ATOP Technologies, Inc.

Protocol Gateway
Modbus Client/Server

Protocol and
eNode Designer configuration

eNode Configuration Manual

V1.5

August 3rd, 2023

This PDF Document contains internal hyperlinks for ease of navigation.
For example, click on any item listed in the Table of Contents to go to that page.

Published by:

ATOP Technologies, Inc.

2F, No. 146, Sec. 1, Dongxing Rd.,
Zhubei City, Hsinchu County,
302058, Taiwan, R.O.C.

Tel: +886-3-550-8137
Fax: +886-3-550-8131
https://www.atoponline.com

Important Announcement

Disclaimer

We have checked the contents of this manual for agreement with the hardware and the software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual is reviewed regularly and any necessary corrections will be included in subsequent editions.
Suggestions for improvement are welcome.
All other product's names referenced herein are registered trademarks of their respective companies.

Documentation Control

Author:	Simon Huang
Revision:	1.5 update
Revision History:	Update copyright and ATOP contact information
Creation Date:	27 May 2016
Last Revision Date:	3 August 2023
Reviewer	Simon Huang
Product Reference:	PG59XX Protocol Gateway
Document Status:	Released

Table of Contents

Table of Figures

1 Introduction

Thank you for Buying ATOP's Protocol Gateway.

The product is bundled with the following three user manuals:

1) Hardware specific installation user manual, **not covered in this document.** It covers ATOP's hardware installation procedure, wiring, power connection etc.
2) Getting started with ATOP's Protocol Gateway user manual – configuration tool introduction, web configuration, software architecture introduction– **not covered in this document**. This manual covers the introduction, installation, network set-up maintenance and using of the configuration tool software, including the procedure to be followed for uploading new configurations to ATOP's device.
3) Protocol specific user manual **(This Manual). One protocol-specific manual will be provided for each protocol installed on the device.** This manual covers:
    a. Basic device network configuration
    b. Step-by-step protocol set-up for in eNode designer
    c. Description of the protocol-specific software features, the device profile and the implementation table of supported functionalities.

This manual is for **Modbus TCP/RTU Client/Server** and describes how to use the **Modbus eNode Designer Module** to configure ATOP's *Modbus ADH Application* within the eNode Designer configuration tool.

1.1 Scope

This document is divided into 3 major sections:

- **General description and configuration theory**
- **Configuration with eNode Designer**
- **Modbus properties**

1.1.1 Document Reference

[1] Document Title: Getting started User Manual: 197-0100
Revision: Version 1.00 or higher

[2] Document Title: Modbus standard
Revision: Edition 2 – 06/2006

1.2 List of Abbreviations

ADH = Application Data Hub
IED = Intelligent Electronic Device
IP = Internet Protocol
TCP = Transmission Control Protocol

2 General Description

The Modbus eNode Module can be used to Modbus ADH Application as a master or slave. For naming consistency across eNode Designer, the Modbus master is called a client, and the Modbus slave is called a server.

The Modbus client can communicate with many Modbus slave IEDs/RTUs, all of whose data point details can be configured using this module.

2.1 Configuration Theory

The configuration properties always describe a *server* (slave). When configuring the Modbus server, you are configuring the properties of the server itself. When configuring the Modbus client, you are describing the properties of all the remote servers (IEDs) with whom the client is communicating.

Configuring the protocol specific information, such as object addresses, is handled in the Modbus module. This is explained in this document.

Communication port properties (such as Baud Rate) are configured on the communication port itself. The Device module handles the communication port properties, so that the heavy-critical details are outside the scope of this document and configuration examples can be found inside eNode Designer general user manual.

However, screenshots of the typical configuration method are shown in section 4. The properties of the communication ports automatically apply to the application. For example, in a TCP Modbus server application, the IP Address the application binds to is taken from the parent Ethernet port.

2.2 General Screen Description

A small example of ModBus configuration is shown below to help describe the layout of the screen.

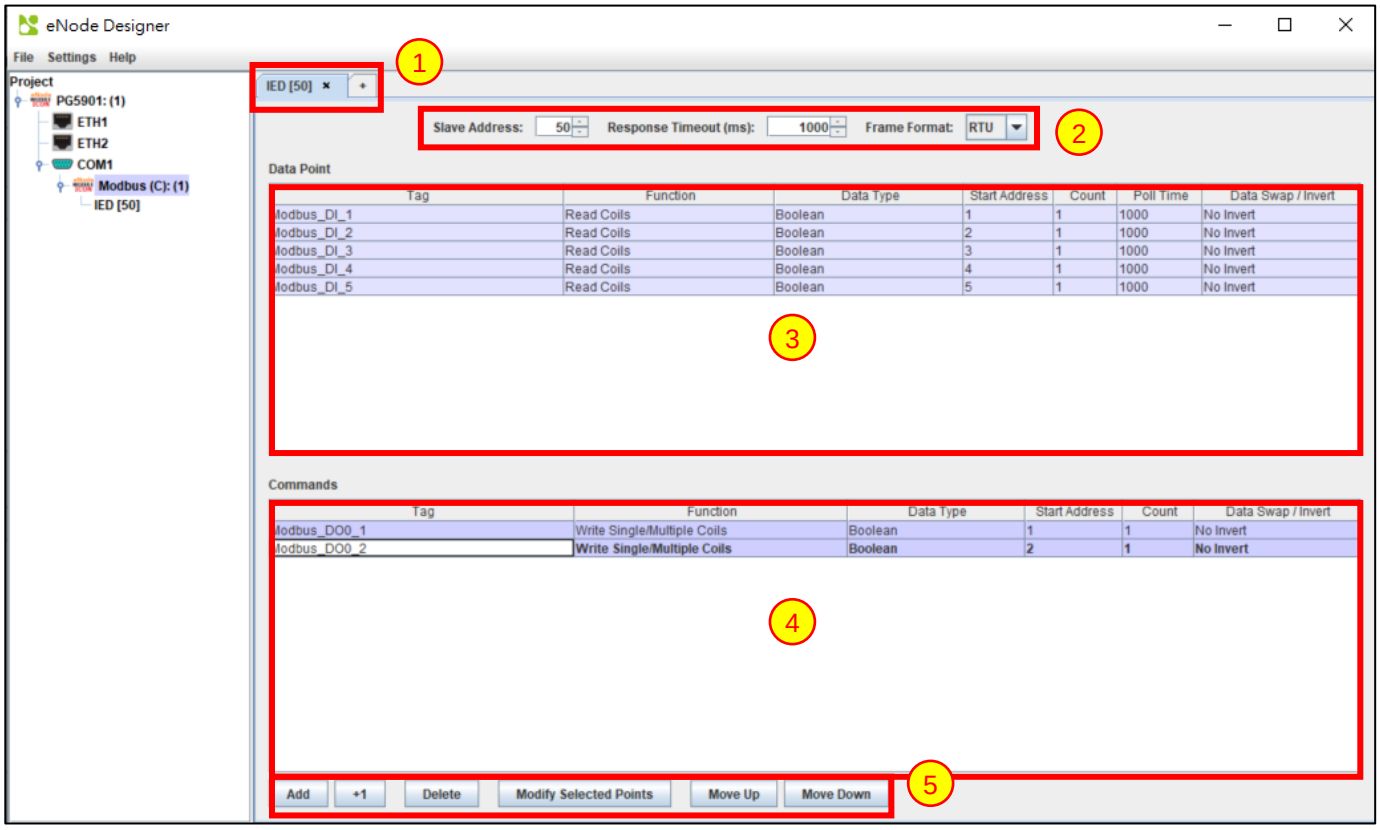


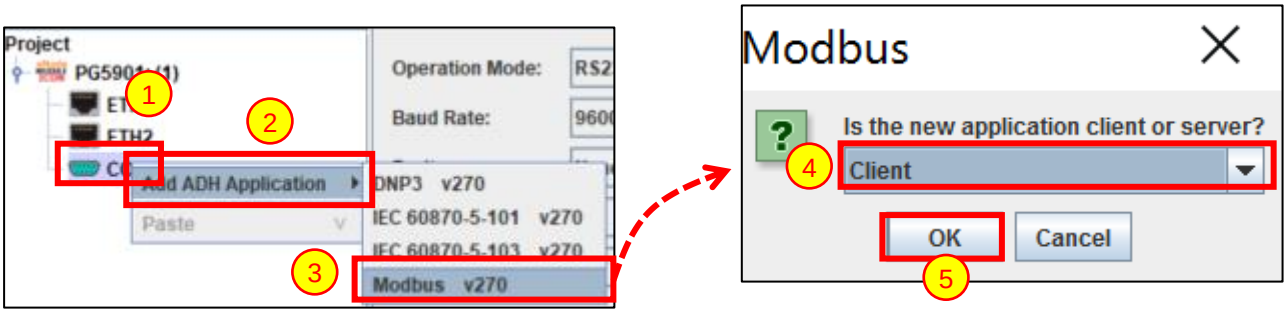
Figure 2-1 - Example Screen

1. **Tabs** – For the client, one tab represents the properties of a single remote Modbus server. eNode server has no tabs – all properties describe the server itself.
2. **Server IED Properties** – Describes the protocol-specific properties of the server IED.
3. **Data Table** – Shows all (information) data associated with the IED.
4. **Commands Table** – Shows all commands associated with the IED.
5. **Buttons** – Used to add, remove and modify data points.

3 Modbus Configuration Guide

3.1 Adding the Module in eNode Designer

The Modbus module can be added to *Ethernet* and *Serial* ports.
The application can be set up as a Client or a Server – the user will be able to choose when adding the Modbus application to the project.

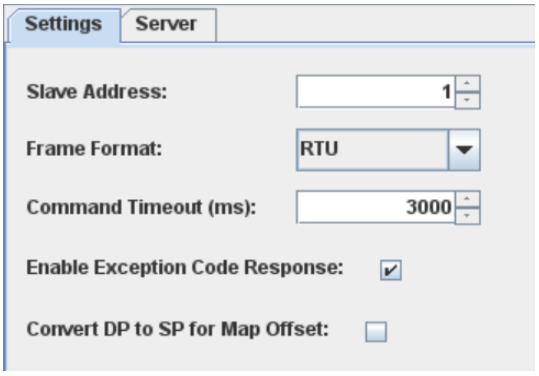


1. Right click the desired **communication port**.
2. Open the **Add ADH Application** menu.
3. Select **Modbus**.
4. Select **Client** or **Server** from the drop-down menu.
5. Click **OK**.

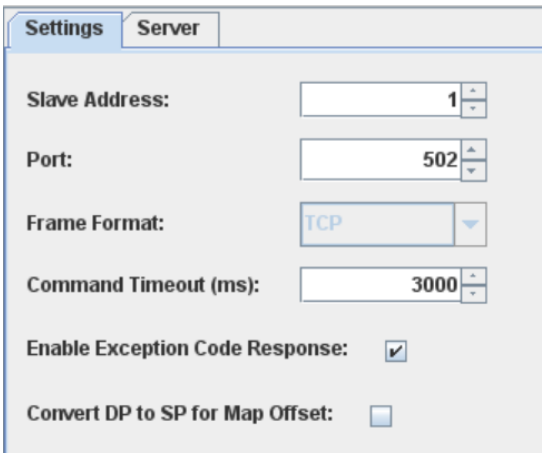
3.2 Server IED Properties

The server IED properties are shown on the top of the module screen. The options available are limited to what is relevant for the communication port.

Serial port example:



Ethernet port example:



Each property is described in detail below.

3.2.1.1 Slave Address

Description	The slave address of the server IED. For servers it describes its own slave address. For clients, it describes the slave address of the remote server.
Data Entry	Integer
Range	*1 to 255*
Input Option	Mandatory

3.2.1.2 Port

Description	The TCP Port used by the Modbus server IED.
Data Entry	Integer
Range	*0 to 65535*
Input Option	Mandatory

3.2.1.3 Frame Format

Description	The Modbus frame format of the communication line. In serial communications, it can be RTU or ASCII. For Ethernet communications, it is automatically set to TCP.
Data Entry	Drop down menu
Choices	*RTU, ASCII or TCP*
Input Option	Mandatory

3.2.1.4 Command Timeout (ms)

Description	The amount of time a master will wait for a slave's response before it is considered an error. Measured in milliseconds.
Data Entry	Integer
Range	*1000 to 10000 (default: 3000)*
Input Option	Mandatory

3.2.1.5 Enable Exception Code Response

Description	Uncheck it to respond a data value with 0xFFFF instead of a Modbus Exception Code.
Data Entry	Checkbox
Type	*Checked or not (default: checked)*
Input Option	Mandatory

3.2.1.6 Convert DP to SP for Map Offset

Description	Check it to convert Double-point's ON (2) to Single-point's ON (1) for Map Offset function
Data Entry	Checkbox
Type	*Checked or not (default: unchecked)*
Input Option	Mandatory

3.3 Client Configuration

Adding a Modbus client application will immediately show a view similar to the following figure. In this example, the Modbus application has been added to a serial port. If we assume this is added to an Ethernet port, there would be some different server IED properties at the top of the screen, as discussed in section 3.2. Each tab shows a single Modbus server with which this client is communicating.

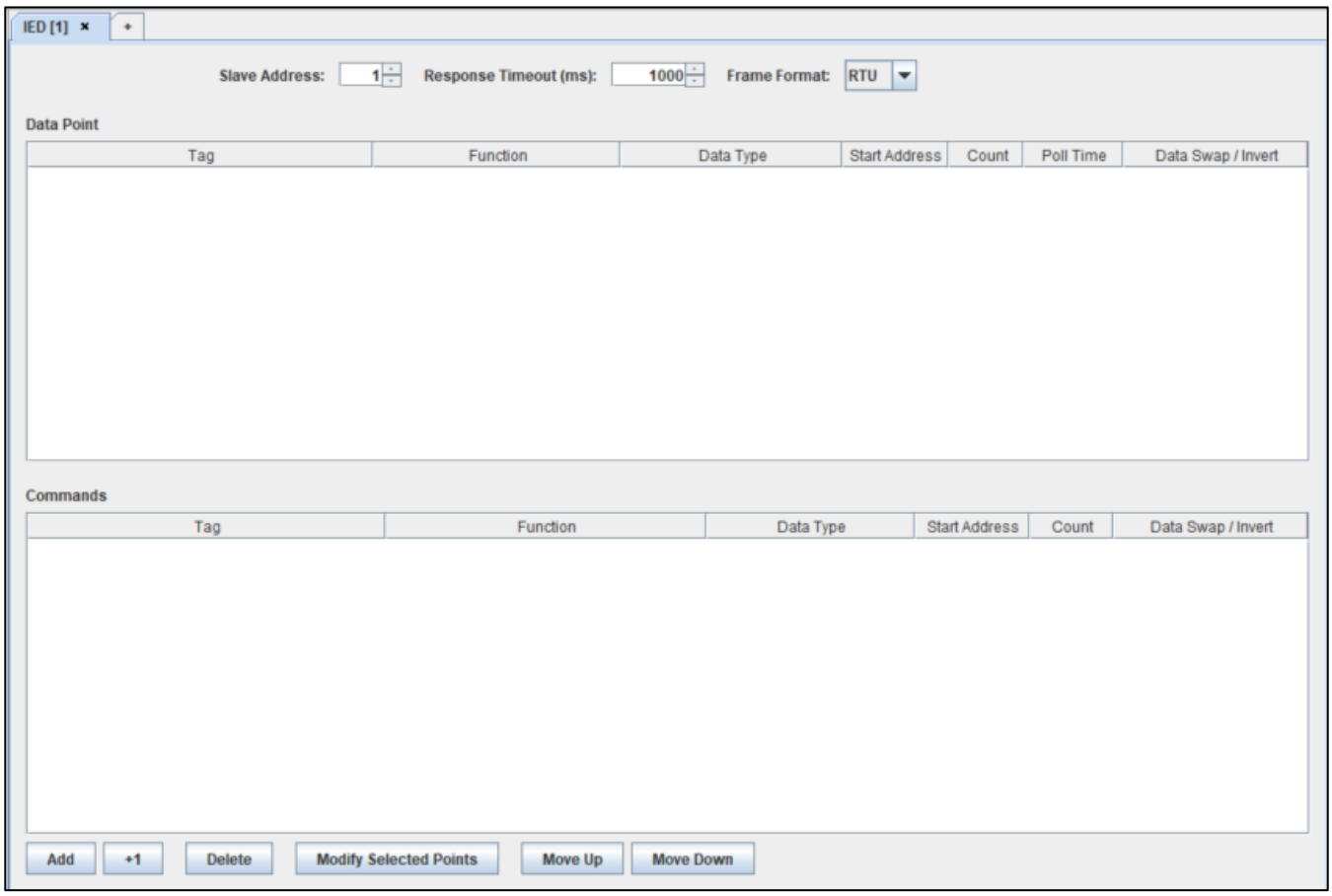


Figure 3-1 - Client has been added.

Here the “Add” button can be used to add data points. Adding data points is explained in the next section, and the other buttons are described in section 6: Reference Guide.

3.3.1 Adding Data Points

To add data points, left click the "**Add**" button beneath the tables in the main view. Doing so will bring up the following window.

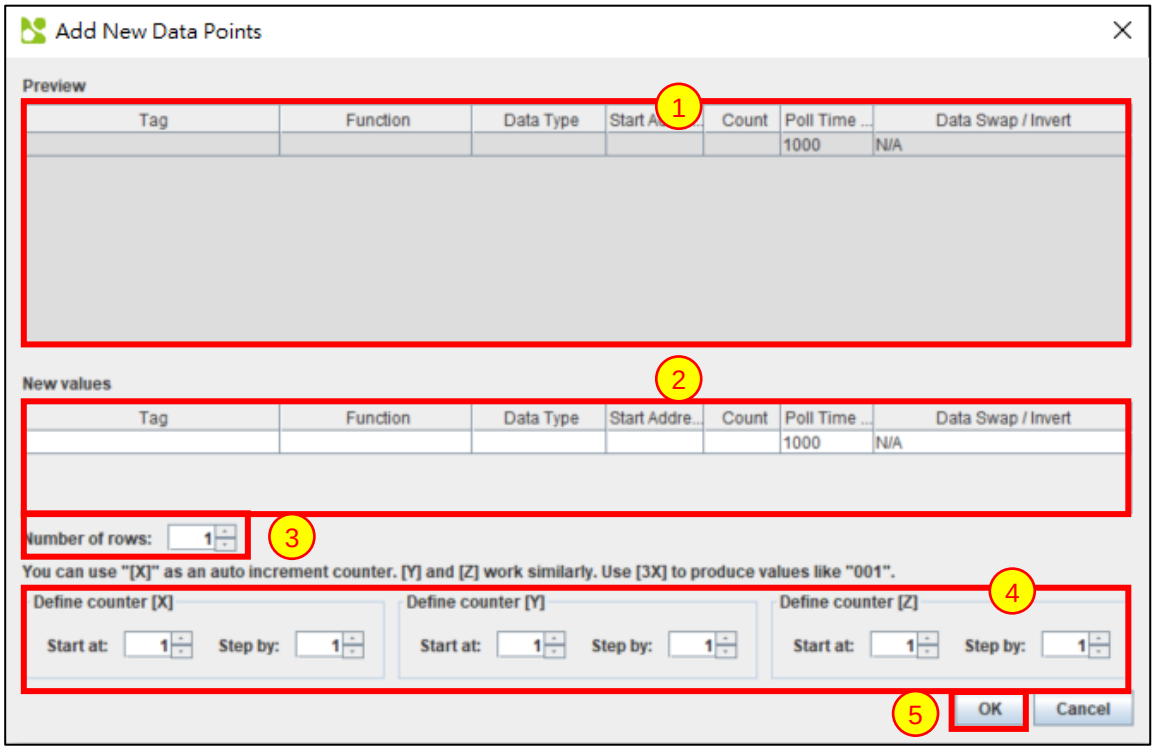


Figure 3-2 - Add data points window.

Type in the new properties of the data points here. Many data points can be added using the value in the "Number of rows" spinner. For details on the meaning of each column, see section 6.2.

1 **Preview Area** – Shows the preview of the data points that will be added.

2 **New values** – The area used to enter values. Tag, description, address and poll time use manual data entry (click the box and type new values), and function and data-type uses a drop-down menu. Entering an integer into the address column will start at that number with an auto- increment feature in each successive point.

3 **Number of rows –** A counter that can be used to add many data points at once.

4 **Automatic Counters** – Counters can be used in the input areas and their values will be substituted in the result and preview area. The starting values and step increment values can be changed in this area.

5 **OK button** – to accept the new data points.

When a command type is selected as the Function (*Force Single Coil* or *Preset Single Register*), the final "Poll Time" column will be removed from the view. In Modbus, poll time is not used for command types.

3.3.2 Connected Servers (Remote IEDs)

Each connected slave IED is represented by a single tab, with the title "IED [{X}]" where {X} is the slave address of the slave. Each IED will also appear in the eNode Designer project tree.

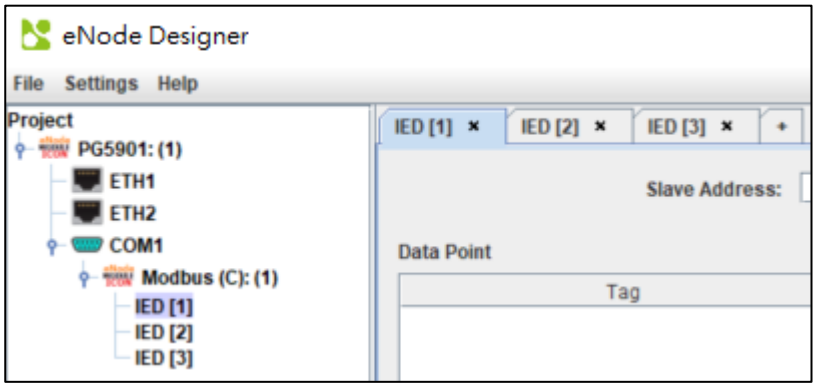


Figure 3-3 - Multiple connected servers example.

How to modify the connected IED list is shown below:

1. To **add** a new remote IED, click the **"+"** tab at the end of the list of existing remote servers.

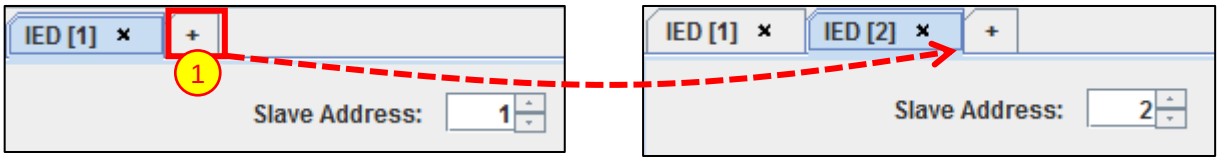


Figure 3-4 - Add a connected server.

In order to remove a remote IED, it should have no data points specified. If there are data points in the table and you still wish to remove the IED, you will have to remove those data points first.

1. To **remove** a remote IED, click the cross on the right side of the tab of the IED you wish to remove.

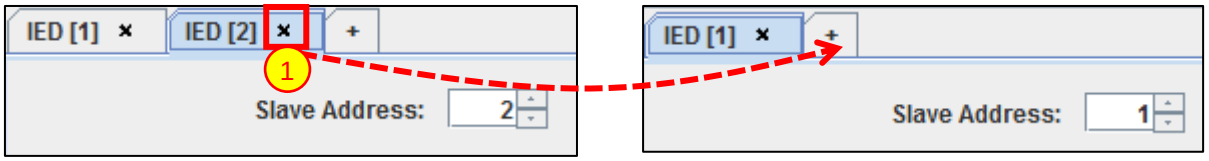


Figure 3-5 - Removed a connected server.

3.4 Server Configuration

A Modbus server application outputs data from the ADH database, receives commands and passes them into the ADH system to command another application to perform the operation. Thus, all server operations involve using data point *references* to already existing data points (created by other application clients or server-clients).

The options for configuring the server are very similar to the client options. The difference is that the options are now being used to describe the local server itself. The following view will be shown when an eNode Modbus server application is added. ATOP protocol gateway supports one server/slave application per protocol per device.

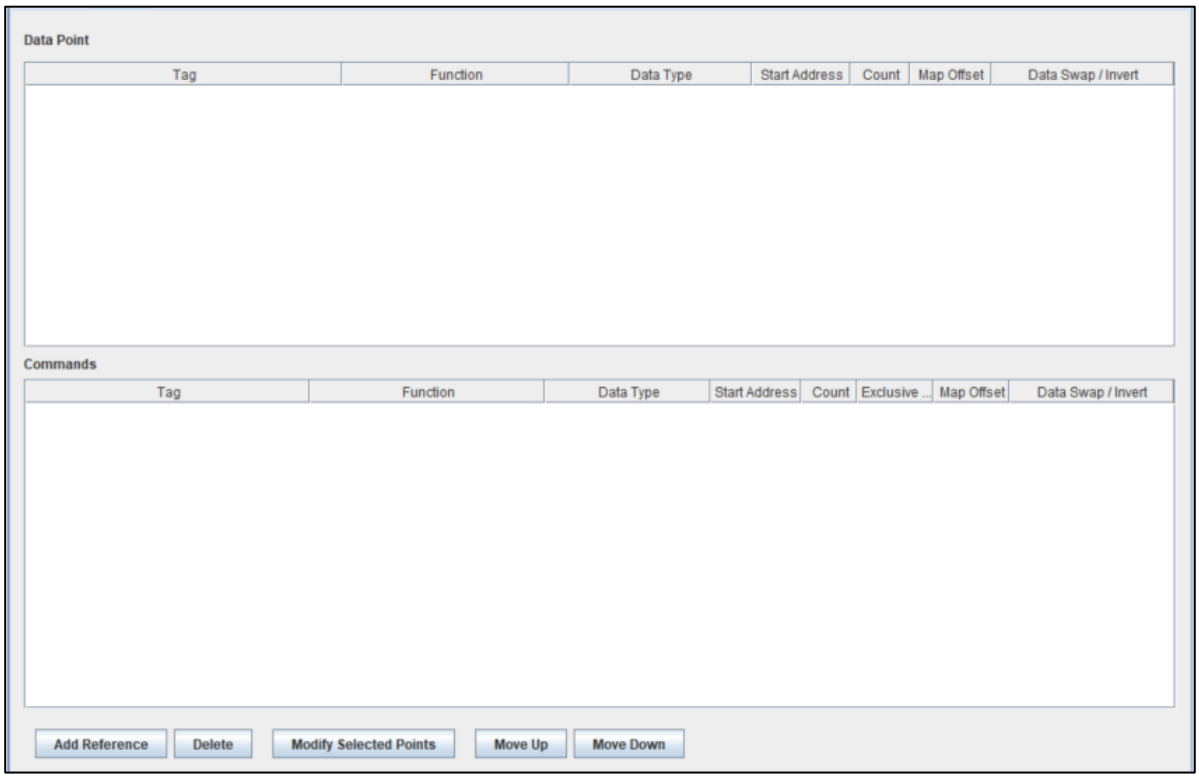


Figure 3-6 – Server has been added.

Here the “Add Reference” button can be used to add data point references. This is explained fully in the next section, and the other buttons are described in section 6: Reference Guide.

3.4.1 Adding Data Point References

To add new data point references, left click the "**Add Reference**" button underneath the tables in the main view. This will bring up the Add References window defined by the eNode Designer main application. The following screen will appear and will report all data points available for referencing purpose:

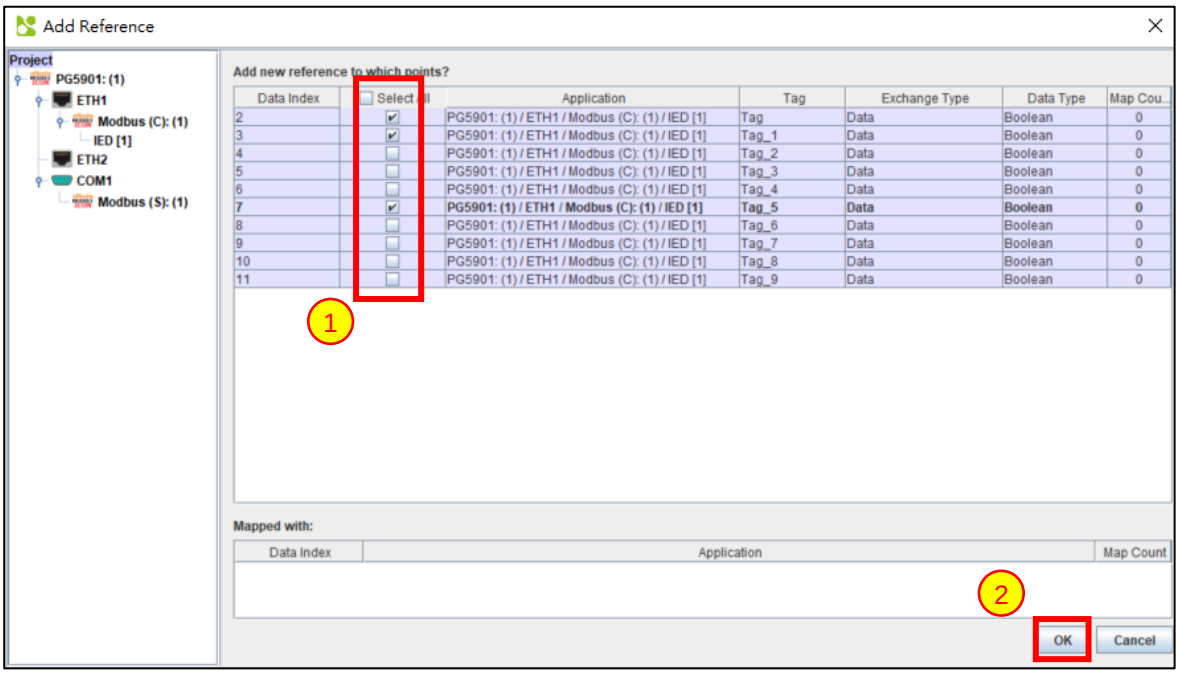


Figure 3-7 - Add new references window.

1. **Select Data Points** – Adding a reference to a point creates a "mapping" to that point. Select which data points the Modbus server application is interested in using.

2. Left Click **OK** when done to accept the new references.

The only data points that will appear in the list and that are available for mapping, are those whose data point type is compatible with the Modbus application. For the table matching Modbus function types to ADH data types, see 錯誤! 找不到參照來源。 錯誤! 找不到參照來源。.

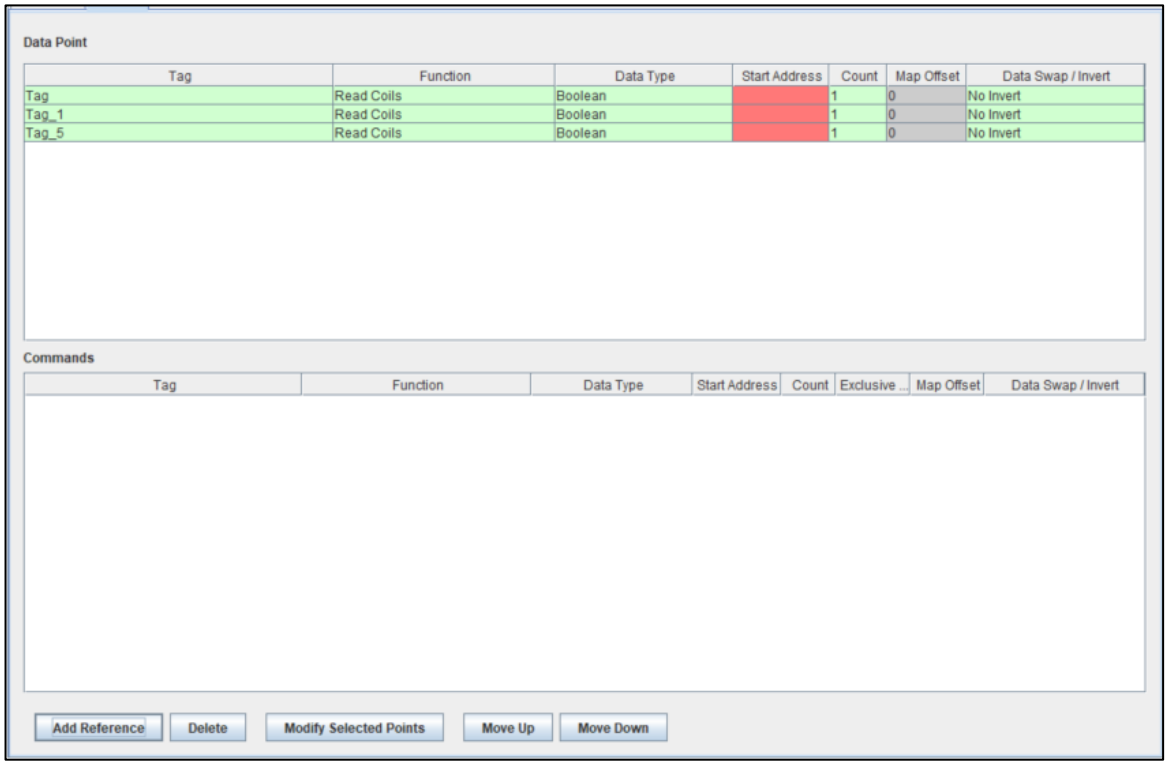


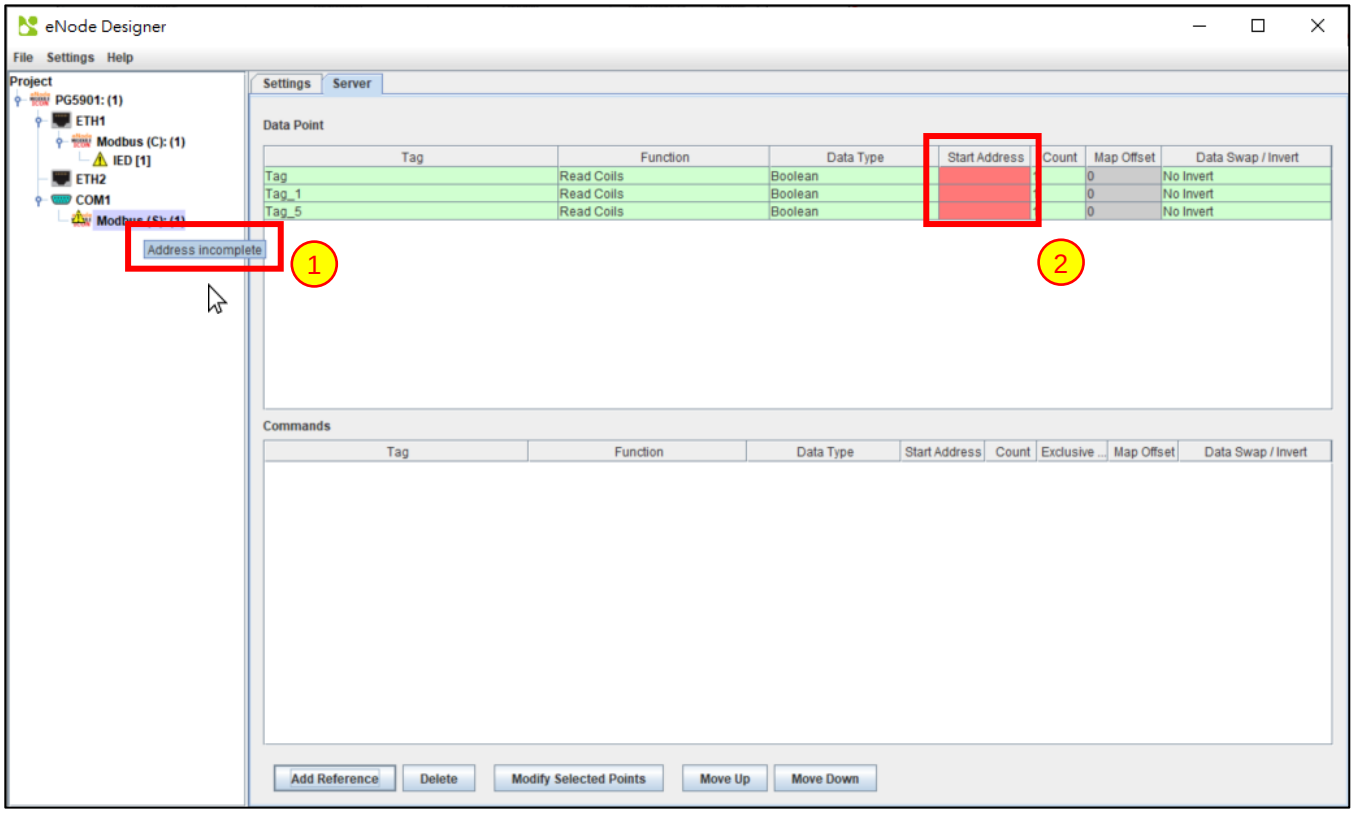
Figure 3-8 - Data point references added.

When the data points have been added, the Modbus Address field has a red background indicating the value that needs to be filled in.

3.5 Miscellaneous Common information

3.5.1 Incomplete or Conflicting Information

Incomplete or conflicting information in the data points are shown in the table with a red cell background, and will show the warning symbol on the tab (in clients) and in the project tree. Hovering over the warning icons will show further details about what is causing the warning. This allows the user to find invalid information quickly.



1 **Mouse-over a warning** to show a tooltip explaining the warning.

2 **Invalid Cells** show in red. The darker red means there is no value entered, and the lighter red means there is an address conflict.

3.5.2 Modify Selected Points Window

The “Modify Selected Points” window is used to change many row properties in one-step.

Select the data points you want to change, and then press the “**Modify Selected Points**” button beneath the tables. It will generate the following window.

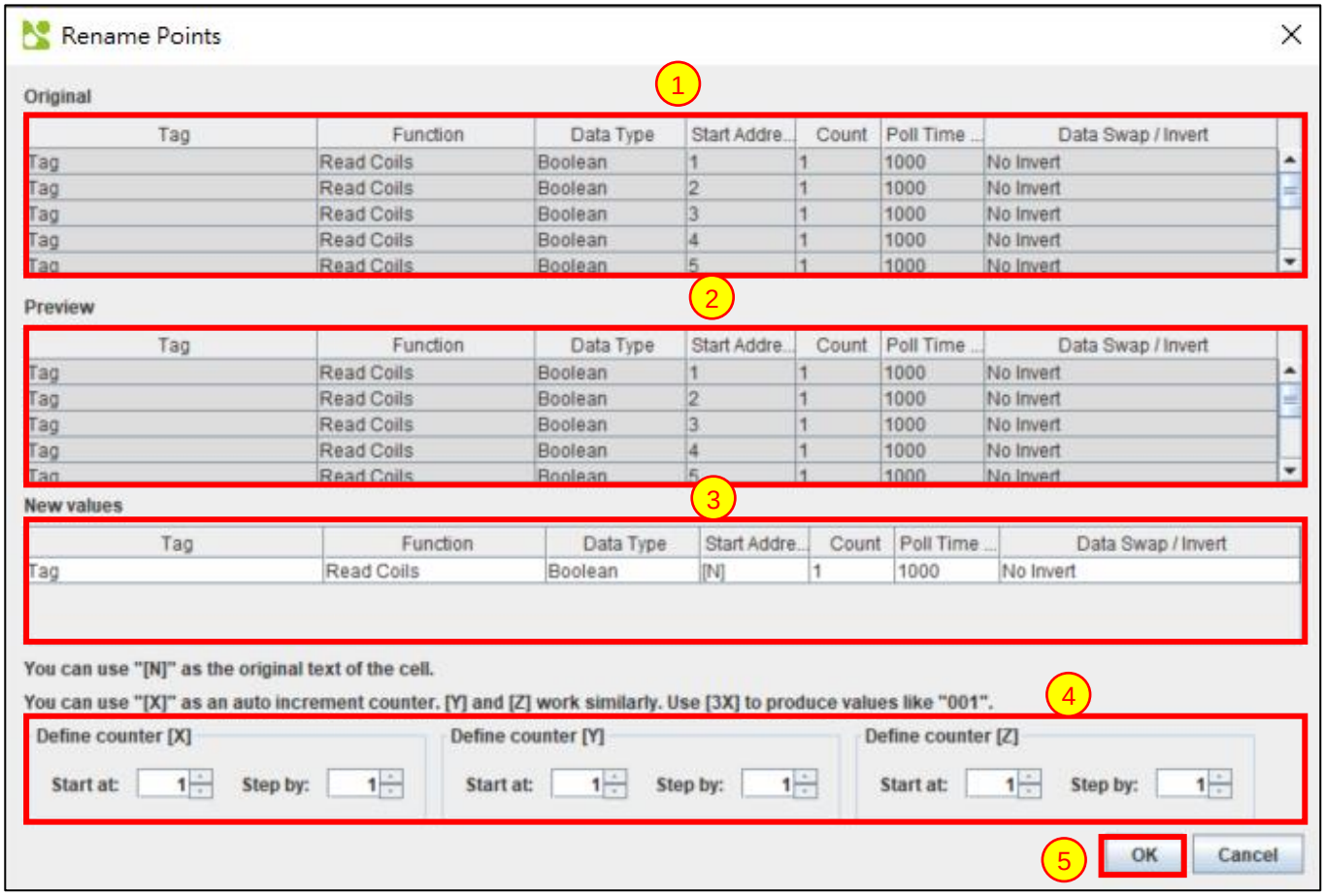


Figure 3-9 - Modify data points window example.

1 **Original table data** – Shows the original table data.

2 **Preview** – Shows the new table data that will be used if the modifications are accepted. These fields update according to the contents of (3).

3 **New values –** Shows the new values for the table cells. "[N]" can be used to maintain the original value of the cell, while the auto-incrementing counters [X], [Y] and [Z] can be used to add numbers. See Using Auto-increment.

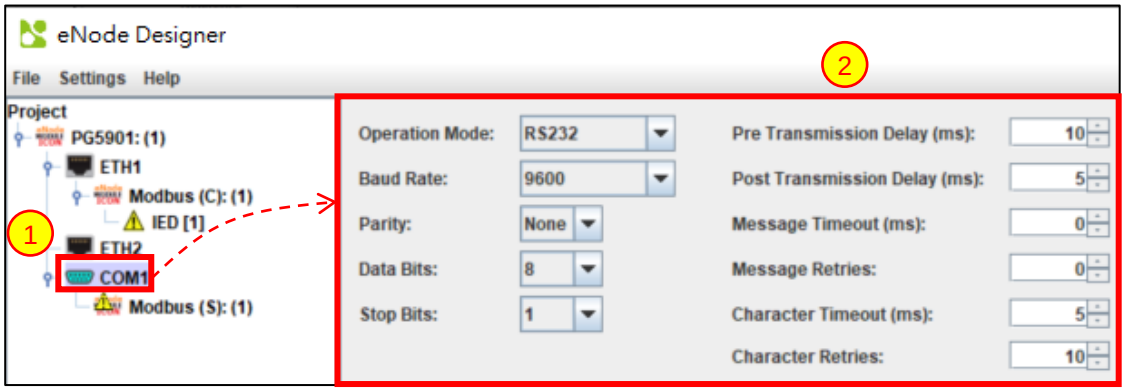
4 **Counter properties** – Sets the initial values and step amounts of the counters [X], [Y] and [Z].

5 **OK button** – to accept the modifications.

Data point references always use the *tag* and *description* of the "real" point. Consequently, these values will not be changed by a server application.

4 Communication Port Properties

The device module handles how the communication port properties are set; however, the typical method is described briefly below.



1 **Select the communication port in the project tree** – This will cause the central panel to show the port's properties.

2 **Properties** – The communication port's properties can be set.

5 Using Auto-increment Counters

The following is a full example showing how auto-increment works. The example given shows the IEC 60870-5-104 window, however the Modbus auto-increment works in the same way.

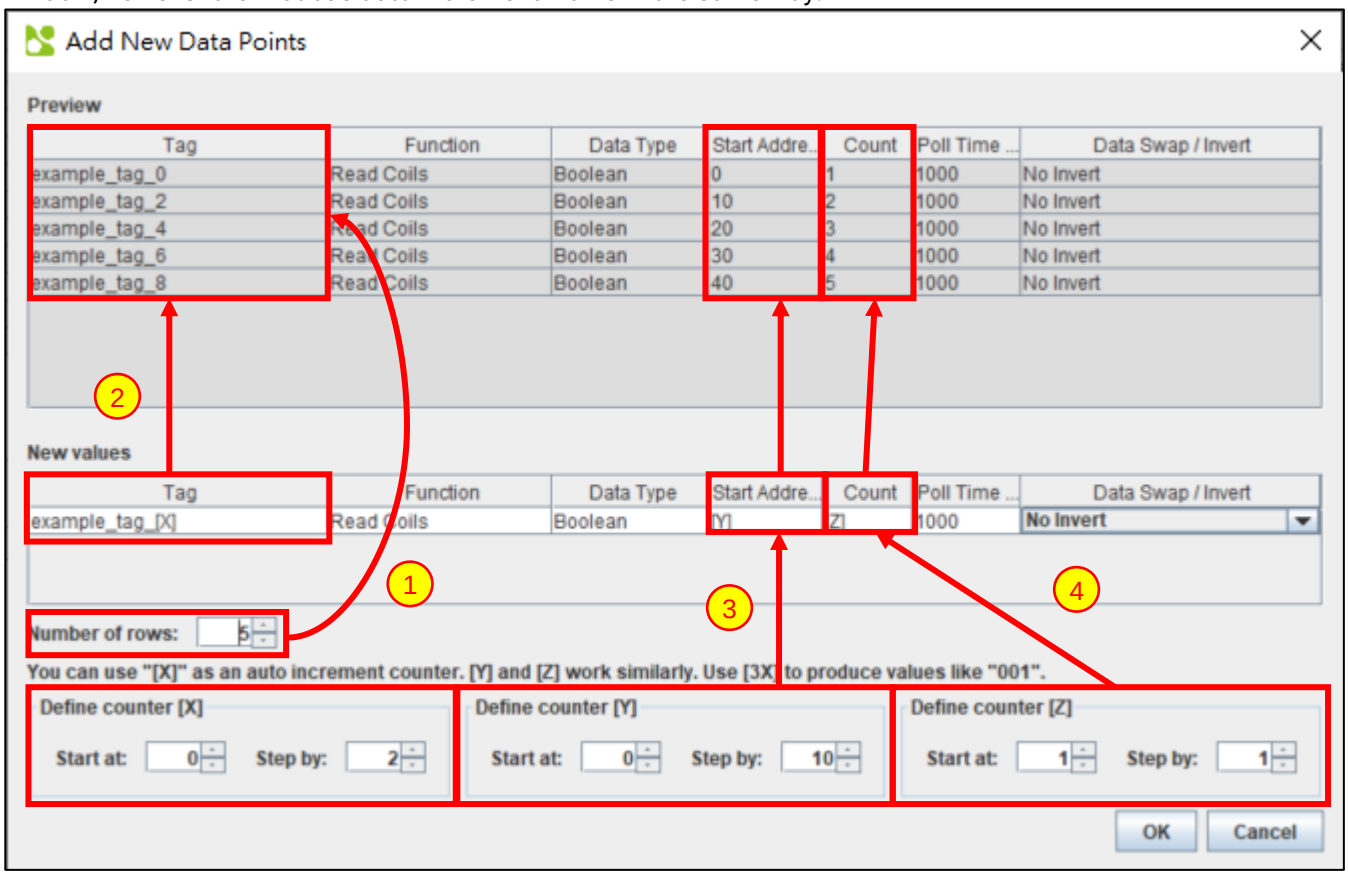


Figure 5-1 – Using Auto Increment when adding Data Points or Commands.

1 *Number of Rows* can be modified to set the number of data points or commands created from the *New values* section. As shown in the example above, five data points/commands are created and shown in the preview section because the *Number of Rows* is set to 5.

When using the auto increment counters, each auto-increment counter can be defined to “start at” and “step by” any integer amount. There is one section per counter where there values can be changed. Adjusting *Start At* will change the value that the first data point receives. Adjusting *Step By* adjusts the value that the second and subsequent values will be incremented by.

2 In this example, the *[X]* counter used The *Start At* value set to 0 and the *Step By* value set to 2. This results in the values seen in the preview section.

It is also possible to include a number within the square brackets and before the X, Y or Z while using auto increment. This will produce values that contain the entered number of digits. Any digits that are not taken up by the value determined by the *Start At* and *Step By* values will be shown as 0s.

3 In this example, the *[Y]* counter used integer 4 to indicate the number structure. This results in the values shown in the preview section.

4 In this example, the *[Z]* counter used the *Start At* and *Step By* values at default, this results in the values shown

If no auto increment value is entered in any field, each data point/command field value will be created the same with the exception of *Tag* and the object address field (in this case the *IOA*). The first new data point/command's *Tag* value will represent what was entered in the *New value* section. However, the subsequent data points/commands will contain the initial *Tag* value followed by an underscore and a number incrementing by one from 1 onwards. (Example: tag, tag_1, tag_2 etc.). This is an artefact of eNode Designer ensuring all data point tag names are unique.

5.1 Automatic Increments in Constant Values

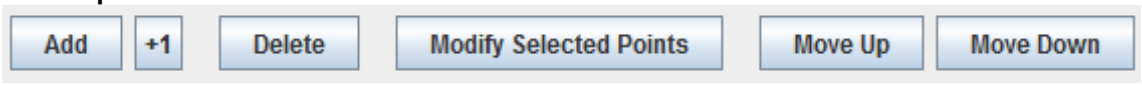
In the Modbus eNode module, the following fields will be automatically increased by one for each row, even if a constant value is entered in the "New value" field.

- Address

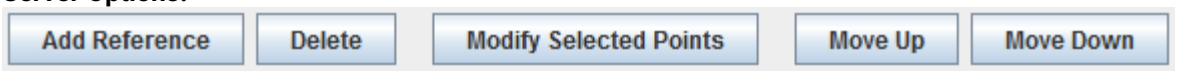
6 Reference Guide

6.1 Table Buttons

Client Options:



Server Options:



Add	Adds new data points in the client. See section 3.3.1.
+1	Adds a single new data point in the client. See section 錯誤! 找不到參照來源。 3.3.1.
Add Reference	Adds a new data point reference in the server. See section 3.4.1
Delete	Deletes the selected data points
Modify Selected Points	Modify the properties of the selected data points. See section 3.5.2.
Move Up	Moves the selected data points up one row in the table
Move Down	Moves the selected data points down one row in the table

6.2 Table Columns

6.2.1.1 Tag

Description	A unique tag name for each data point
Data Entry	String
Min Length	1
Max Length	N/A
Input Option	Mandatory

6.2.1.2 Function

Description	The Modbus function type
Data Entry	Drop Down Menu
Types	*Read Coil, Read Discrete Input, Read Holding Registers, Read Input Registers, Write Single/Multiple Coils, Write Single/Multiple Registers*
Input Option	Mandatory

6.2.1.3 Data Type

Description	The data type
Data Entry	Drop Down Menu
Types	*Boolean, Double Point, Int8U, Int8, Int16U, Int16, Int32U, Int32, Int64U, Int64, F32*
Input Option	Mandatory

6.2.1.4 Start Address

Description	The Modbus object address.
Data Entry	Integer
Range	*0 to 65535*
Input Option	Mandatory

6.2.1.5 Count

Description	Number of data unit
Data Entry	Integer
Range	*0 to 2000*
Input Option	Mandatory

6.2.1.6 Poll Time (ms)

Client Only

Description	The data will be polled at regular periods. This value sets the length of that period in milliseconds.
Data Entry	Integer
Range	*0 to 65535*
Input Option	Mandatory

6.2.1.7 Data Swap / Invert

Description	The data type
Data Entry	Drop Down Menu
Types	*No Invert or Invert for Boolean* *No Swap (01b) or Swap (10b) for Double Point* *No Swap (00ABh) or Swap (AB00h) for Int16U and Int16* *No Swap (00ABh CDEFh), Swap Word (CDEFh 00ABh), Swap Word and Byte (EFCDh AB00h), Swap Byte for Word (AB00h EFCDh) for Int32U or Int32 or F32*
Input Option	Mandatory

7 Modbus Specifications

7.1 Functional specifications

Here below protocol information and supported functions:

Protocol Specification	
Operation Mode	Client or Server
Frame Format	Serial Port: Modbus RTU/ASCII Ethernet Port: Modbus TCP
Function Codes	01: Read Coils 02: Read Discrete Inputs 03: Read Holding Registers 04: Read Input Registers 05: Write Single Coil 06: Write Single Register 15: Write Multiple Coils 16: Write Multiple registers 43/14: Read Device Identificaton
Node Address	1-255
Serial Specification	
Operate Mode	RS-232, RS-422, RS-485 (2 wires), RS-485 (4-wire)
Baud Rate	110 to 115200
Parity	None, Odd, Even, Mark, Space
Data Bits	7 or 8
Stop Bits	1 or 2
Ethernet Specification	
TCP port	1~65535 (default: 502)
Error Checking	
Modbus RTU	CRC-16
Modbus ASCII	LRC
Modbus Client	
Maximum number of Modbus Servers be connected to	64
Modbus Server	
Maximum number of Modbus Server on a single port	1
Maximum number of accepted Modbus Clients	1

7.2 Modbus Client (Ethernet) or Master (Serial)

With pre-defined commands, Modbus client/master supports the communication with multiple Modbus servers either through serial ports or TCP/IP connections. Each server/slave can have independent commands and slave definition to be reached by client/master.

7.2.1 IED Slaves

The Modbus server that the client communicates with. Each server can have different slave address and response time out configurations. If connection type is Modbus TCP/IP, up to 64 servers can be connected.

7.2.2 Slave address

The address of slave. The valid value is from 1 to 255.

7.2.3 Response timeout

Timeout for client to wait the response from the server after sending the command. The valid value is 1 to 65535 in milliseconds.

7.2.4 Frame format (for serial port)

The transmission mode that is being used. In serial port transmission, RTU mode and ASCII mode can be used.

7.2.5 IP Address and Port Number (for TCP/IP)

The IP address and TCP port of Modbus server. This is only used when transmission is through TCP/IP.

7.3 Modbus Client Commands and data points

Each Modbus request command in a client device can map one or more data points. When a client sends commands to the server and received the response, it will automatically update the values of mapped data points.

7.3.1 Function

The function code of Modbus command. The following function codes are supported:

01: Read Coils
02: Read Discrete Inputs
03: Read Holding Registers
04: Read Input Registers
05: Write Single Coil
06: Write Single Register
15: Write Multiple Coils
16: Write Multiple registers

7.3.2 Data Type

Each mapped data point can associate with one data type, the following data types are supported:

- Single Point
- Double Point
- Unsigned Byte
- Signed Byte
- Unsigned Word
- Signed Word
- Unsigned Double Word
- Signed Double Word
- Float32

7.3.3 Start Address

The starting address of Modbus command. The valid value is 0 to 65535.

7.3.4 Quantity

The quantity field of Modbus command. The valid value is 0 to 2000 for function code 01 and 02, for function code 03 and 04 is 0 to 125.

7.3.5 Poll Time

The interval to send the command by client. The valid value is 0 to 65535

7.4 Modbus Server (Ethernet) or Slave (Serial)

The device can be configured as a server to response the commands from client. The commands are defined to map data points, if the commands are read functions, the value of mapped data point is returned to client. If the commands are write functions, the value of data point is updated by that command.

7.4.1 Slave Address

The slave address of Modbus server. The valid value is 1 to 255. For Modbus TCP/IP transmission, there are up to 64 Modbus servers to accept connections from client. For Modbus RTU / ASCII transmission, only one Modbus server in one serial port is available.

7.4.2 Frame Format

The transmission mode that is being used. In serial port transmission, RTU mode and ASCII mode can be used.

7.4.3 Port

The TCP listen port in Modbus TCP mode. The valid value is 1 to 65535, default value is 502.

7.5 Modbus Server/Slave Reference Data points

When device is configured as a server, the device will receive requests from client and response the value of mapped data point. Users have to define the Modbus requests to map the existed data points to act with read or write function codes.

7.5.1 Function

The function code of Modbus command. The following function codes are supported:

01: Read Coils
02: Read Discrete Inputs
03: Read Holding Registers
04: Read Input Registers
05: Write Single Coil
06: Write Single Register
15: Write Multiple Coils
16: Write Multiple registers

7.5.2 Start Address

The starting address of Modbus command. The valid value is 0 to 65535.

7.5.3 Count

Number of referenced points started from the reference point. The valid value is 1 to 65535.

7.6 Limitation of total points

Limitation of total number of reading points: 8000 points.
Limitation of total number of writing points: 4000 points.

ATOP Technologies, Inc.

www.atoponline.com

TAIWAN HEADQUARTERS

2F, No. 146, Sec. 1, Dongxing Rd.,
Zhubei City, Hsinchu County,
302058, Taiwan, R.O.C.
Tel: +886-3-550-8137
E-mail: info@atop.com.tw
Fax: +886-3-550-8131

ATOP INDIA OFFICE & GLOBAL INQUIRIES

Prashant Mishra
No.3M-217, East of NGEF Layout, Kasturi Nagar,
Bengaluru- 560043, Karnataka, India
Tel: +91-80-492-06308
E-mail: prashant.m@atop.com.tw

ATOP CHINA BRANCH

Sam Xia
3F, 75th, No. 1066 Building,
Qingzhou North Road,
Shanghai, China
Tel: +86-21-64956231
E-mail: info@atop.com.tw

ATOP INDONESIA OFFICE

Jopson Li
Wisma Slipi, Kav. 12 Lt. 3 Unit 308
Jl. Let. Jend. S. Parman 11480 - Indonesia
Tel: +6221-5326171
E-mail: jopsonli@atop.com.tw

ATOP JAPAN OFFICE

易傑 Takashi Eki
東京都千代田区丸の内 1-1-3
日本生命丸の内ガーデンタワー3F
Tel: +81-3-4530-3390
E-mail: eki@atop.com.tw

ATOP COLOMBIA OFFICE

Brenda Solano Sarmiento
Calle 35 No. 19 - 41 - Oficina 315 Torre Norte,
Bucaramanga, Santander, Colombia
Tel: +57 322 848 9815
E-mail: Brenda.solano@atop.com.tw