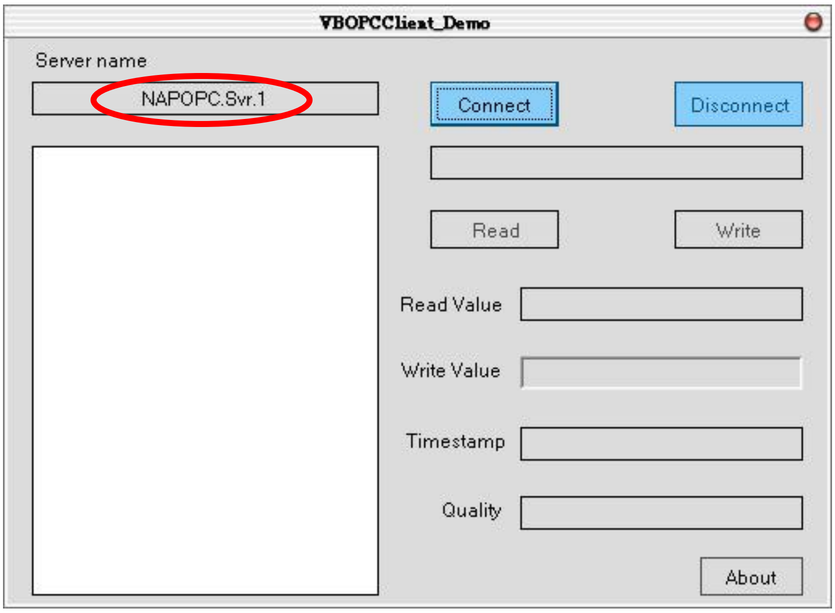
Step 5:

Now, you can add codes for the “Read” button, “Write” button and “TreeView” AfterSelect function. Please refer to the btnRead_Click, btnWrite_Click and tvTags_AfterSelect subroutine of VB.Net demo program. In these three functions, the *Grp.Read*, the *Grp.Write and the* TagTree.TagName are three key methods.

```
'Read the OPCTag value after the read button press
Private Sub btnRead_Click ()
'Write the value in the text box after the write button press
Private Sub btnWrite_Click ()
‘The action after selecting the tag
Private Sub tvTags_AfterSelect ()
```

Step 6:

You can build the project and you will see the UI as below.



Server name: NAPOPC.Svr.1

Read Value:

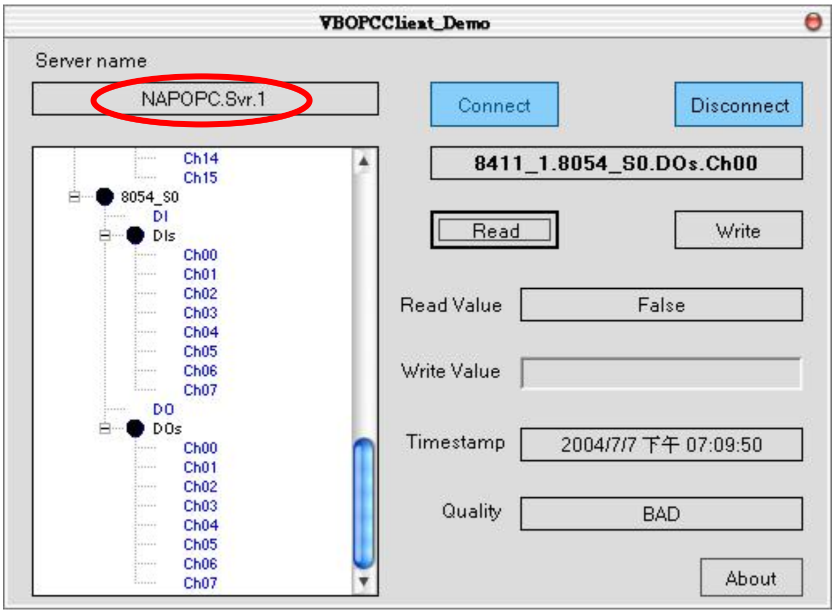
Write Value:

Timestamp:

Quality:

Step 7:

After you click on the "Connect" button, you will see the OPC Server tree list. You can choose one of them and click on the "Read" button. You will see the tag value at the "Read Value" field as below. You can also type the value you want to write in the "Write Value" and click on the "Write" button. (Refer to 4.2 .Net Client Demo Program)



Server name: NAPOPC.Svr.1

8411_1.8054_S0.DOs.Ch00

Read Value: False

Write Value:

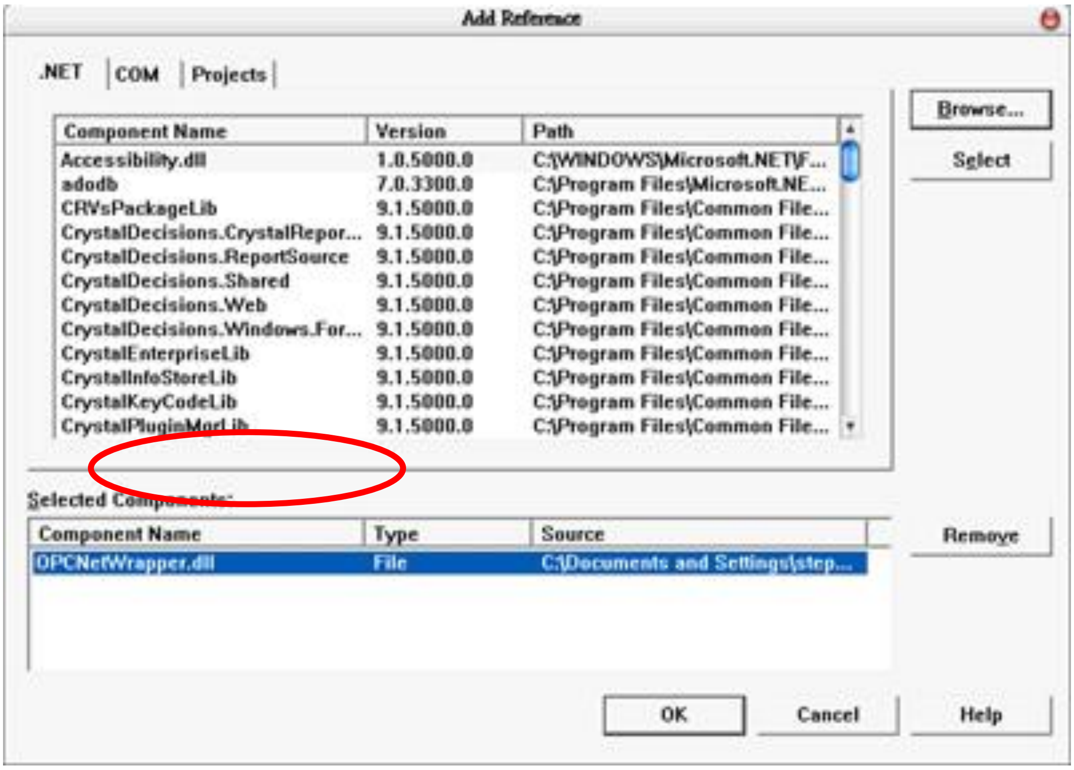
Timestamp: 2004/7/7 下午 07:09:50

Quality: BAD

6.2.4 Building Your VC#.Net Client – Step By Step

Step 1:

- Install OPC DA Component 2.0 on your PC
- Start a new VC#.Net project with "Windows Application Program"
- In VC#.Net, click Project -> Add References on the VC#.Net menu bar
- Click Browse… to select "OPCNetWrapper.dll" as shown below



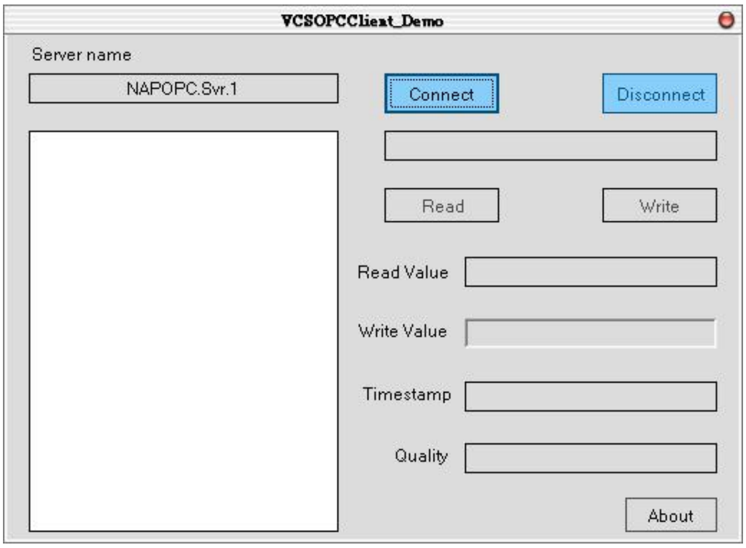
Component Name	Version	Path
Accessibility.dll	1.0.5000.0	C:\WINDOWS\Microsoft.NET\F...
adodb	7.0.3300.0	C:\Program Files\Microsoft.NE...
CRVsPackageLib	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.CrystalRepor...	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.ReportSource	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.Shared	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.Web	9.1.5000.0	C:\Program Files\Common File...
CrystalDecisions.Windows.For...	9.1.5000.0	C:\Program Files\Common File...
CrystalEnterpriseLib	9.1.5000.0	C:\Program Files\Common File...
CrystalInfoStoreLib	9.1.5000.0	C:\Program Files\Common File...
CrystalKeyCodeLib	9.1.5000.0	C:\Program Files\Common File...
CrystalPluginMgrLib	9.1.5000.0	C:\Program Files\Common File...

Selected Components:

Component Name	Type	Source
OPCNetWrapper.dll	File	C:\Documents and Settings\step...

Step 2:

First, you must using directive as following declarations: "ICPDAS.OPC","ICPDAS.OPCDA" and "ICPDAS.OPC.NET". Second, you could design your UI (User Interface). You can refer to the UI of VC#.Net program shown as below. Third, you need to declare some variables at the General Declarations area of VC#.Net code window. The most important types of variables are ICPDAS_OPCServer, ICPDAS_SyncIOGroup, and ServerTreeBrowser. As the declaration, we can use several functions to read/write tag values through Server and Group variables.



Server name
NAPOPC.Svr.1

```
'Declare a new ICPDAS_OPCServer object
ICPDAS_OPCServer                              Svr = null;
'Declare two new ICPDAS_SyncIOGroup objects
ICPDAS_SyncIOGroup                            ReadWriteGroup, Grp;
'Declare a new ServerTreeBrowser object
ServerTreeBrowser                             TagTree;
```

```
'Declare two new RefreshGroup objects
RefreshGroup                        AsyncRefrGroup, rGrp;
‘Declare a new TagDef object
TagDef                              TagData;
‘Declare a new String variable for OPC Server ProgID
string                              ServerName;
```

Step 3:

You can set ServerName as “NAPOPC.Svr.1” and pass it to ICPDAS_OPCServer for connecting.

```
'Set a ProgID to ServerName
ServerName = "NAPOPC.Svr.1;"
'Create a new OPC Server object
Svr = new ICPDAS_OPCServer ();
'Connect to NAPOPC DA Server
Svr.Connect (ServerName);
```

Step 4:

Now, you’ll go ahead and add the code rights after you get your connection to the NAPOPC Server. Please refer to following TagReadWrite and DataChangeHandler subroutine of VC# demo program.

```
 ‘Handles of data change callbacks
public void DataChangeHandler ( object sender, DataChangeEventArgs e )

public void TagReadWrite ( ICPDAS_SyncIOGroup grp, RefreshGroup rgrp,
ServerTreeBrowser iTree, string tagId)
```

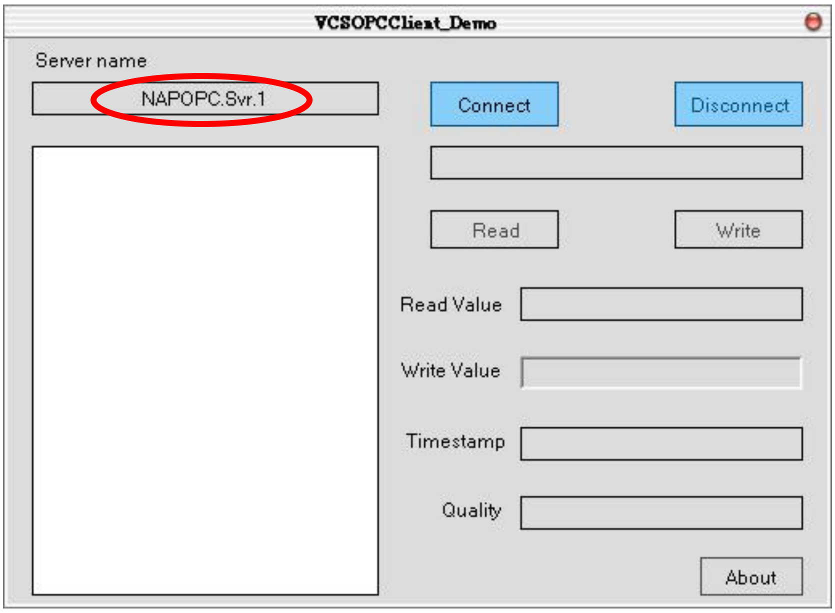
Step 5:

Now, you can add codes for the “Read” button, “Write” button and “TreeView” AfterSelect function. Please refer to the btnRead_Click, btnWrite_Click and tvTags_AfterSelect subroutine of VC#.Net demo program. In these three functions, the *Grp.Read*, the *Grp.Write and the* TagTree.TagName are three key methods.

```
'Read the OPCTag value after the read button press
private void btnRead_Click (object sender, System.EventArgs e)
'Write the value in the text box after the write button press
private void btnWrite_Click(object sender, System.EventArgs e)
‘The action after selecting the tag
private void tvTags_AfterSelect(object sender,
                                   System.Windows.Forms.TreeViewEventArgs e)
```

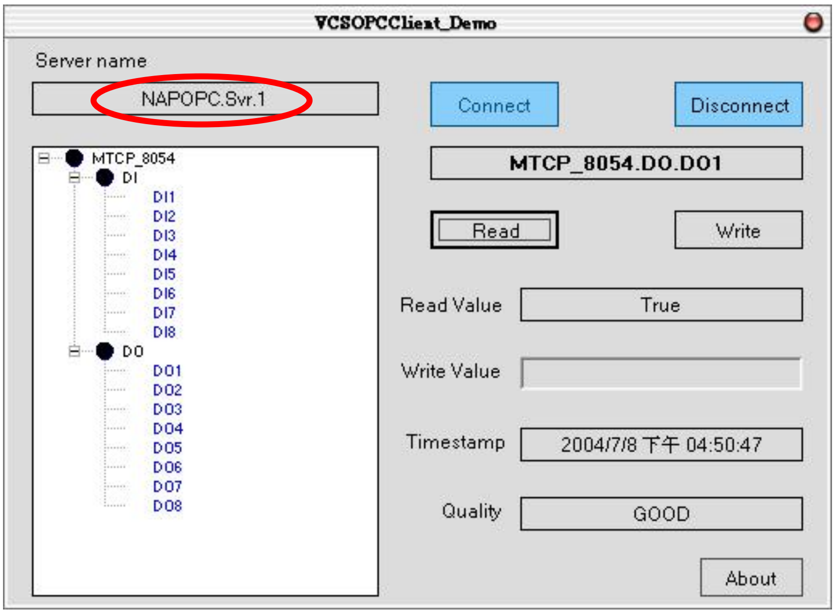
Step 6:

You can build the project and you will see the UI as below.



Step 7:

After you click on the "Connect" button, you will see the OPC Server tree list. You can choose one of them and click on the "Read" button. You will see the tag value at the "Read Value" field as below. You can also type the value you want to write in the "Write Value" field and click on the "Write" button. (Refer to 4.2 .Net Client Demo Program)



MTCP_8054.DO.DO1

Read Value: True

Write Value:

Timestamp: 2004/7/8 下午 04:50:47

Quality: GOOD

7 NAPOPC_XPE Changes List

7.1 Features of NAPOPC_XPE version 1.01

The new features of NAPOPC_XPE version 1.01 are briefly described in this section.

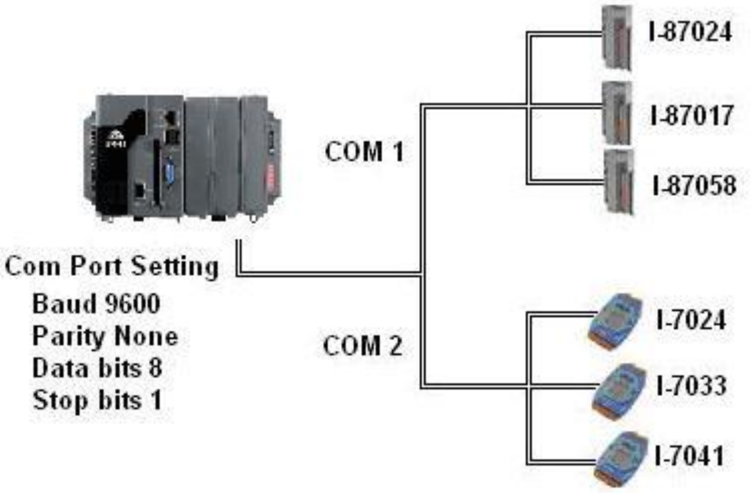
7.1.1 Customized Module/Device Polling Time

NAPOPC_XPE version 1.01 allows customers to optimize their communication by setting "Pending Time" in "Device Dialog ". Customers have chance to spread more time resource to other modules which are connected with each other. Please refer to "3.1 Optimize Your Communication" in detail.

7.1.2 Multi-Thread Communication

NAPOPC_XPE version 1.01 uses multi-thread architecture to organize module and device communication. Before NAPOPC_XPE version 1.01, NAPOPC_XPE uses single-thread to deal with module and device communication. Therefore, if NAPOPC_XPE connects many modules and devices, accessing time will increase observably. By NAPOPC_XPE version 1.01, customers have chances to group modules and devices into several parts, and connect via different COM port. One COM port works in one thread. All TCP communications belong to one thread.

Here is an example to show the difference of average scan time between NAPOPC_XPE version 1.01 and before NAPOPC_XPE version 1.01.



Under the same architecture, the average scan time in NAPOPC_XPE version 1.01 is 375 ms and 539 ms separately. Before NAPOPC_XPE version 1.01, the average scan time is 1109 ms.

7.1.3 Support Remote Procedure Call with Quicker/UPC

NAPOPC_XPE version 1.01 supports RPC(Remote Procedure Call) communication with Quicker and UPC(ICP DAS Universal Protocol Converter). Based on RPC service, NAPOPC_XPE can synchronize OLE data with Quicker and

UPC automatically. The behavior of changing data between "NAPOPC_CE5/CE6" and NAPOPC_XPE is not polling from NAPOPC_XPE but sending from "NAPOPC_CE5/CE6". This mechanism can transfer data effectively and shorten response time.

7.1.4 Switch of Single-Thread and Multi-Thread

NAPOPC_XPE version 1.01 supports "Communication Mechanism" option at "Options/ Configurate Initial Status". This option lets user define the communication behavior of NAPOPC_XPE. Generally, "Multi-Thread" is the best choice for high performace. However, for some particular OPC clients which can not work smoothly under "Multi-Thread" communication, user can choose "Single-Thread" instead.

7.2 Features of NAPOPC_XPE version 1.02

7.2.1 Bug Fix

- Fix "Unsupported_Device" bug when searching RU87PN
- Fix modbus tags showing error when tag property length is 4 bytes
- Fix coil type error in "OPC to Modbus client" program

7.3 Features of NAPOPC_XPE version 1.10

The new features of NAPOPC_XPE version 1.10 are briefly described in this section.

7.3.1 Support ET-7000 Search

NAPOPC_XPE version 1.10 supports "ET-7000 Search" option at "Add/Search Modules". This option lets NAPOPC_XPE can search not only the modules communicating via COM port but also ET-7000 modules via Ethernet automatically.

7.3.2 Support ZB-2K I/O

NAPOPC_XPE version 1.10 supports "ZB-2K" I/O modules at "Select Device" dialog. User can easily choose ZB-2K module ID to add ZB-2K module. And then use "Generate Tags" function to generate ZB-2K properties automatically.

7.3.3 Support FRnet Module

NAPOPC_XPE version 1.10 supports "FRnet" I/O modules at "Select Device" dialog. User can easily choose FRnet module ID to add FRnet module. And then use "Generate Tags" function to generate FRnet properties automatically.

7.3.4 Support Account Selection

NAPOPC_XPE version 1.10 supports "Recent File Source" selection at "Options/Configurate Initial Status" dialog. User can choose "Administrator Account" or "Current User Account" to decide registry location for some DCOM application scenario.

7.3.5 UI Modification

For more protocol support, NAPOPC_XPE version 1.10 modifies "Select Device" user interface. NAPOPC_XPE version 1.10 separates three groups of "DCON", "FRnet", and "Modbus" from one property sheet to clarify the user interface of device settings. Also, it limits the dialog size to an allowable boundary.

7.4 Features of NAPOPC_XPE version 1.12

The new features of NAPOPC_XPE version 1.12 are briefly described in this section.

7.4.1 Support FRnet module via RPC communication

NAPOPC_XPE version 1.12 supports "FRnet" I/O modules via RPC communication. User can easily create RPC device to synchonize "FRnet" I/O modules plugged in WinPAC(See 1.6.2 Adding A New RPC Controller).

8 Reference

FAQ.txt

The frequently asked questions and answers.
http://opc.icpdas.com/faq_xpe.htm

GetStart.PDF

This manual can be downloaded from below link.
http://www.icpdas.com/download/7000/manual.htm

OPCNetWrapper.PDF

This document describes how to use OPC .NET wrapper for .NET programming.

OPCOvw.PDF

The OPC overview - written by the OPC Foundation.

OPCCommn.PDF

The OPC common interface - specifications by the OPC Foundation.

OPCDA20_Cust.PDF

The OPC DA v2.0 custom interface - specifications by the OPC Foundation.

OPCDA20_Auto.PDF

The OPC DA v2.0 automation interface - specifications by the OPC Foundation.

OPC Foundation Web Site

http://www.opcfoundation.org/