ECAT-2601 User Manual

Modbus/OPC UA to EtherCAT Main Device Accelerator

Ver. 1.1, March 2026

WARRANTY

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

WARNING

ICP DAS assumes no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

COPYRIGHT

TRADEMARKS

Names are used for identification purposes only and may be registered trademarks of their respective companies.

CONTACT US

If you have any questions, please feel free to contact us via email at:

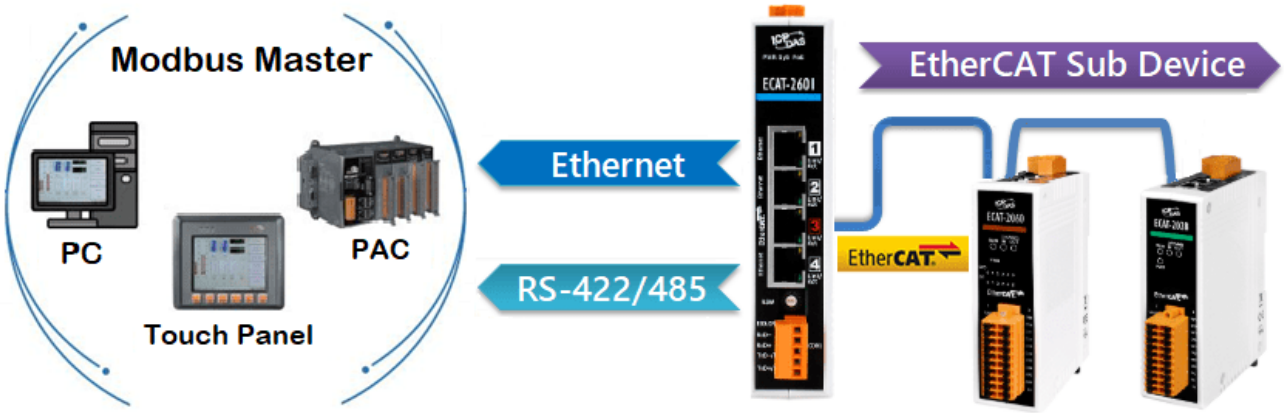
service@icpdas.com

TABLE OF CONTENTS

1. Introduction

The ECAT-2600 is a Modbus/OPC to EtherCAT Main Device Gateway with a built-in web server (HTTP/HTTPS), allowing users to configure the module, monitor, and control I/O directly through a standard web browser, enabling remote control systems. The operation is as simple as regular web browsing. It supports logic function rule settings with IF, THEN, ELSE for configuring logical judgments between I/O and software points. It also features a scheduling function, allowing predefined rules to be executed at specific times. Equipped with dual-port Ethernet switches, it supports the construction of a Daisy Chain network topology, simplifying wiring configuration and reducing maintenance complexity, cabling costs, and the need for additional switches while enhancing network scalability.

In the event of a power failure in an individual ECAT-2600 module, the automatic LAN Bypass function ensures that the Daisy Chain network continues to communicate, maintaining the stability of Ethernet communication.



The ECAT-260x series is equipped with multiple cybersecurity mechanisms, offering comprehensive protection for industrial communications. Its built-in web server supports authentication and HTTPS encrypted communication, ensuring the security of web communication and RESTful API data, preventing information leakage. It supports various industrial IoT communication protocols, including Modbus, OPC UA Server/Client, MQTT Client, and LwM2M Client, with the option to use TLS/DTLS communication certificates and encryption mechanisms. This provides additional protection for valuable data and devices, preventing unauthorized access, interception, or tampering during transmission. Additionally, the ECAT-260x series includes an IP address filtering feature, allowing users to permit or block access from specific IP addresses. In case of network

anomalies, the system can activate defense mechanisms to minimize potential damage, providing comprehensive cybersecurity protection. Here's a comparison between the ECAT-260x series and traditional Ethernet I/O:

Series / Features	ECAT-260x	Traditional Gateway
Modbus	Modbus TCP/UDP/TLS	Modbus TCP/UDP
OPC UA	Account/Password, Anonymous, Certificate	-
MQTT	Account/Password, Anonymous, Certificate	-
LwM2M	Account/Password, Anonymous, Certificate	
Web/Restful API	HTTP/HTTPS	HTTP
Transmission Encryption	AES-128、AES-256、SHA-256	-
Active Transmission	Yes	-
IEEE 802.1X	EAP、PEAP	-

The ECAT-260x series supports IPv4/IPv6 Dual-stack technology, allowing it to simultaneously hold both IPv4 and IPv6 addresses and communicate with either IPv4 or IPv6 devices. It automatically analyzes packet sources and determines network protocols without the need for additional configuration by the user. The module supports SLAAC (Stateless Address Auto configuration) for automatic IPv6 address assignment, enabling true plug-and-play functionality for IPv6.

Additionally, the ECAT-260x series features built-in 802.1Q VLAN technology, providing enhanced cybersecurity and real-time industrial communication. Through the VLAN function, network communication resources can be partitioned or reserved, improving flexibility in network management.

1.1 Features

1. Comprehensive Cybersecurity Mechanisms

The ECAT-260x series has a built-in HTTP/HTTPS web server, with HTTPS providing a secure web service interface. Users can configure module settings, control output channels, and monitor connections and I/O status via the web interface or RESTful API, all without the need for additional software installation. The ECAT-260x series supports industrial IoT communication protocols such as Modbus, OPC UA Server/Client, MQTT Client, and LwM2M Client, with the option to enable SSL/TLS certificates and encryption. This ensures that sensor data is encrypted during transmission, preventing unauthorized access. Additionally, the ECAT-260x series includes an IP address filtering function, allowing users to set specific IP addresses for permitted or blocked access. The Netstat feature helps users check the real-time connection status of listening TCP ports and connected clients.

2. Denial-of-Service (DoS/DDoS) Attack Defense

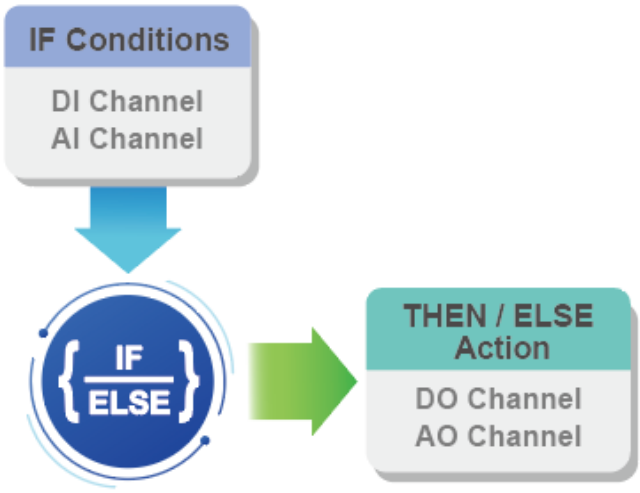
The ECAT-260x series is equipped with built-in defense mechanisms against DoS/DDoS attacks. It can actively regulate network traffic to mitigate the interference caused by a large volume of abnormal network packets on Ethernet I/O operations. When the network experiences a surge in abnormal packets, the ECAT-260x series detects it in real-time and activates its defense mechanisms, managing and controlling network traffic to prevent the negative impact of packet floods. This ensures the stable operation of the Ethernet I/O system.

3. Support for IPv4/IPv6 Dual-Protocol

The ECAT-260x series utilizes Dual-stack technology, allowing it to monitor packets from both IPv4 and IPv6 networks and maintain both IPv4 and IPv6 addresses simultaneously. Users do not need to perform additional configurations, as the system automatically analyzes packet sources and determines the appropriate network protocol. The ECAT-2600 is equipped with SLAAC (Stateless Address Auto configuration), enabling true plug-and-play functionality for IPv6. Despite the complexity of IPv6 addresses, the module automatically requests IP information from the router using SLAAC technology, eliminating the need for manual configuration while ensuring IPv6 plug-and-play convenience.

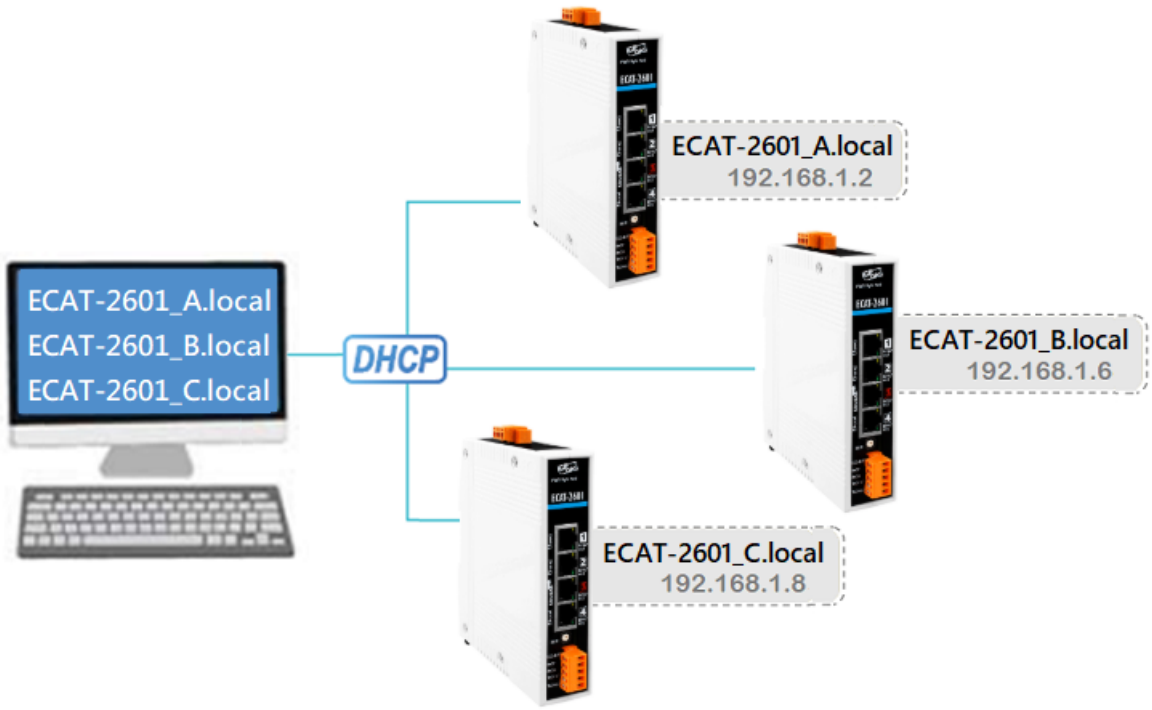
4. Rule Logic Engine

The ECAT-260x series has a built-in logic function engine that supports IF-THEN-ELSE rules for configuring logical decisions between physical I/O and software points. This allows for stable and rapid execution of automated monitoring procedures. The logic engine also supports encrypted (or unencrypted) email notifications. Users can incorporate email sending actions into the IF-THEN-ELSE logic to promptly send email messages to relevant personnel when specified events occur.



5. Support for mDNS Hostname Resolution

The ECAT-260x series supports the mDNS (Multicast DNS) protocol, which provides easy-to-remember and fixed domain names ending in **.local** (e.g., EthernetIO.local) for local network communication. Browsers or software that support mDNS can communicate with the ECAT-2600 using its mDNS domain name. Regardless of whether the network uses a static IP address or DHCP, if the IP address changes, users can still communicate with the ECAT-260x series using the fixed mDNS domain name. This avoids communication problems that may occur due to IP address changes.

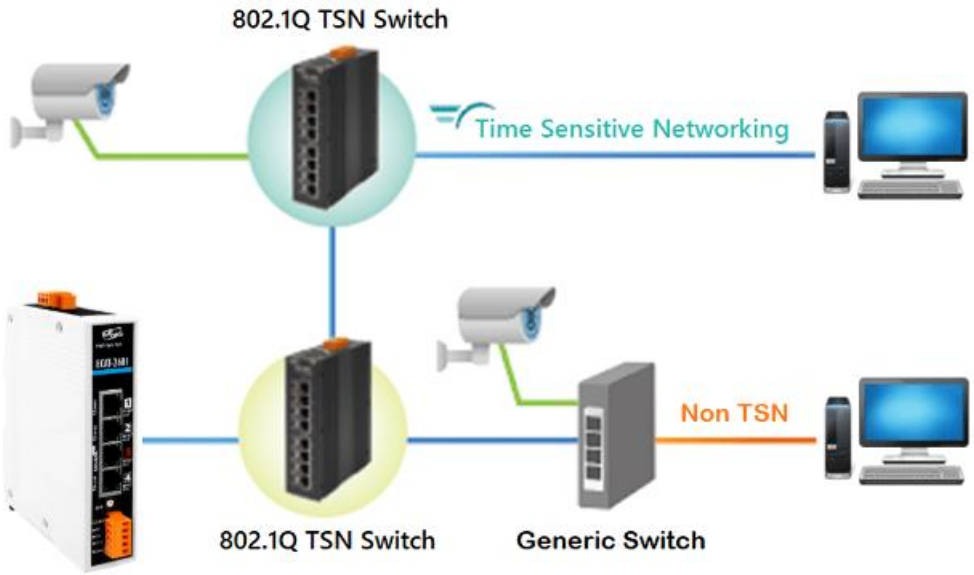


6. Data Logging and Backup Functionality

The ECAT-2600 features a Data Logger function that can record real-time values from I/O module channels either periodically (by time intervals) or based on event triggers. The recorded data is stored in .csv format files, which can be downloaded. Additionally, data log files can be automatically sent back to a backend management center via FTP/OneDrive/Google Drive for data consolidation and analysis.

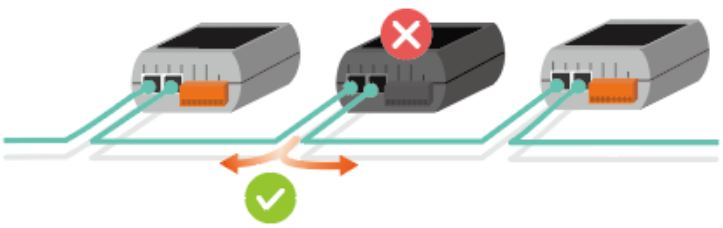
7. 802.1Q and 802.1p Priority Tagging

Supporting 802.1Q and 802.1p priority tagging, the ECAT-2600 module tags data frames on selected network protocols for use with 802.1Q compliant switches. In environments with limited network bandwidth, this configuration effectively preserves network resources for time sensitive network communication, ensuring low-latency and high-reliability transmission.



8. Dual Ethernet Ports, Supporting Daisy-Chain and LAN Bypass

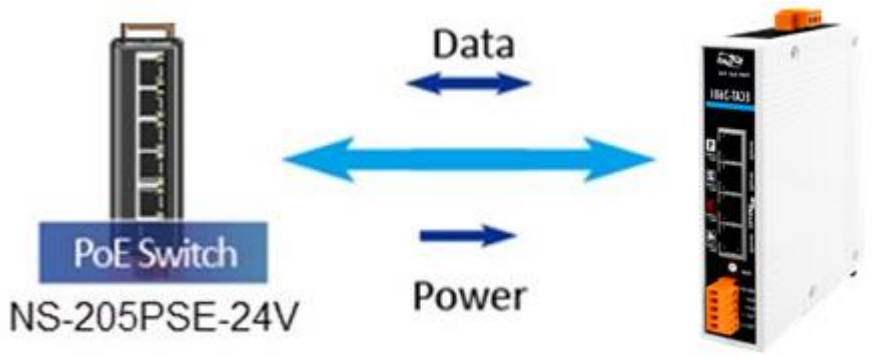
The ECAT-260x series are equipped with a dual-port Ethernet switch, supporting the construction of a Daisy-Chain network topology. The benefits of using a Daisy-Chain topology include simplified wiring and maintenance complexity, reduced wiring and maintenance costs, and improved network scalability. Additionally, the ECAT-260x is equipped with a LAN Bypass function, which ensures Ethernet communication stability. If any ECAT-2600 module loses power, this function automatically activates to maintain network connectivity.



9. Power over Ethernet (PoE)

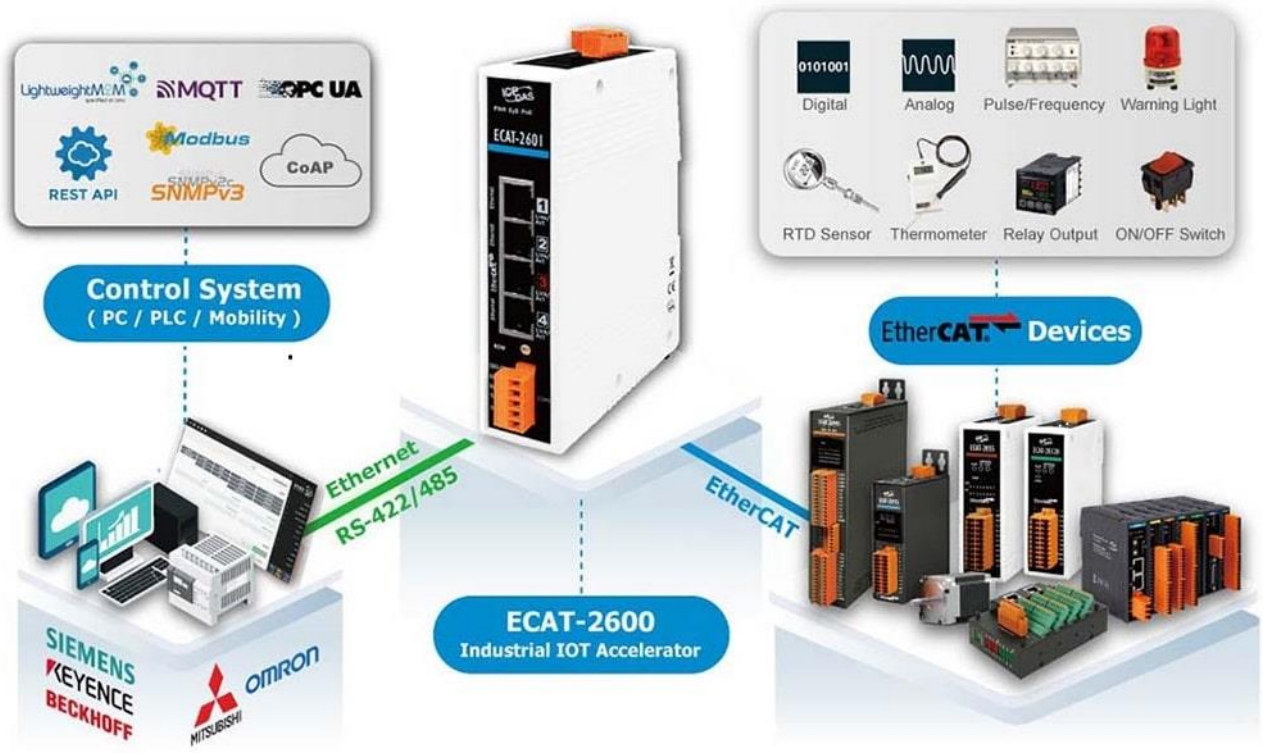
The module has integrated Power-over-Ethernet (PoE), it allows power and data to be carried over a single Ethernet cable, so a device can operate solely from the power it receives through the data cable. This innovation allows greater flexibility in office design, higher efficiency in systems design, and faster turnaround time in set-up and implementation. The module feature true IEEE 802.3af-compliant (classification, Class 1) Power over Ethernet (PoE) using both Ethernet pairs (Category 5 Ethernet cable).

The module can receive power from an auxiliary power sources like AC adapters and battery in addition to the PoE enabled network. This is a desirable feature when the total system power requirements exceed the PSE's (power sourcing equipment) load capacity. Furthermore, with the auxiliary power option, the module can be used in a standard Ethernet (non-PoE) system.



1.2 Applications

- ✓ Factory Automation
- ✓ Building Automation
- ✓ Home Automations
- ✓ Remote Diagnosis and Management



More Information

- **Software- eSearch Utility**

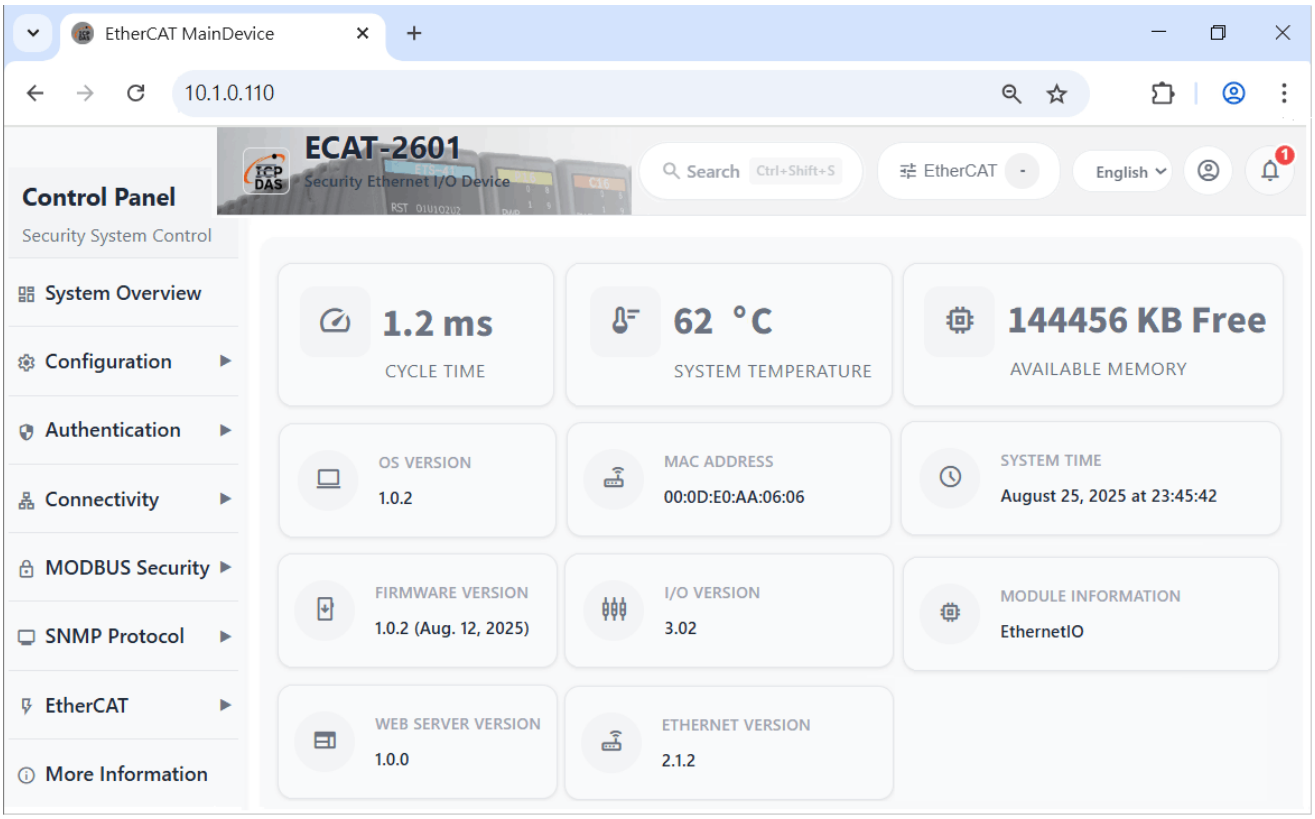
https://www.icpdas.com/en/product/guide+Software+Utility_Driver+eSearch__Utility

- **ECAT-2601 download center (Documentation, Firmware..., etc.)**

https://www.icpdas.com/en/product/ECAT-2601

1.3 Web Server Technology

Web server technology enables the ECAT-2601 to be configured via a standard web browser interface, e.g., Google Chrome, Internet Explorer, or Firefox, etc. This means that it is easy to check the configuration of the ECAT-2601 via an Ethernet network without needing to install any other software tools, thereby reducing the learning curve required for maintaining the device.



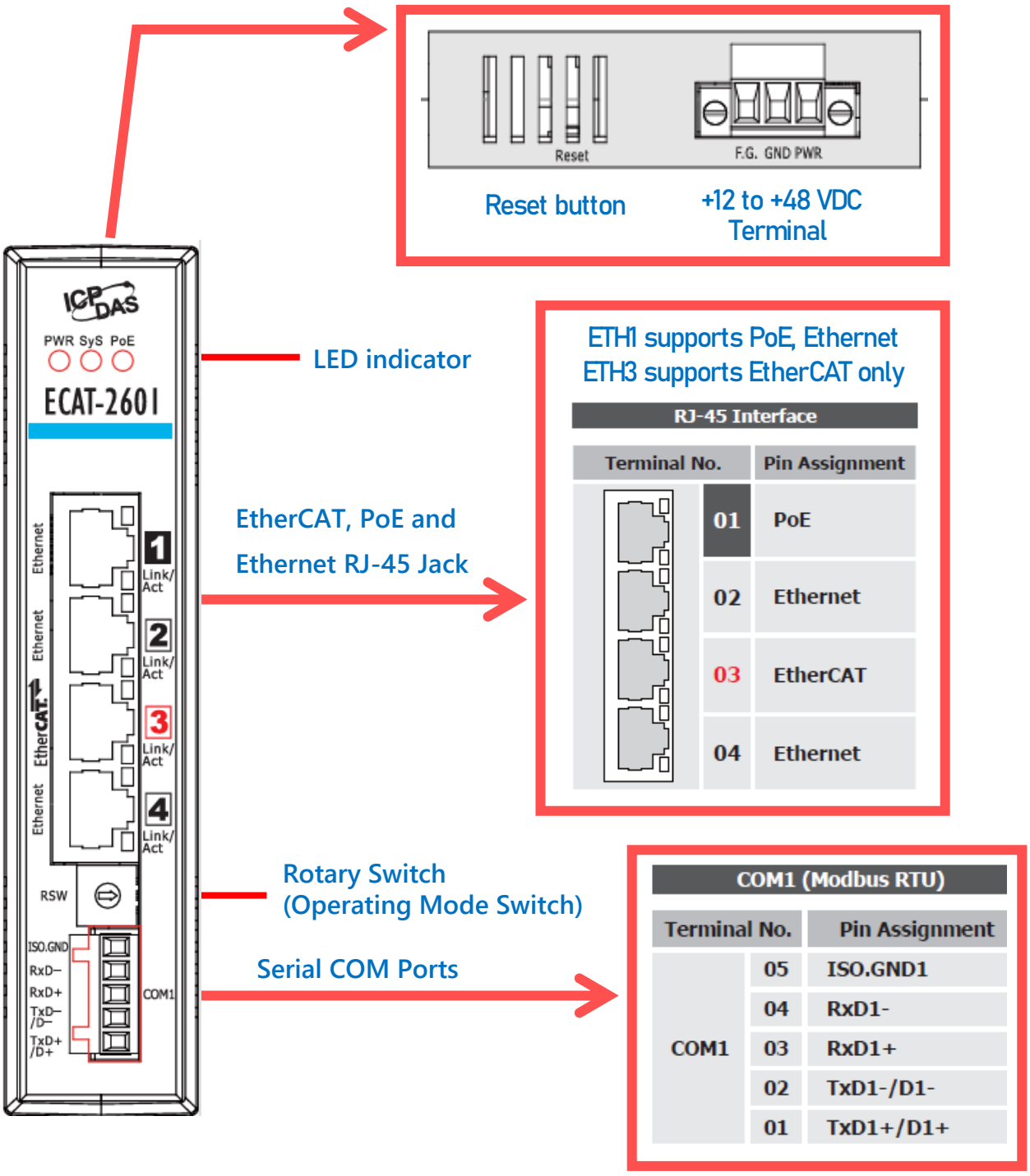
2. Hardware Information

This chapter provides a detailed description of the front panel, the hardware specifications, the pin assignments, the wiring notes and the dimensions for the ECAT-2601 series modules.

2.1 Specifications

Model	ECAT-2601
Software	
Function	Web Interface for Configuration
COM Ports	
Ports	1 x 422/485
Protocol	Modbus RTU (Salve)
EtherCAT	
Ports	RJ-45 x 1 Distance between Stations: Max. 100 m (100BASE-TX) Data Transfer Medium: Ethernet/EtherCAT Cable (Min. CAT 5e)
Cycle Time	0.125 ~ 32ms
Distributed Clocks	Yes
Ethernet	
Ports	RJ-45 x 3 2 x Distance between Stations: Max. 100 m (100BASE-TX) 1 x PoE (IEEE 802.3af, Class 2)
Protocol	Modbus TCP, Modbus TCP/TLS, Modbus UDP, OPC UA, SNTP, HTTP/HTTPS
Power	
Consumption	0.22 A @ 24 VDC
Powered from PoE	IEEE 802.3af, Class 2
Powered from Terminal Block	+12 ~ +48 VDC
Mechanical	
Casing	Plastic
Dimensions (W x L x H)	119 mm x 31 mm x 157 mm
Installation	DIN-Rail Mounting
Environment	
Operating Temperature	-25 ~ +75 °C
Storage Temperature	-30 ~ +80 °C
Humidity	10 ~ 90% RH, non-condensing

2.2 Appearance



PoE and Ethernet RJ-45 Jack

The ECAT-2601 series module is equipped with two RJ-45 jacks that are used as the 10/100 Base-TX Ethernet port and features networking capabilities, only ETH1 supports PoE power supply. When an Ethernet link is detected and an Ethernet packet is received, the **Link/Act LED (Orange)** indicator will be illuminated. When power is supplied via PoE (Power-over-Ethernet), the **PoE LED** indicator will be illuminated.

EtherCAT Port

The ECAT-2601 module is equipped with one RJ-45 jacks that are used as the EtherCAT port and features networking capabilities, only ETH3 supports EtherCAT connection. When an EtherCAT link is detected and an EtherCAT packet is received, the **Link/Act LED (Orange)** indicator will be illuminated.

+12 to +48 VDC Jack

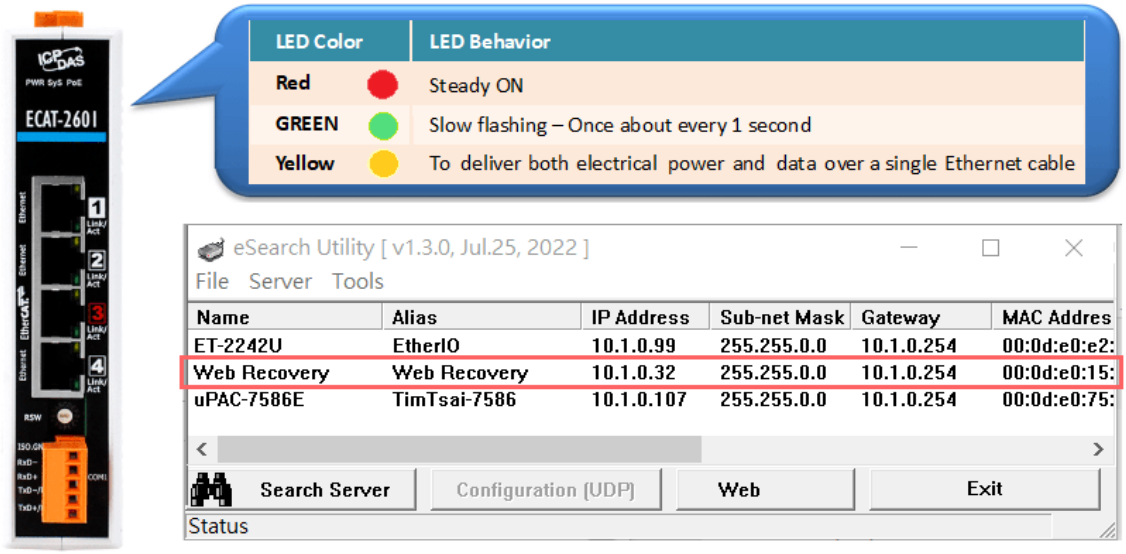
The ECAT-2600 series is equipped with a +12V_{DC} to +48 V_{DC} terminal that can be used to connect a power supply. If no PoE switch is available on site, a DC adapter can be used to power the ECAT-2601 module.

LED Indicator

Once power is supplied to the ECAT-2601 series module, the system LED indicator will illuminate. An overview of the system LED functions is given below:

LED Function	LED Color	LED Behavior
Power on (PWR)	**Red**	Steady ON
Running Firmware (Sys.)	**GREEN**	Slow flashing – Once about every 1 second
Power over Ethernet (PoE)	**Yellow**	To deliver both electrical power and data over a single Ethernet cable.

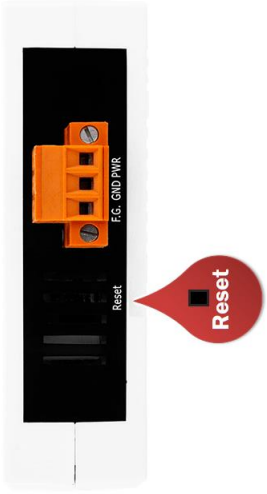
Reset button

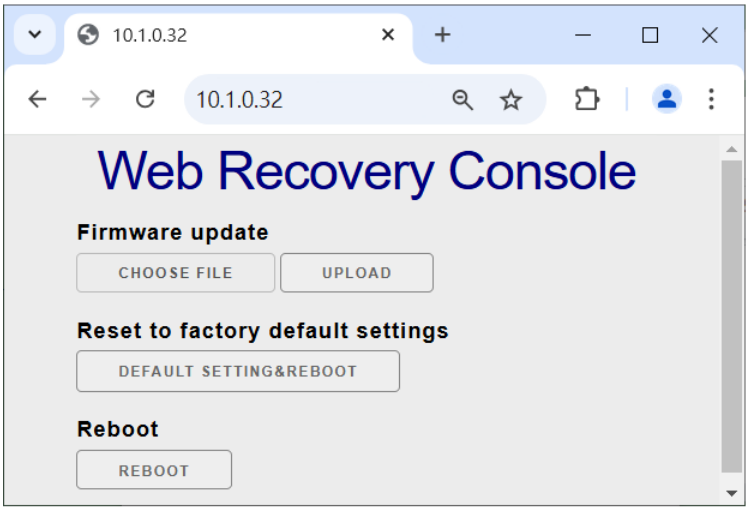


It can help you to upload the firmware in rescue mode when your ECAT-2601 fails to recovery/update firmware in a normal way.

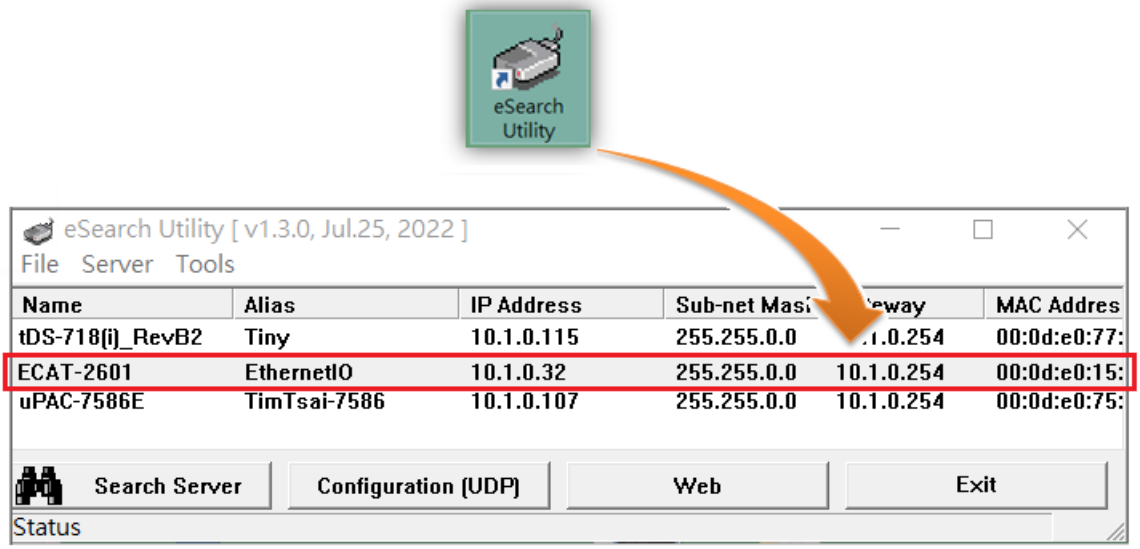
There are 6 recovery processes as shown below:

1) Turn off the Power.
2) Keep pressing the reset button then turn on the power.
3) Hold the reset button for about 6 seconds, the **Sys.** LED will flashing.
4) Open the eSearch Utility and scan devices, you will see that the name is "**Web Recovery**".
5) Open the Web browser, such as Google Chrome, enter the IP address of the ECAT-2601 module in the address bar and then click "Enter", you will see the Recovery menu.
6) Click "***Firmware update***" or "***Reset to factory default settings***" button to begin the restore process.





Clicking **"Search Server"** button on eSearch Utility again, you will see that the name is "ECAT-2601".



Rotary Switch

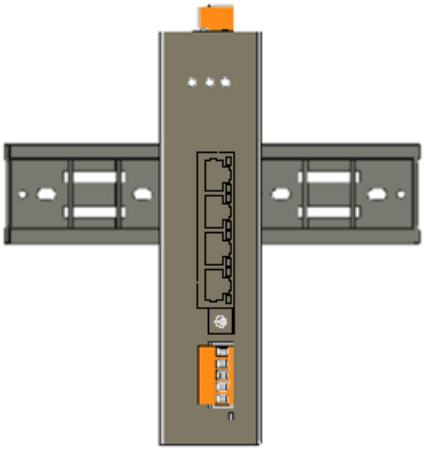
Rotary switch is a mode selector. All selectors can be set by the user (range 0~15).

Serial COM1 Ports

The number of serial COM Ports available depends on the type of ECAT-2601 module (Modbus RTU (Salve)). For more detailed information regarding the pin assignments for the Serial COM1 port.

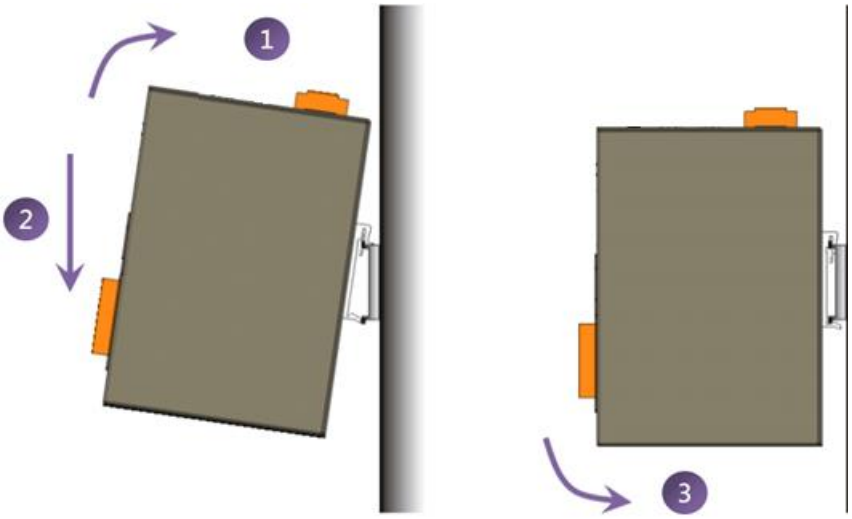
ECAT-2601		
Terminal No.		**Pin Assignment**
COM1	05	ISO.GND
	04	RxD1-
	03	RxD1+
	02	TxD1-/D1-
	01	TxD1+/D1+

DIN-Rail Mounting

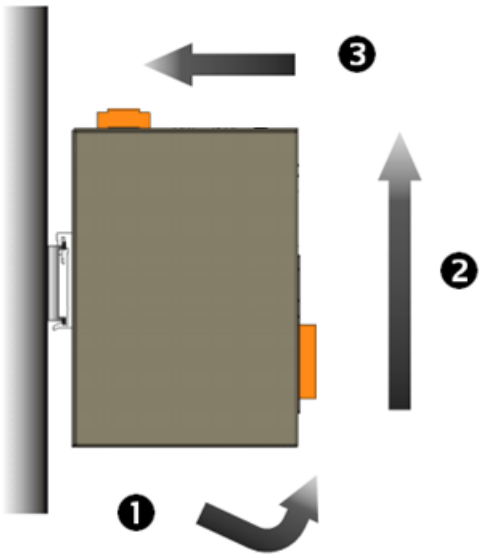


The ECAT-2600 series modules include simple rail clips on the bottom of the chassis that allow them to be reliably mounted on a DIN-Rail or a wall. For more detailed information regarding DIN-Rail Mounting, refer to the illustration in figure below.

Mounting on a DIN-Rail

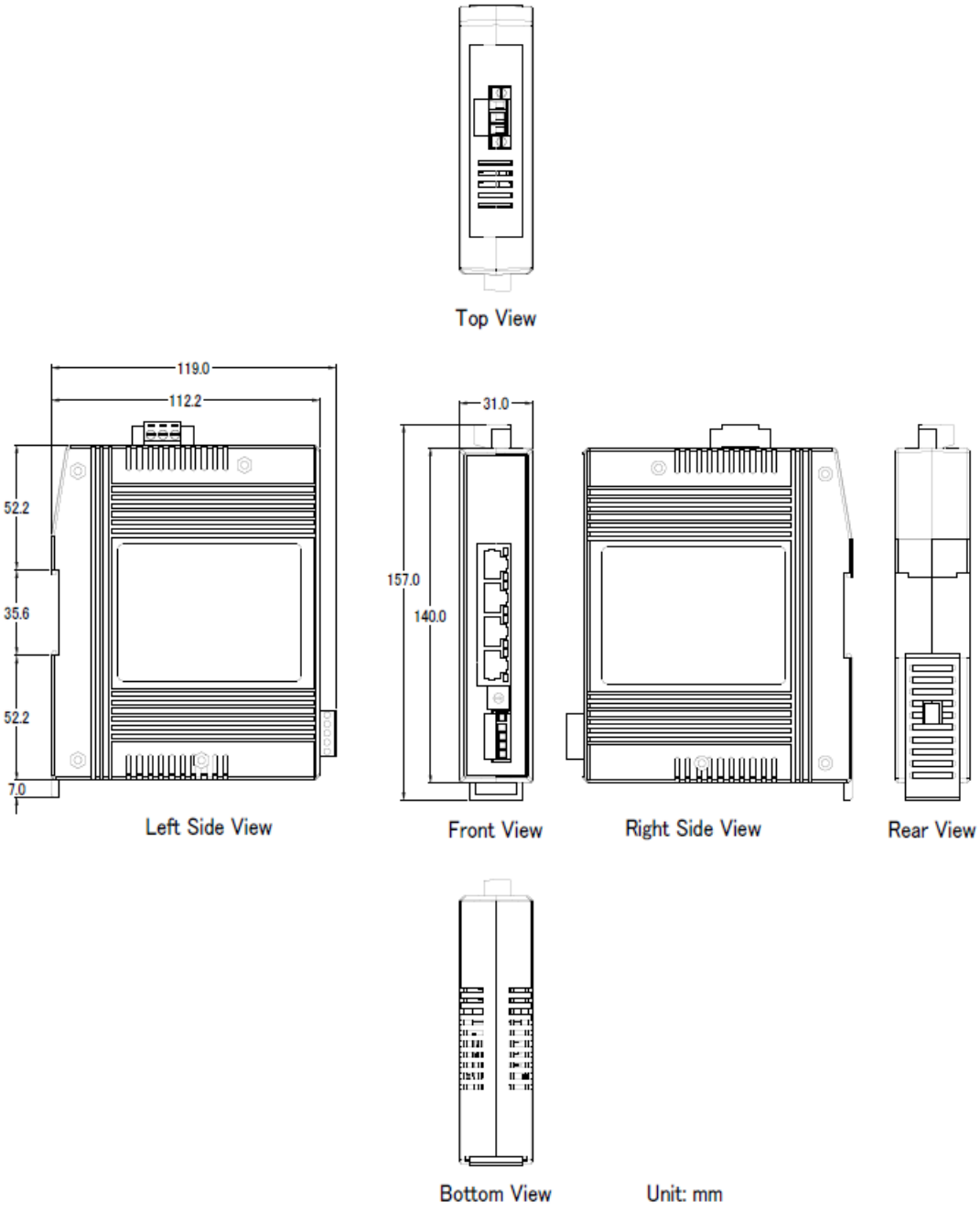


Dismounting form a DIN-Rail



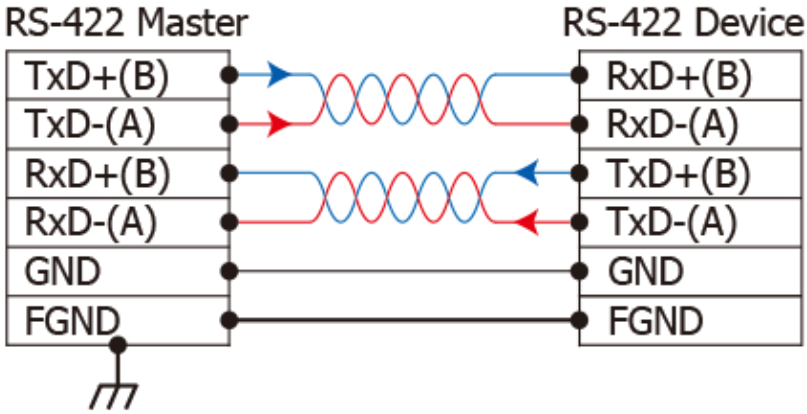
2.3 Dimensions

The following diagrams provide the dimensions of the ECAT-2601 series module. All dimensions are in millimeters.

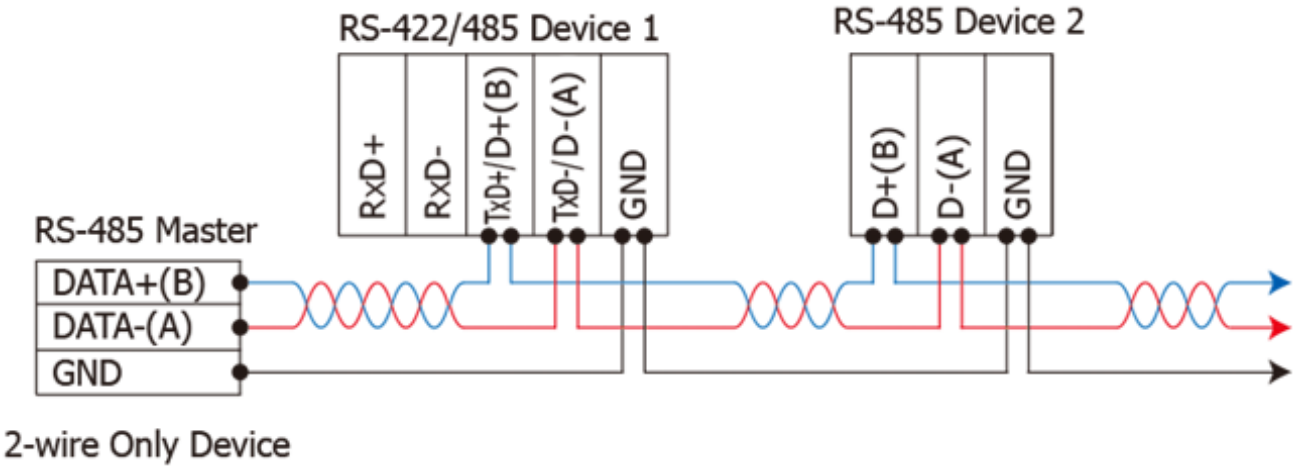


2.4 Notes for RS-485/422 Interfaces

RS-422 Wiring



RS-485 Wiring



Notes:

1. **Usually, you have to connect all signal grounds of RS-422/485 devices together to reduce common-mode voltage between devices.**
2. **Twisted-pair cable must be used for the D+/- wires.**
3. **Both two ends of the cable may require a termination resistor connected across the two wires (D+ and D-). Typically 120 Ω resisters are used.**
4. **The D+ and B pins are positive-voltage pins, and D- and A pins are negative-voltage pins in the above figure. The B/A pins may be defined in another way depending on devices, please check it first.**

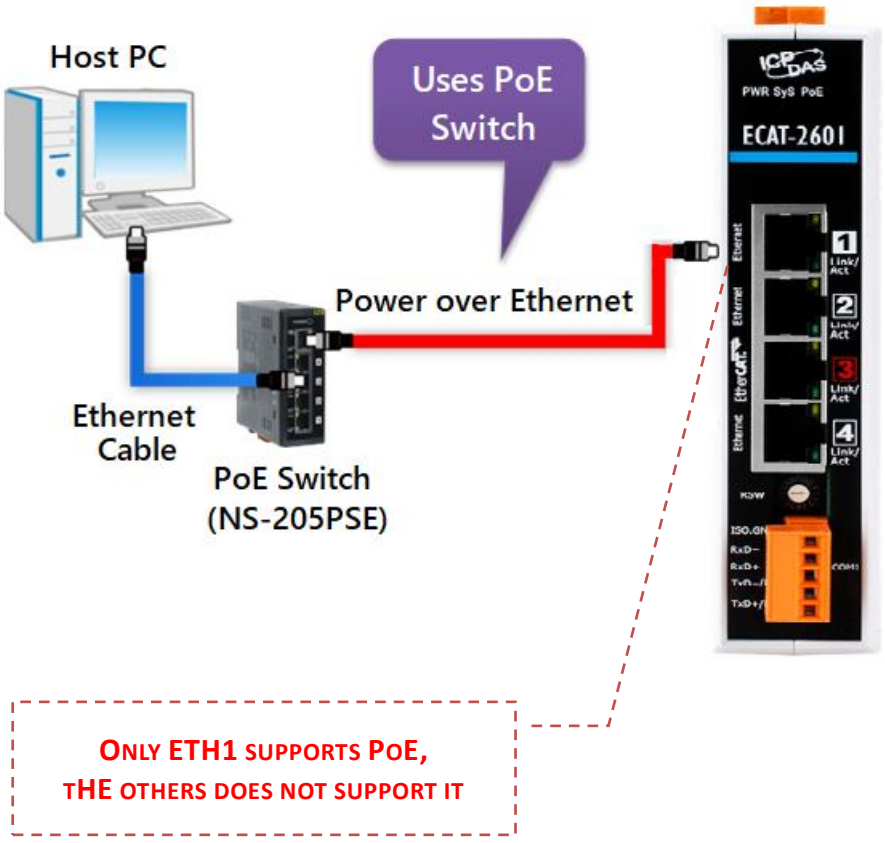
2.5 Connecting the Power and Host PC

1. Ensure that the network settings on your PC are configured correctly.

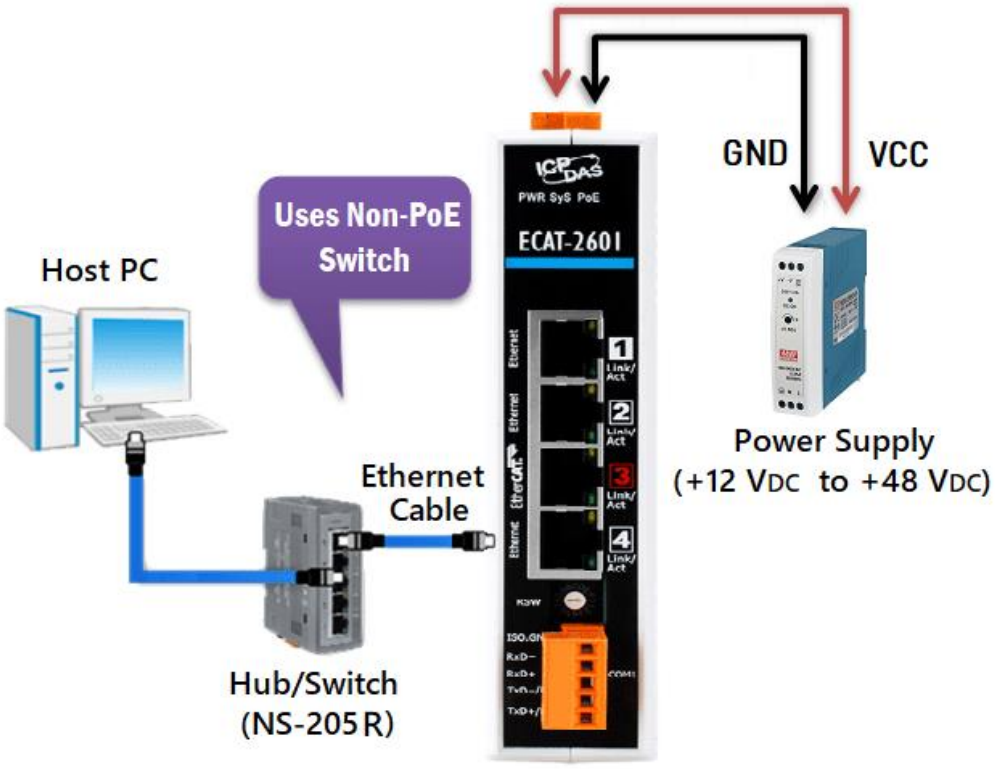
2. Ensure that the Windows firewall or any Anti-Virus firewall software is correctly configured or temporarily disable these functions; otherwise the **"Search Servers"** function in the eSearch Utility may not work as required. You may need to contact your System Administrator for more details of how to do this.

3. Connect both the ECAT-2601 and the Host computer to the same sub-network or the same Ethernet Switch, and then supply power (PoE or +12 to +48 VDC) to the ECAT-2601.

PoE Power Supply



+12 to +48 VDC Jack Power Supply (Non-PoE)



4. Verify that the System (Sys.) LED indicator is flashing.

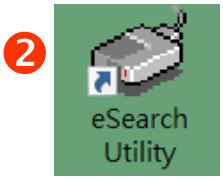
3. Getting Started for ECAT-2601 on IPv4

This chapter provides detailed information about the “Self-Test” process, which is used to confirm that the ECAT-2601 series module is operating correctly. Before beginning the “Self-Test” process, the wiring test, Ethernet configuration and search/Modbus utility driver installation procedures must first be fully completed. Follow the procedure described below:

Note: This chapter is based on IPv4 environment.

3.1 Configuring Network Settings

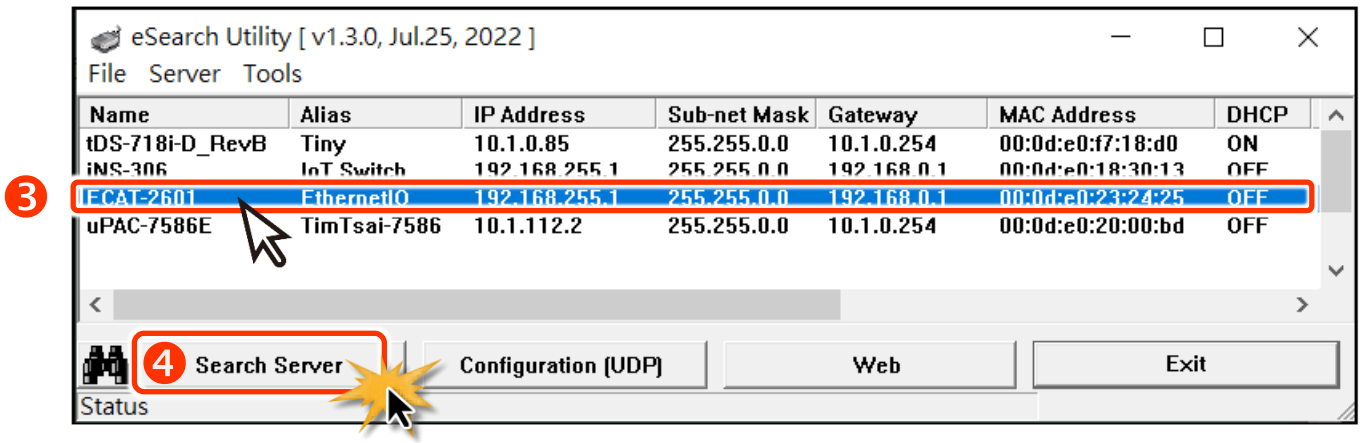
1. Download the **eSearch Utility** and install it according to the installation instructions.
   The eSearch Utility can be obtained from the ICP DAS web site.
   The location of the download link is shown below:
   https://www.icpdas.com/en/product/guide+Software+Utility_Driver+eSearch__Utility



2. Double click the **eSearch Utility** shortcut on the desktop.

3. Click the **“Search Servers”** button to search your ECAT-2601.

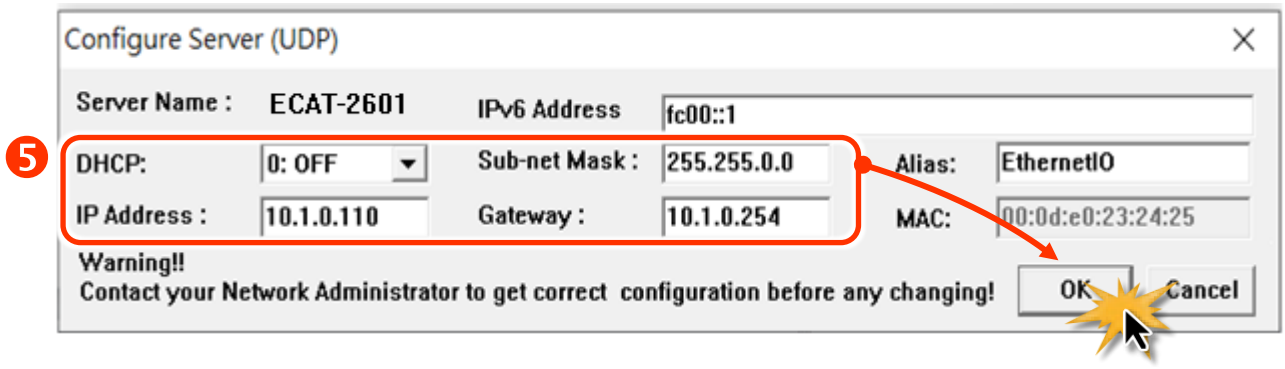
4. Once the search process is complete, double-click on the module name to open the **“Configure Server”** dialog box.



Factory Default Settings of ECAT-2601 Series Module:

IPv4 settings		**Writable**
IP Address	192.168.255.1	✓
Subnet Mask	255.255.0.0	✓
Gateway	192.168.0.1	✓
IPv6 settings		**Writable**
User-defined	fc00::1	✓
Link-Local	EUI-64 format	×
SLAAC	Auto-Configure	×

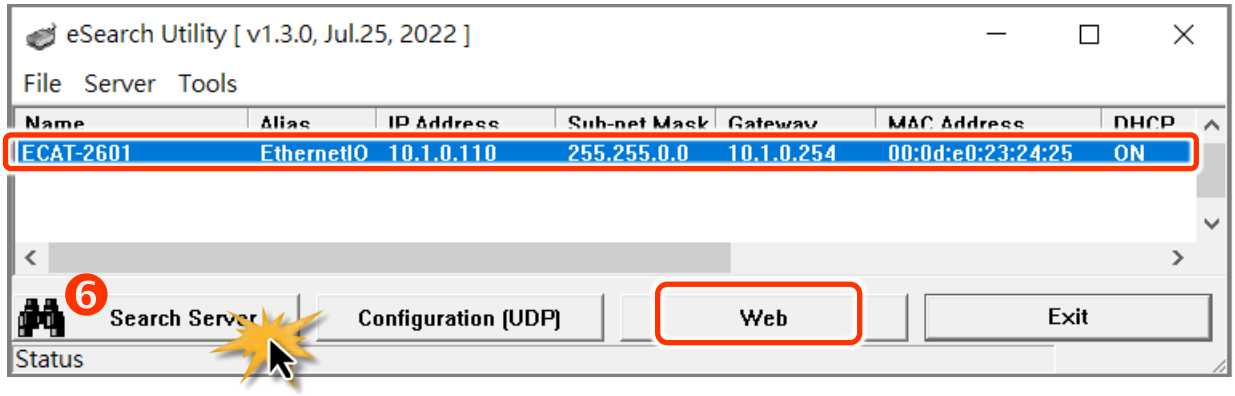
5. Enter the network settings information, including the **IP, Mask, Gateway addresses**, and then click "**OK**" button. The new settings for the ECAT-2601 will take effect within 2 seconds.
   If you don't know the correct network configuration information, contact your Network Administrator to obtain the details.



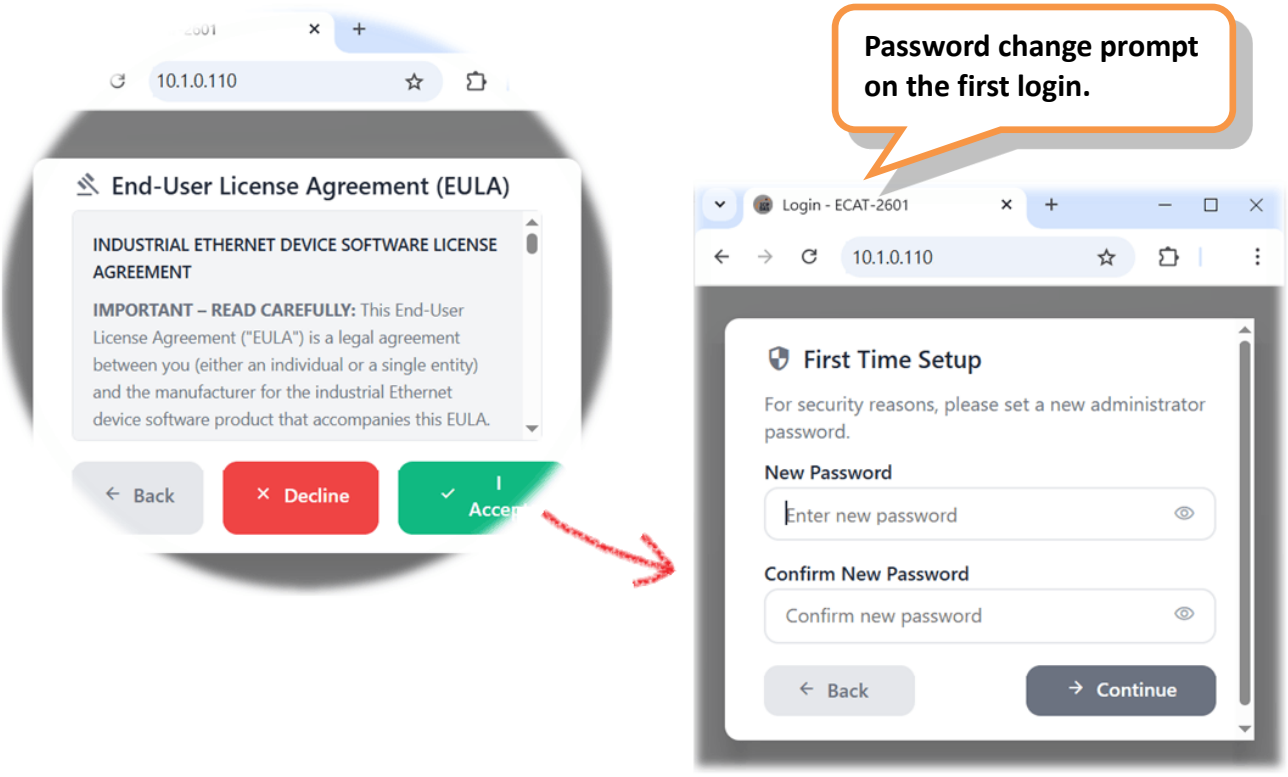
Note: The DHCP feature is only valid on IPv4.

6. Wait 2 seconds and click **"Search Servers"** button again to ensure the ECAT-2601 is working well with new configuration.

7. Click the **"Web"** button to log in to the web configuration pages.
   (Or enter the URL address of the ECAT-2601 in the address bar of the browser.)



8. When the ECAT-2601 web page appears, a pop-up box containing an **End-User License Agreement (EULA)** will appear. You will need to read this and click "**I Accept**" before you can continue.



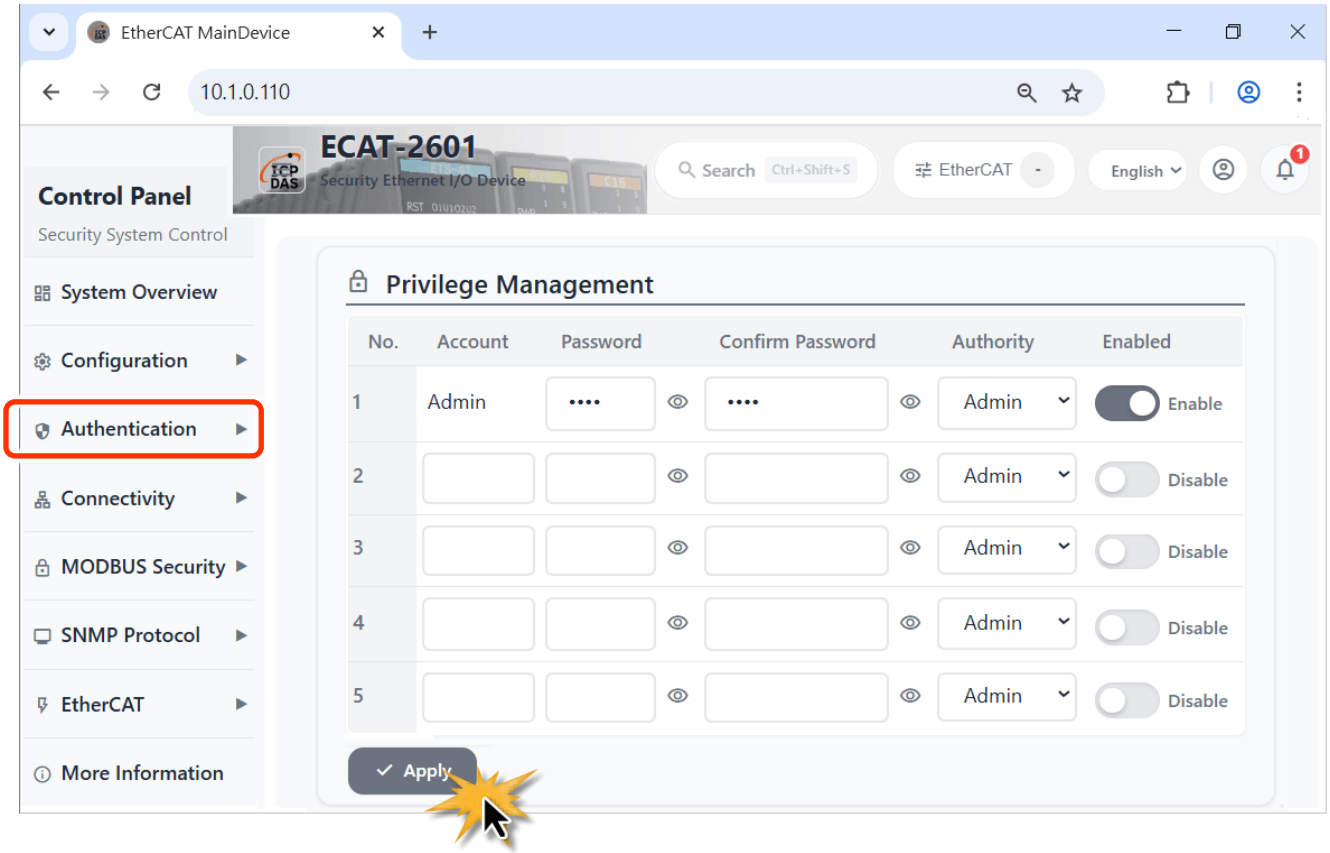
9. Once you have set the new password in the pop-up box, the user will be able to log in to the ECAT-2601 homepage.

3.2 Configuring the Password

1. Open a web browser, such as Google Chrome, Internet Explorer, or Firefox, enter the IP address of the ECAT-2601 module in the address bar and then click “Enter”, or click the **“Web”** button in the **eSearch Utility**.

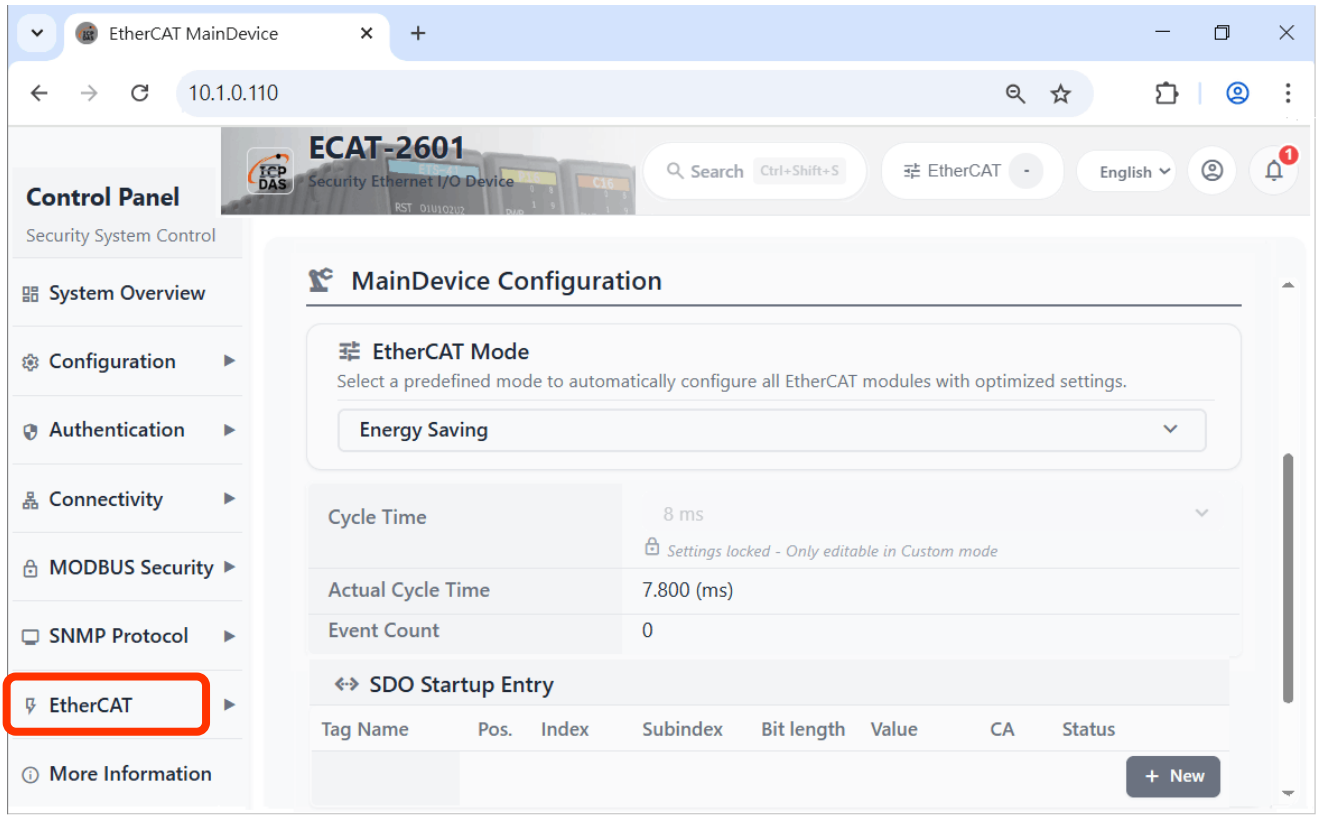
2. To enhance the security, you are prompted to change the password when you login to the module for the first time.

3. Go to the “**Authentication**” setting, click the “**Account Management**”, user can modify the default password: **Admin** in the ***Current password*** field. Next, input your new password in the ***Password*** and ***Confirm password*** fields, then click the **“Apply”** button.

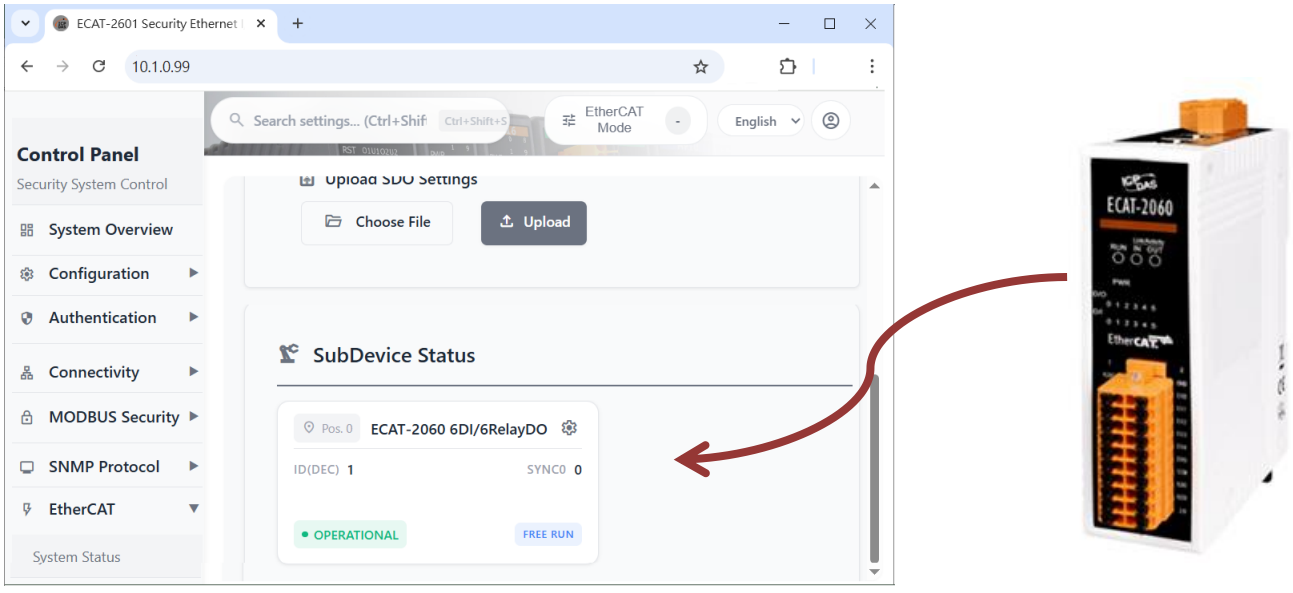


3.3 Configuring the EtherCAT

Click the **“EtherCAT”** tab to display the EtherCAT Settings page, an then Click the **“System Status”** tab to display the EtherCAT I/O Settings page.

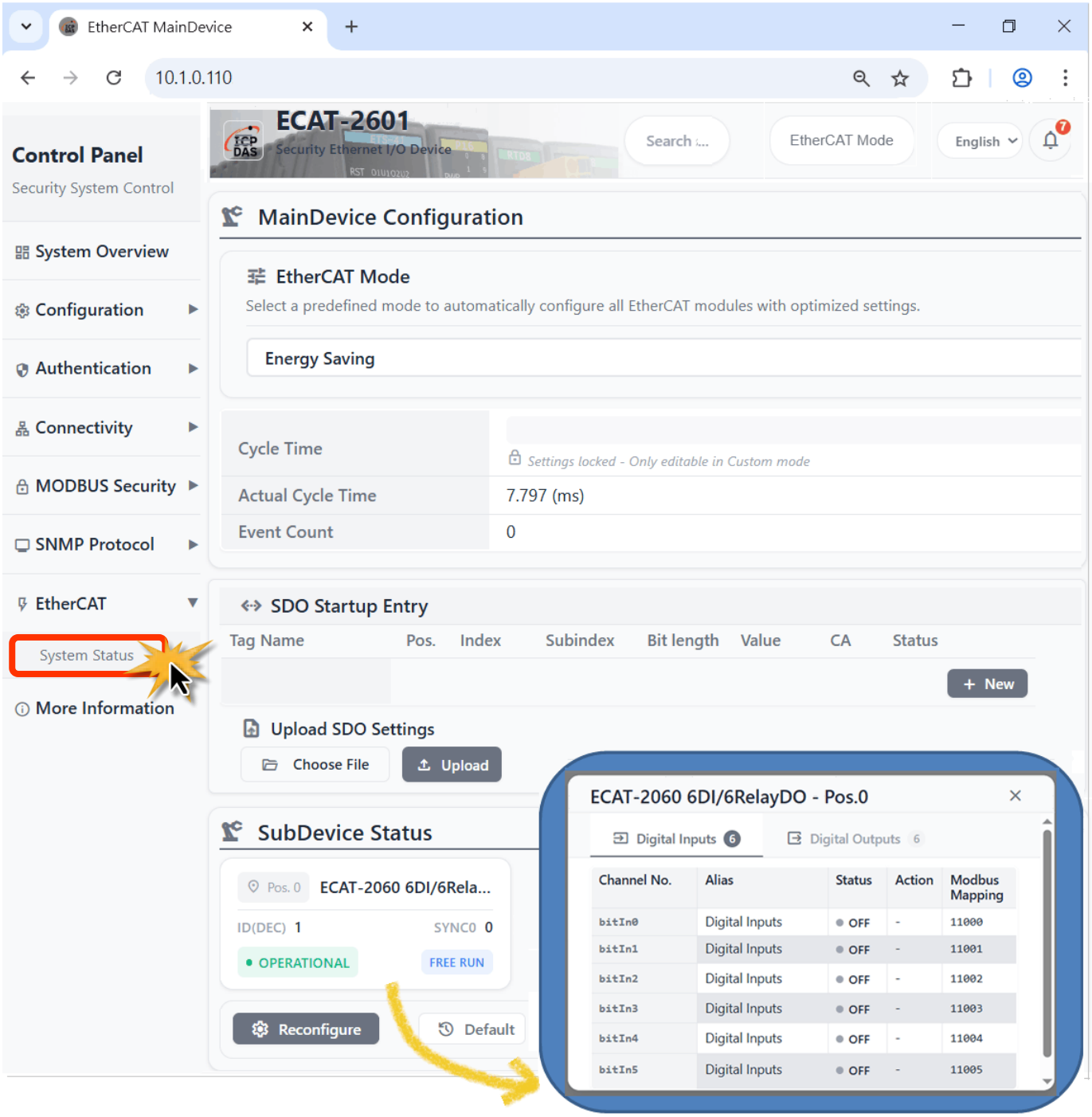


If the EtherCAT port (LAN3) connect with the ECAT-2060 device, it will display as below:



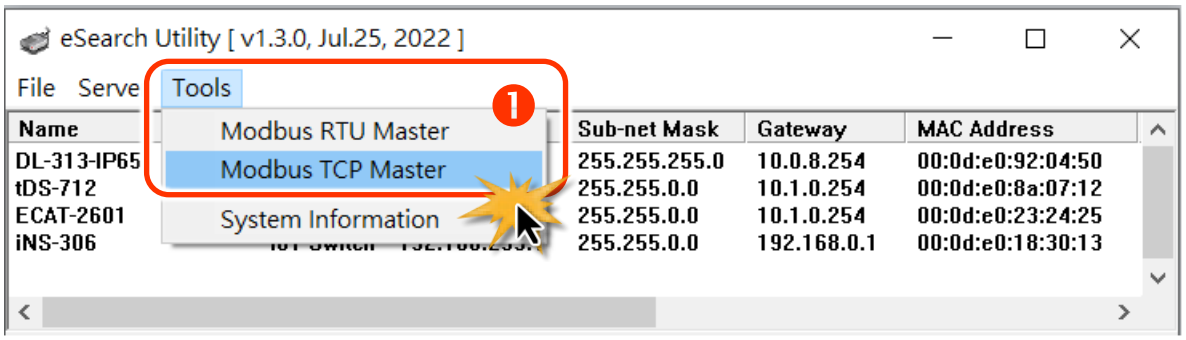
Change the '**EtherCAT Mode**' to the Custom mode, and select the appropriate Cycle Time from the relevant drop down options.

In the '**SubDevice Status**' section, click on the device name (ECAT-2060) to display the configuration pages for I/O settings.



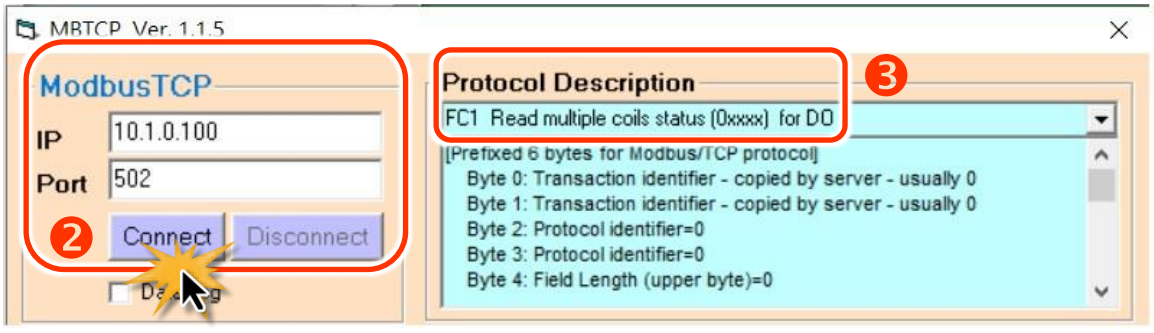
3.4 Self-Test

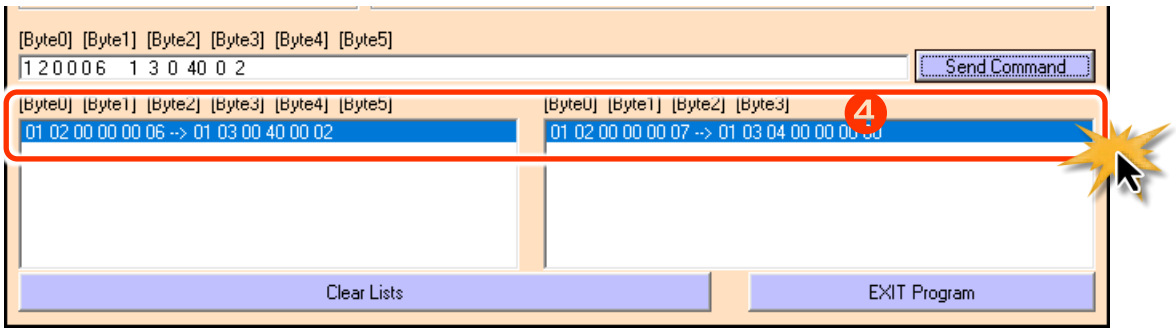
1. In the eSearch Utility, select the **"Modbus TCP Master"** item from the **"Tools"** menu to open the Modbus TCP Master Utility.



2. In the Modbus TCP Master Utility, enter the **IP address of ECAT-2601** in the **"Modbus TCP"** section, and then click the "**Connect**" button to connect to the ECAT-2601.
3. Refer to **"Protocol Description"** section and type the Modbus command in the **"Command"** field then click the **"Send command"** button.
4. If the response data is correct, it means the test is success.

Note: The Modbus command settings depends on your Modbus device.





4. Getting Started for ECAT-2601 on IPv6

This chapter provides detailed information about the “Self-Test” process, which is used to confirm that the ECAT-2601 series module is operating correctly. Before beginning the “Self-Test” process, the wiring test, Ethernet configuration and search/Modbus utility driver installation procedures must first be fully completed. Follow the procedure described below:

Note: This chapter is based on IPv6 environment.

4.1 Configuring Network Settings

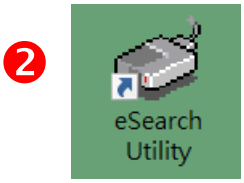
1. Download the **eSearch Utility** and install it according to the installation instructions.

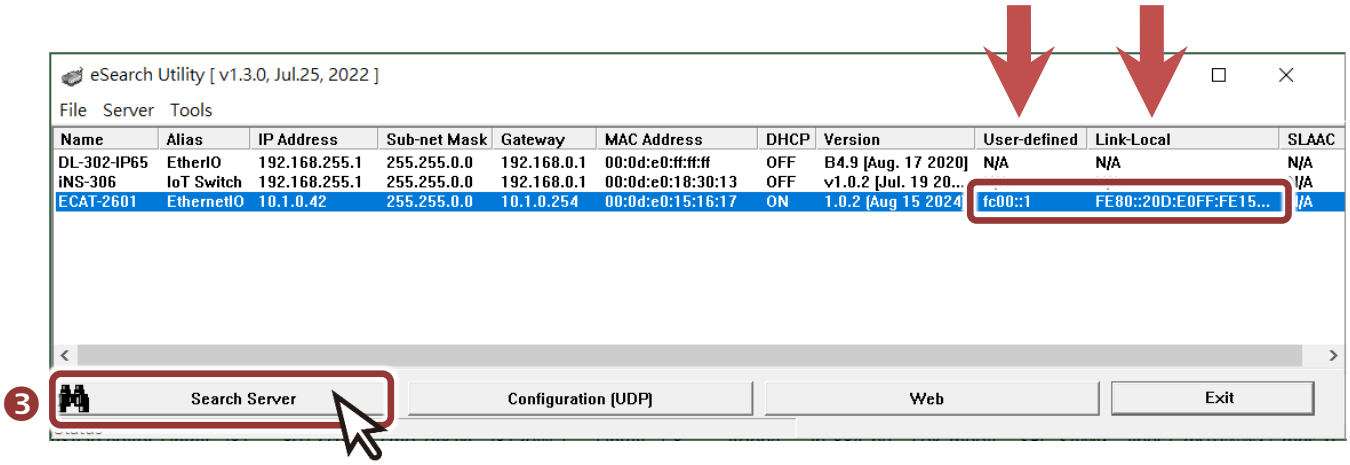
The eSearch Utility can be obtained from the link below:

https://www.icpdas.com/en/product/guide+Software+Utility_Driver+eSearch__Utility

2. Double click the **eSearch Utility** shortcut on the desktop.

3. Click the **“Search Servers”** button to search your ECAT-2601.





4. Every IPv6 device has the Link-Local address. You can view the Link-Local address of the ECAT-2601 module in the "**Link-Local**" field without configuring. If your environment supports Stateless Address Auto-configuration (SLAAC), the SLAAC field will display the SLAAC address when the SLAAC configuration is completed. You can click the **"Search Servers"** button again to update the state.

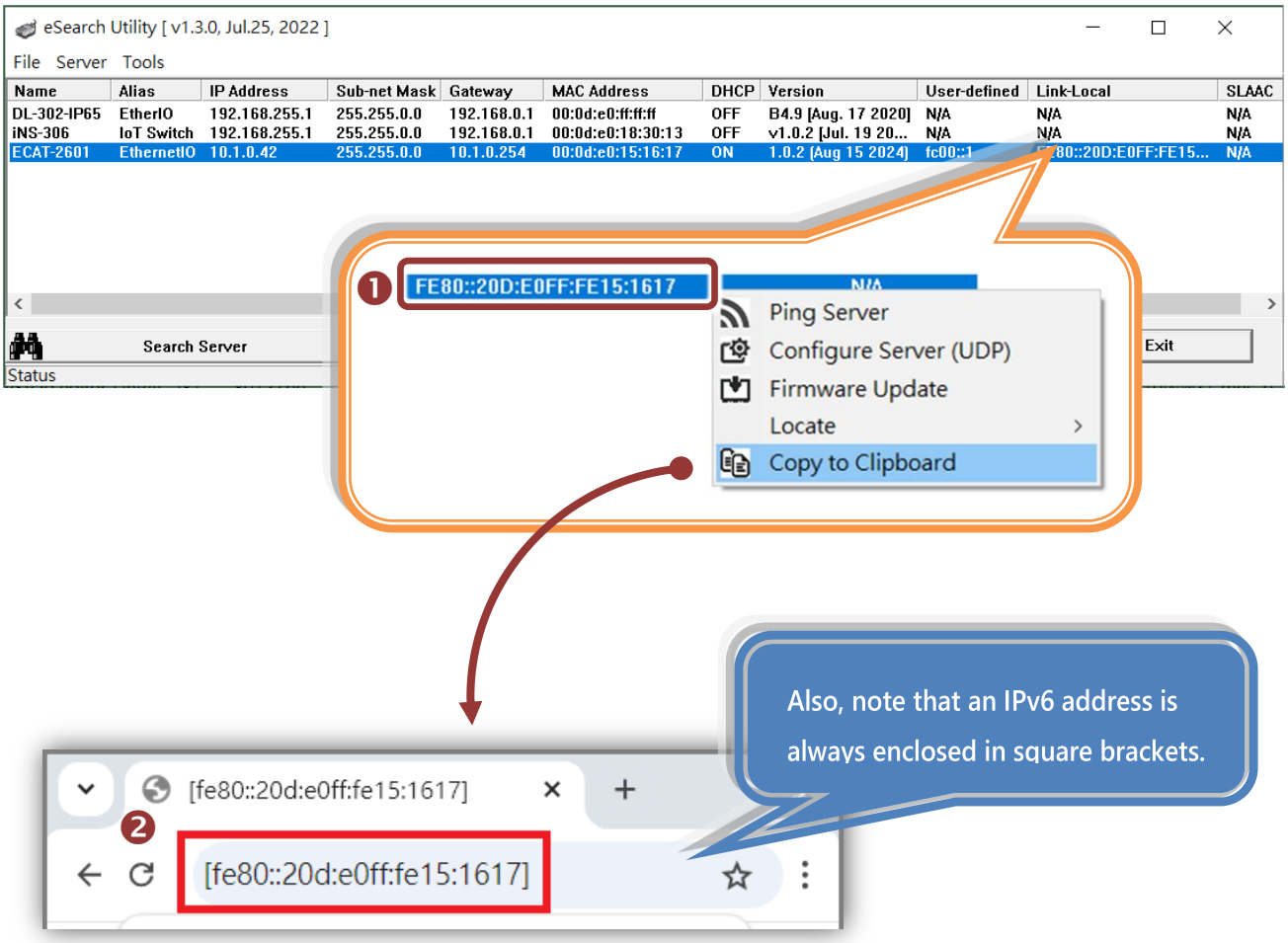
Factory Default Settings of ECAT-2601 Series Module:

IPv4 settings		**Writable**
IP Address	192.168.255.1	✓
Subnet Mask	255.255.0.0	✓
Gateway	192.168.0.1	✓
IPv6 settings		**Writable**
User-defined	fc00::1	✓
Link-Local	EUI-64 format	✗
SLAAC	Auto-Configure	✗

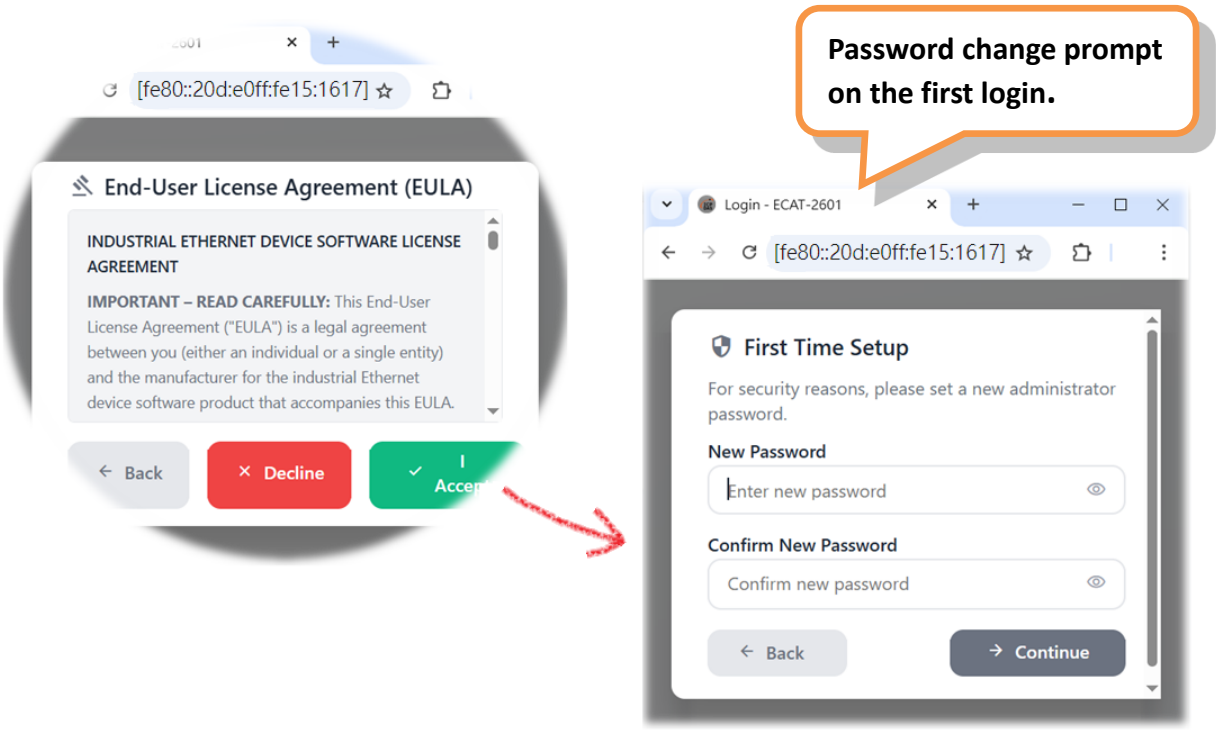
4.2 Configuring the Password

1. Right Click on the Link-Local field and click the **"Copy to Clipboard"** to copy the "**Link-Local address"** of the ECAT-2601 module.

2. Paste the "**Link-Local address"** of the ECAT-2601 module in the address bar of the browser and add the brackets, i.e., [**Link-Local address**].

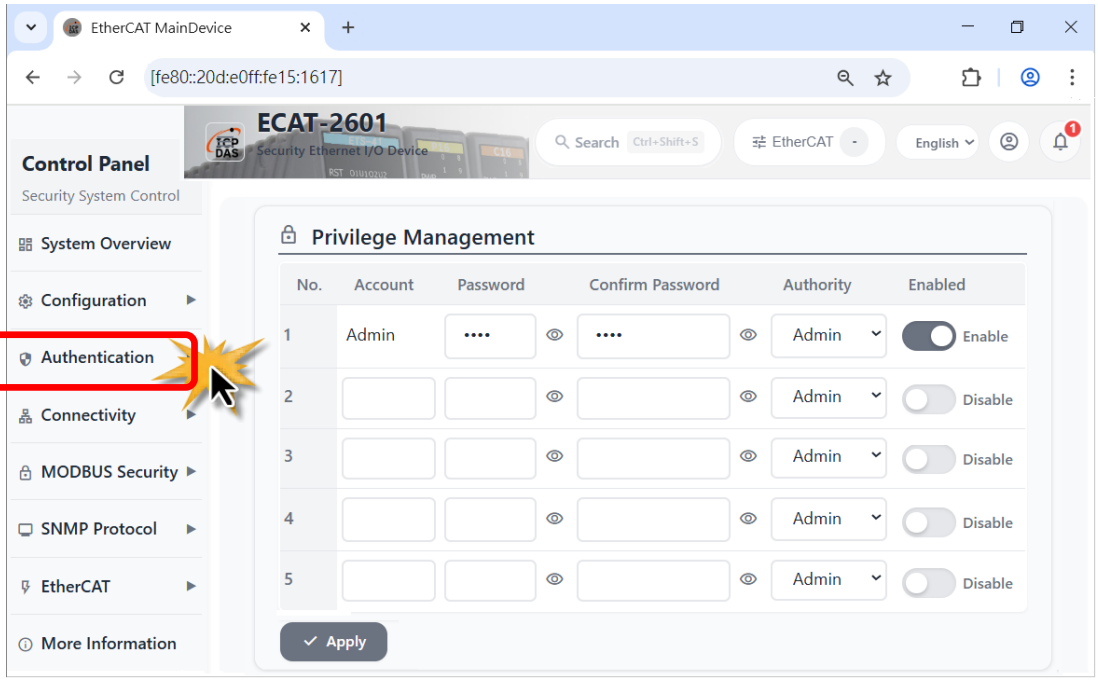


Note: The Web button use the IPv4 address to access the Web Server, not IPv6 address.



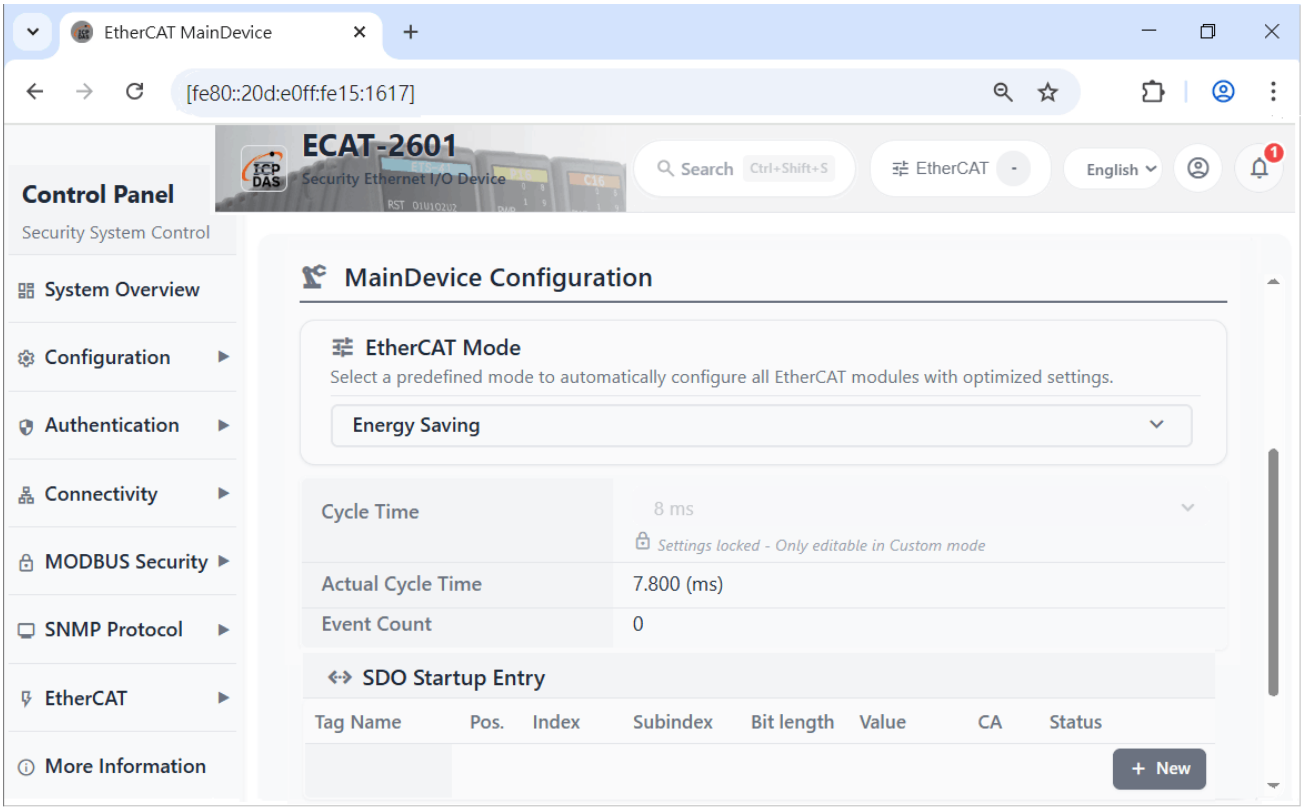
3. To enhance the security, you are prompted to change the password when you login to the module for the first time.

4. Go to the "**Authentication**" setting, click the "**Account Management**", user can modify the default password: **Admin** in the ***Current password*** field. Next, input your new password in the ***Password*** and ***Confirm password*** fields, then click the **"Submit"** button.

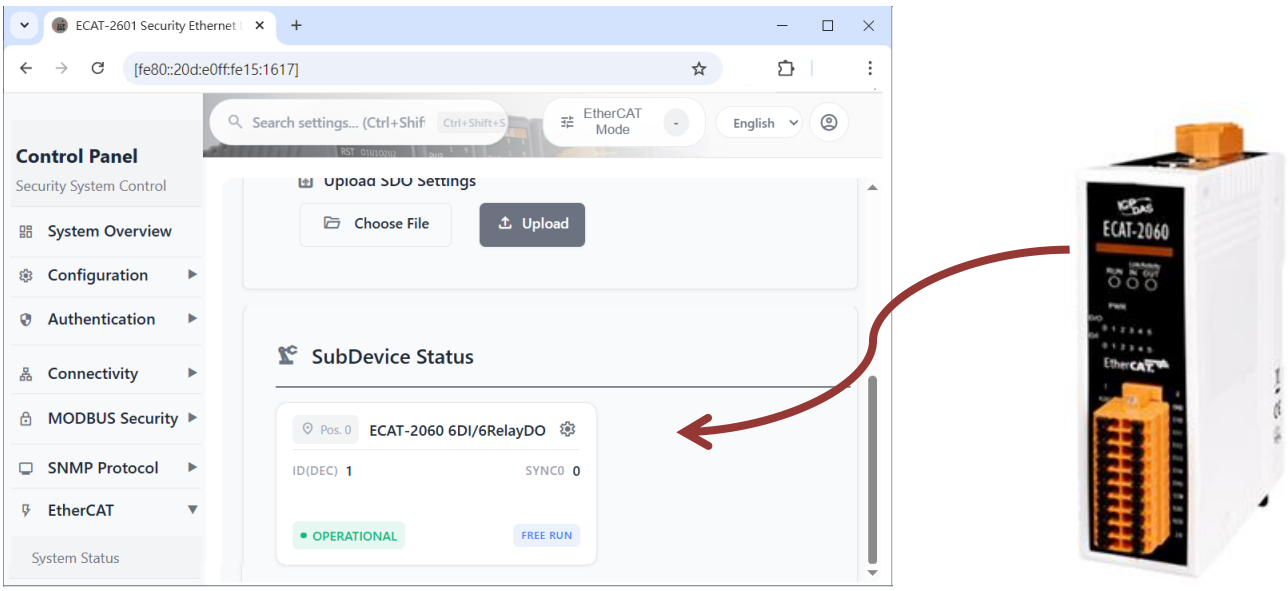


4.3 Configuring the EtherCAT

Click the **"EtherCAT"** tab to display the EtherCAT Settings page.

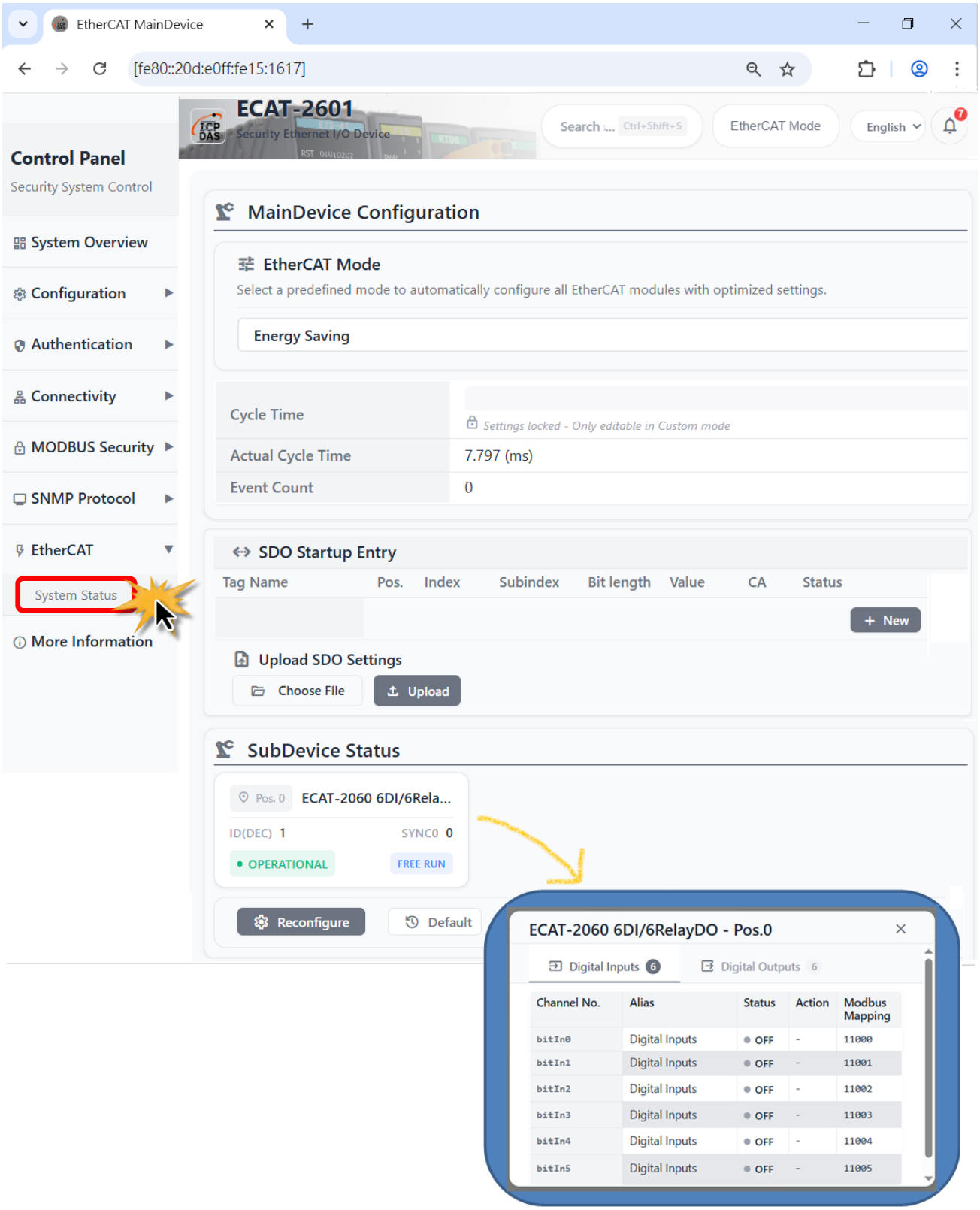


If the EtherCAT port (LAN3) connect with the ECAT-2060 device, it will display as below:



Change the '**EtherCAT Mode**' to the Custom mode, and select the appropriate Cycle Time from the relevant drop down options.

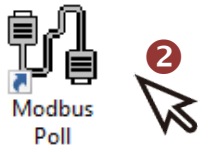
In the '**SubDevice Status**' section, click on the device name (ECAT-2060) to display the configuration pages for I/O settings.



4.4 Self-Test

1. Download and install the “Modbus Poll” test program at below link.
   https://www.modbustools.com/download.html

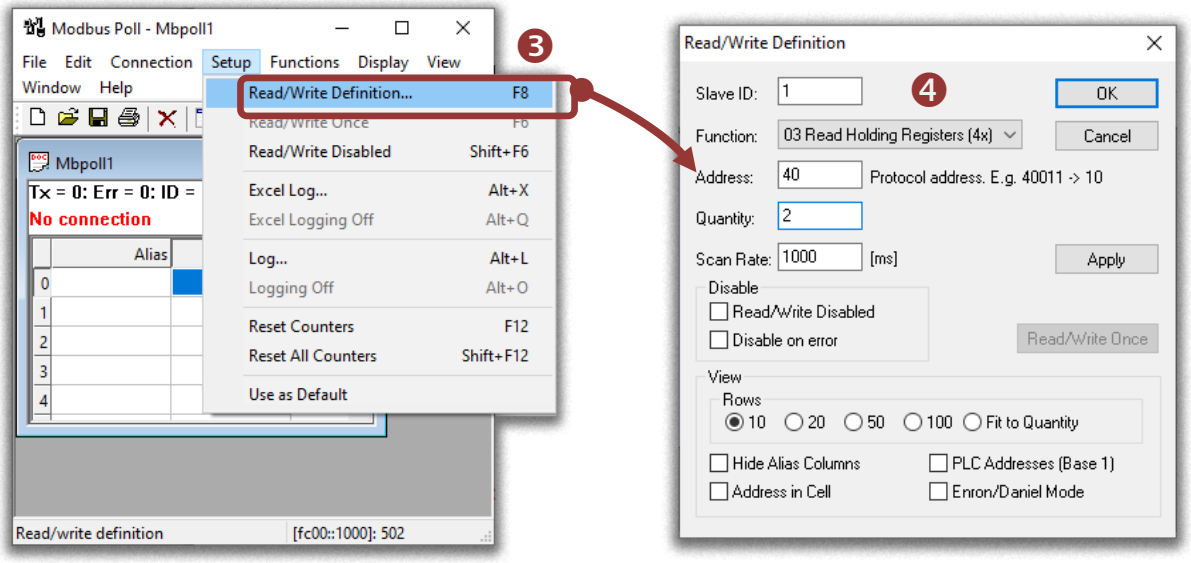
2. Double-click the Modbus Poll shortcut to open.



3. Select the “**Read/Write Definition…**” item from the “Setup” menu to open the “Read/Write Definition” dialog box.

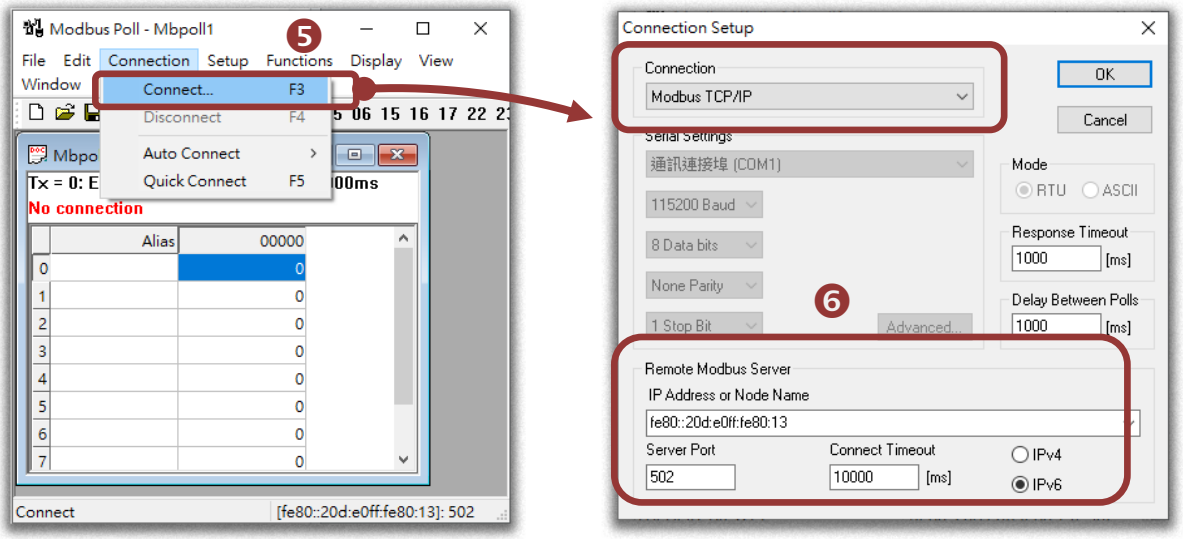
4. Configure the settings for the Slave.

Note: The Modbus Slave settings depends on your Modbus device.

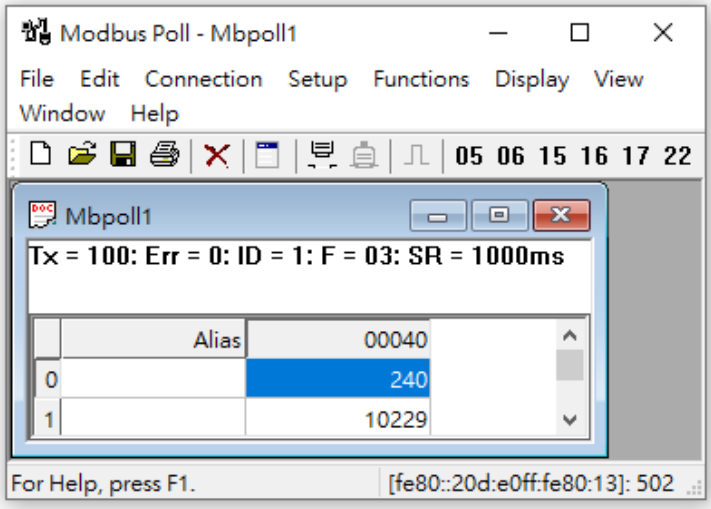


5. Select the “Connect…” item from the “Connection” menu to open the “Connection Setup” dialog box.

6. Configure the IPv6 address and TCP port (default: 502) of ECAT-2601 and click “**OK**” to connect the ECAT-2601 for testing.



7. If the response data is correct, it means the test is success.



5. Web Configuration

Once the ECAT-2601 series module has been correctly configured and is functioning normally on the network, the configuration details can be retrieved or modified using either the eSearch Utility described above, or via a standard web browser.

5.1 Logging in to the ECAT-2601 Web Server

The embedded ECAT-2601 series web server can be accessed from any computer that has an Internet connection.

Step 1: Open a new browser window

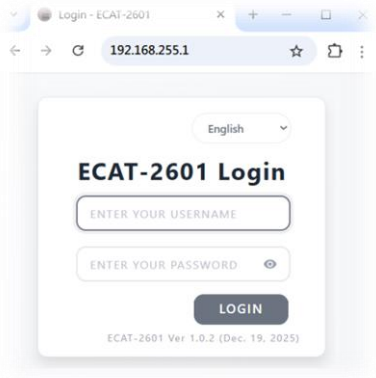
Open a web browser, for example, Google Chrome or Firefox, which are reliable and popular Internet browsers that can be used to configure ECAT-2601 series module.

Step 2: Enter the URL for the ECAT-2601 web server

Ensure that you have correctly configured the network settings for the ECAT-2601 series module, and then enter the URL for the ECAT-2601 web server in the address bar of the browser.

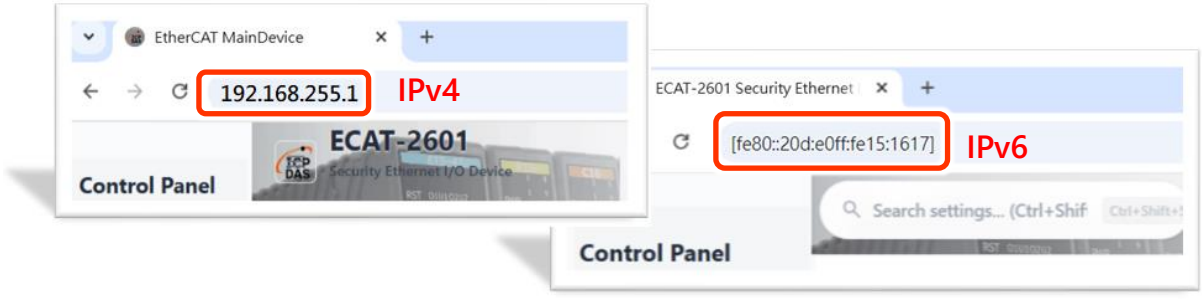
Step 3: Enter the Password

After connecting to the IP address, the login page is displayed. Enter your username and password, and then click the **"LOGIN"** button to continue. (If this is the first time you login, refer to Section 3.4 to change the factory default password.)



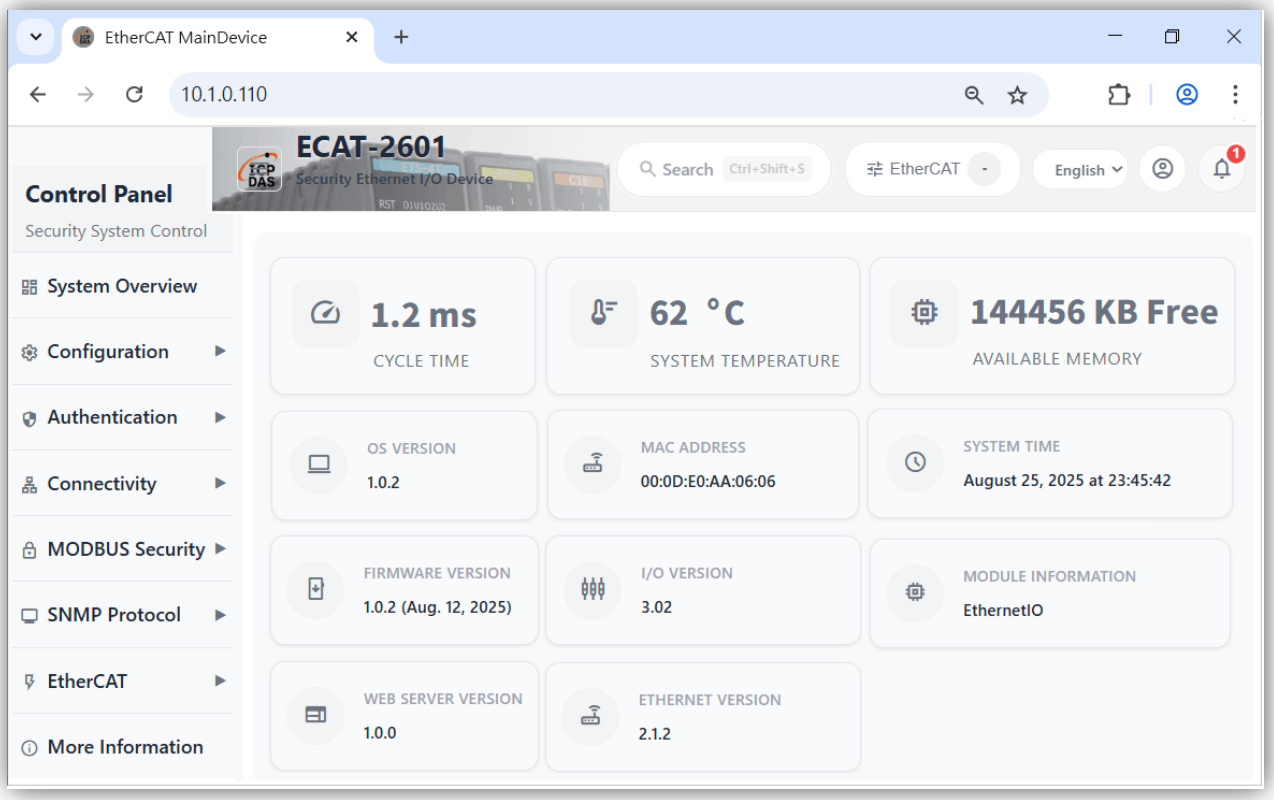
Step 4: Log in to the ECAT-2601 Web Server

After logging into the ECAT-2601 web server, the main page will be displayed.



5.2 Home Page

The **Home** link connects to the **System Overview** page, which contains two parts.



The left part of the page illustrates the primary function, it's a control panel.

The right section provides basic information about the ECAT-2601 hardware and software. The software and hardware information section includes information related to the system temperature, the current Firmware version, the IP Address, the available memory, the Alias, the MAC Address, and the System cycle time. If the firmware for the ECAT-2601 module is updated, you can check the version information here.

5.3 Configuration Page

After clicking the '**Configuration**' item will display the EtherCAT configuration page on the right. User can configure the network settings, basic settings, advanced settings, system maintenance and the network status for the ECAT-2601 module, each of which will be described in more detail below.

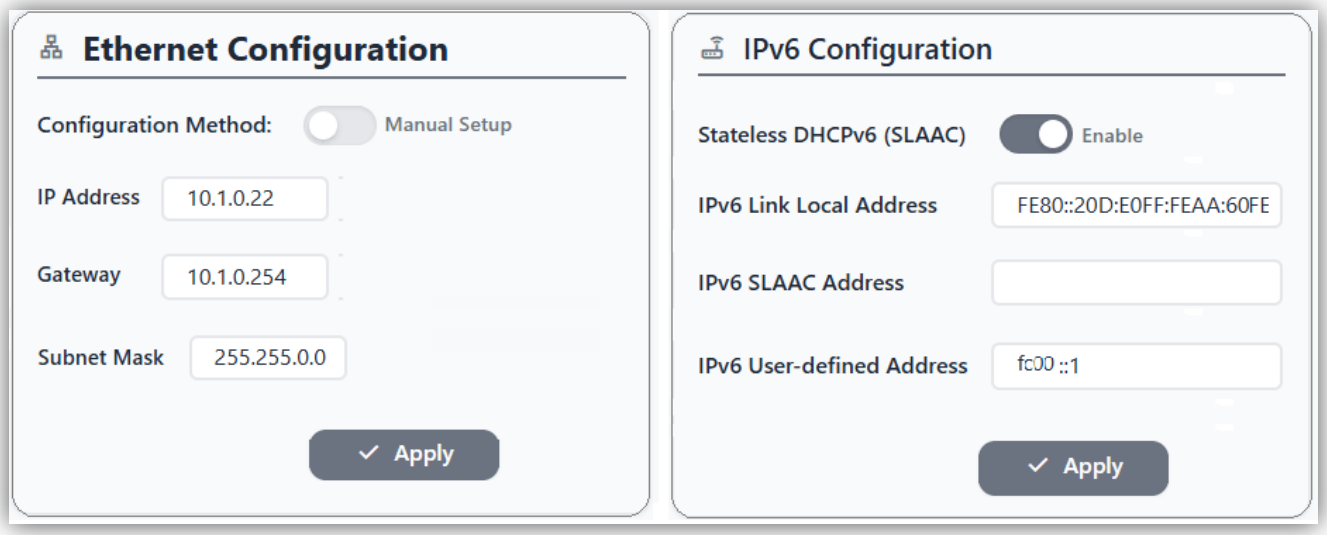
5.3.1 Network Settings

The network settings had three main parts: Ethernet, IPv6 and WireGuard VPN configuration.

The Address Type, Static IPv4 Address, Subnet Mask and Default Gateway values are the most important network settings and should always correspond to the LAN configuration. If they do not match, the ECAT-2601 module will not operate correctly in IPv4.

The IPv6 Link Local Address always effective in the same link layer. The IPv6 SLAAC Address will be configured by the Router automatically, and the default router is the link-local address of the router.

The IPv6 User-defined Address can be defined by the User. It's more convenient for using in the simple application. The IPv6 User-defined Address can be routed if its prefix as same as IPv6 SLAAC Address. If the settings are changed while the module is operating, any connection currently in use will be lost and an error will occur. If your environment doesn't support the IPv6, please ignore the IPv6 Address fields.



The following is an overview of the parameters contained in the **IP Address Selection** section:

Item	Description
Address Type	**Static IP:** If no DHCP server is installed on the network, the network settings can be configured manually. Refer to Section "Manual Configuration" for more details.
	DHCP: The Dynamic Host Configuration Protocol (DHCP) is a network application protocol that automatically assigns an IPv4 address to each device. Refer to Section "Dynamic Configuration" for more details. **Note: The DHCP function doesn't effective on IPv6.**
Static IPv4 Address	Each ECAT-2601 connected to the network must have its own unique IP address. This parameter is used to assign a specific IPv4 address.
Subnet Mask	This parameter is used to assign the subnet mask for the ECAT-2601. The subnet mask indicates which portion of the IPv4 address is used to identify the local network or subnet.
Default Gateway	This parameter is used to assign the IPv4 Address of the Gateway to be used by the ECAT-2601. A Gateway (or router) is a device that is used to connect an individual network to one or more additional networks.
MAC Address	This parameter is used to set a user-defined MAC address, which must be in the format FF-FF-FF-FF-FF-FF.
IPv6 Link Local Address	Each IPv6 device connected to the network must have a link-local address. It always effective in the same link layer. It is auto-configured by EUI-64. If your environment doesn't support the IPv6, please ignore this field.
IPv6 SLAAC Address	The ECAT-2601 supports stateless address auto configuration (SLAAC). It will be configured by the Router automatically, and the default router is the link-local address of the router. The router should provide the prefix and the ECAT-2601 will calculate the Host ID by EUI-64. If your environment doesn't support the SLAAC, please ignore this field.
SLAAC Timeout (SLAAC Watchdog)	This parameter is used to configure the SLAAC timeout value. If the SLAAC address doesn't be configured in the specified time, the system will be rebooted to configure the SLAAC address again. Timeout value range: 30 to 65535 (seconds); Disable = 0;
IPv6 User-defined Address	It can be defined by the User. It's more convenient than using the link-local address in the simple application. This address can be routed if the prefix as same as IPv6 SLAAC Address. If your environment doesn't support the IPv6, please ignore this field.
Apply	Click this button to save the revised settings to the ECAT-2601.

Note: The IPv6 Link Local Address is auto-configured by EUI-64.

Note: The IPv6 SLAAC Address is auto-configured by Router. The router should provide the prefix and the ECAT-2601 will calculate the Host ID by EUI-64.

Select the '**Configuration**' page from the dropdown menu on the left. You will then see the '**Ethernet Configuration**' section for setting IPv4 and IPv6.

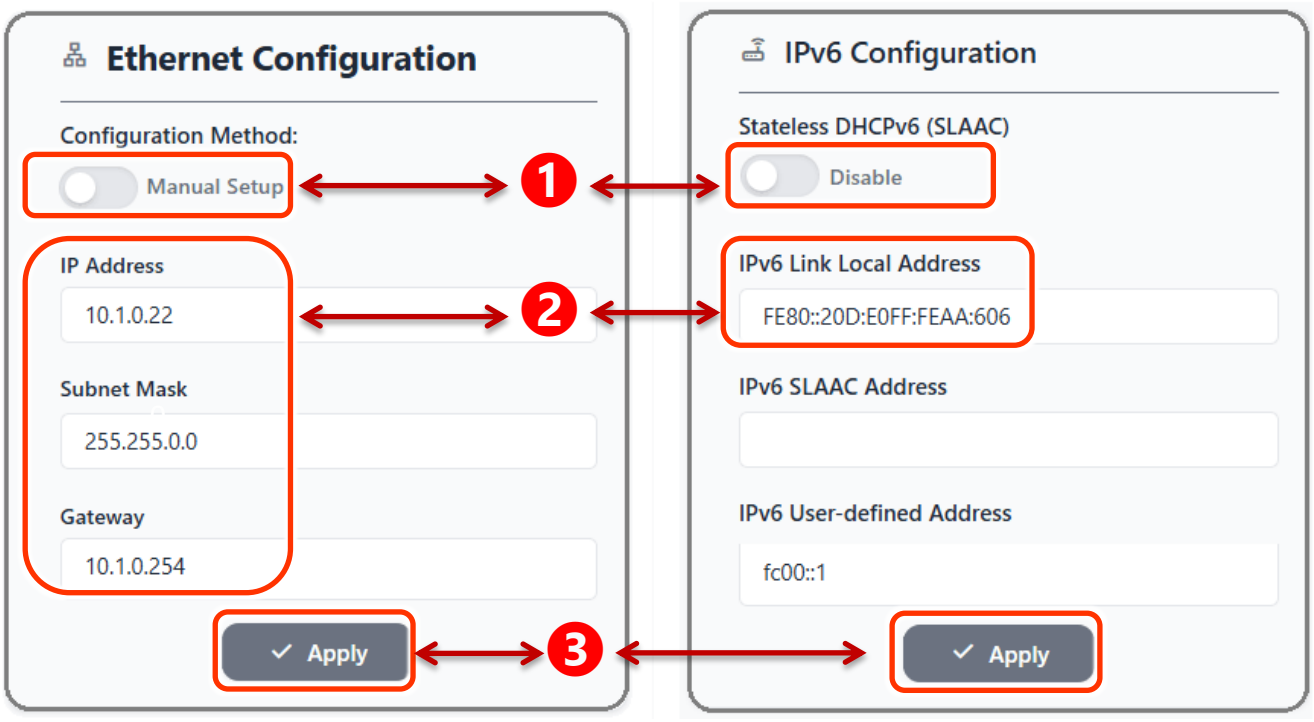
Manual Configuration

When using manual configuration, the network settings should be assigned in the following manner:

Step 1: Select the **"Manual Setup"** option from the **"Configuration Method"** toggle switch.

Step 2: Enter the relevant details in the respective **network settings** fields**.** If your environment doesn't support the IPv6, please ignore the **"IPv6 User-defined Address"** fields.

Step 3: Click the **"Apply"** button to complete the configuration.

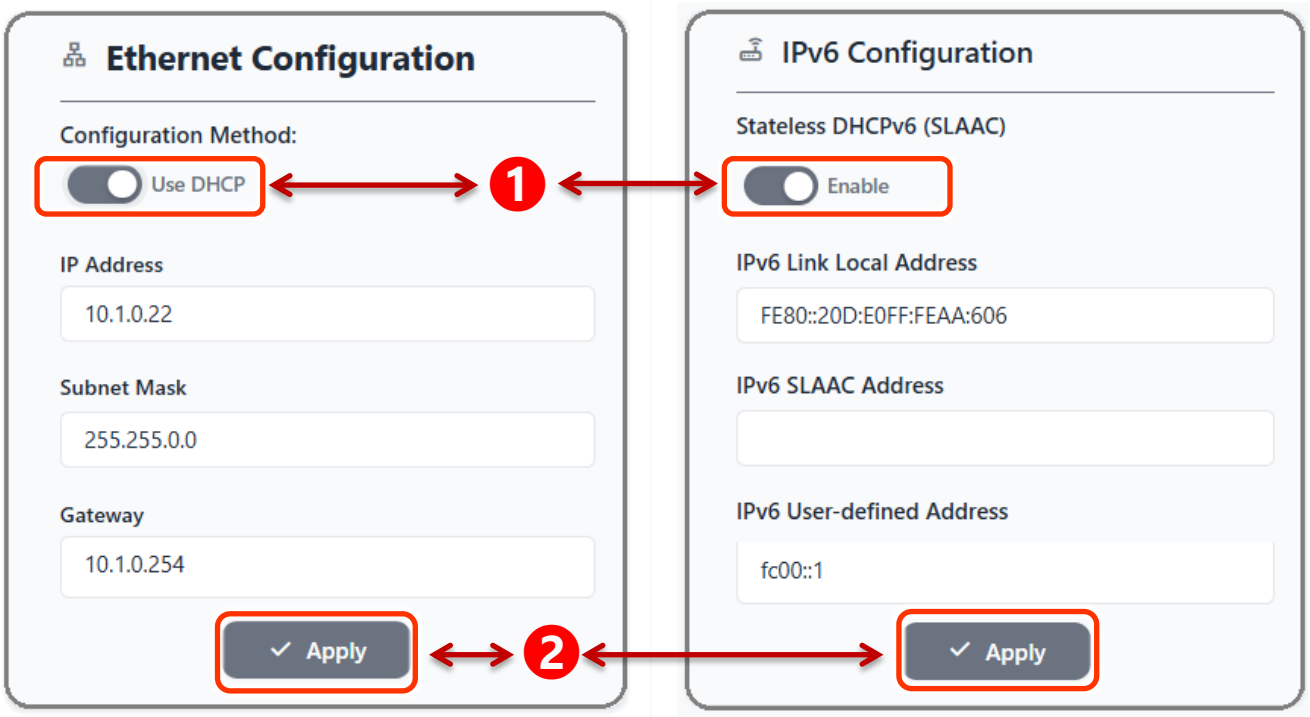


Dynamic Configuration

Dynamic configuration is very easy to perform. If a DHCP server is connected to you network, an IPv4 network address can be dynamically configured by using the following procedure:

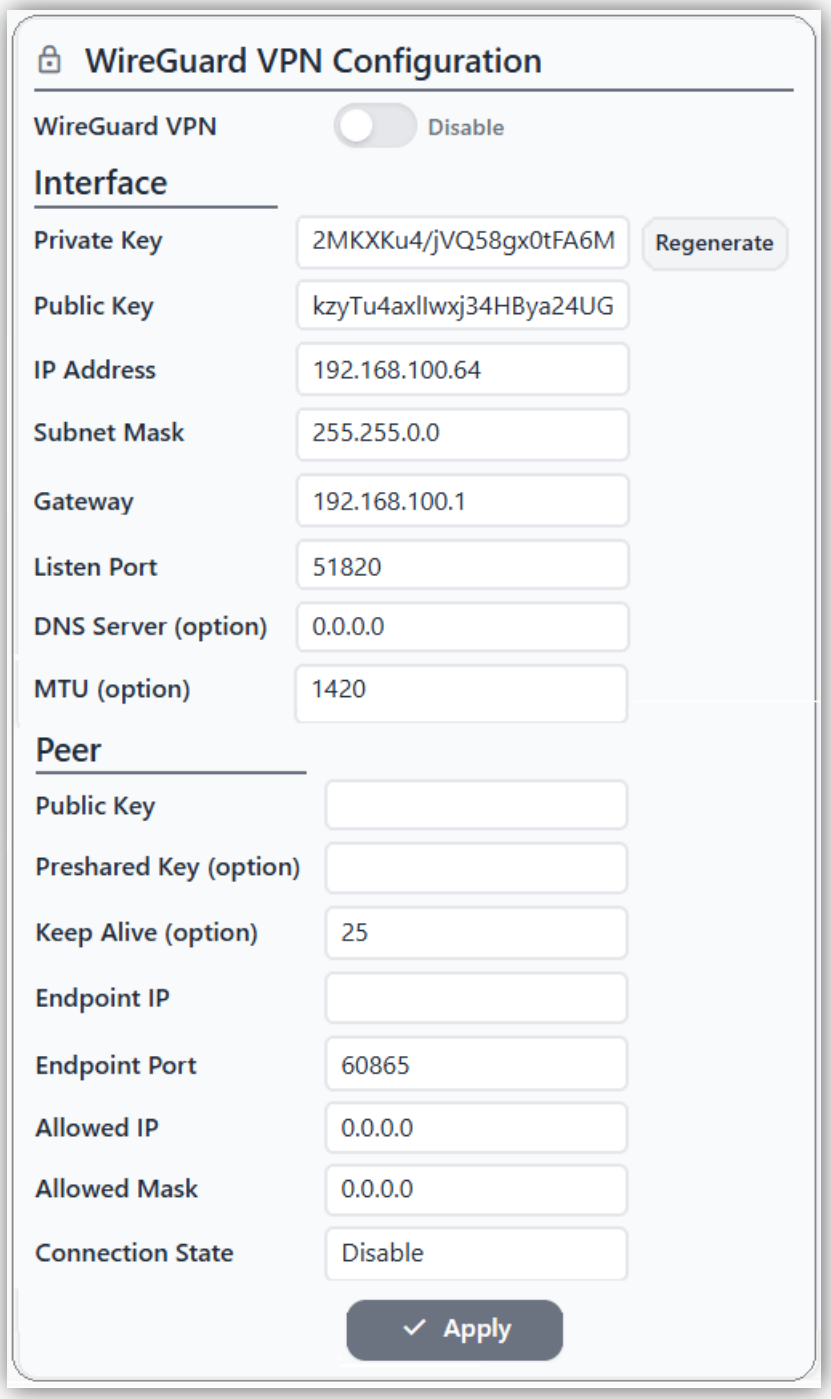
Step 1: Enable this function. Select the **"DHCP"** option from the **"Configuration Method"** toggle switch.

Step 2: Click the **"Apply"** button to complete the configuration.



WireGuard VPN Configuration

Select the '**Configuration**' page from the dropdown menu on the left. You will then see the '**WireGuard VPN Configuration**' section.



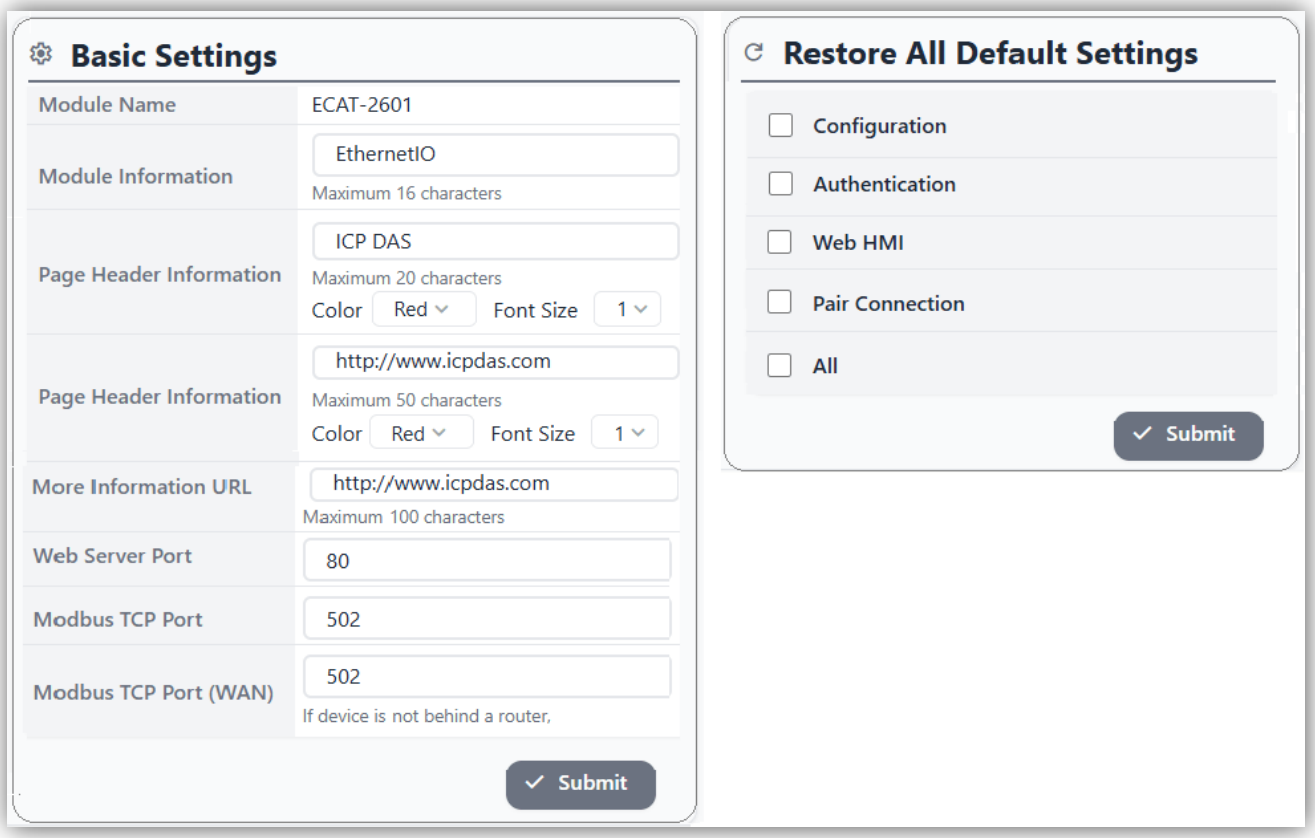
The following is an overview of the parameters contained in the **WireGuard VPN Configuration** section:

Item	Description
WireGuard VPN Type	Enable/Disable: Check the toggle switch to activate WireGuard or not.
Private Key	System will auto-generate the Private Key.
Public Key	Interface: System will auto-generate the Public Key.
	Peer: Enter the public key generated in the WireGuard application in the VPN client device.
IP Address	Enter a fixed IP subnet for the VPN server.
Subnet Mask	Specify the Remote subnet IP address.
Gateway	This parameter is used to assign the IPv4 Address of the Gateway to be used by the ECAT-2601. A Gateway (or router) is a device that is used to connect an individual network to one or more additional networks.
Listen Port	The listening port is used to listen to WireGuard packets and the data forwarding port as the corresponding WireGuard interface. The default number is 51820. (Range: 1~65535)
DNS Server (option)	Each IPv6 device connected to the network must have a link-local address. It always effective in the same link layer. It is auto-configured by EUI-64. If your environment doesn't support the IPv6, please ignore this field.
MTU (option)	Max Transimission Unit (MTU) of this WireGuard interface. Can be blank, or set according to the actual requirement. Specify the MTU 1420
Pre-shared Key (option)	Check the Enable box if a Pre-Shared Key is required.
Keep Alive (option)	Configured to send keepalive packets periodically to maintain the connection with the Peer. Default is 25.
Endpoint IP	The actual IP address or domain of peer.
Endpoint Port	The listening port of peer.
Allowed IP	Determines which traffic enters the tunnel and limits which source IPs are accepted from the peer.
Allowed Mask	The mask length (/24 or /32) sets the default AllowedIPs, and can be left to its default value.
Connection State	Disable or Enable
Apply	Click this button to save the revised settings to the ECAT-2601.

5.3.2 Basic Settings

The **Basic settings** had two main parts: Basic settings and Restore all default settings.

Select the '**Configuration**' page from the dropdown menu on the left. You will then see the 'Basic Settings' section.



The following is an overview of the parameters contained in the **Basic Settings** section:

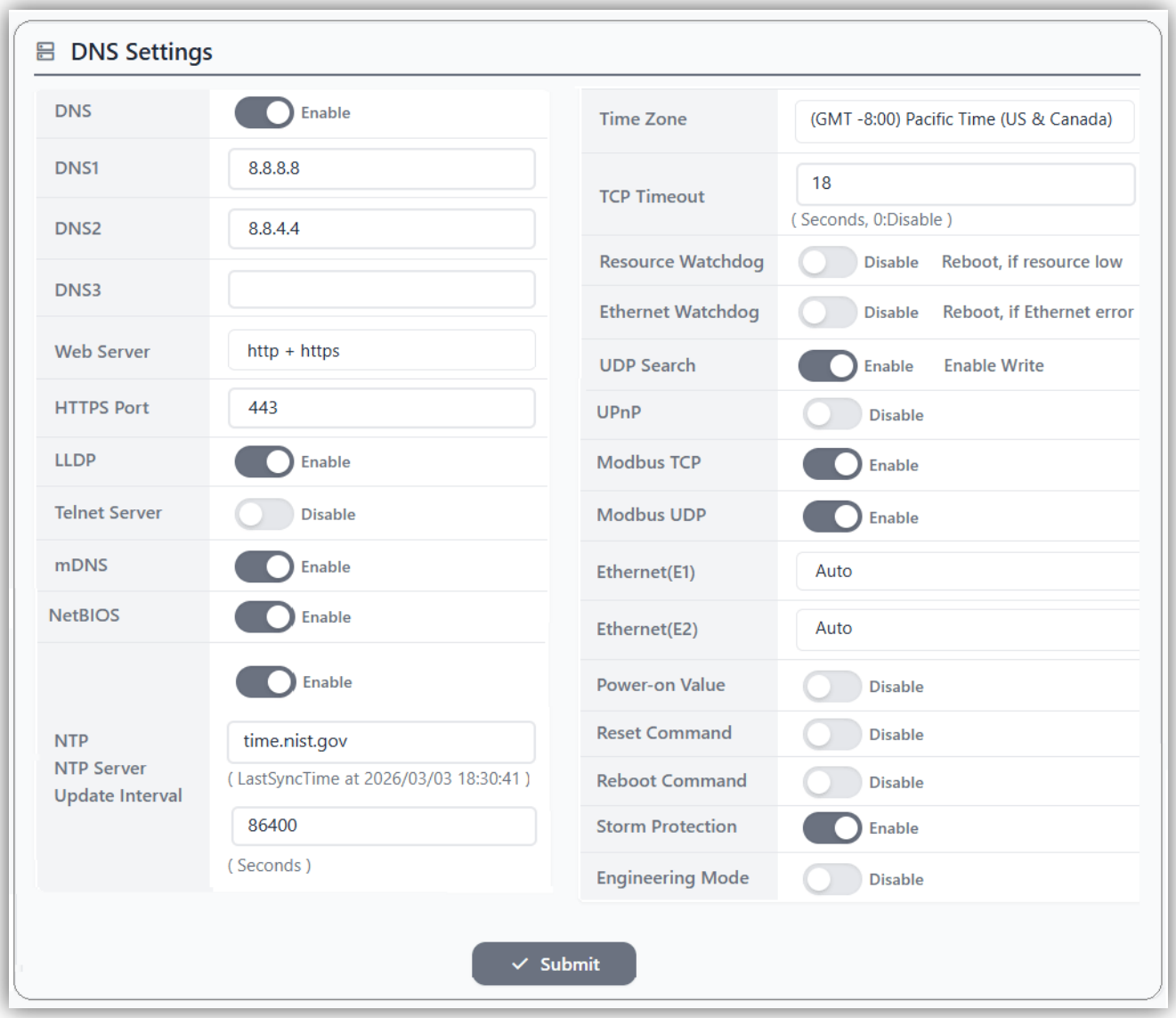
Item	Description
Module information	User can enter the information for the ECAT-2601.
Page Header information	User can enter the URL for visit to page header webpage, and setup color and font size.
More information URL	User can enter the URL address for visit to more information. Maximum 100 characters.
Web Server Port	The communication port number of the Web Server Port.
Modbus TCP Port	The communication port number of the Modbus TCP Port.
Modbus TCP Port (WAN)	The communication port number of the Modbus TCP Port for WAN. If device is not behind a router, you can ignore this setting.
Submit	Click this button to save the revised settings to the ECAT-2601.

5.3.3 Advanced Settings

The **Advanced settings** had six main parts: DNS settings, Module RTC, TSN settings, 802.1X settings and HTTPS settings.

5.3.3.1 DNS settings

Select the '**Configuration**' page from the dropdown menu on the left. You will then see the 'DNS Settings' section.

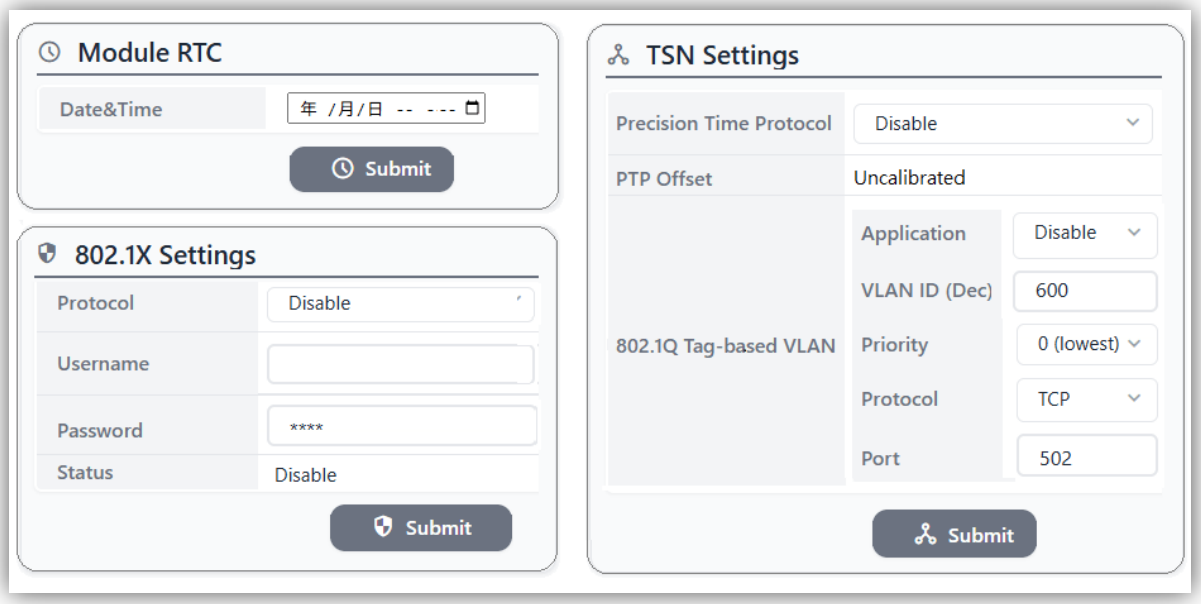


The following is an overview of the parameters contained in the **DNS Settings** section:

Item	Description
DNS Type	Enable/Disable: Check the toggle switch to activate DNS Type or not.
DNS1 (Primary DNS)	Enter DNS server addresses. The first resolver queried by the client. It holds the master zone files or acts as the main resolver to minimize latency.
DNS2 (Secondary DNS)	Used if DNS1 is unavailable, providing redundancy
DNS3 (Tertiary DNS)	A third backup resolver.
Web Server	Select Web Server Type.
HTTPS Port	Enter a port number for enable HTTPS (secured web interface) by setting the web server port to 443.
LLDP	Enable/Disable: Check the toggle switch to activate **LLDP** or not.
Telent Server	Enable/Disable: Check the toggle switch to activate **Telent Server** or not.
mDNS	Enable/Disable: Check the toggle switch to activate **mDNS** or not.
NetBIOS	Enable/Disable: Check the toggle switch to activate **NetBIOS** or not.
NTP	Enable/Disable: Check the toggle switch to activate **NTP** or not.
NTP Server	Configure an NTP server address (e.g., time.nist.gov) in the Time Synchronization Section to ensure accurate time stamping.
Update Interval	Defines the time interval (seconds) that the NTP client requests for a time update.
Time Zone	Select your local time zone. This setting affects all time-related functions and logging timestamps in the system.
TCP Timeout	Enter a timeout value. (0: Disable)
Resource Watchdog	Enable/Disable: Check the toggle switch to activate **Resource Watchdog** or not.
Ethernet Watchdog	Enable/Disable: Check the toggle switch to activate **Ethernet Watchdog** or not.
UDP Search	Enable/Disable: Check the toggle switch to activate **UDP Search** or not.
UPnP	Enable/Disable: Check the toggle switch to activate **UPnP** or not.
Modbus TCP	Enable/Disable: Check the toggle switch to activate **Modbus TCP** or not.
Modbus UDP	Enable/Disable: Check the toggle switch to activate **Modbus UDP** or not.
Ethernet (E1)	Select Ethernet (E1) Link Speed.
Ethernet (E2)	Select Ethernet (E2) Link Speed.
Power-on Value	Enable/Disable: Check the toggle switch to activate **Power-on Value** or not.
Reset Command	Enable/Disable: Check the toggle switch to activate **Reset Command** or not.
Reboot Command	Enable/Disable: Check the toggle switch to activate **Reboot Command** or not.
Strom Protection	Enable/Disable: Check the toggle switch to activate **Strom Protection** or not.
Engineering Mode	Enable/Disable: Check the toggle switch to activate **Engineering Mode** or not.
Submit	Click this button to save the revised settings to the ECAT-2601.

5.3.3.2 HTTPS Authentication Certificate

Select the '**Configuration**' page from the dropdown menu on the left. You will then see the ' **HTTPS Authentication Certificate** ' section.



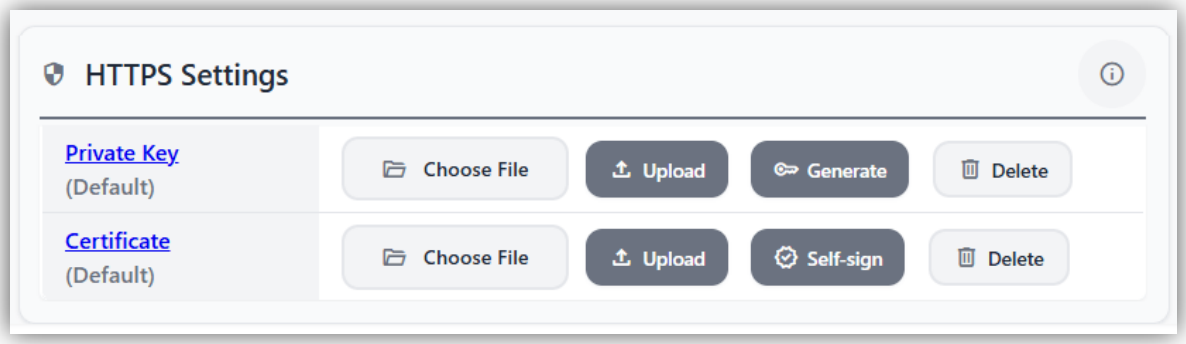
The following is an overview of the parameters contained in the **RTC, TSN** and **802.1X** section:

Item	Description
Date & Time	Select the Date and Time information for the ECAT-2601.
Precision Time Protocol	There are four types for the Precision Time Protocol, which is IEEE 1588 UDP, IEEE 1588 Ethernet, 802.1AS and Disable.
PTP Offset	Uncalibrated, Listening, Initializing or Slave mode.
802.1Q Tag-based VLAN	Application – You can select: Disable, server(Source Port) or Client (Destination Port)
	VLAN ID(Dec) – use a 12-bit field, allowing decimal values ranging from 1 to 4094. IDs 0 and 4095 are reserved.
	Priority – the range of priority is 0~7 (highest)
	Protocol – You can select: TCP , UDP or TCP+UDP
	Port – input the port number
802.1X settings	Protocol – You can select: Disable, EAP MD5, EAP GTC, EAP MSCHAPv2, PEAP MD5, PEAP GTC or PEAP MSCHAPv2
	Username – You can setup the username
	password – You can setup the password
	Status – show the status
Submit	Click this button to save the revised settings to the ECAT-2601.

5.3.3.3 HTTPS Settings

Select the '**Configuration**' page from the dropdown menu on the left. You will then see the ' **HTTPS** ' section.

This is a cryptographic key that identifies your server. Keep this secure and never share it.



The HTTPS function provides secure encrypted communication for your HTTPS connections, which is consists of the following sections:

- Private Key – The private key is a secure entity and should be stored in a file with restricted access.
- Certificate – The certificates are an important component of Transport Layer Security (TLS, sometimes called by its older name SSL), where they prevent an attacker from impersonating a secure website or other server.

NOTE:

- ★ The HTTPS server key/certificate is used for encryption when communicating with browsers.
- ★ The ECAT-2601 is supplied with a pre-installed default certificate that enables direct HTTPS communication.

Private Key

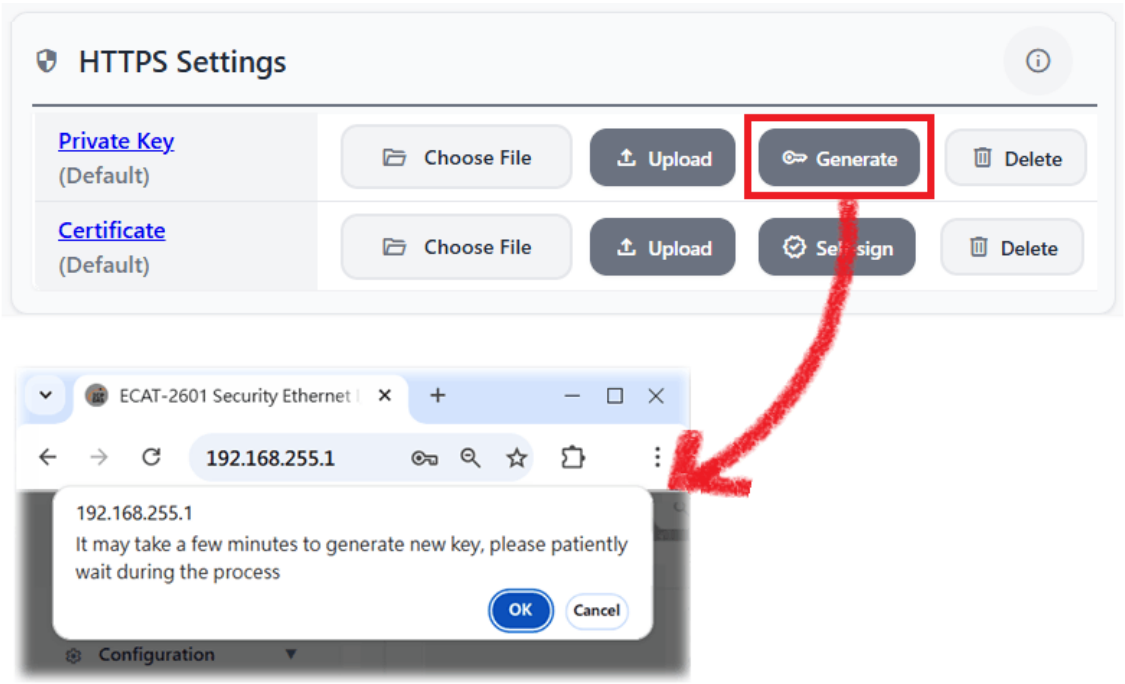
The following is an overview of the functions contained in the **Private Key** section:

Item		Description
1	Choose File	Click “Choose File” and select your Private Key (file names often use extensions like .key, .pem, .crt, .cer or .pfx).
2	Upload	After choose a private key file, you could upload your own private key.
3	Generation	A unique private key is generated on the server. You could generate a new private key.
4	Remove	Remove the current private key.

Users can follow these instructions to generate a private key:

STEP 1

To click the [**GENERATION**] button in the "Private Key" section, the ECAT-2601 will show the message as below.



STEP 2

Once you click the [**OK**] button, the ECAT-2601 will generate a new key.

Certificate

This is a certificate that proves your server's identity to connecting clients.

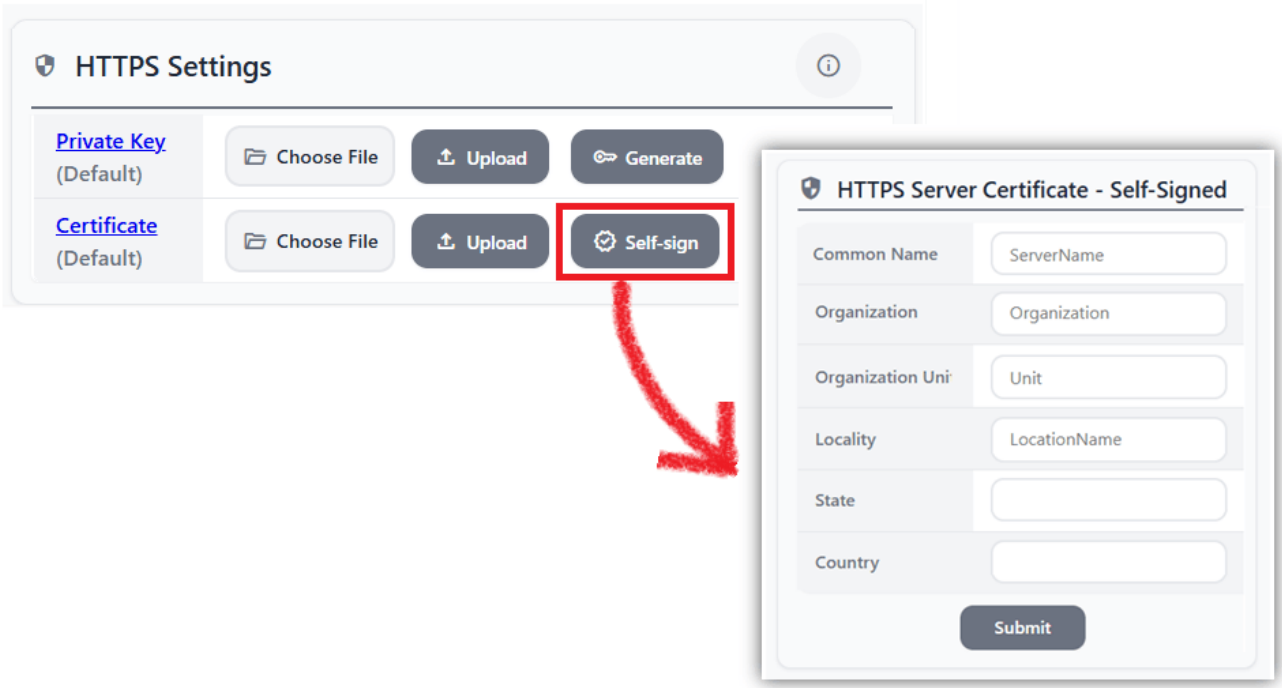
The following is an overview of the functions contained in the **Self-Signed** section:

Item		Description
1	Choose File	Click "Choose File" and select your CA certificate (file names often use extensions like .key, .pem, .crt, .cer or .pfx).
2	Upload	After choose a certificate file, you could upload your own certificate files.
3	Self-signed	You could create a self-signed certificate.
4	Remove	Remove the current Certificate.

Users can follow these instructions to create a self-signed certificate:

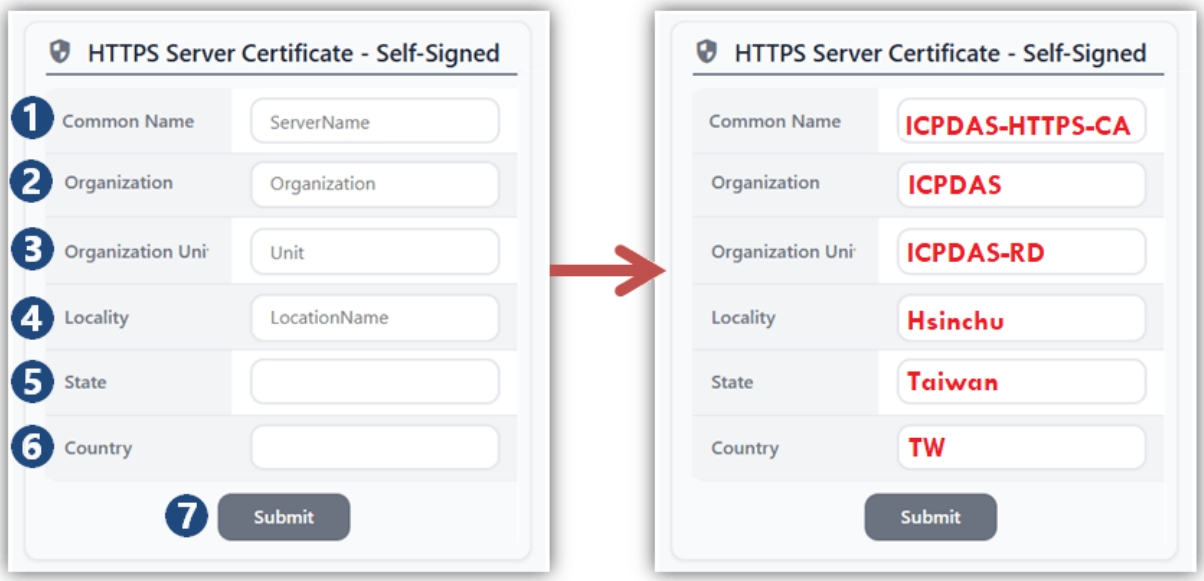
STEP 1

To click the [**SELF-SIGNED**] button in the "Certificate" section, and then you will see the "**HTTPS Server Certificate – Self-Signed**" page. The screen will be as follows:



STEP 2

Enter the following information for the self-signed certificate manually. The screen will be as follows:



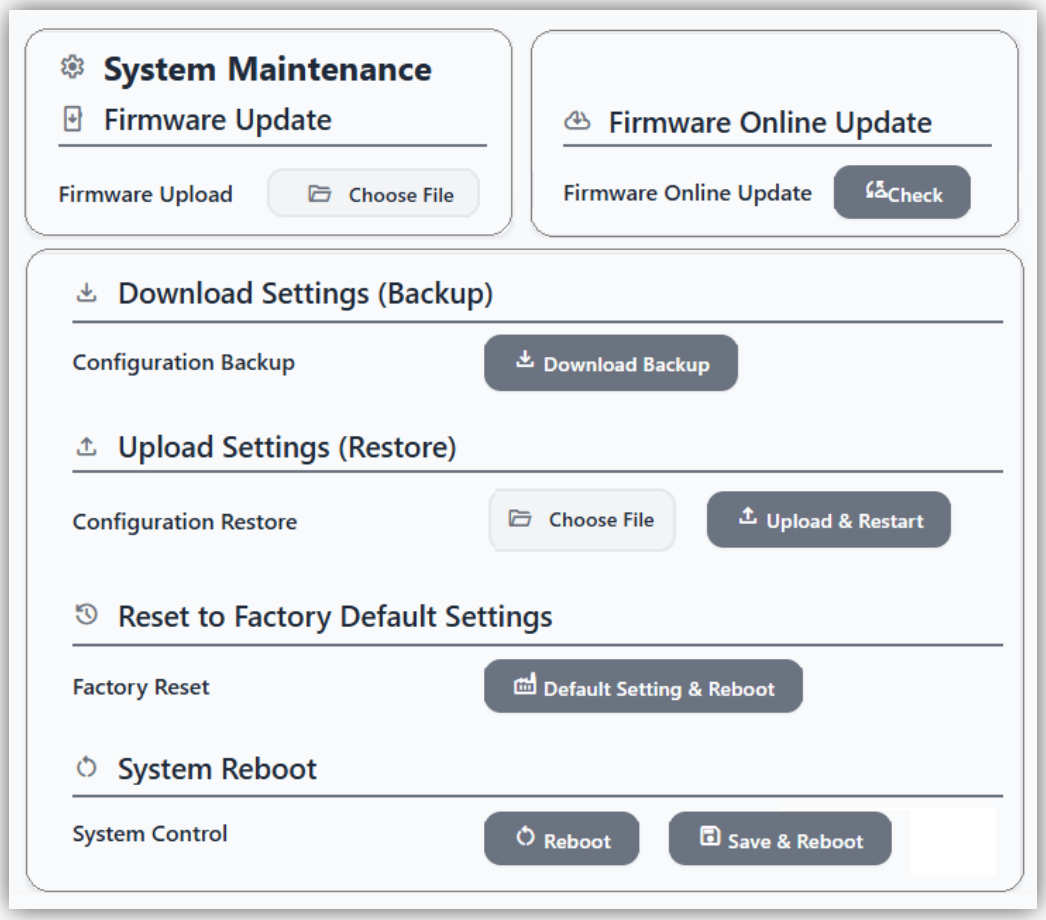
The following is an overview of the parameters contained in the **Self-Signed** section:

Item		Description	For example
1	Common Name	The fully qualified server name.	ICPDAS-HTTPS-CA Or gw.icpdas.com
2	Organization	This parameter is used to set the name of your organization.	ICPDAS
3	Organization Unit	This parameter is used to set the name of your organization unit.	ICPDAS-RD
4	Locality	This parameter is used to set the location where your organization is located.	Hsinchu
5	State	This parameter is used to set the state where your organization is located.	Taiwan
6	Country	This parameter is used to set the country where your organization is located.	TW
7	Submit	Click this button to save the revised settings to the ECAT-2601.	

Finally, click [**Submit**] button in order to save the certificate and commit your changes. If you leave the changes uncommitted, it will get lost and the signed certificate cannot be installed.

5.3.4 System Maintenance

The **System Maintenance** had six main parts: Firmware update, Firmware Online update check, Download Settings(Backup), Upload settings (Restore), Reset to factory default settings and System reboot.



Firmware update requires initialization and local network operations. Traditional firmware update requires adjusting the Reset button and reboots the module manually for the initialization of firmware update, while new way allows user to initialize the module via web interface without adjusting the hardware switch. Initialization via web is useful when module is installed in remote site and can be accessed by a remote PC.

For detailed information regarding how to use this function to update the Firmware for ECAT-2601 series module, refer to the “ECAT-2601 Firmware Update Manual (EN)”. The download address is shown below:

ECAT-2601 :

https://www.icpdas.com/en/download/show.php?num=2750

5.3.4.1 Reset to Factory Default Settings

Use the following procedure to reset all parameters to their original factory default settings:

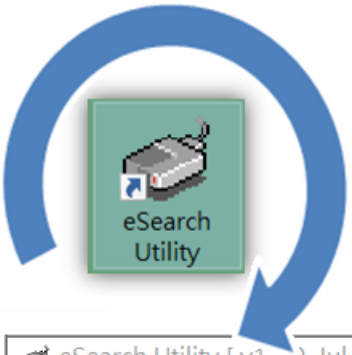
Step 1: Click the **"Default Settings & Reboot"** button to reset the configuration.

Step 2: Click the **"OK"** button in the message dialog box.

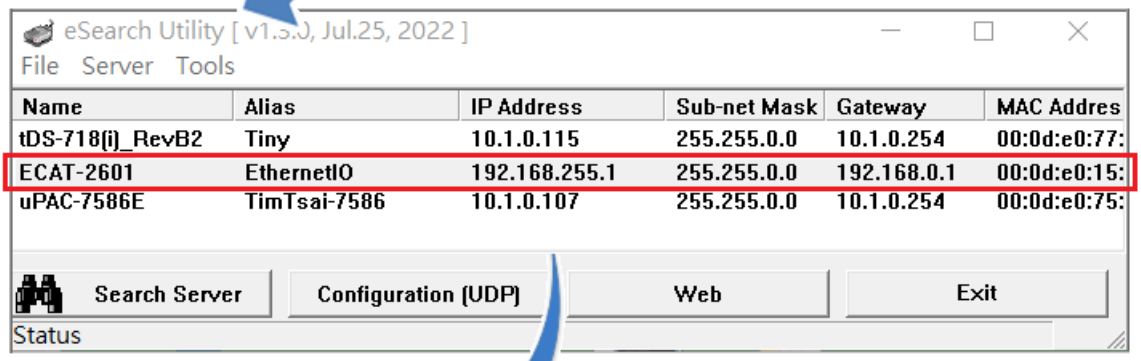
The following is an overview of the factory default settings:

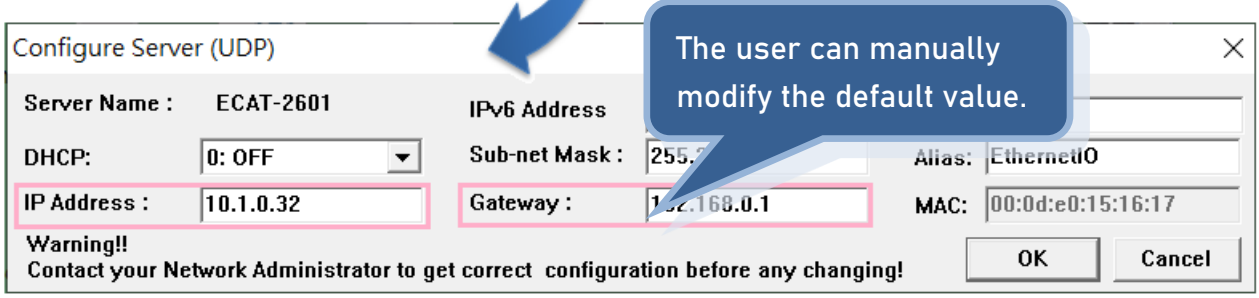
IPv4 Network Settings		IPv6 Network Settings	
IPv4 Address	192.168.255.1	User-defined	fc00::1
Gateway Address	192.168.0.1	Link-Local	Auto-configure
Subnet Mask	255.255.0.0	SLAAC	Auto-configure
DHCP	Disabled		

Step 3: Check whether the module has been reset to the original factory default settings for use with the eSearch Utility.



The **Reboot** function: can be used to force the ECAT-2601 to reboot or to remotely reboot the module. After the ECAT-2601 module has rebooted, the original login screen will be displayed requesting that you enter your Login Password before continuing.



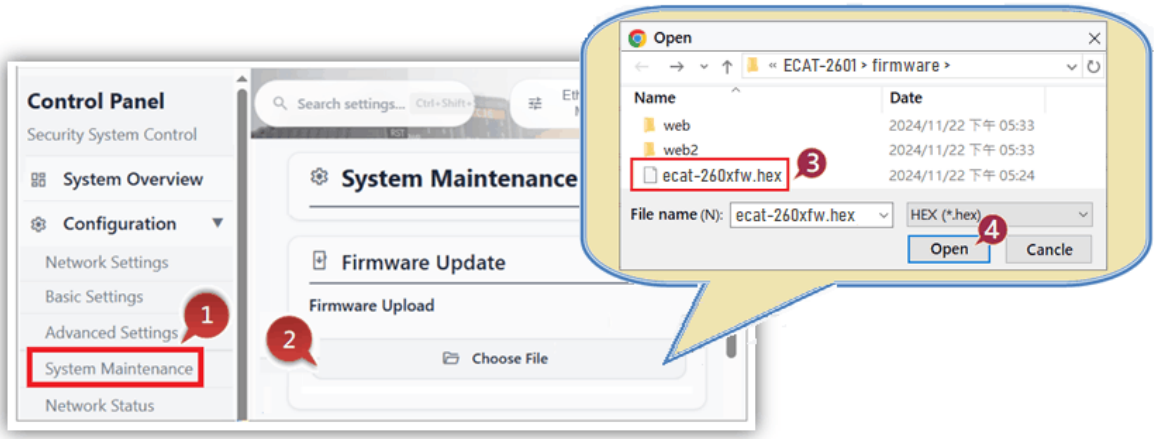


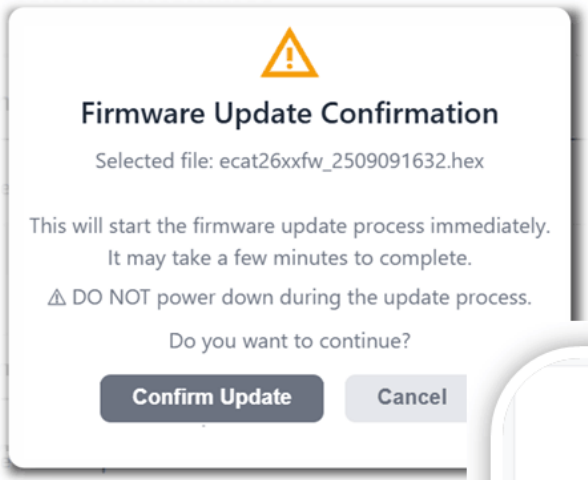
5.3.4.2 Firmware Update

Use the following procedure to update firmware:

Step 1~2: Click the **"Choose File"** button to select the firmware file.

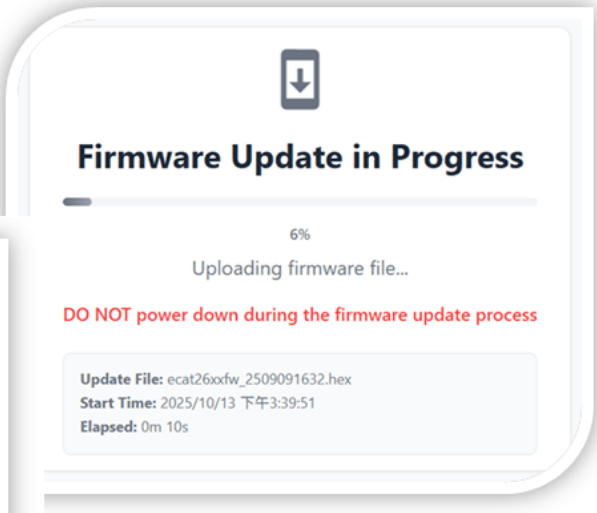
Step 3~4: Click the **"Open"** button in the message dialog box.

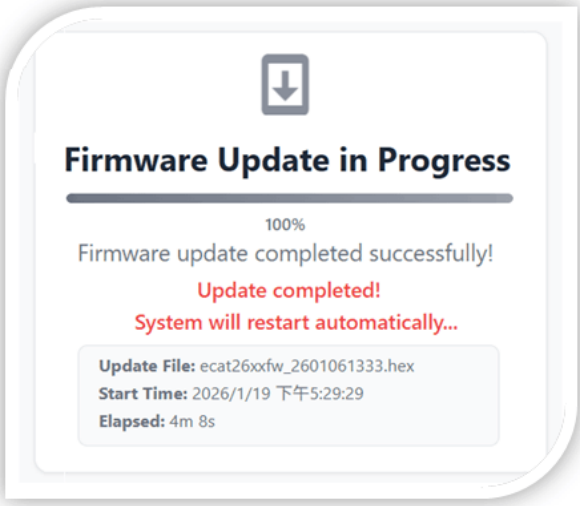




Note:

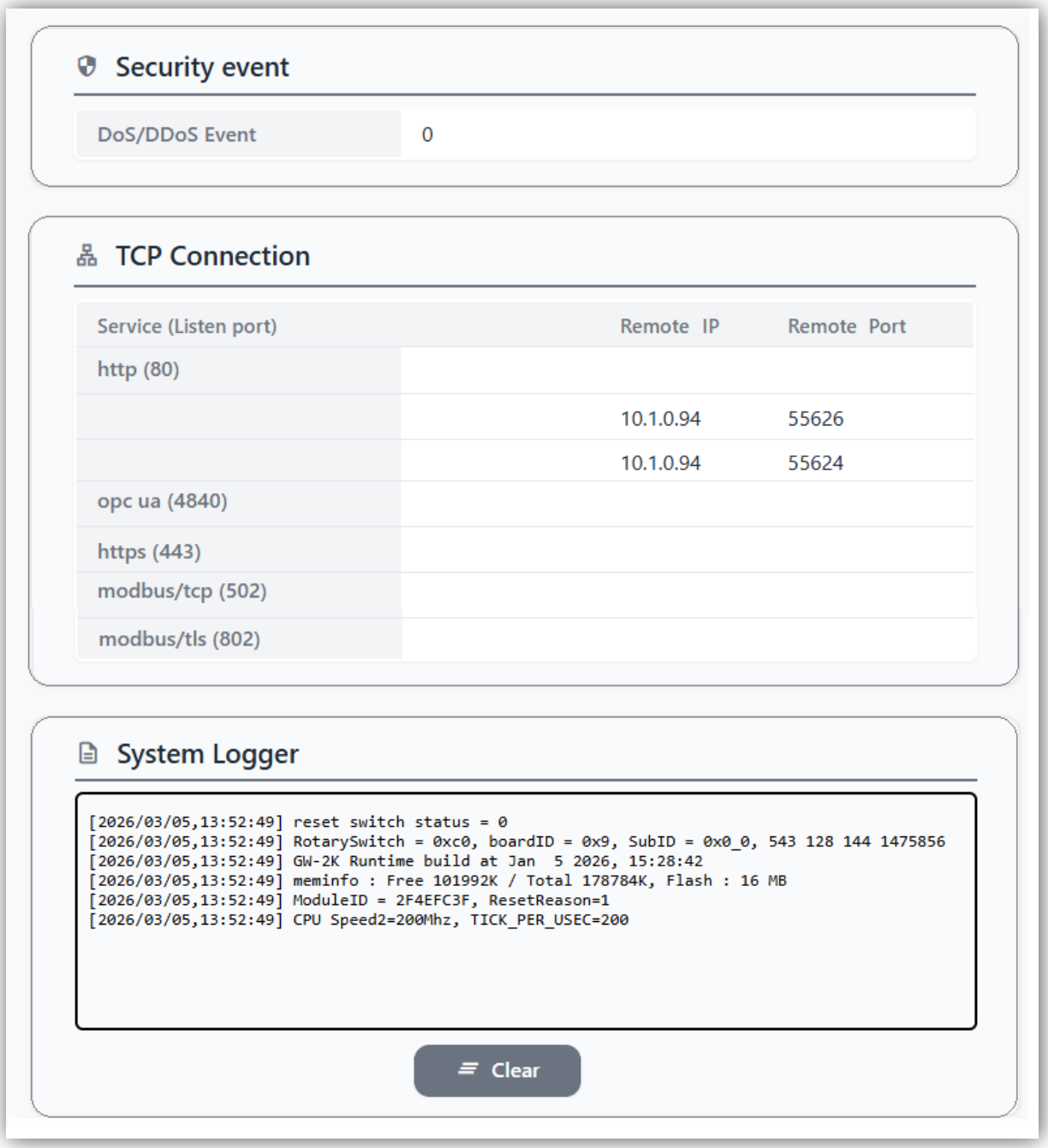
If the remote firmware update is failed, then the traditional firmware update (Local) is required to make the module working again.





5.3.5 Network Status

The **Status Network** had three main parts: Security event, TCP Connection and System logger.

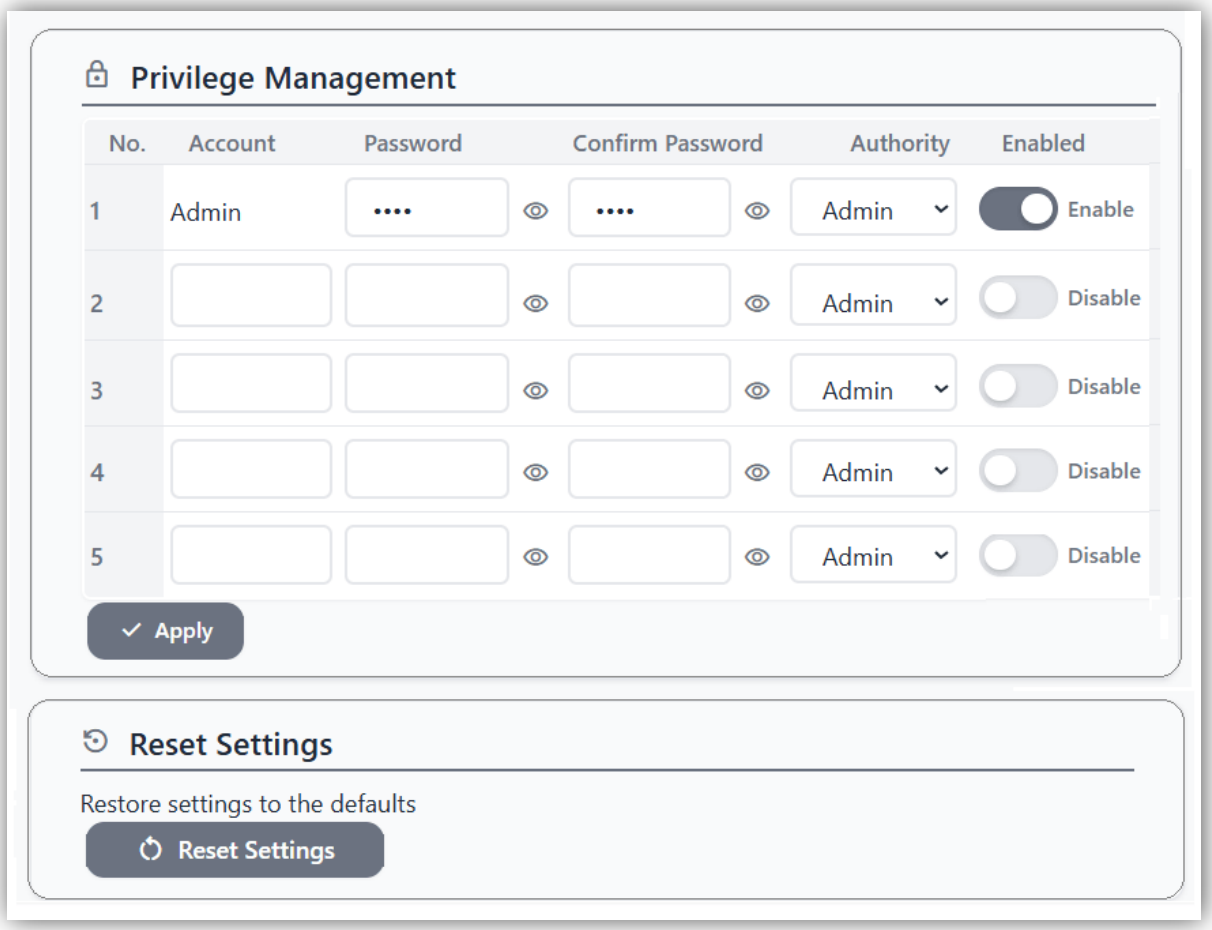


5.4 Authentication Page

After clicking the '**Authentication**' item will display the EtherCAT Authentication page on the right. User can configure the account management and the IP access control for the ECAT-2601 module, each of which will be described in more detail below.

5.4.1 Account Management

The **Account Management** had two main parts: Privilege management and Reset settings.



After clicking the **Password** tab, the **Privilege Management** page will be displayed.

Step 1: Enter the account and password in the field. (Use the default password "**Admin**" for the first login)

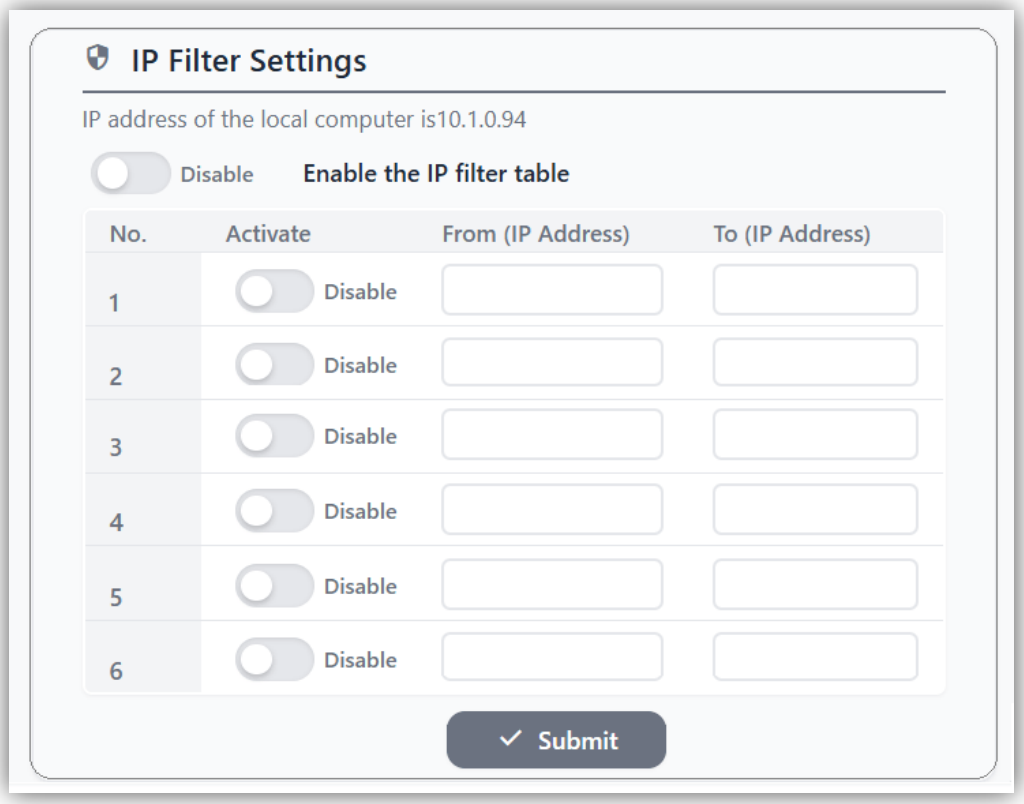
Step 2: Enter the new password in the "***Confirm password***" fields and make it enable.

Step 3: Click the "**Enable**" toggle switch to activate the function.

Step 4: Click the "**Apply**" button to update the account and password.

5.4.2 IP Access Control

The **Filter** page is used to query or edit the IP Filter List. The IP Filter List restricts the access of packets based on the IP header. If the filter function is enabled, only clients whose IP is specified in the IP Filter List can access the ECAT-2601.

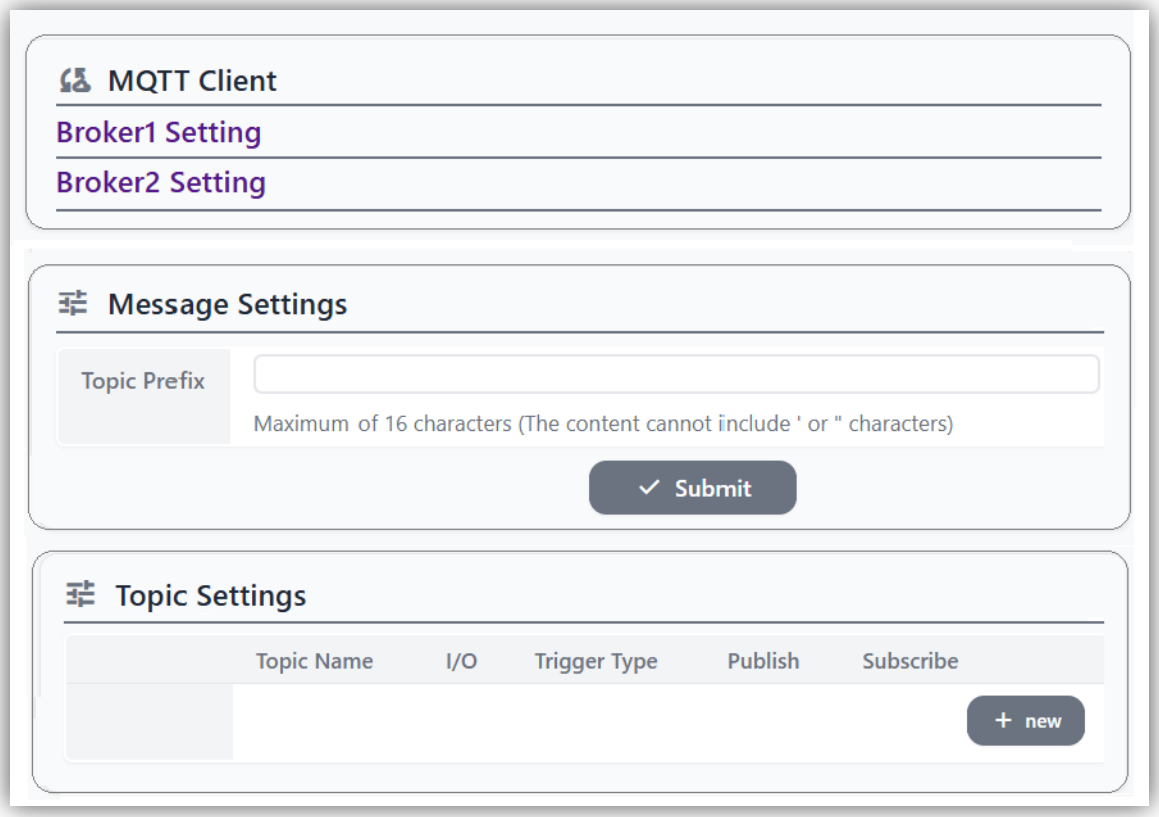


5.5 Connectivity Page

After clicking the '**Connectivity**' item will display the EtherCAT Connectivity page on the right. User can configure the MQTT protocol, OPC UA server and the REST API for the ECAT-2601 module, each of which will be described in more detail below.

5.5.1 MQTT Protocol

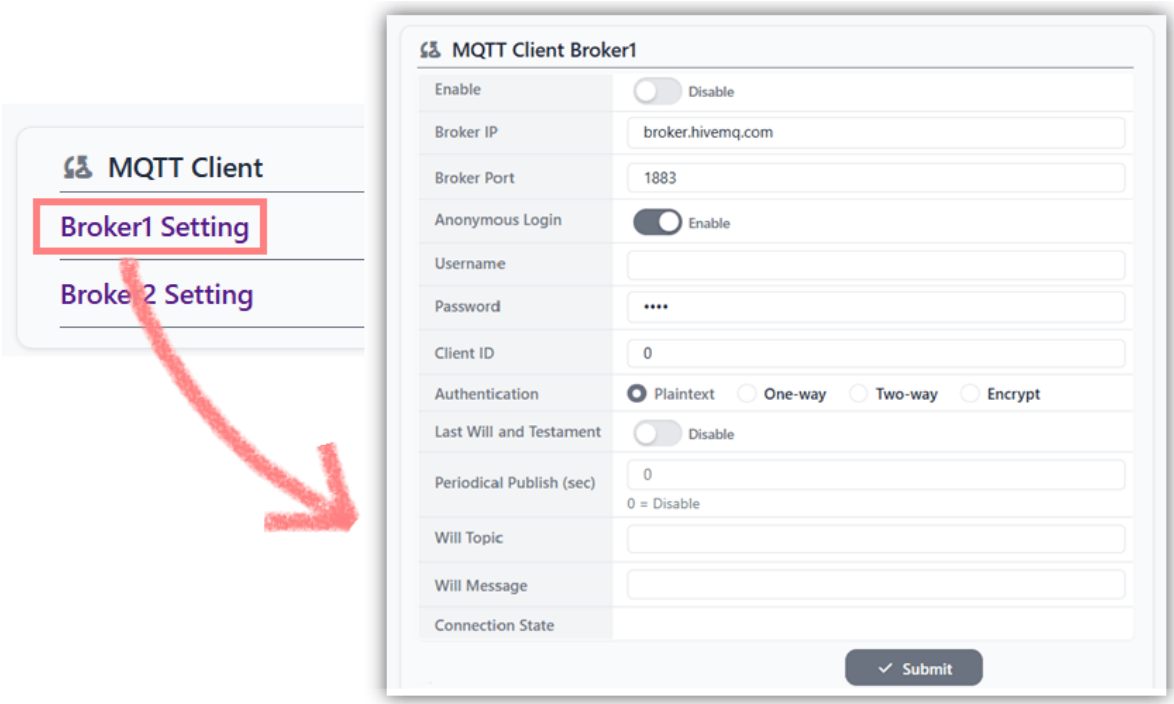
The **MQTT Protocol** had three main parts: MQTT Client, Message settings and Topic settings.



- Message settings

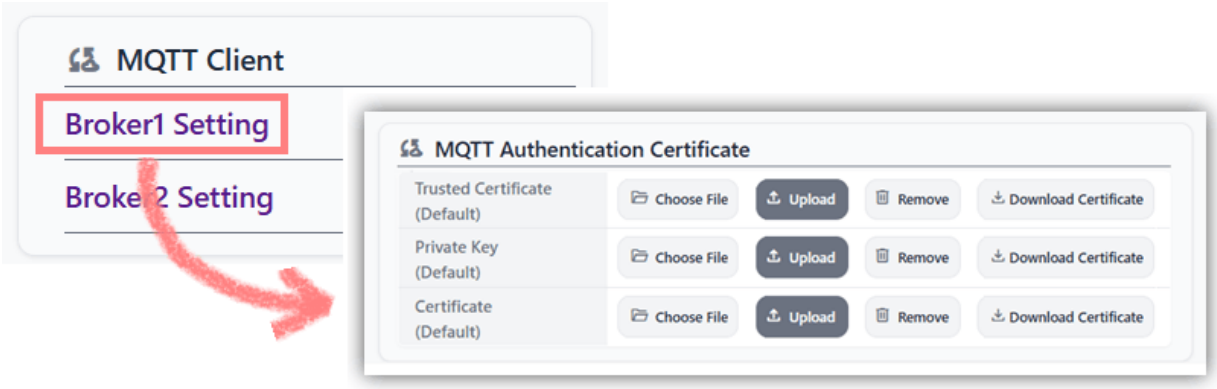
Specify the message content for the Manual publish message, and the ***Topic prefix*** is use to add the predefined topic prefix into a MQTT message

■ MQTT Client



The following is an overview of the parameters contained in the **MQTT Client broker** section:

Item	Description
Enable	Enable/Disable: Check the toggle switch to activate MQTT broker connection or not.
Broker IP	Specifies the IP address or domain name of the MQTT broker server.
Broker Port	Specify a port as the port for MQTT connection. System default: 1883.
Anonymous Login	Enable/Disable: Check the toggle switch to activate Anonymous Login or not.
Username	The client has the option to send a username and password when it connects to an MQTT broker
Password	The client has the option to send a username and password when it connects to an MQTT broker
Client ID	Specify a unique ID for the MQTT client.
Authentication	There are four types: Plaintext, One-way, Tow-way or Encrypt.
Last Will and Testament	Enable/Disable: Check the toggle switch to activate or not.
Periodical Publish (sec)	The time interval which all Publish Topics with the “Periodical Publish” attribute will be sent automatically periodically. If the value of the “Periodical Publish Interval” field is 0, it means to disable the “Periodical Publish” operation.
Will Topic	The Will Topic is not used for normal messaging, but strictly for communicating client state upon unexpected disconnection.
Will Message	When the client unexpectedly disconnects, the server sends this Will Message to other clients who have subscribed to the corresponding topic. Will Message is an important feature in MQTT, which solves the problem that only the server can know whether clients are offline. It allows us to gracefully take follow-up actions for unexpectedly offline clients.
Connection State	Displays the current connection status of the MQTT client.
Submit	Click this button to save the revised settings to the ECAT-2601.



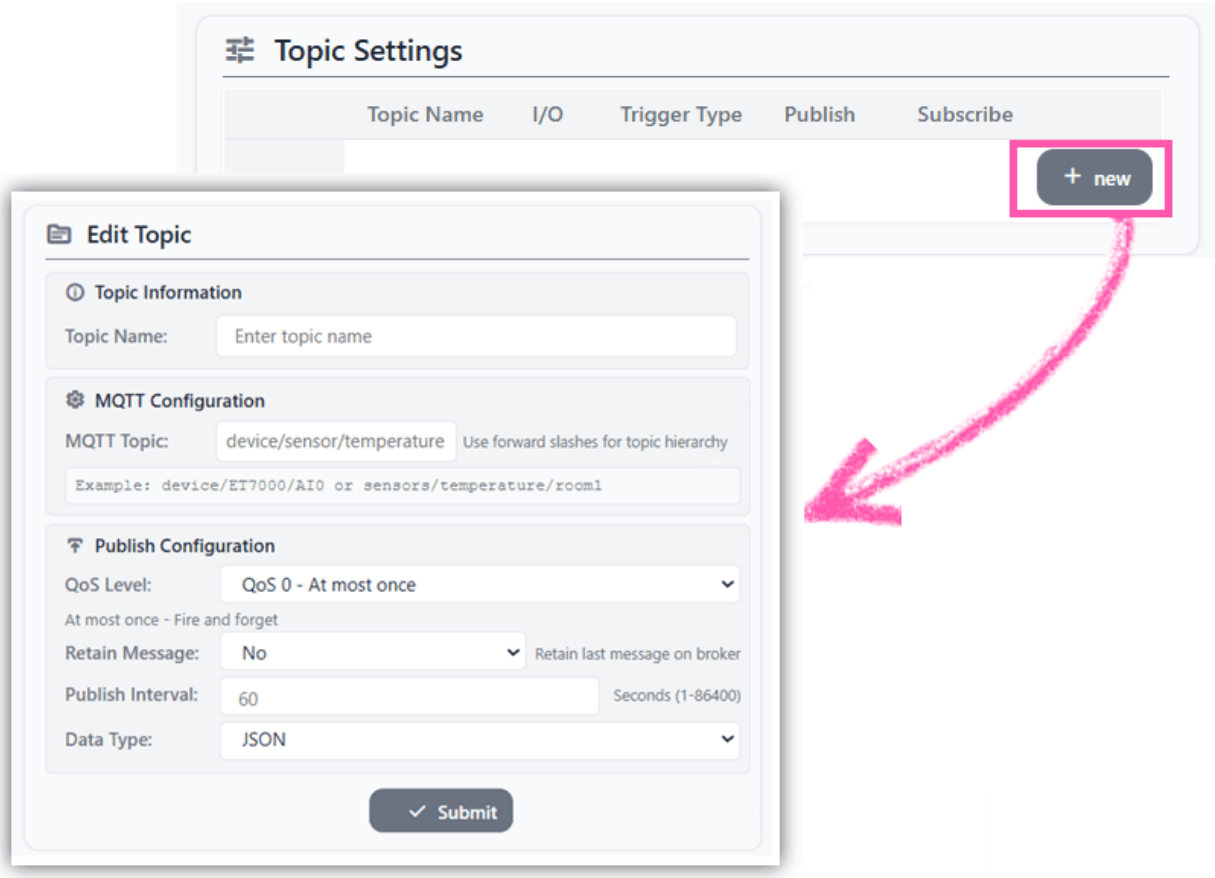
The [MQTT Authentication Certificate] is for setting up security communications to upload the MQTT Trusted Certificate, Certificate and Private Key.
The users upload the file to the ECAT-2601 according to the type of obtained certificate. If you want to perform Broker authentication, you need to upload the Trusted Certificate.
If you want to perform the Broker/Client two-way authentication, you need to upload the Credential and Private Key additionally.
The user can skip this step if the user project does not use certificate transmission security.

The following is an overview of the parameters contained in the **MQTT Authentication Certificate** section:

Item	Description
Trusted Certificate	Select File: select the MQTT Trusted Certificate file of the device.
	Upload: upload the MQTT Trusted Certificate file to the UA controller.
	Remove: remove the file of the device. File format must be PEM. Extension name must be “pem / cer / crt”.
	Download: Download the Certificate file to the PC.
Certificate	Select File: select the MQTT Certificate file of the device.
	Upload: upload the MQTT Certificate file to the UA controller.
	Remove: remove the file of the device. File format must be PEM. Extension name must be “pem / cer / crt”.
	Download: Download the Certificate file to the PC.
Private Key	Select File: select the MQTT Private Key of the device.
	Upload: upload the MQTT Private Key file to the UA controller.
	Remove: remove the file of the device. File format must be PEM. Extension name must be “.key”.
	Download: Download the Certificate file to the PC.

■ Topic settings

The user can click "**new**" button to add Topic settings.

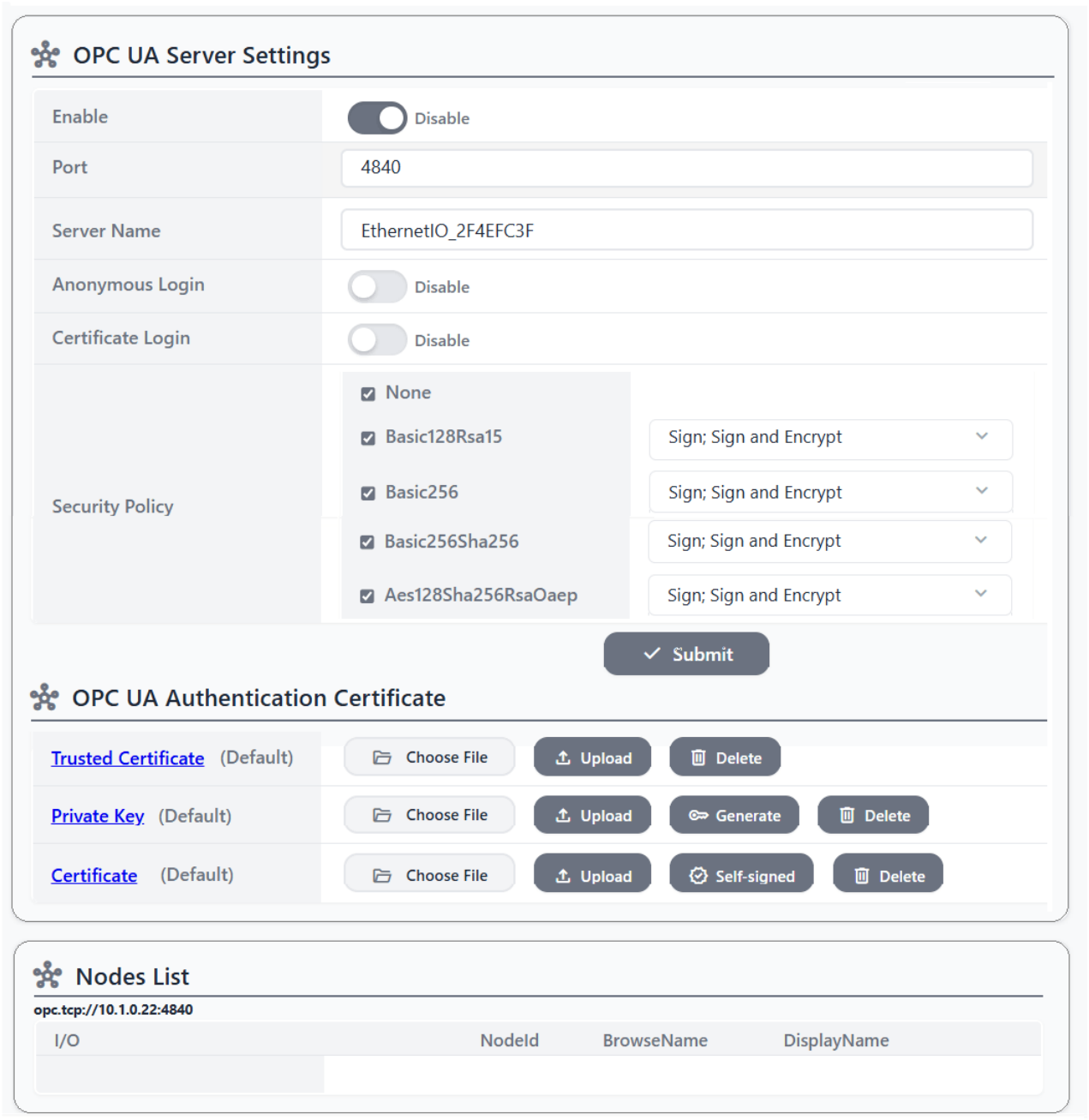


The following is an overview of the parameters contained in the **Topic Settings** section:

Item	Description	Default
Topic Name	Specify the topic for the message to be published	-
MQTT Topic	Use forward slashes for topic hierarchy. Example: Device/sensor/temperature	-
QoS Level	0 - at most once The message will be published only once, and the broker and subscribed client(s) take no additional steps to acknowledge the delivery, no matter if it is received or not. 1 - at least once 2 - exactly once	0
Retain Message	Yes or No (Retain last message on broker)	No
Publish interval	To publish a message with specified time interval. For seconds. Range: 1~86400	60
Data Type	There are three types: JSON, Plain Text or Binary.	JSON
Submit	Click this button to save the revised settings to the ECAT-2601.	

5.5.2 OPC UA Server

The **OPC UA Server** section is used to configure the setting as below:



The following is an overview of the parameters contained in the **OPC UA Server Settings** section:

Item	Description	Default
Enable	Enable/Disable: Check the toggle switch to activate **OPC UA** or not.	Enable
Port	The communication port number of the OPC UA Server.	4840
Server Name	User defined.	-
Anonymous Login	Check to enable the user password login from OPC UA clients. The following are the defaults for username and password: Username: Admin Password: Admin It is necessary that user use the latest ones, if you change your username and password.	Disable
Certificate Login	Check to enable the certificate login from OPC UA clients. (refer to next section, Server Certificate)	Disable
Security Policy	None	Enable
	Basic128Rsa15 Security Mode: Sign, Sign and Encrypt	Enable
	Basic256 Security Mode: Sign, Sign and Encrypt	Enable
	Basic256Sha256 Security Mode: Sign, Sign and Encrypt	Enable
	Aes128Sha256RsaOaep Security Mode: Sign, Sign and Encrypt	Enable
Submit	Click this button to save the revised settings to the ECAT-2601.	

To connect the ECAT module (ECAT-2028) with ECAT port, after clicking the **OPC UA** tab, and the **OPC UA Server Settings** page will be displayed.

Secure Encrypted Connection: OPC UA Authentication Certificate

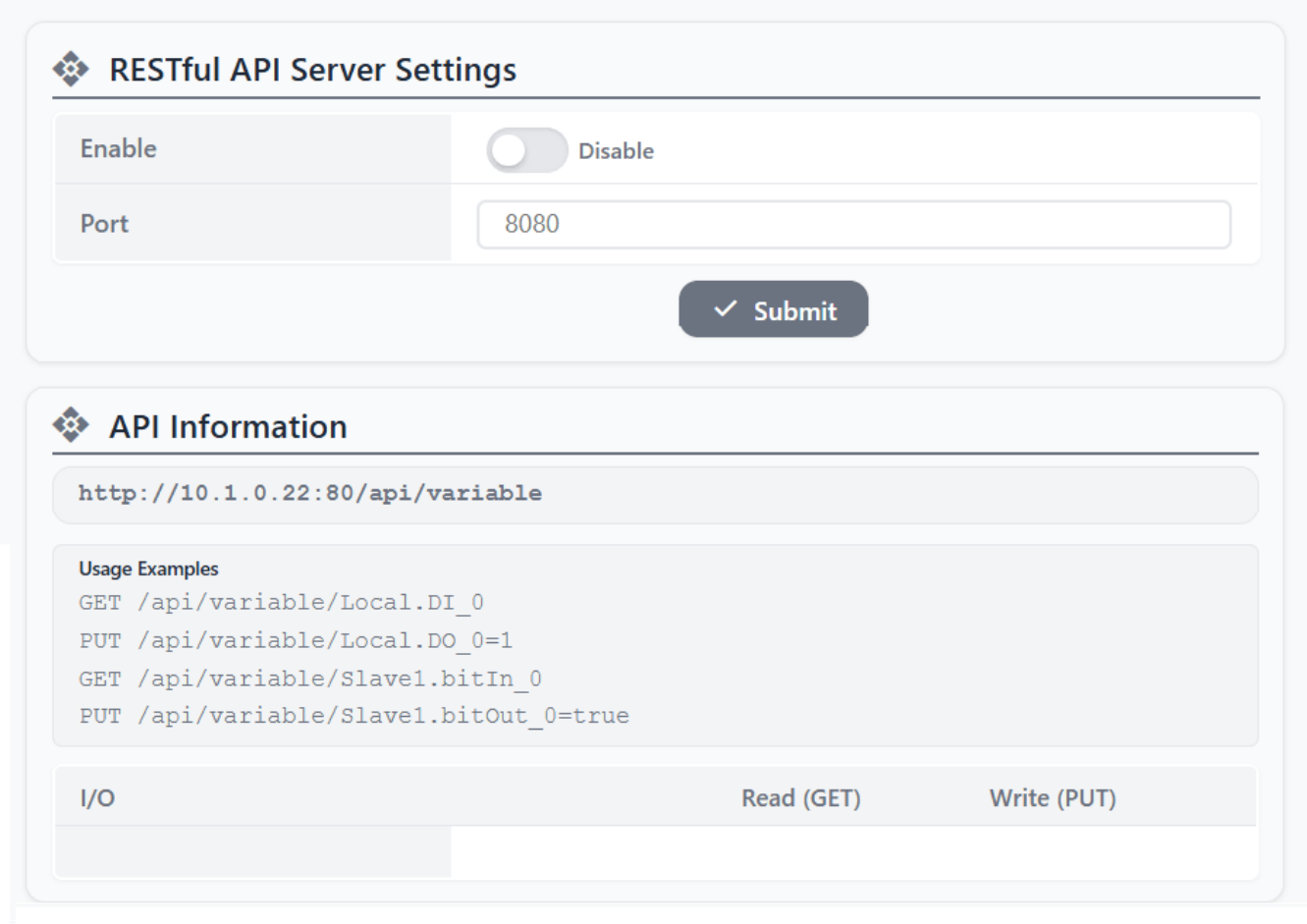
When using the OPC UA connection, in addition to the account login for security, users can also enable the certificate login to double the protection by the secure encryption.

When enabling the OPC UA certificate login, the Server/Client both sides of the connection need to add certificates to each other's trust zones.

User can also create a Self-Signed OPC UA Compliant Certificate by clicking the "**SELF-SIGNED**" button.

5.5.3 REST API

The **RESTful API Server Settings** section is used to configure the setting as below:



The following is an overview of the parameters contained in the **RESTful API Server Settings** section:

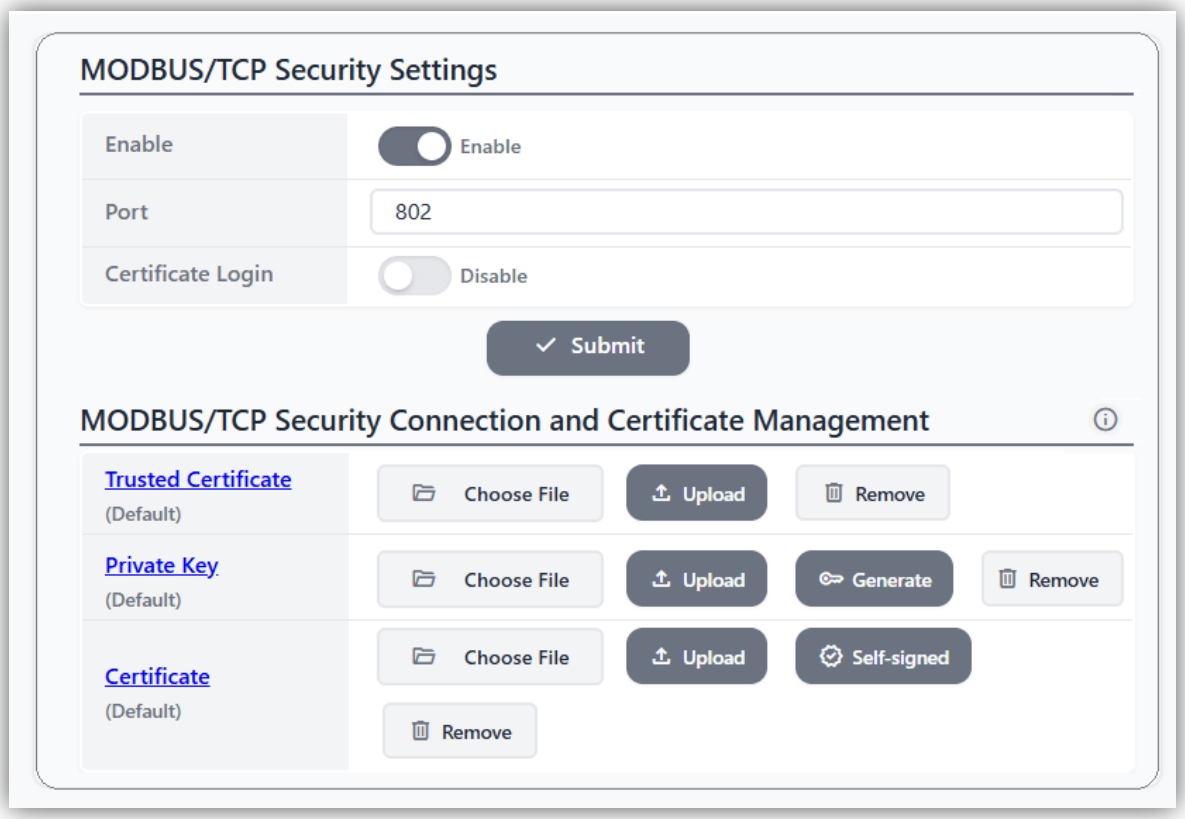
Item	Description	Default
Enable	Enable/Disable: Check the toggle switch to activate **RESTful API Server** or not.	Disable
Port	Setup the port number	8080
Submit	Click this button to save the revised settings to the ECAT-2601.	

5.6 MODBUS Security Page

After clicking the '**MODBUS Security**' item will display the EtherCAT Connectivity page on the right. User can configure the security settings and the I/O mapping for the ECAT-2601 module, each of which will be described in more detail below.

5.6.1 Security Settings

The **Security Settings** section is used to configure the setting as below:



The following is an overview of the parameters contained in the **Modbus/TCP Security Settings** section:

Item	Description	Default
Enable	Enable/Disable: Check the toggle switch to activate **Modbus/TCP** security function or not.	Enable
Port	Allows you to enter MODBUS/TCP security port.	802
Certificate Login	Enable/Disable: Check the toggle switch to activate **Certificate Login** or not.	Disable
Submit	Click this button to save the revised settings to the ECAT-2601.	

The users upload the file to the ECAT-2601 according to the type of obtained certificate. There are 3 types of Modbus/TCP Security connection and certificate management:

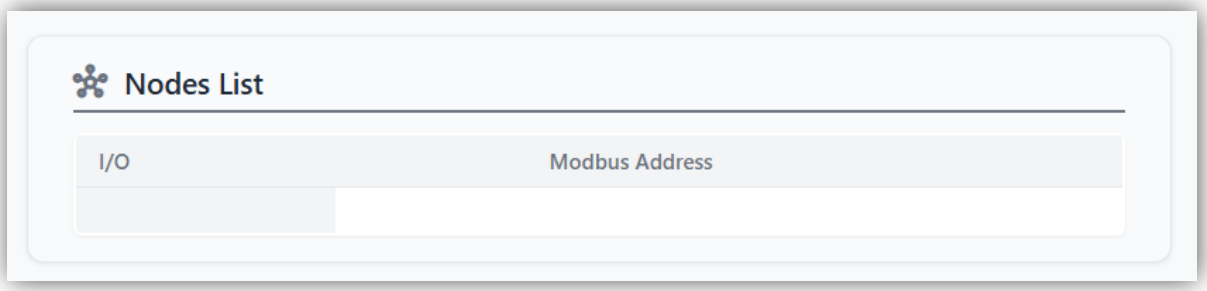
- Trusted Certificate
- Certificate
- Private Key (A private key must be paired with the certificate for encryption.)

The user can click the “CHOOSE FILE”, “UPLOAD” and “REMOVE” button to finish setting.

5.6.2 I/O Mapping

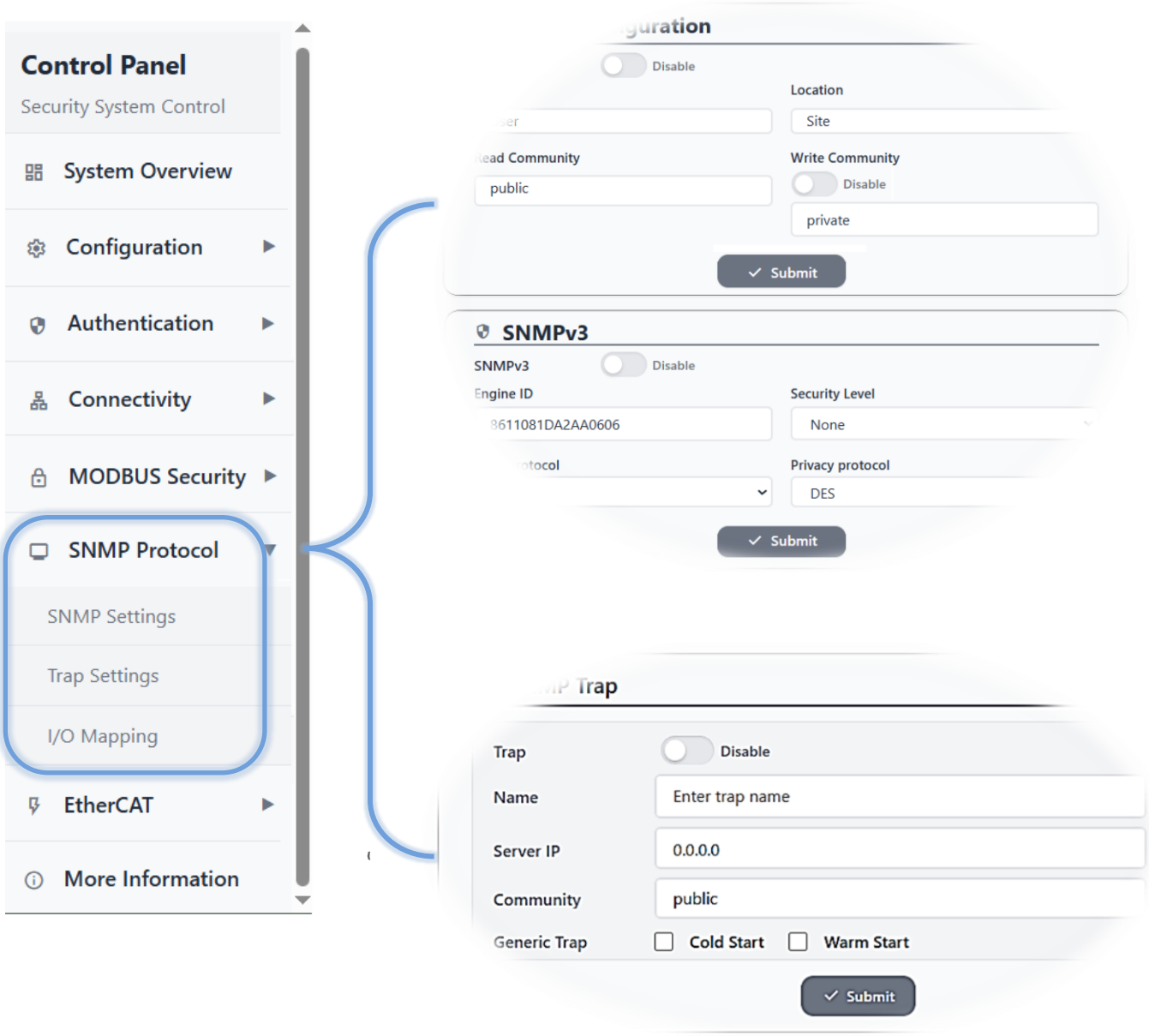
The “I/O Mapping” page displays the mapping relationship between the device’s internal I/O nodes and their corresponding Modbus addresses.

The table shows the I/O nodes and their Modbus address assignments.



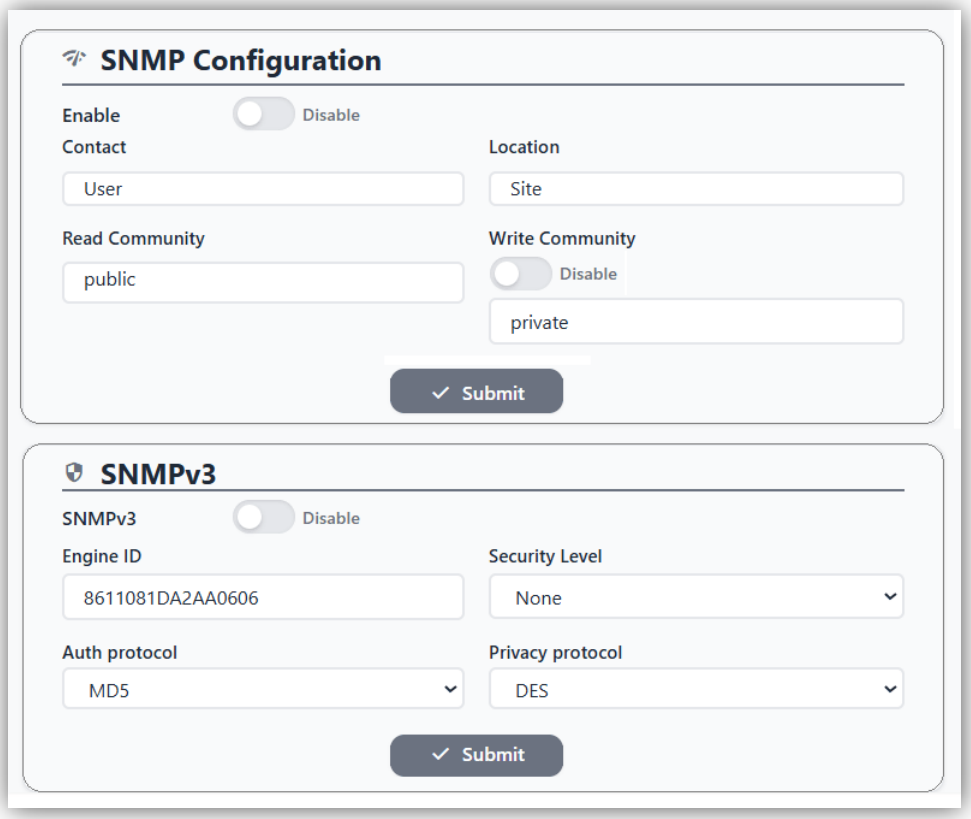
5.7 SNMP Protocol Page

After clicking the "**SNMP Protocol**" item will display the EtherCAT Connectivity page on the right. User can configure the SNMP settings, trap settings and the I/O mapping for the ECAT-2601 module, each of which will be described in more detail below.



5.7.1 SNMP Settings

The **SNMP Configuration Settings** section is used to configure the setting as below:

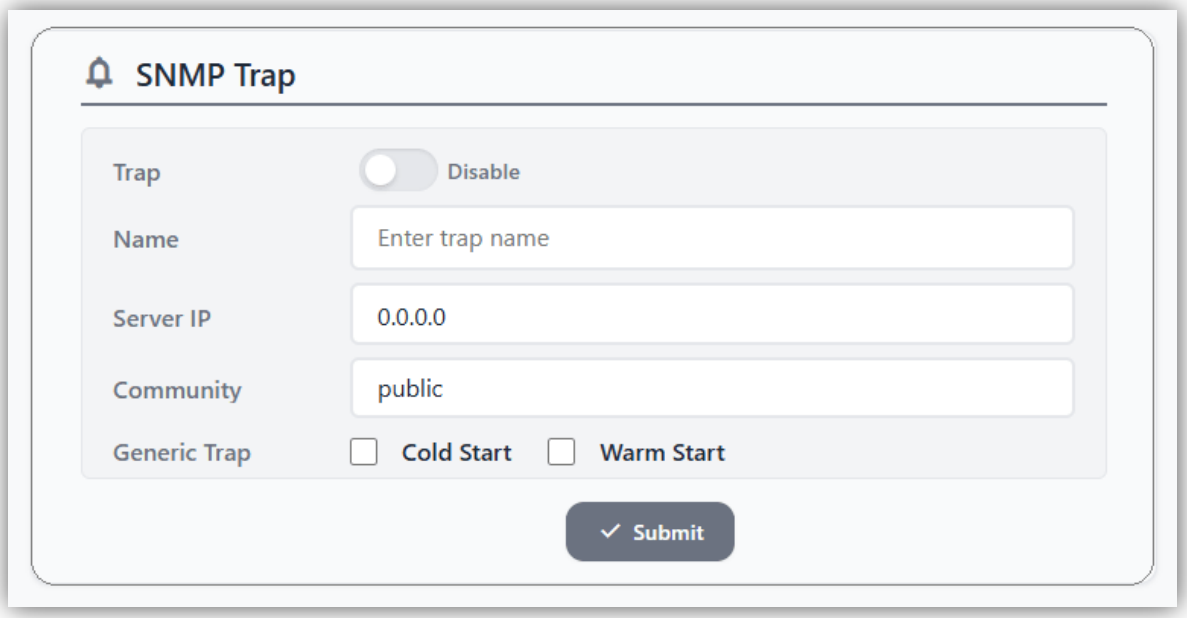


The following is an overview of the parameters contained in the **SNMP Configuration Settings** section:

Item	Description
Enable	Enable/Disable: Check the toggle switch to activate **SNMP** service or not.
Contace	Specifies the contact information of the device administrator.
Location	Defines the physical location of the device.
Read Community	Sets the community string with read-only access permission.
Wire Community	Sets the community string with read-write access permission.
SNMPv3	Enable/Disable: Check the toggle switch to activate **SNMPv3** service or not.
Engine ID	Specifies the unique SNMP engine identifier of the device.
Security Level	Defines the security level for SNMPv3 communication.
Auth Protocol	Specifies the authentication protocol used for user verification.
Privacy Protocol	Specifies the encryption protocol used to protect data transmission.
Submit	Click this button to save the revised settings to the ECAT-2601.

5.7.2 Trap Setting

The **SNMP Trap Settings** section is used to configure the setting as below:

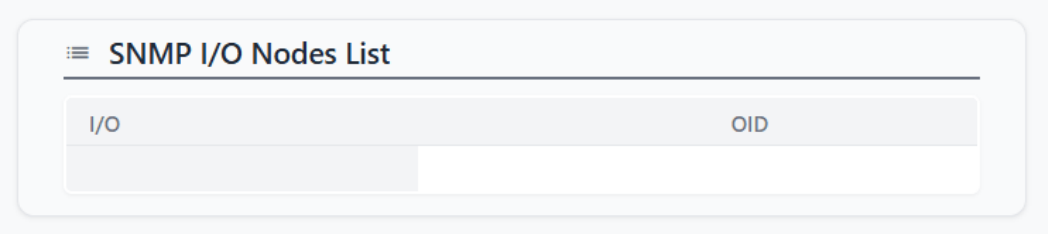


The following is an overview of the parameters contained in the **SNMP Trap Settings** section:

Item	Description	Default
Trap	Enable/Disable: Check the toggle switch to activate **SNMP Trap** or not.	Disable
Name	Specifies a user-defined name for this trap configuration entry.	-
Server IP	Defines the IP address of the SNMP management station that will receive trap notifications.	0.0.0.0
Community	Specifies the community string used for SNMP Trap authentication (SNMP v1/v2c).	public
Generic Trap	Specifies the generic trap type that will be sent when an event occurs.	-
Submit	Click this button to save the revised settings to the ECAT-2601.	

5.7.3 I/O Mapping

The “I/O Mapping” page displays the SNMP Node List supported by the device:

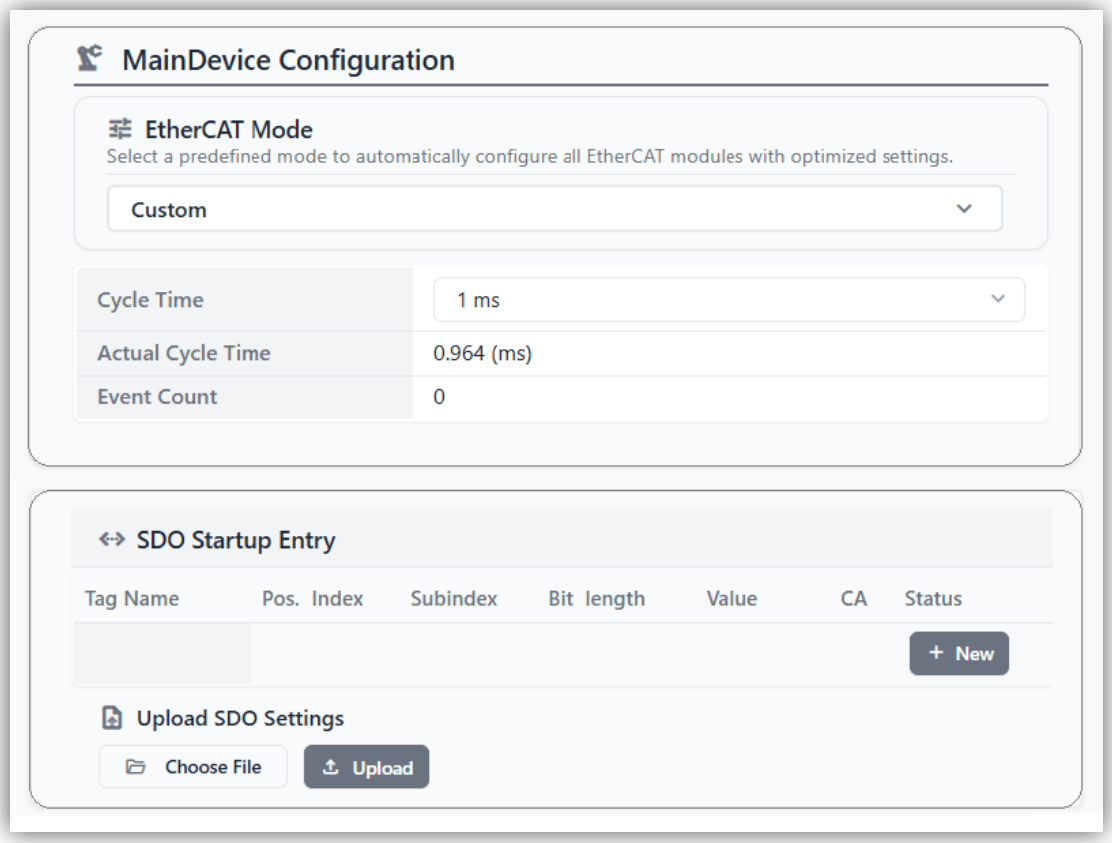


The table represents a manageable I/O object and its corresponding SNMP Object Identifier (OID).

5.8 EtherCAT Page

After clicking the "**EtherCAT**" item will display the EtherCAT Connectivity page on the right. User can configure the system status for the ECAT-2601 module, each of which will be described in more detail below.

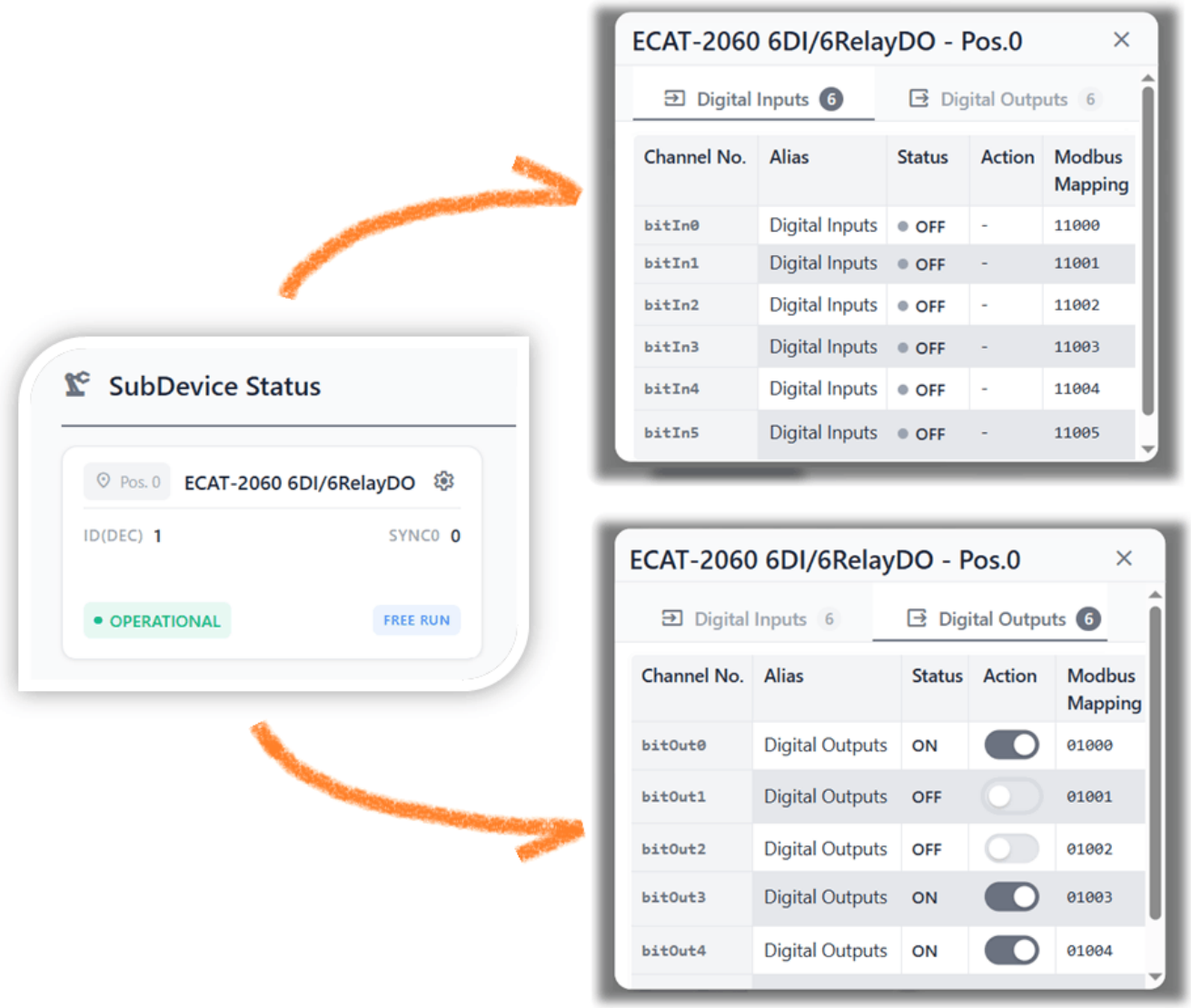
The **MainDevice Configuration** is used to configure the setting as below:



The following is an overview of the parameters contained in the **MainDevice Configuration** section:

Item	Description
EtherCAT Mode	Select a predefined mode to automatically configure all EtherCAT modules with optimized settings. ■ Custom: Manual configure - all settings can be customized individually ■ Energy Saving: Optimized for power consumption - longer cycle times, reduced CPU usage ■ Auto: Automatic optimization - balanced performance and power consumption ■ High Performance: Maximum performance - shortest cycle times, highest precision ■ Synchronization: Synchronization mode - optimized for multi-axis coordination
Cycle Time	The predefined, constant time interval at which the EtherCAT Master is programmed to send cyclic process data frames.
Actual Cycle Time	The real-time, measured duration between two consecutive occurrences of a synchronization event or the actual arrival of EtherCAT frames at a slave.
Event Count	A set of diagnostic counters within the Slave Controller or application that record the frequency of specific synchronization or communication triggers.

If the EtherCAT port (LAN3) connect with the ECAT-2060 device, the **SubDevice Status** section is used to configure the setting as below:

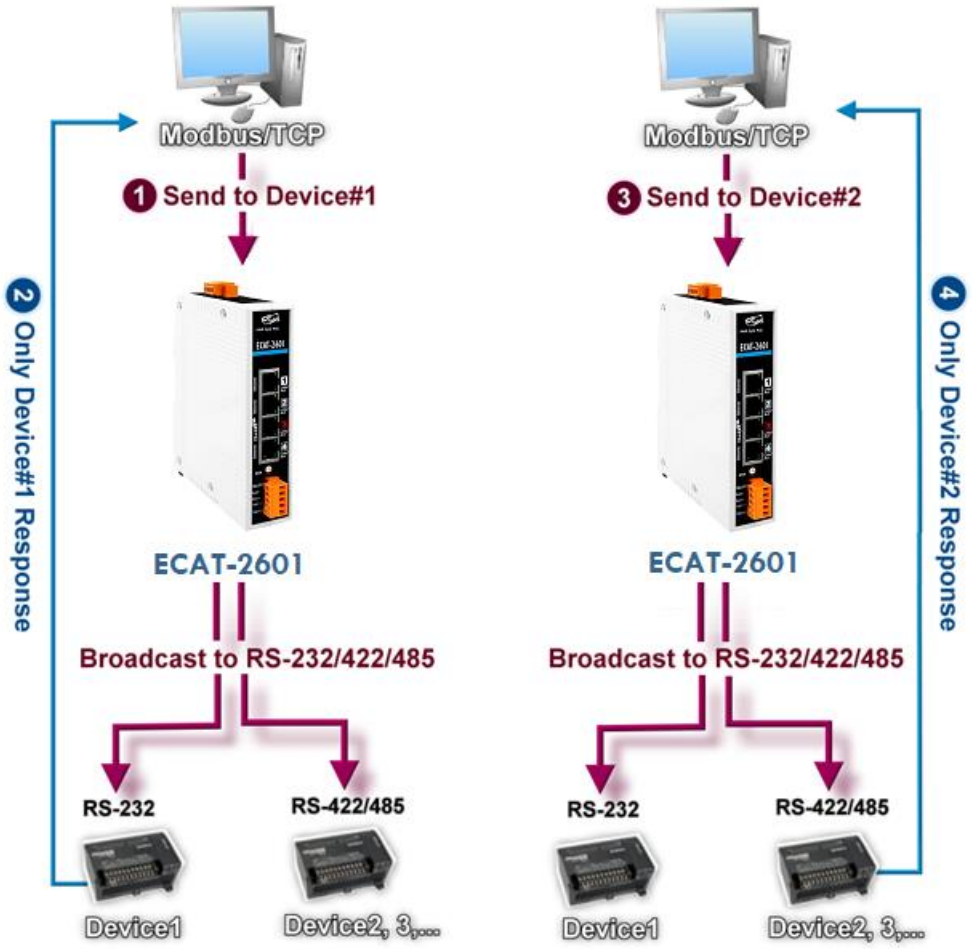


6. Typical Applications

This chapter provides some examples of typical scenarios for the ECAT-2601 module, including applications focused on the Modbus Gateway, Modbus Net ID, Pair-connection and TCP Client Mode, etc...

6.1 Modbus Gateway

The ECAT-2601 module is a Modbus TCP/UDP to RTU/ASCII gateway that enables a Modbus TCP/UDP host to communicate with serial Modbus RTU/ASCII devices through an Ethernet network, and eliminates the inherent cable length limitations of legacy serial communication devices.

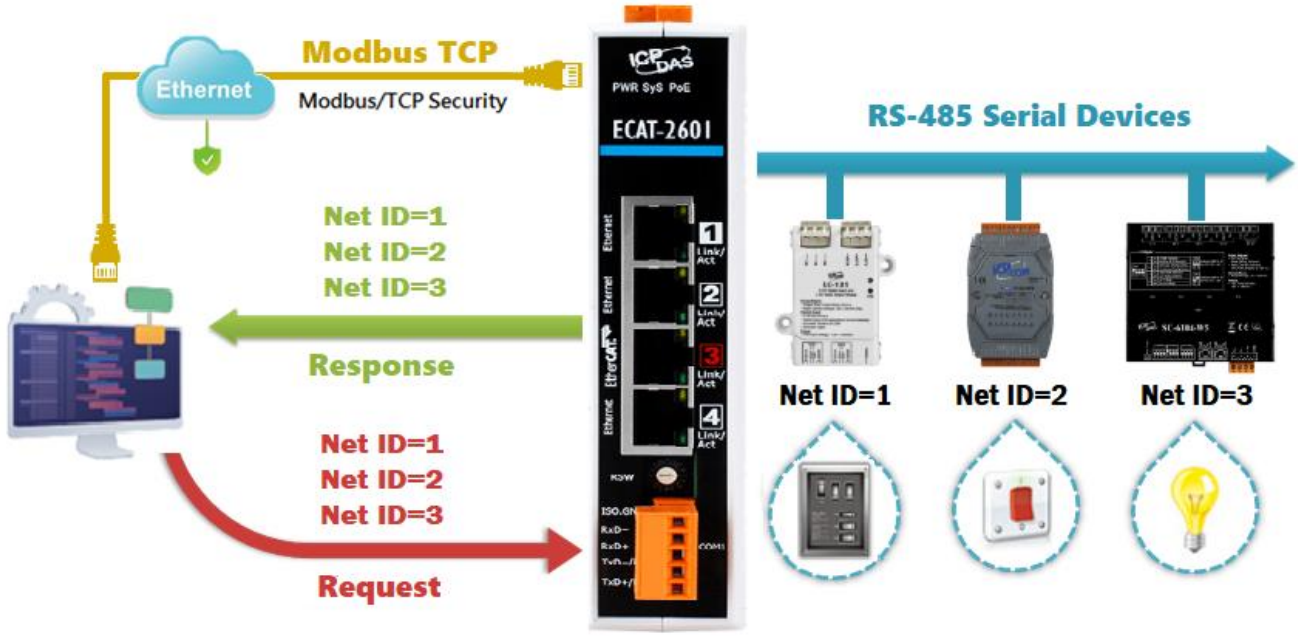


6.2 Modbus Net ID

The ECAT-2601 module is a gateway that can be used to convert between the Modbus TCP/UDP protocol and the Modbus RTU/ASCII protocol.

Consequently, SCADA/HMI applications is able to access each Modbus RTU/ASCII slave device via the ECAT-2601 gateway by specifying correct Net ID of the intended slave device in each Modbus TCP request.

Note that the Net ID of the ECAT-2601 gateway is reserved for specific control purposes, and is not used to access slave devices.



7. Modbus Information

What is Modbus TCP/IP?

Modbus is a communication protocol developed by Modicon in 1979. You can also visit http://www.modbus.org to find more valuable information.

The Different versions of Modbus used today include Modbus RTU (based on serial communication interfaces such as RS485 and RS232), Modbus ASCII and Modbus TCP, which is the Modbus RTU protocol embedded into TCP packets.

Modbus TCP is an internet protocol. The protocol embeds a Modbus frame into a TCP frame so that a connection oriented approach is obtained, thereby making it reliable. The master queries the slave and the slave responds with the reply. The protocol is open and, hence, highly scalable.

7.1 Modbus Message Structure

Modbus devices communicate using a master-slave (client-server) technique in which only one device (the master/client) can initiate transactions (called queries). The other devices (slaves/servers) respond by supplying the requested data to the master, or by taking the action requested in the query.

A query from a master will consist of a slave address (or broadcast address), a function code defining the requested action, any required data, and an error checking field. A response from a slave consists of fields confirming the action taken, any data to be returned, and an error checking field.

Modbus TCP Message Structure

Byte 00~05	Byte 06~11
6-byte header	RTU Data

Leading 6 bytes of Modbus TCP protocol:

Byte 00	Byte 01	Byte 02	Byte 03	Byte 04	Byte 05
Transaction identifier		Protocol identifier		Length field (upper byte)	Length field (lower byte)

Transaction identifier: Assigned by Modbus TCP master (client)

Protocol identifier: 0

Length field (upper byte): 0 (since all messages are smaller than 256)

Length field (lower byte): Number of following RTU data bytes

RTU Data Structure

Byte 06	Byte 07	Byte 08-09	Byte 10-11
Net ID (Station number)	Function Code	Data Field	
		Reference number (Address Mapping)	Number of points

- **Net ID (Station Number):** specifies the address of the receiver (Modbus TCP slave).

  The first byte in the message structure of Modbus is the receiver's address. The valid addresses are in the range of 0 to 247. Addresses 0 is used for broadcast, while addresses 1 to 247 are given to individual Modbus devices.

- **Function Code:** specifies the message type.

  The second byte in the frame structure of the Modbus RTU is the function code. The function code describes what the slave is required to do. Valid function codes are between 1 and 255. The slave uses the same function code as the request to answer it. Only when an error occurs in the system will the highest bit of the function code is set to '1'. Hence the master will know whether the message has been transmitted correctly or not.

Code	Function	Reference (Address)
01 (0x01)	Read the Status of the Coils (Readback DOs)	0xxxx
02 (0x02)	Read the Status of the Input(Reads DIs)	1xxxx
03 (0x03)	Read the Holding Registers (Readback AOs)	4xxxx
04 (0x04)	Read the Input Registers (Reads AIs)	3xxxx
05 (0x05)	Force a Single Coil (Writes DO)	0xxxx
06 (0x06)	Preset a Single Register (Writes AO)	4xxxx
15 (0x0F)	Force Multiple Coils (Writes DOs)	0xxxx
16 (0x10)	Preset Multiple Registers (Writes AOs)	4xxxx

- **Data Field:** is the data block.

Data is transmitted in 8-, 16- and 32-bit format. The data for 16-bit registers is transmitted in high-byte first format. For example: 0x0A0B ==> 0x0A, 0x0B. The data for 32-bit registers is transmitted as two 16-bit registers, and is low-word first. For example: 0x0A0B0C0D ==> 0x0C, 0x0D, 0x0A, 0x0B.

The data field of messages sent between a master and a slave contains additional information about the action to be taken by the master or any information requested by the slave. If the master does not require this information, the data field can be empty.

Reference (Address)	Description
0xxxx	Read/Write Discrete Outputs or Coils. A 0x reference address is used to output device data to a digital output channel.
1xxxx	Read Discrete Inputs. The ON/OFF status of a 1x reference address is controlled by the corresponding digital input channel.
3xxxx	Read Input Registers. A 3x reference register contains a 16-bit number received from an external source, e.g. an analog signal.
4xxxx	Read/Write Output or Holding Registers. A 4x register is used to store 16bits of numerical data (binary or decimal), or to send the data from the CPU to an output channel.

Note: For details regarding address mapping (Reference Number) depends on your slave device.

01(0x01) Read the Status of the Coils (Readback DOs)

This function code is used to read either the current status of the coils or the current digital output readback value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x01
02-03	Starting DO Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Number of Points (Channels)	2 Bytes	Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x01
02	Byte Count	1 Byte	Byte Count of the Response (n = (Points+7)/8)
03	Data	n Bytes	n= 1; Byte 03 = data bit 7 to 0 n= 2; Byte 04 = data bit 15 to 8 ..... n= m; Byte m+2 = data bit (8m-1) to 8(m-1)

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x81
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details

02(0x02) Read the Status of the Input (Read DIs)

This function code is used to read the current digital input value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x02
02-03	Starting DI Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Number of Points (Channels)	2 Bytes	Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x02
02	Byte Count	1 Byte	Byte Count of Response (n =(Points+7)/8)
03	Data	n Bytes	n= 1; Byte 03 = data bit 7 to 0 n= 2; Byte 04 = data bit 15 to 8 ..... n= m; Byte m+2 = data bit (8m-1) to 8(m-1)

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x82
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details

03(0x03) Read the Holding Registers (Readback AOs)

This function code is used to readback either the current values in the holding registers or the analog output value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x03
02-03	Starting AO Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Number of 16-bit Registers (Channels)	2 Bytes	Word Count Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x03
02	Byte Count	1 Byte	Byte Count of the Response (n=Points x 2 Bytes)
03~	Register Values	n Bytes	Register Values n= 2; Byte 03 = high byte Byte 04 = low byte ..... n= m; Byte 03 = high byte Byte 04 = low byte ..... Byte m+1 = high byte Byte m+2 = low byte

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x83
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details

04(0x04) Read the Input Registers (Read AIs)

This function code is used to read either the input registers or the current analog input value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x04
02-03	Starting AI Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Number of 16-bit Registers (Channels)	2 Bytes	Word Count Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x04
02	Byte Count	1 Byte	Byte Count of the Response (n=Points x 2 Bytes)
03~	Register Values	n Bytes	Register Values n= 2; Byte 03 = high byte Byte 04 = low byte ..... n= m; Byte 03 = high byte Byte 04 = low byte ..... Byte m+1 = high byte Byte m+2 = low byte

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x84
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details.

05(0x05) Force a Single Coil (Write DO)

This function code is used to set the status of a single coil or a single digital output value.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x05
02-03	DO Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Output Value	2 Bytes	0xFF 00 sets the output to ON. 0x00 00 sets the output to OFF. All other values are invalid and will not affect the coil. Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x05
02-03	DO Address	2 Bytes	The value is the same as Bytes 02-03 of the Request
04-05	Output Value	2 Bytes	The value is the same as Bytes 04-05 of the Request

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x85
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details.

06(0x06) Preset a Single Register (Write AO)

This function code is used to set a specific holding register to store the configuration values.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x06
02-03	AO Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Register Value	2 Bytes	Register Value Byte 04 = high byte Byte 05 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x06
02-03	AO Address	2 Bytes	The value is the same as Bytes 02-03 of the Request
04-05	Register Value	2 Bytes	The value is the same as Bytes 04-05 of the Request

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x86
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details.

15(0x0F) Force Multiple Coils (Write DOs)

This function code is used to set multiple coils status or write multiple digital output values.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x0F
02-03	Starting DO Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Number of Output Channels (Points)	2 Bytes	Byte 04 = high byte Byte 05 = low byte
06	Byte count	1 Byte	n = (Points +7)/8
07	Output value	n Bytes	A bit corresponds to a channel. A value of 1 for a bit denotes that the channel is ON, while a value of denotes that the channel is OFF. n= 1; Byte 07 = data bit 7 to 0 n= 2; Byte 08 = data bit 15 to 8 ..... n= m; Byte m+6 = data bit (8m-1)to 8 (m-1)

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x0F
02-03	Starting DO Address	2 Bytes	The value is the same as Bytes 02-03 of the Request
04-05	Number of Output Channels (Points)	2Bytes	The value is the same as Bytes 04-05 of the Request

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1to 247
01	Function Code	1 Byte	0x8F
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details.

16(0x10) Preset Multiple Registers (Write AOs)

This function code is used to set multiple holding registers that are used to store the configuration values.

[Request]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x10
02-03	Starting AO Address	2 Bytes	Refer to the Modbus address depends on your slave device for more details. Byte 02 = high byte Byte 03 = low byte
04-05	Number of 16-bit Registers (Channels)	2 Bytes	Word Count. Byte 04 = high byte Byte 05 = low byte
06	Byte Count	1 Byte	n =Points x 2 Bytes
07	Register Values	n Bytes	Register Values. n= 2; Byte 03 = high byte Byte 04 = low byte ..... n= m; Byte 03 = high byte Byte 04 = low byte ..... Byte m+1 = high byte Byte m+2 = low byte

[Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x10
02-03	Starting AO Address	2 Bytes	The value is the same as Bytes 02-03 of the Request
04-05	Number of 16-bit Registers (Channels)	2 Bytes	The value is the same as Bytes 04-05 of the Request

[Error Response]

Byte	Description	Size	Value
00	Net ID (Station Number)	1 Byte	1 to 247
01	Function Code	1 Byte	0x90
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details.

7.2 Exception Codes

If an exception occurs during Modbus communication, the slave device will return an Exception Code in the response message. The following is an explanation of the Exception Codes:

➢ **Exception Codes:**

Code	Name and Description
0x01	ILLEGAL FUNCTION
	Indicates that the function code received in the query is not an allowable action for the slave. If not an allowable action for the slave. If a Poll Program Complete command was issued, this code indicates that no program function preceded it.
0x02	ILLEGAL DATA ADDRESS
	Indicates that the data address received in the query is not an allowable address for the slave.
0x03	ILLEGAL DATA VALUE
	Indicates that a value contained in the query data field is not an allowable value for the slave.
0x04	SLAVE DEVICE FAILURE
	Indicates that an unrecoverable error occurred while the slave was attempting to perform the requested action.
0x05	ACKNOWLEDGE
	Indicates that the slave has accepted the request and is processing it, but it will take an extended period of time to do so. This response is returned to prevent a timeout error from occurring in the master. The master can issue a Poll Program Complete message later to determine whether the processing is complete.
0x06	SLAVE DEVICE BUSY
	Indicates that the slave is engaged in processing a long–duration program command. The master should retransmit the message later when the slave is free.
0x07	NEGATIVE ACKNOWLEDGE
	Indicates that the extended file area failed to pass a consistency check, and the slave cannot perform the program function received in the query. This code is returned when a programming request using function code 13 or 14 decimal was unsuccessful. The master should request diagnostic or error information from the slave.
0x08	MEMORY PARITY ERROR
	Indicates that the slave attempted to read extended memory, but detected a parity error in the memory. The master can retransmit the request, but maintenance may be required on the slave device.

➢ **Defined Exception Codes for ECAT-2601:**

Code	Name and Description
0x0B	GATEWAY TARGET DEVICE FAILED TO RESPOND Timeout. The slave device does not respond within the timeout value, the ECAT-2601 will return this code.
0x4B	GATEWAY TARGET DATA FAILED TO RESPOND Timeout. The slave device is still sending data when timed out, the ECAT-2601 will return this code. Please use larger Slave Timeout value for the serial port of the ECAT-2601 module.
0x41	MODBUS PROTOCOL FORMAT ERROR The ECAT-2601 will return this code when slave response is invalid Modbus message.
0x42	WRONG DATA LENGTH The ECAT-2601 will return this code when ECAT-2601 received wrong data length. Please use larger Slave Timeout value for the serial port of the ECAT-2601 module.
0x43	CRC ERROR The ECAT-2601 will return this code when the CRC of the slave response is wrong.

Appendix A: Glossary

1. ARP (Address Resolution Protocol)

The Address Resolution Protocol (ARP) is a telecommunication protocol that is used to convert an IP address to a physical address, such as an Ethernet address.

Consider two machines A and B that share the same physical network. Each has an assigned IP address IP_A and IP_B, and a MAC address, MAC_A and MAC_B. The goal is to devise a low-level software application that hides the MAC addresses and allows higher-level programs to work only with the IP addresses. Ultimately, however, communication must be carried out by the physical networks using whatever MAC address scheme the hardware supplies.

Suppose machine A wants to send a packet to machine B across a physical network to which they are both attached, but A only has the Internet address for B, IP_B. The question arises: how does A map that address to the MAC address for B, MAC_B?

ARP provides a method of dynamically mapping 32-bit IP address to the corresponding 48-bit MAC address. The term dynamic is used since the mapping is performed automatically and is normally not a concern for either the application user or the system administrator.

2. Clients and Servers

The client-server paradigm uses the direction of initiation to categorize whether a program is a client or server. In general, an application that initiates peer-to-peer communication is called a client. End users usually invoke client programs when they use network services.

By comparison, a server is any program that waits for incoming requests from a client program. The server receives a request from a client, performs the necessary action sand returns the result to the client.

3. Ethernet

The term Ethernet generally refers to a standard published in 1982 by Digital Equipment Corp., Intel Corp. and Xerox Corp. Ethernet is the most popular physical layer Local Area Network (LAN) technology in use today.

4. Firmware

Firmware is an embedded software program or set of instructions programmed on a device that provides the necessary instructions for how the device communicated with other computer hardware, and is located or stored in a semi-permanent storage area, e.g., ROM, EEPROM, or Flash memory. Firmware can often be updated by downloading a file from the manufacturer's web site or FTP.

5. Gateway

Computers that interconnect two networks and pass packets from one to the other are called Internet Gateways or Internet Routers. Gateways route packets that are based on the destination network, rather than the destination host.

6. ICMP (Internet Control Message Protocol)

ICMP provides a method of communicating between the Internet Protocol software on one machine and the corresponding software on another. It allows a gateway to send error or control messages to other gateways, or allows a host to diagnose problems with the network communication.

7. Internet

Physically, the Internet is a collection of packet switching networks interconnected by gateways that together with the TCP/IP protocol, allows them to perform logically as a single, large and virtual network. The Internet recognizes hosts using 32-bit IP address.

8. IP (Internet Protocol) Address

Each interface on the Internet must have a unique IP address (also called an Internet address). These addresses are 32-bit numbers, and are normally written as four decimal numbers, one for each byte of the address for example “192.168.41.1**”**. This is called dotted-decimal notation.

9. MAC (Media Access Control) Address

To allow a computer to determine which packets are meant for it, each device attached to an Ethernet network is assigned a 48-bit integer known as its MAC address (also called the Ethernet address, the hardware address or the physical address). A MAC address is normally written as eight hexadecimal numbers, for example “**00:71:88: AF: 12:3e:0f:01**”. Ethernet hardware manufacturers purchase blocks of MAC addresses and assign them in sequence as they manufacture Ethernet interface hardware. Thus, no two hardware interfaces can have the same MAC address.

10. Packet

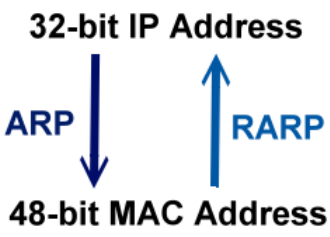
A packet is the unit of data sent across a physical network. It consists of a series of bits containing data and control information, including the source and the destination node (host) address, and is formatted for transmission from one node to another.

11. Ping

Ping is a network administration utility used to test the whether a host on an Internet network is active, and to measure the round-trip time for messages sent from the originating host to a destination computer. Ping operates by sending an ICMP echo request message to a host, expecting an ICMP echo reply to be returned. Normally, if a host cannot be pinged, Telnet or FTP cannot be used to connect to the host. Conversely, if Telnet or FTP cannot be used to connect to a host, Ping is often the starting point to determine the nature of the problem.

12. RARP (Reverse Address Resolution Protocol)

RARP provides a method of dynamically mapping 48-bit MAC address to the corresponding 32-bit IP address. RARP has now been replaced by the Bootstrap Protocol (BOOTP) and the modern Dynamic Host Configuration Protocol (DHCP).



13. Socket

Each TCP segment contains a source and destination port number that can be used to identify the sending and receiving application. These two values, along with the source and destination IP addresses in the IP header, uniquely identify each connection. The combination of an IP address and a port number is called a socket.

14. Subnet Mask

A Subnet mask, often simply called the “Mask”, is a 32-bit number that masks and IP address, and divides the IP address into the network address and the host address. Given its own IP address and its subnet mask, a host can determine whether a TCP/IP packet is destined for a host that is (1) on its own subnet, or (2) on a different network. If (1), the packet will be delivered directly; otherwise it, will be delivered via a gateway or a router.

15. TCP (Transmission Control Protocol)

TCP is a set of rules used in combination with the Internet Protocol to send data in the form of message units between computers over the Internet. TCP provides a reliable flow of data between two hosts and is associated with tasks such as dividing the data passed to it from an application into appropriately sized chunks for the network layer below, acknowledging received packets, setting timeouts to make certain that the other end acknowledges packets that are sent, and so on.

16. TCP/IP

The Transmission Control Protocol (TCP) and the Internet Protocol (IP) is standard network protocols that are almost always implemented and used together in a formation are known as TCP/IP. TCP/IP can be used to communicate across any set of interconnected networks.

17. UDP (User Datagram Protocol)

UDP is an internet protocol that provides a much simpler service to the application layer as it only sends packets of data from one host to another, but there is no guarantee that the packets will reach the destination host. UDP is suitable for purposes where error checking and correction is either not necessary or is performed in the application.

Appendix B: Actual Baud Rate Measurement

Ideal Baud Rate (bps)	Actual Baud Rate (bps)	Error
300	298.48	0.51%
600	597.04	0.49%
1200	1197.6	0.20%
2400	2395.2	0.20%
4800	4790.4	0.20%
9600	9568.0	0.33%
14400	14392	0.05%
19200	19136	0.33%
38400	38464	0.17%
57600	57552	0.08%
115200	114960	0.21%
128000	128240	0.18%
230400	229920	0.21%
250000	250000	0.00%
256000	256400	0.15%
460800	459760	0.22%
921600	921600	0.00%

Note:
Recommended max baud rate is 115200 bps or below.
Because the loading of the module, we don't guarantee a proper operation if using a larger baud rate (over 115200 bps).

Appendix C: Revision History

This chapter provides revision history information to this document.

The table below shows the revision history.

Revision	Date	Description
1.0	Jan. 2025	Initial issue
1.1	Mar. 2026	Website redesign