Ethernet/IP Protocol

Examples of RTU32M/N Controller Setup

Application Note

April 2022

Brodersen RTU32M/N Series Hardware

- Allows rapid connection of Allen Bradley controllers with modern 'future-proof' Brodersen RTUs
- Runs on Brodersen RTU32M/RTU32N hardware
- All configuration is made via easy to use tables, that also include online diagnostics for viewing of PLC data
- Allows for staged upgrades of systems ie. maintain old protocols and comms technologies in parallel with new deployments and upgrades of networks / host SCADA systems
- Brodersen RTUs include a range of IP functionality such as in-built web server (for config and HMI / data presentation), SNMP, SNTP, FTP and VPN security
- Advanced RTU security features include DNP Secure Authentication (master and slave implementations)
- Open programming interface using IEC61131-3 logic with a range of program languages - including the ability to translate from one language to another eg. convert a program developed in structured text to ladder diagram
- A proven solution – Brodersen have been supplying industrial automation controllers and RTUs for more than 50 years!

Overview

Brodersen RTU32 hardware is used in a wide range of applications for the remote management of assets that form the critical infrastructure that keeps our cities and towns operational and safe. These applications include the monitoring and control of wind farms, electrical substations, airports, railways, tunnels, broadcast equipment, IT facilities, building automation, water distribution systems and oil/gas pipelines.

The RTU32M and RTU32N Series controllers are both DIN rail mount in modular and 'brick' style formats to suit the varied remote monitoring and control requirements of infrastructure applications. The controllers include a range of serial, USB and Ethernet ports and communicate using both IT and/or SCADA industry protocols. The suite of protocols include SNMP (with Traps), FTP, SNTP, Modbus, Profibus, BACnet, Ethernet/IP, Siemens S7, DNP3, IEC60870-5-101/103/104 and IEC61850. For large applications additional IO expansion modules can be easily connected via the IO bus interface. Other advanced features include data logging, control logic and an inbuilt web server for management of RTU configuration and display of data in a graphical format. The RTU32M/N can also run/host embedded SCADA applications with advanced local and remote data management and display.

BRODERSEN
simplifying systems

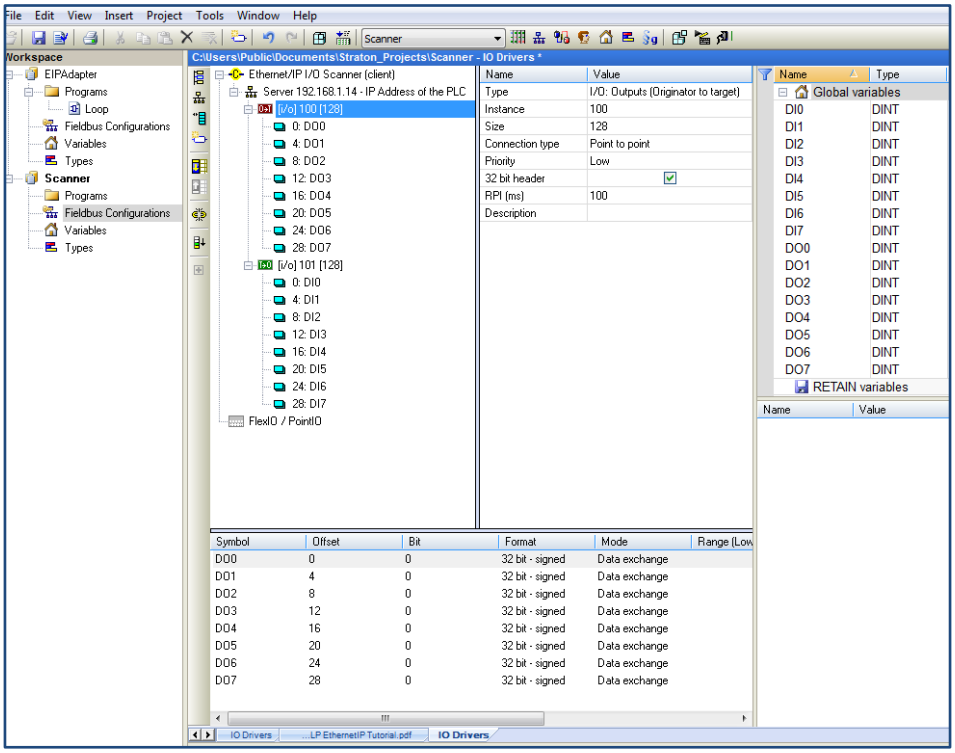
Ethernet/IP Protocol

Overview

The Ethernet/IP protocol is implemented as a user configurable ‘fieldbus driver’ within the WorkSuite logic application of the Brodersen controller. The Ethernet/IP driver allows the Brodersen controller to use four different methods for communications with AB ControlLogix and CompactLogix PLCs. The methods are;

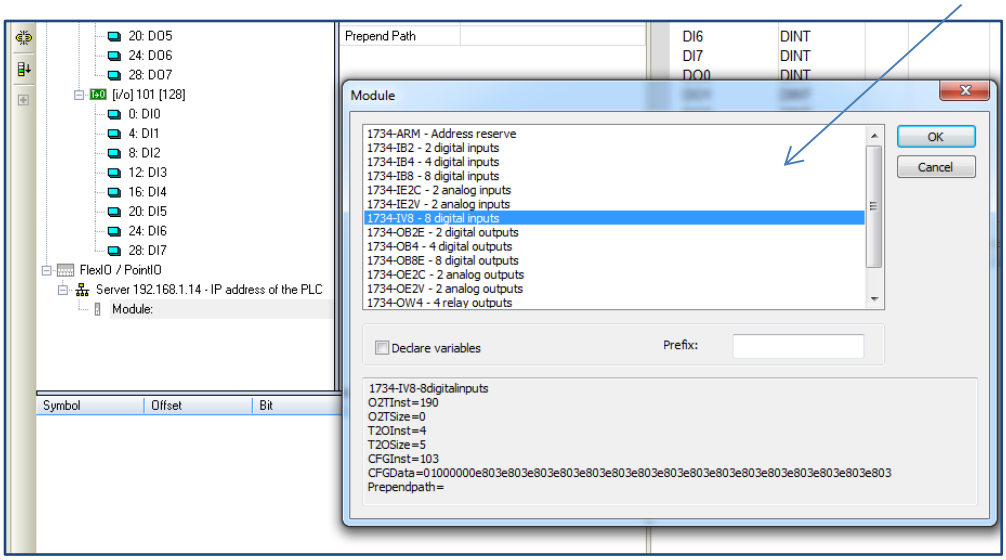
1) FlexIO/PointIO – allows the RTU to directly address AB I/O
2) Ethernet/IP Adapter – allows the RTU to act as a server using I/O and vendor specific objects
3) Ethernet/IP Scanner – allows the RTU to act as a client to the AB PLC
4) Ethernet/IP Tag Client – allows the RTU to read and write from/to AB PLC tags (easiest method to use)

All configuration is managed from the WorkSuite Logic – a modern IEC61131-3 compliant interface that allows creation of communications and logic applications.



Example showing setup of the Brodersen RTU as an Ethernet/IP Scanner

Easily map IO points to/from AB IO hardware using the FlexIO/PointIO setup tools

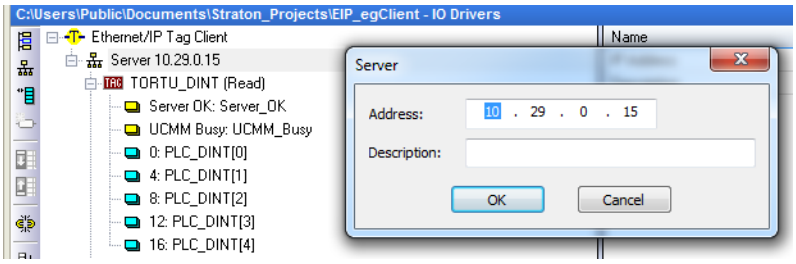


Ethernet/IP Protocol

Example setup of an Ethernet/IP Tag Client

The Ethernet/IP Tag Client interface is the easiest method to use.

The example here shows the RTU32 setup to communicate with a PLC (the Server) at IP address 10.29.0.15.

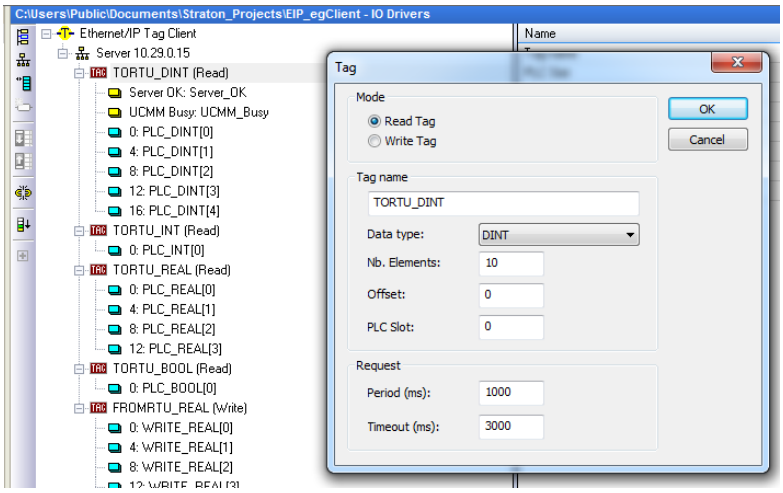


Defining the PLCs IP address

The RTU32 can communicate with multiple PLCs/servers and each server can have groups/arrays of tags that are segregated by their Read or Write mode and by their data type.

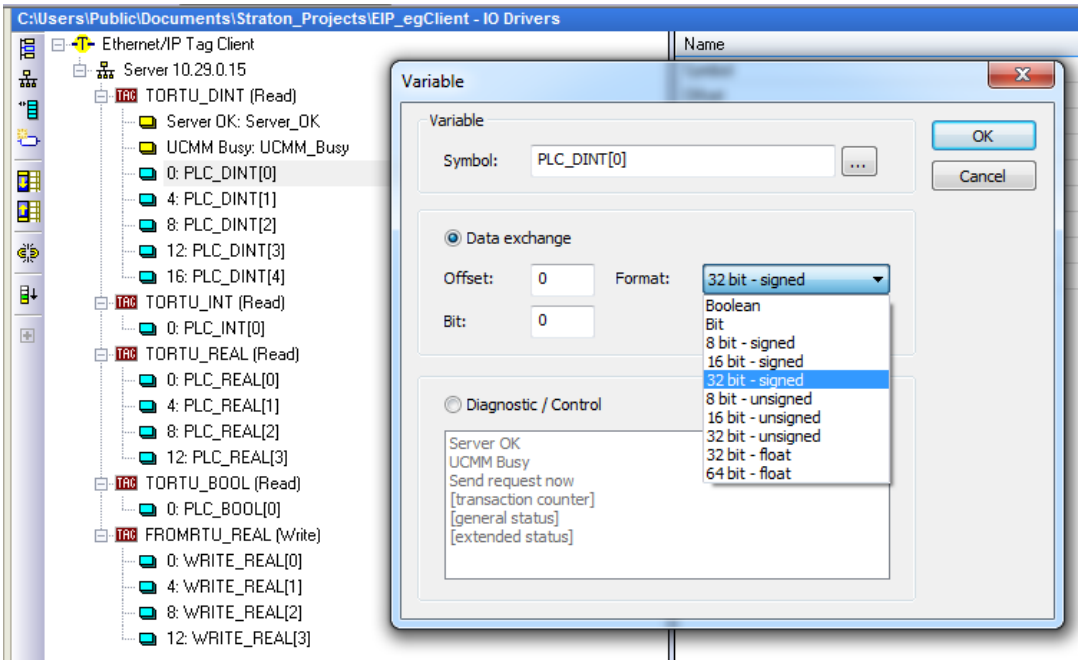
In the example shown here the RTU is setup to read 4x groups and write 1x group. The PLC must have the same 5x tag names (arrays of tags) setup in standard PLC memory.

The example has the PLC CPU in slot 0.



Setting up the tag name, read/write mode, data type, array size

Each RTU32 variable that is associated to the PLCs data must have its offset (index within the PLCs data array) defined. In the example below the RTU is also using array style tags, so each tag must have its [index] set. Here the RTU variable 'PLC_DINT[index=0]' is associated to PLC tag name'TORTU_DINT' at offset 0 (PLC s first 32bit value)



Associating the RTUs variable to the PLC tag name and setting the offset and data type

This example application has 5x tags in the PLC that the RTU is continuously reading and writing.

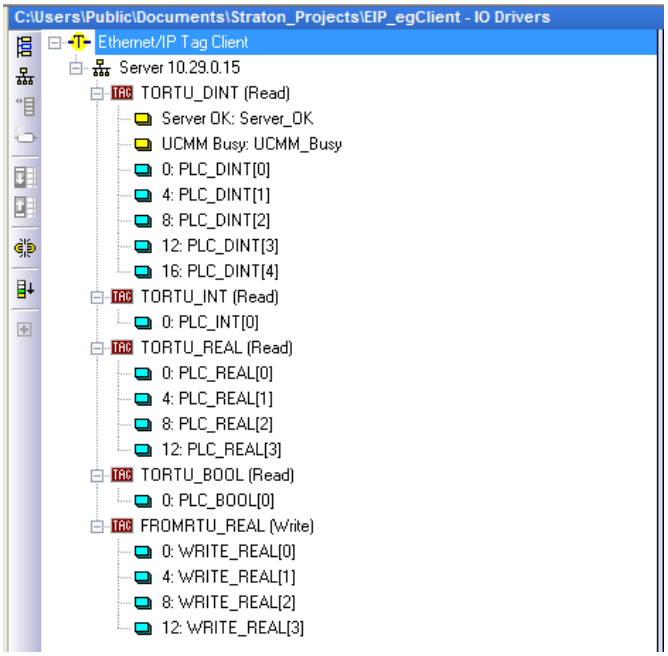
PLC Tags (that are read by the RTU32)
TORTU_DINT - array of double integer tags
TORTU_INT - array of integer tags
TORTU_REAL - array of floating point/real tags
TORTU_BOOL - array of Boolean/digital tag)

PLC Tags (that are written to by the RTU32)
FROMRTU_REAL - array of floating point/real tags

RTU Tags (location for data read from the PLC)
PLC_DINT[index 0-4] - array of double integer tags
PLC_INT[index 0] - array of integer tags
PLC_REAL[index 0-3] - array of real tags
PLC_BOOL[index 0] - array of boolean tags

RTU Tags (location for data sent to the PLC)
WRITE_REAL[index 0-3] - array of real tags

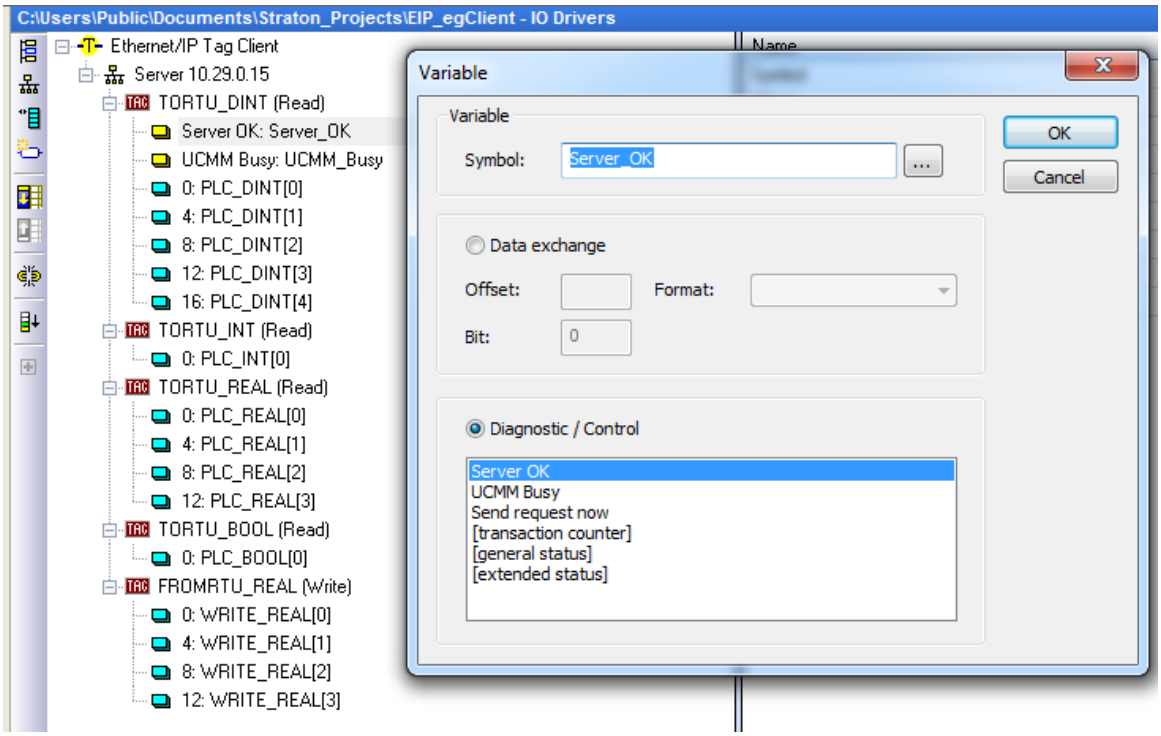
Note: each tag is offset by 4x bytes (32bit PLC value)



Fieldbus Configuration Overview

The RTU32 is also able to monitor the status and manage the communications link with the PLC for each tag name / group.

The example below shows 2x Boolean tags defined in the RTUs variables that are associated to the Ethernet/IP Tag Client tag name 'TORTU_DINT'. These tags can be used to provide feedback to the logic of Sever OK and UCMM Busy (link busy) status.



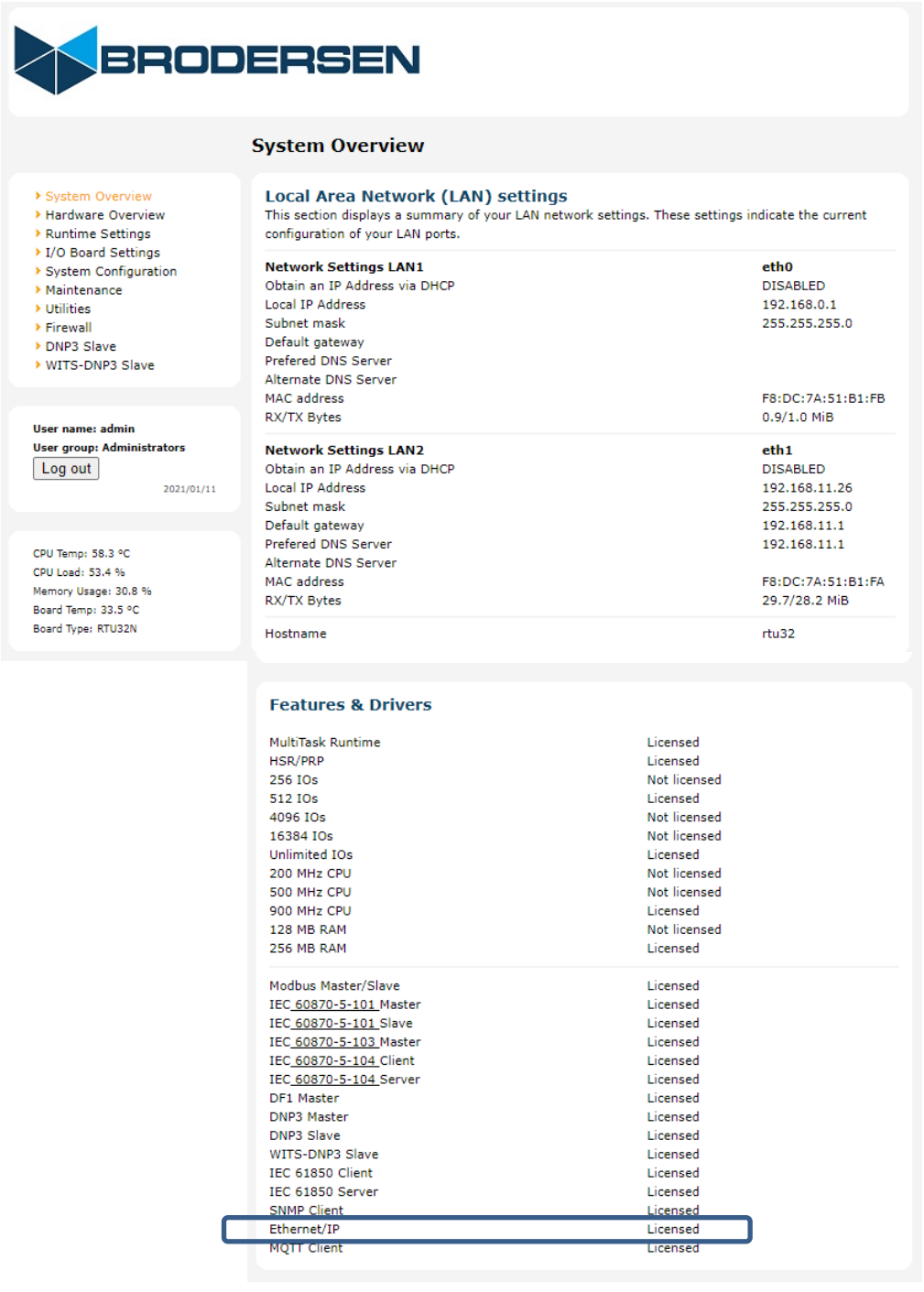
Associating RTU32 variables to the communications status of the PLC interface

Ethernet/IP Protocol Implementation

Steps to setup an Ethernet/IP Tag Client

1) Protocol Enabled and RTU32 IP Addressing

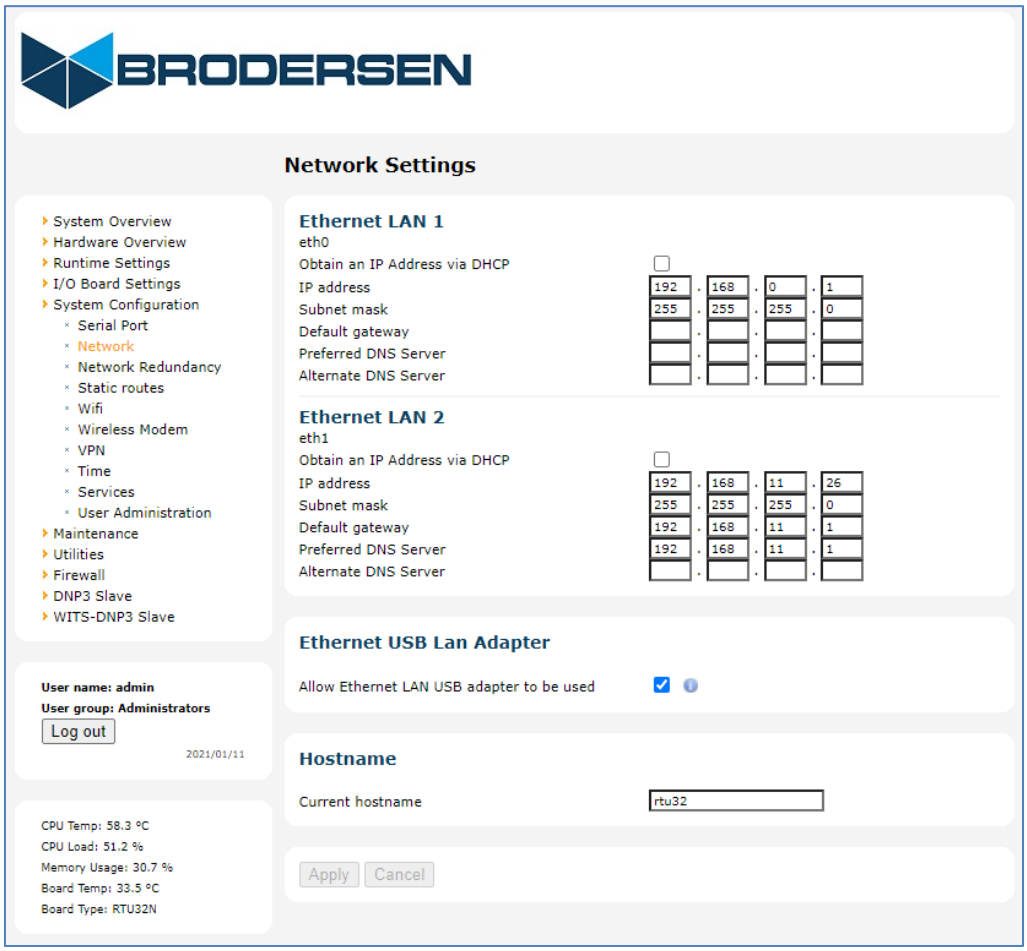
Before you get frustrated with your setup not working as expected – be sure to first check that your RTU32 has the Ethernet/IP protocol option installed and that you have an appropriate LAN port setting! Connect to the RTUs web interface and review the network settings and list of installed drivers as shown below.



RTU32 Settings Overview – showing LAN setup and list of drivers installed

The Network Settings menu allows access to the setup of the RTUs LAN ports.

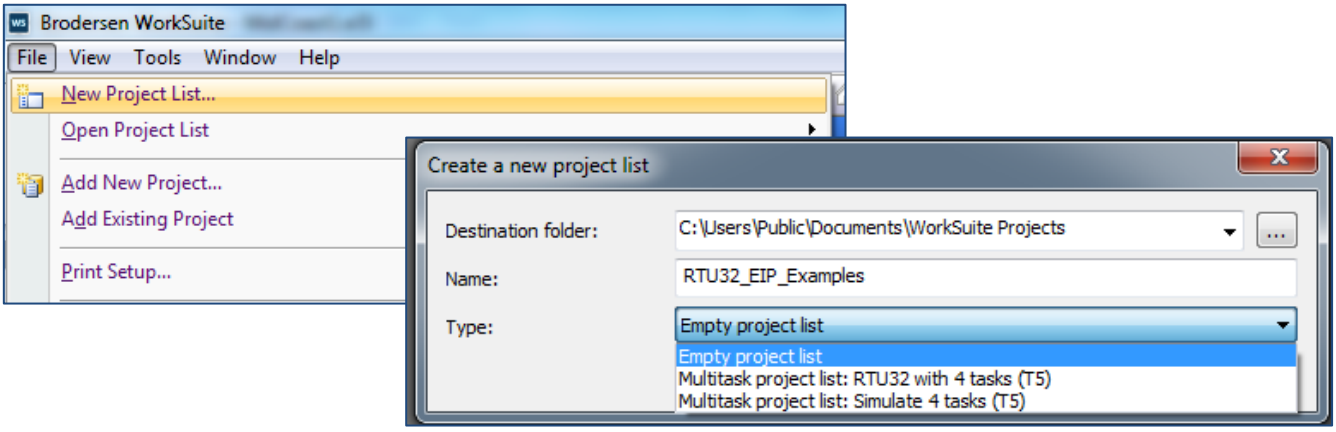
Note: you must select apply for the changed network settings to be applied (RTU restart not required).



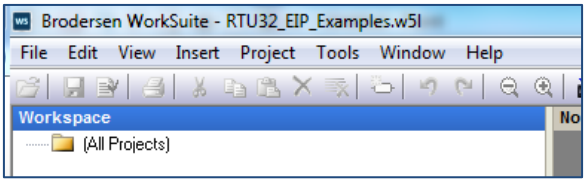
If you do not have the Ethernet/IP driver loaded contact your distributor for further details.

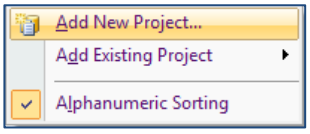
2) Creating an Example Ethernet/IP Tag Client Project in WorkSuite

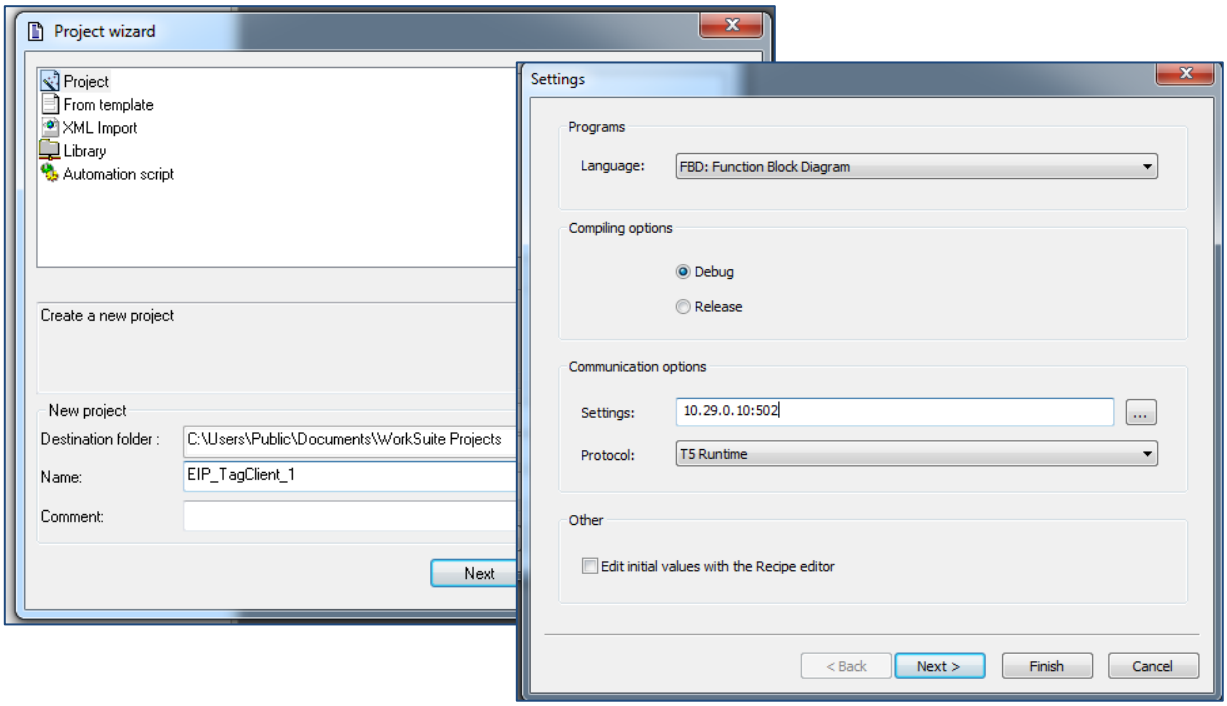
Start WorkSuite - if you have an existing Project List/Workspace that you would like to use then open it. If you want to add a new Project List then select 'New Project List' from the File menu and enter an appropriate name eg. RTU32_EIP_Examples (using an Empty project list). If you have an existing application then jump to step 3…



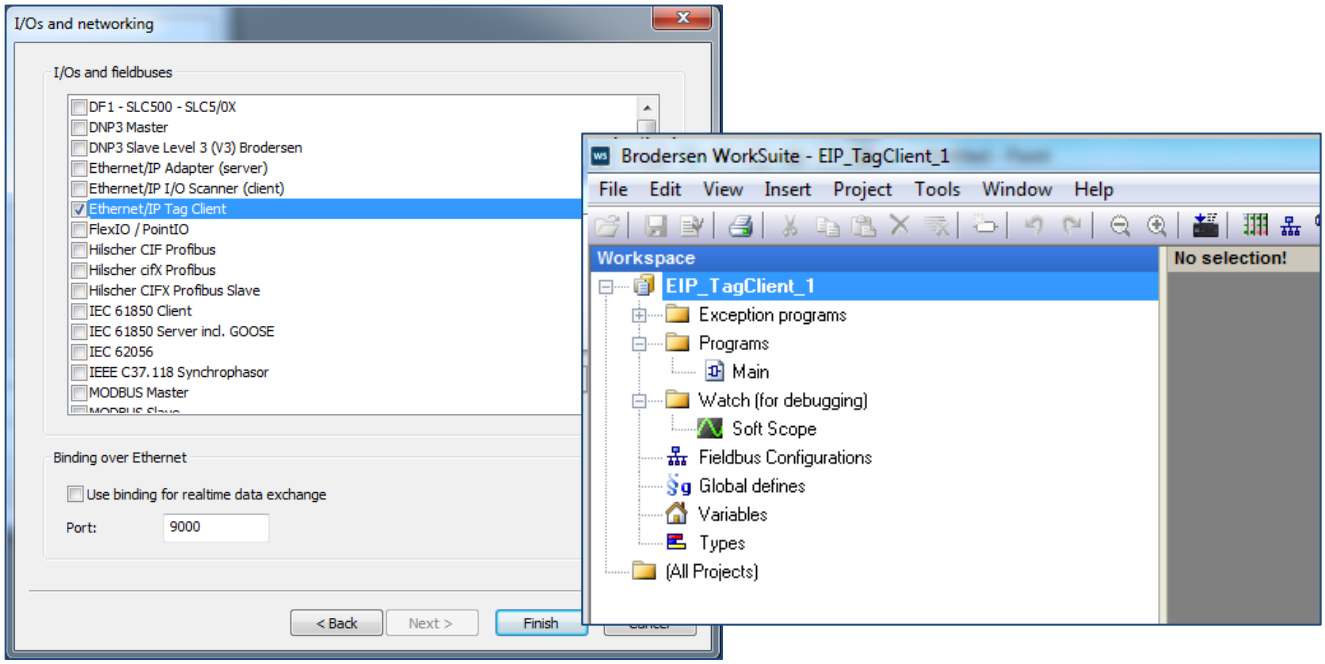
Then either right-click in the empty Workspace region or use the File menu to select ‘Add New Project’ and enter an appropriate Project Name in the project wizard eg. EIP_TagClient_1 then select Next. Then enter the IP address of your RTU32 eg. 10.29.0.10:502 and then select Next (include the port number of 502, as this is the default port always open on the RTU32 that allows MODBUS and configuration activities such as project downloads to occur).





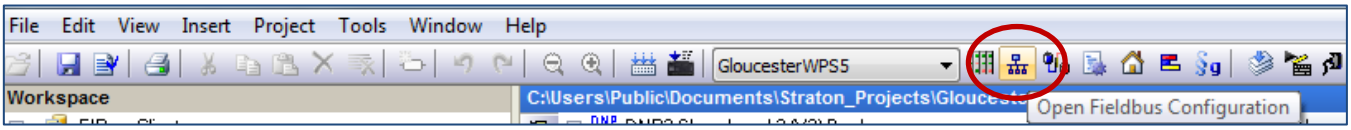


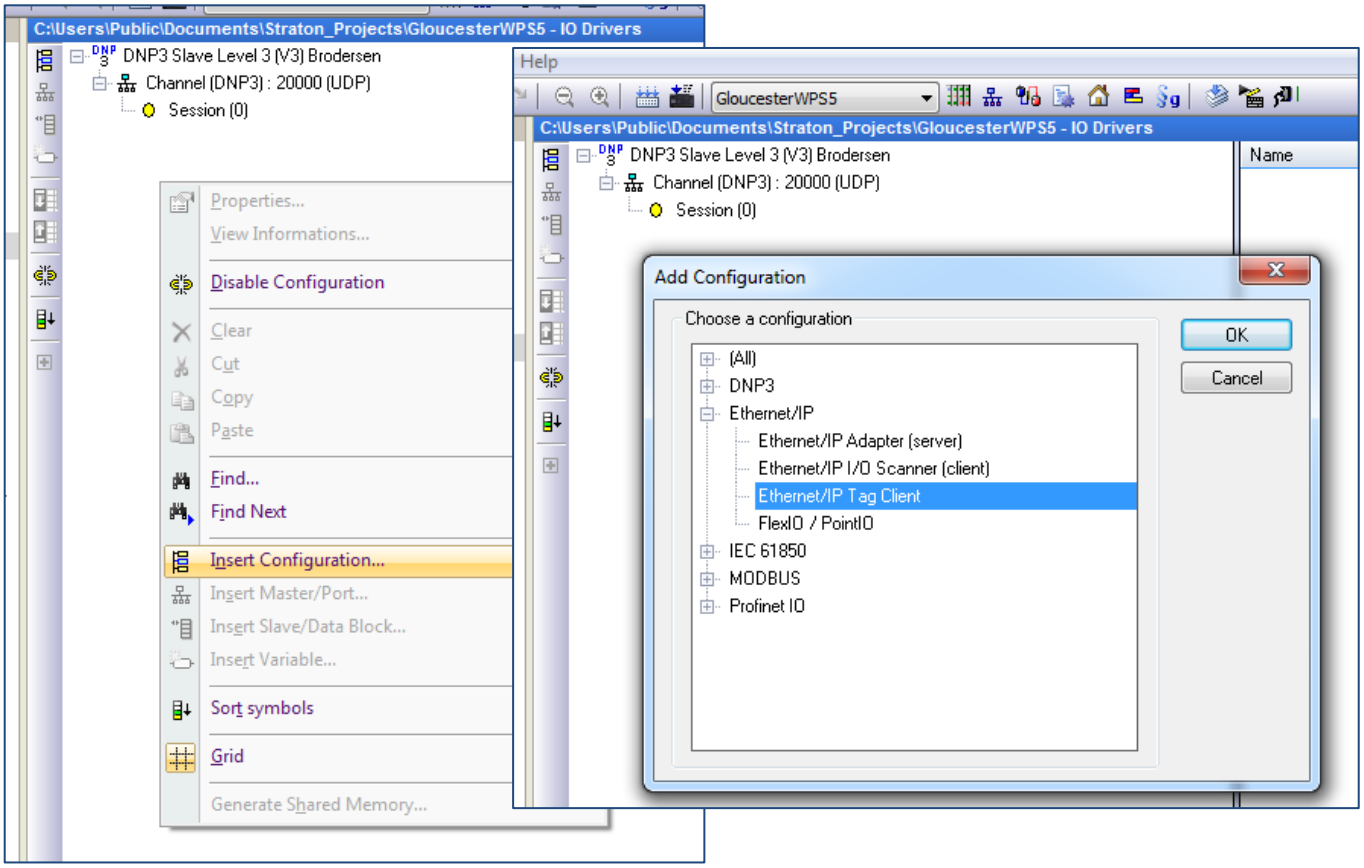
Then select the Ethernet/IP Tag Client from the protocol list and then click Finish and you will then have a project framework ready to use that includes the Ethernet/IP Tag Client interface. Now jump to step 4…



3) Optional Step – only if you need to add Ethernet/IP Tag Client to an existing RTU application

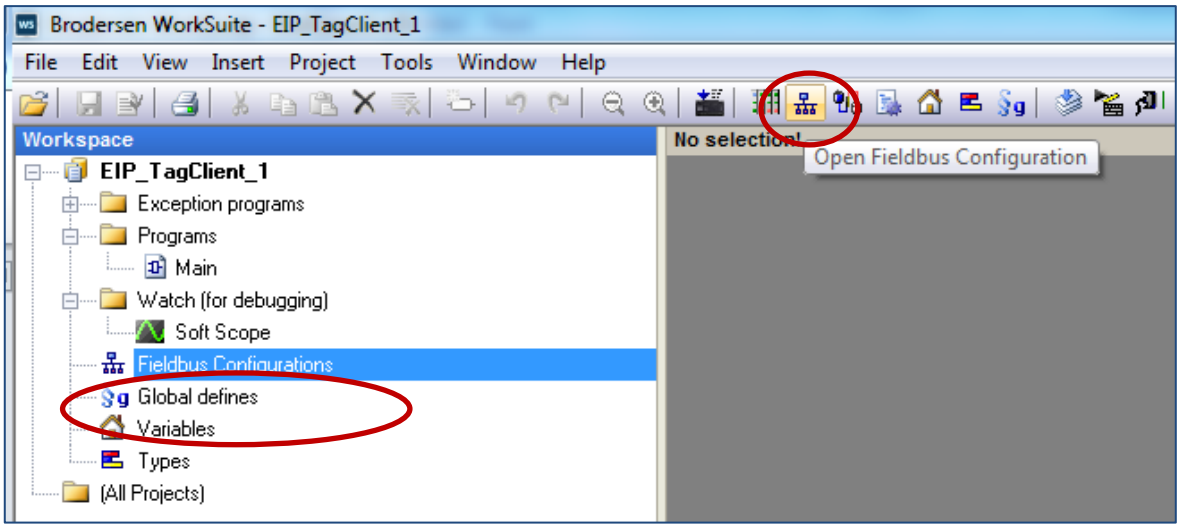
If you have an existing RTU32 project open in WorkSuite and simply want to add an Ethernet/IP Tag Client to your application then you would open the Fieldbus Editor using the Toolbar icon as shown below.



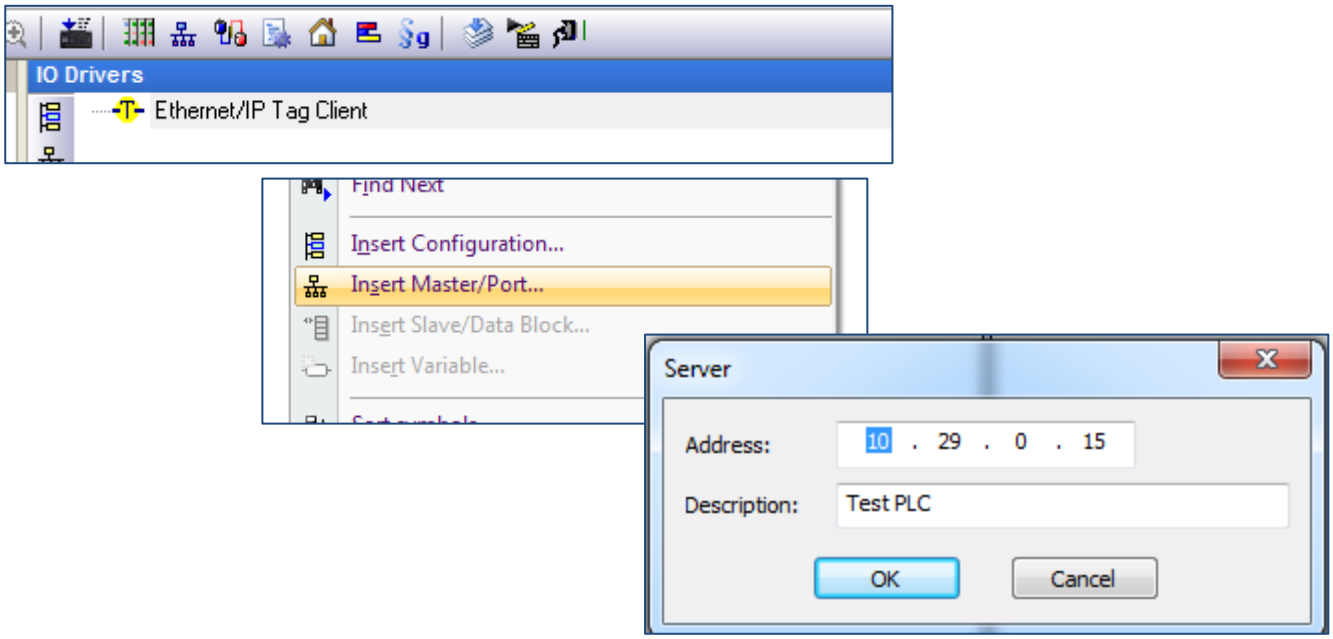


4) Setting up the Ethernet/IP Tag Client interface

The Fieldbus Configuration editor is used to manage the RTU communications interfaces. Open the Fieldbus Configuration editor from the shortcut in the projects tree, or from the icon in the toolbar.

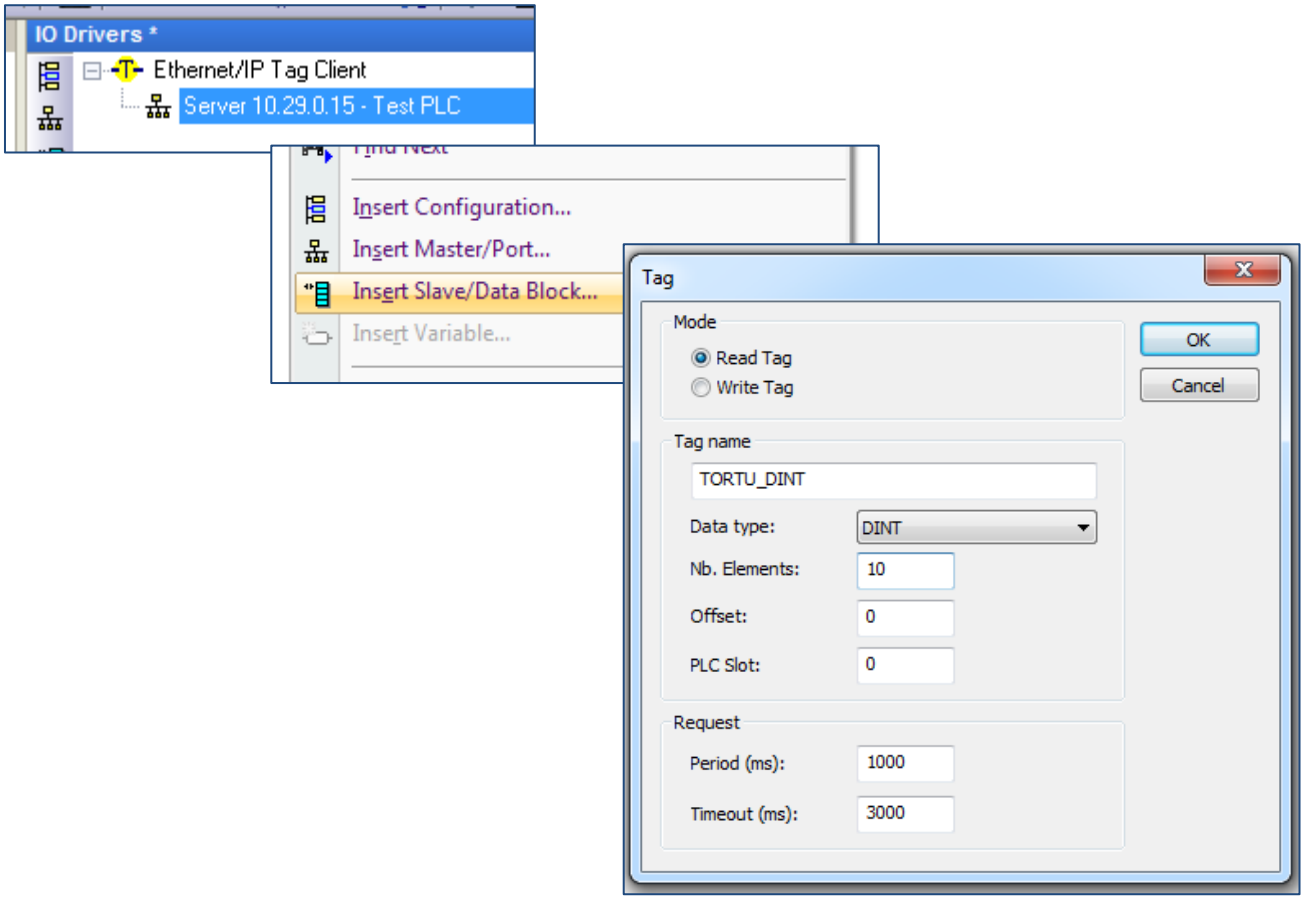


Right click on the protocol heading ‘Ethernet/IP Tag Client’ and select Insert Master Port. You will then need to enter the IP address of your PLC eg. 10.29.0.15 is being used here.

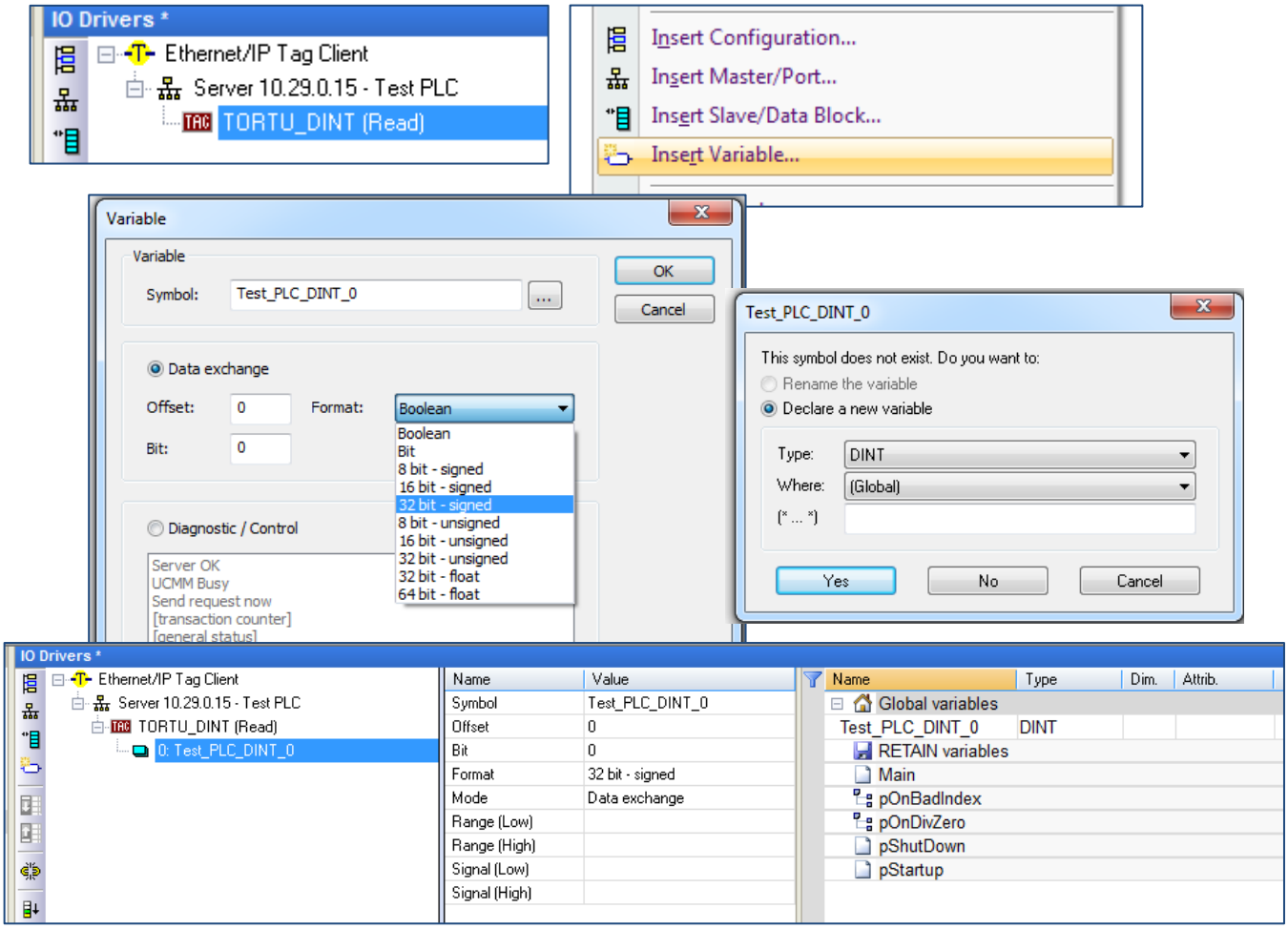


5) Reading a PLC DINT tag

To read a tag from the PLC you must add a tag definition to the server interface defined above. Right click on the Server heading for the Test PLC and select ‘Insert Slave/Data Block’. You will then need to enter the Tag name of the variable in the PLC that you want to read (must be exactly the same name as used in the PLC) eg. ‘TORTU_DINT’ is the name of an array tag in the PLC that has 10x DINT values.

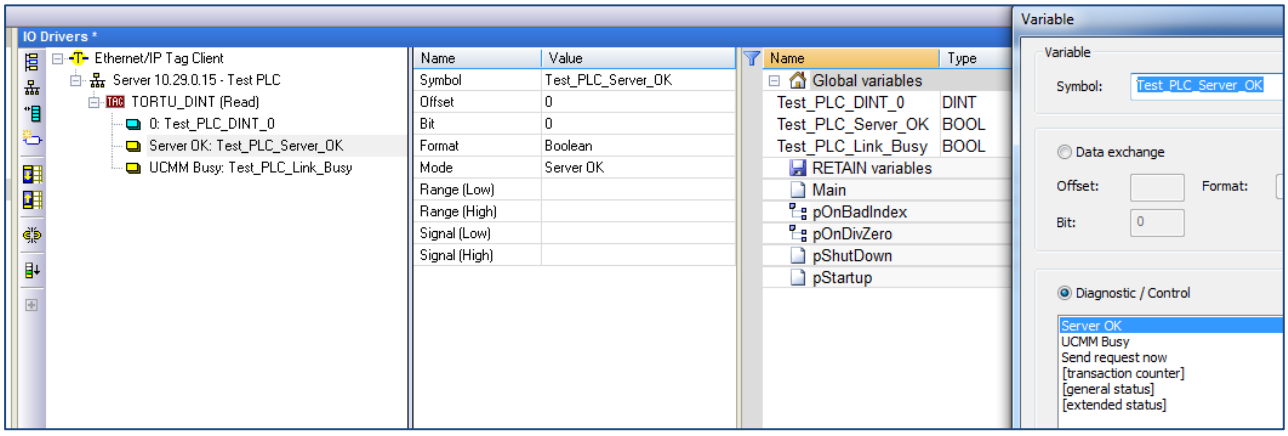


You then need to create one or more variables in the RTU that will be used to store/expose the data read from the PLC. Right click on the Tag header and select 'Insert Variable'. Enter a variable name for the RTU (symbol field) and select 32bit – signed for the format. The RTUs variable can be selected from a list of existing variables(if they exist) or created 'on the fly' by entering an appropriate variable name eg. 'Test_PLC_DINT_0' was used below. Because Test_PLC_DINT_0 did not exist the WorkSuite interface then allowed declaration of the variable.



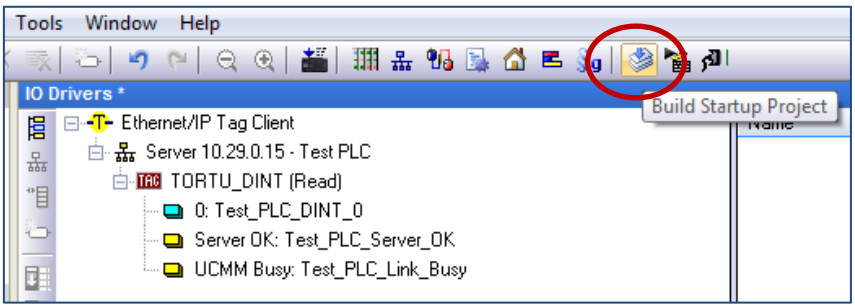
Just about ready to test! If the RTU project was downloaded to the RTU now and a PLC with IP address of 10.29.0.15 was available that included a DINT tag/array named 'TORTU_DINT' data would be 'flowing'. But before getting too daring (or excited) – let's add some communication monitoring.

Right click on the tag header again and add 2x Diagnostic/Control variables to monitor the Server and Link status as shown below...

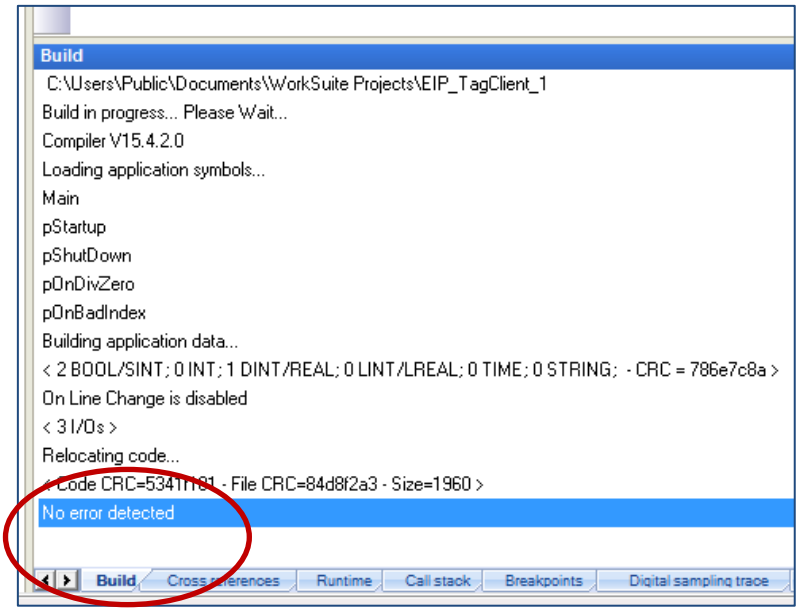


6) Writing the WorkSuite Application to the RTU

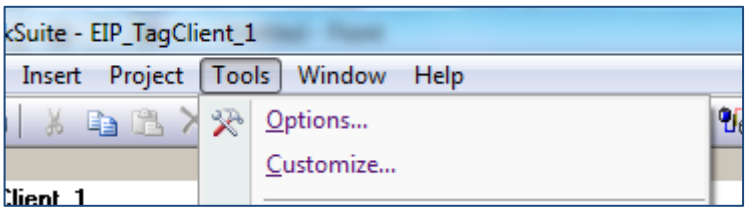
Before writing the application to the RTU, it must first be Compiled/Built without errors.

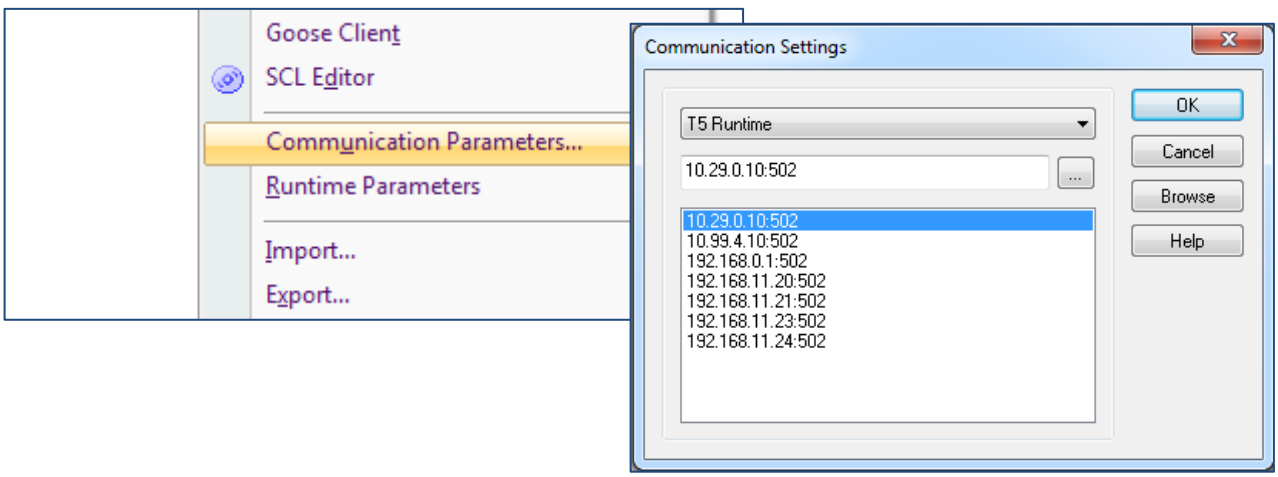


Use the 'Build' icon to compile the project and check that it has compiled without errors. Any errors must be corrected before continuing.

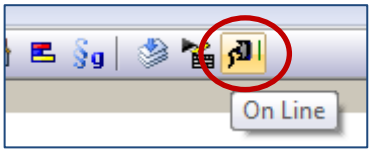


Before you can download the compiled project to the RTU the IP address of the RTU must be configured. Select the Tools menu item 'Communication Parameters' and enter the RTUs IP address (192.168.0.1:502 is the default setting for the RTUs LAN port 1). In the example below the RTU has an IP address on LAN port 2 of 10.29.0.10.

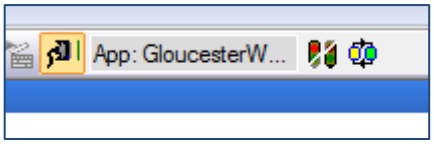




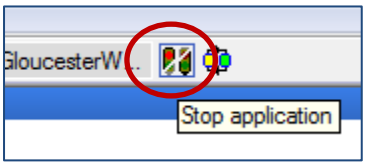
Once the IP address of the RTU has been set, select the On Line icon in the Toolbar to establish communications with the RTU32.



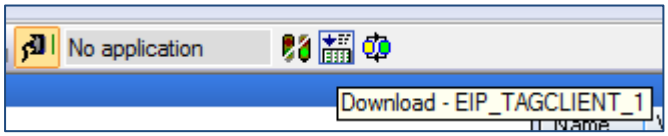
If the RTU is already running a logic application with a different application name, the runtime window will show the name of the application that the RTU is running. Eg. 'GloucesterWPS5'



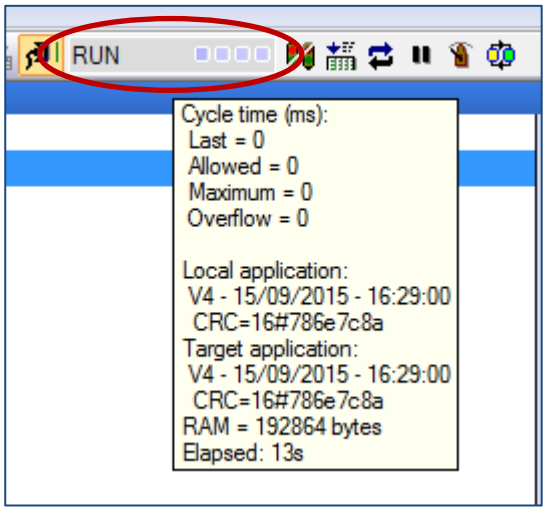
To change/overwrite the application that is running in the RTU the existing/old application must first be stopped



Then use the download icon to write the new application to the RTU32.



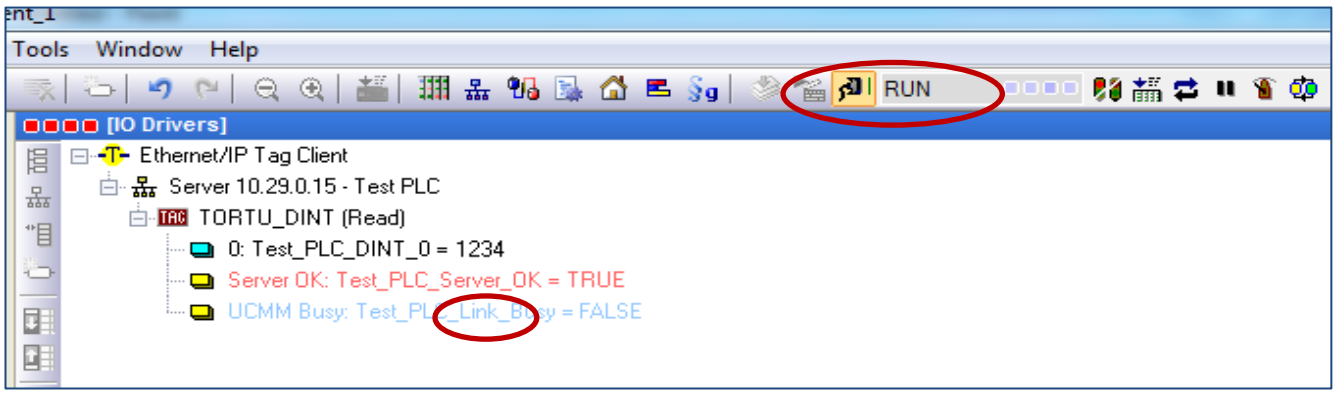
If the download is successful the RTU will then run the new application.



Hover over the runtime window area and information about the project active in the RTU will be displayed.

The example here is showing that the project is taking less than 1ms to run and has been running for 13s since loaded.

If the Fieldbus Configuration editor is opened while On Line to the RTU32, the value read from the PLC and the status of the communications link to the PLC will be shown. – if it works the job is almost done!



Change the value in the PLC and check the new value in the RTU. eg. 1234 above was read from the PLC.

Additional product information is available from our website, or from the authorised distributor in your region.

www.brodersen.com
Brodersen Systems Asia Pacific P/L
832 High Street, Kew East, VIC 3102, Australia
Tel: +61 3 9249 9505, Email: sales-ap@brodersen.com

BRODERSEN
simplifying systems