User Manual

GW-2439M

(BACnet/IP to Modbus TCP & RTU Gateway)

ICP DAS

Table of Contents

Important Information

Warranty Description

All products manufactured by ICP DAS are under warranty regarding defective materials for a period of one year, beginning from the date of delivery to the original purchaser.

Disclaimer

ICP DAS assumes no liability for any damage resulting from 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 Notice

Trademarks

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

Contact Us

If the user has any problems, please feel free to contact us. The user can count on us for a quick response.

mail：service@icpdas.com

Version Description

Manual Version	Firmware Version	Date	Author	Description
1.1.0	1.20	2026/01/14	Johney	1. Modify Modbus start address 2. Check BACnet connection status
1.0.0	1.00	2025/06/05	Johney	Initial release.

1. Introduction

1.1 Overview of GW-2439M

The GW-2439M is a network gateway device that acts as a BACnet/IP Client, capable of accessing BACnet/IP Server devices and converting the retrieved BACnet object data into Modbus TCP Server and Modbus RTU Slave formats for external devices to read.

The GW-2439M is primarily used to convert device data from the BACnet/IP protocol—such as AI, AO, AV, BI, BO, BV, MSI, MSO, and MSV BACnet objects—into the Modbus register format, allowing Modbus-compatible monitoring systems to directly access information from BACnet devices. Users can configure the module and map objects using the utility tool provided by ICP DAS.

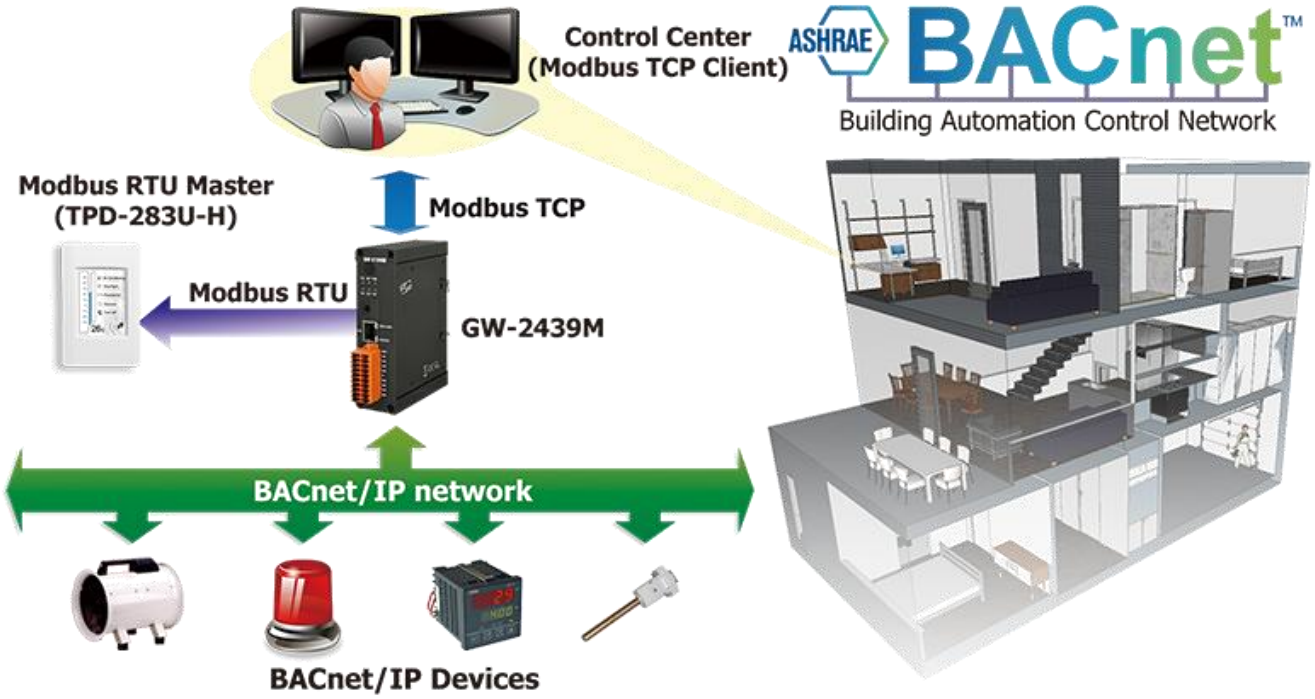
The GW-2439M supports essential BACnet interoperability services including:

- DS-RP-A (Read Property)
- DS-RPM-A (Read Property Multiple)
- DS-WP-A (Write Property)
- DS-WPM-A (Write Property Multiple)

These functions offer flexible data access and control, making the GW-2439M a critical bridge between BACnet/IP and Modbus systems.

1.2 Application Architecture

The GW-2439M can be widely applied in building automation systems, enabling seamless integration of HVAC and lighting control systems through BACnet/IP and Modbus protocol conversion. Whether it's temperature monitoring, fan coil unit control, or centralized lighting switch management, this device allows integration of heterogeneous communication protocol devices into a unified platform, enhancing overall control efficiency and management convenience.



1.3 Features

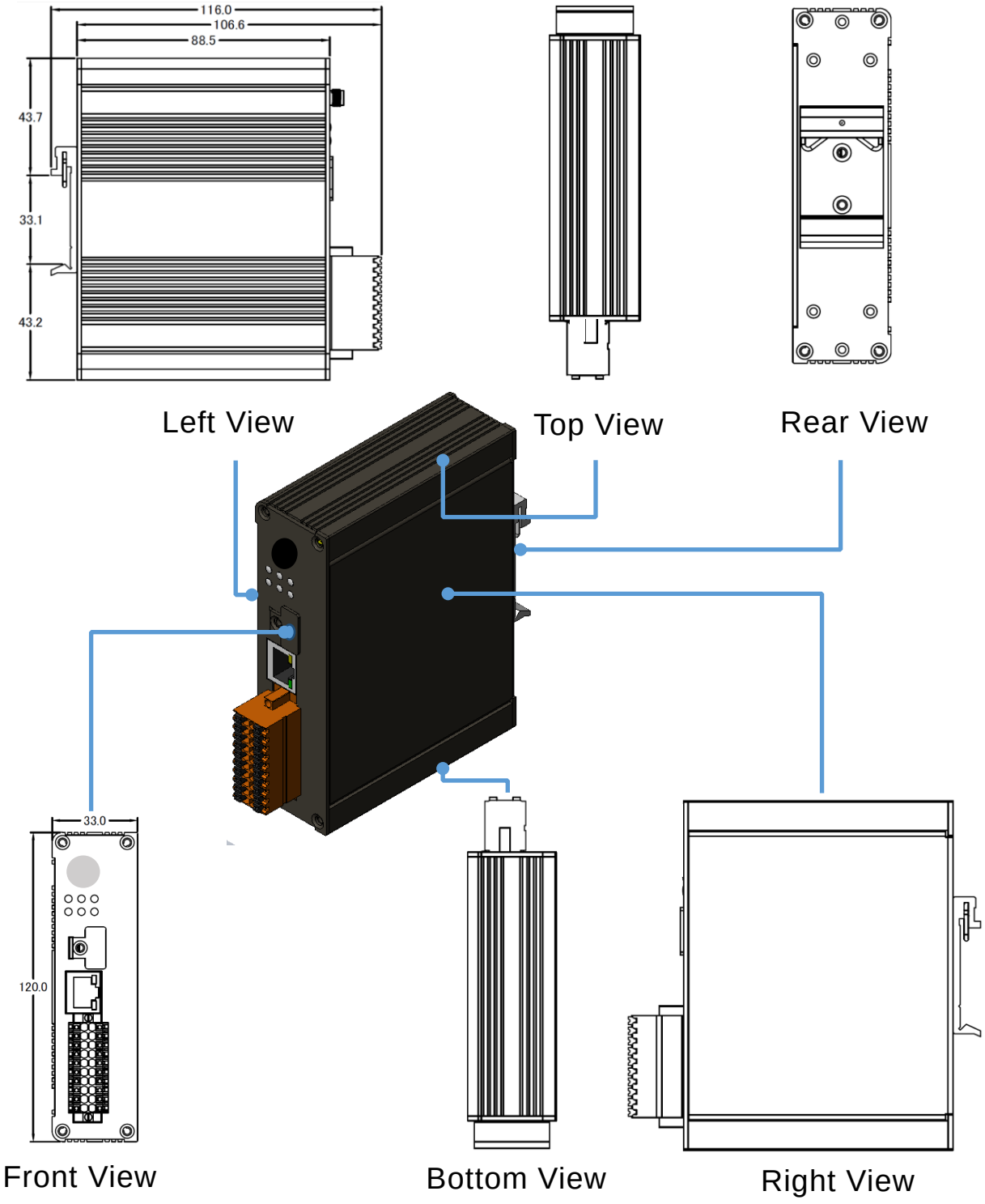
- Supports configuration of up to 10 BACnet/IP Clients.
- Simultaneously supports Modbus TCP Server and Modbus RTU Slave.
- Supports object types: AI, AO, AV, BI, BO, BV, MSI, MSO, MSV.
- Utility software supports BACnet/IP Server discovery and object mapping.
- Supports conversion of up to 9000 BACnet objects to Modbus registers.
- Auto-generates Modbus conversion tables based on BACnet object configuration.

1.4 Specifications

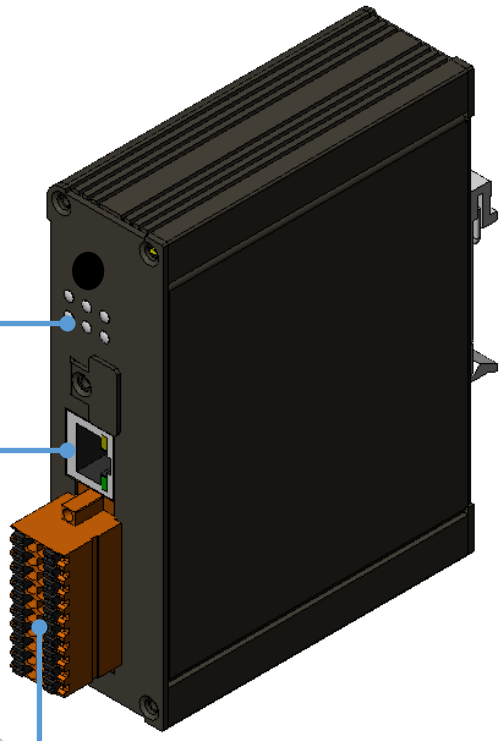
Ethernet	
Controller	10/100Base-TX Ethernet (Auto-negotiating, Auto_MDIX)
Connector	RJ-45 with Ethernet indictor
Protocol	
Modbus	Modbus TCP Server and Modbus RTU Slave (up to 2 TCP connections)
Function Code	0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x10, 0x0F
BACnet	BACnet/IP Client (Support up to 10 BACnet/IP Servers)
BACnet Objects	AI, AO, AV, BI, BO, BV, MSI, MSO, MSV
BIBBs	BIBBs DS-RP-A, DS-RPM-A, DS-WP-A, DS-WPM-A, DM-DDB-A, DM-DOB-A, DM-D CC-A, DM-RD-A
Power	
Protection	Power reverse polarity protection
EMS Protection	ESD, Surge, EFT
Supply Voltage	+10 VDC ~ +30 VDC
Power Consum.	5 W @ 24 VDC
LED Indicator	
LED (Round)	Power (1), BACnet/IP Status (1), BACnet/IP Net(1), Modbus TCP TxD / RxD / Link (3)
Mechanism	
Installation	DIN-Rail
Casing	Metal
Dimensions	33 x 120 x 116 mm (W x L x H)
Environment	
Operating Temp.	-25℃ ~ +75℃
Storage Temp.	-30℃ ~ +80℃
Humidity	10 ~ 90% RH, non-condensing

2. Appearance and Dimensions of the GW-2439M

2.1 Dimensions(Unit : mm)



2.2 Front Panel

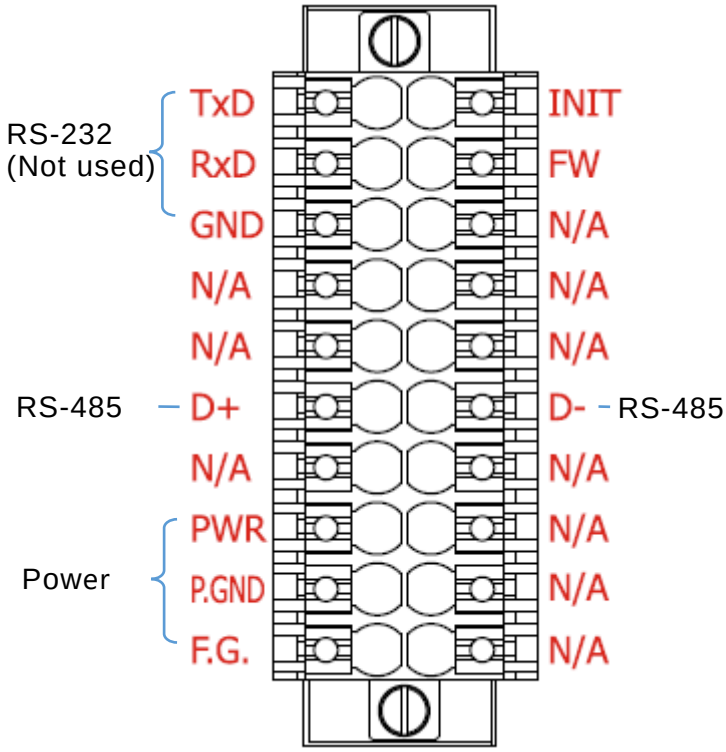


LED Indicators

Ethernet Port

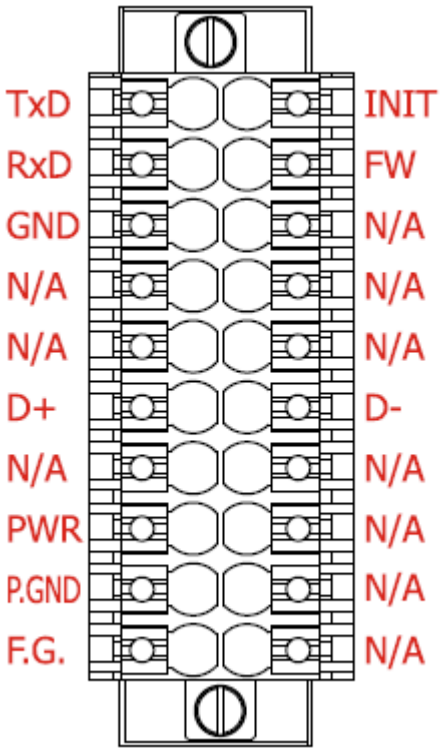
The GW-2439M provides an RJ-45 Ethernet port supporting 10/100 Mbps bandwidth. When operating in 100BASE-TX mode, the bottom-right LED will light orange. When operating in 10BASE-T mode, the LED will turn off. The bottom-left green LED blinks when a network connection is detected and packets are being received.

Terminal Block



2.3 Pin Assignment

The following are the definitions of the 20-pin terminal block:



Pin Name	Function Description
TxD	Not used
RxD	Not used
GND	Signal Ground
D+	RS-485 D+
D-	RS-485 D-
INIT	Used to restore factory settings
FW	Used for firmware update
PWR	Power Input (+10 VDC to +30 VDC)
P.GND	Power Ground
F.G.	Frame Ground (Chassis Ground)

2.4 LED Indicators

There are six LED indicators on the GW-2439M, showing different module statuses.

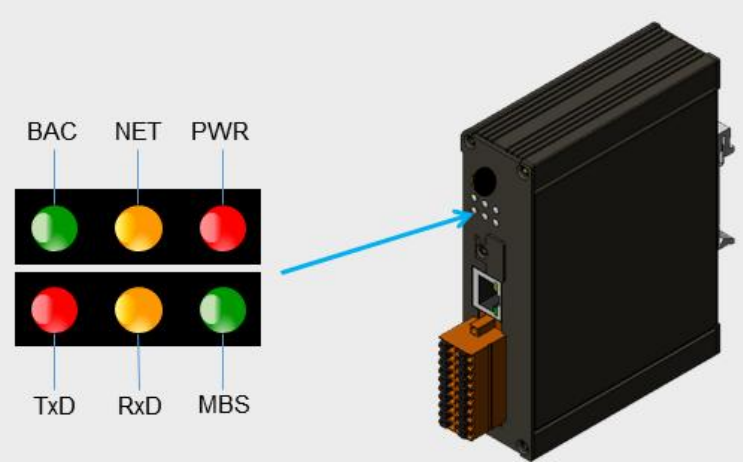


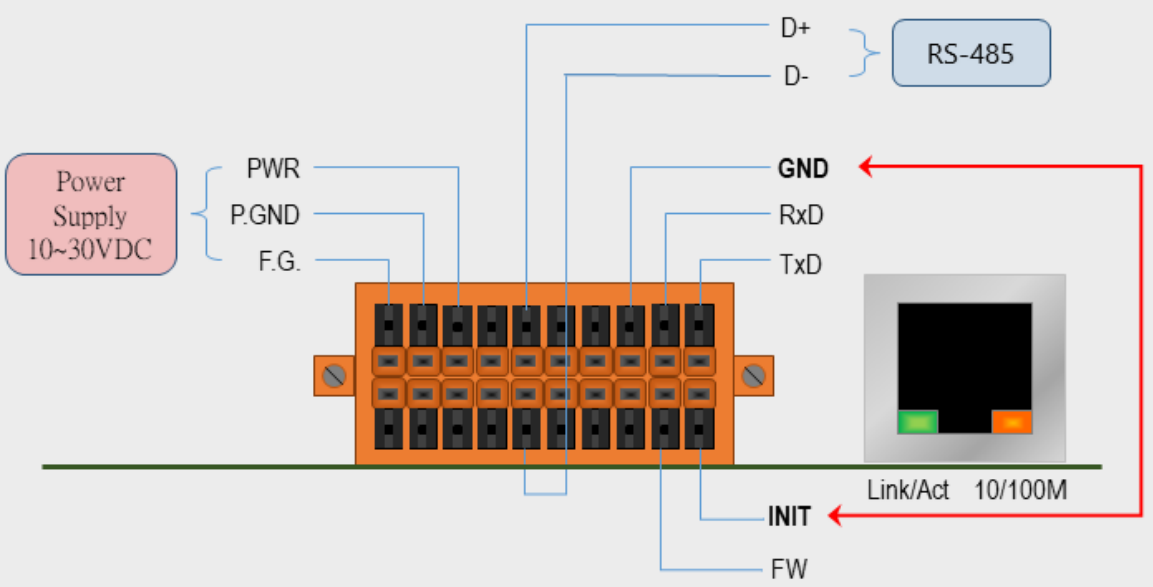
Figure 2.1 Six LEDs on the GW-2439M

LED Name	LED Status	GW-2439M Status
All LEDs	Flashing in sequence	Firmware update mode
	Flash every 500ms	Initialization mode or factory default mode
PWR (Power)	Solid ON	Power is normal
	Off	Power failure or module fault
NET (Ethernet)	Solid ON	Ethernet cable is connected
	Off	Ethernet cable is not connected
	Flash every 200ms	IP address conflict detected
BAC (BACnet/IP)	Solid ON	All BACnet Servers are online
	Off	No BACnet Servers are configured
	Flash every 200ms	At least one BACnet Server is offline
MBS (Modbus Server)	Solid ON	At least one Modbus Client is connected
	Off	No Modbus Clients connected
	Flash every 200ms	Max. Modbus connections (2) reached
RxD (Modbus)	Off	Not getting any Modbus packets
	Flash every 200ms	Receiving Modbus packets
TxD (Modbus)	Off	Not sending any Modbus packets.
	Flash every 200ms	Sending Modbus response packets

2.5 Factory Reset and【INIT】Pin Description

The【INIT】pin is used to reset all parameters to factory defaults (e.g., IP, subnet mask, BACnet object configuration). Follow these steps:

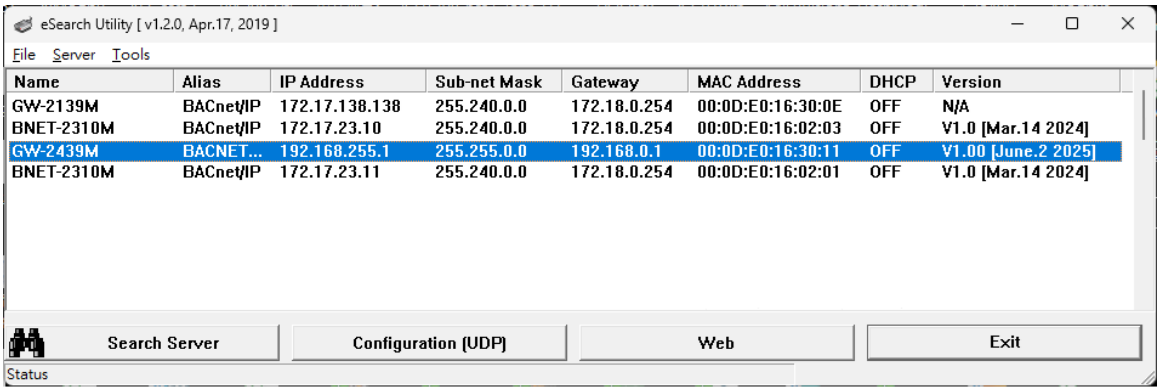
1. Power off the module while keeping the RJ-45 Ethernet cable connected.
2. Connect the【INIT】pin to the【GND】pin. As shown below.



3. Power on the module. All six LEDs will blink simultaneously, indicating that the factory reset was successful.

GW-2439M Factory Default Settings			
Network	IP	192.168.255.1	Parameters can be changed to meet your application.
	Mask	255.255.0.0	
	Gateway	192.168.0.1	
Login	Account	ADMIN	Modify after logging in.
	Password	ADMIN	

4. After reset, use the eSearch tool to find the GW-2439M and change the IP/mask/gateway as needed.



When the modification is complete, press the OK button, then the modification is complete.

5. Disconnect【INIT】from【GND】and restart the module.

2.6 Firmware Update and【FW】Pin Description

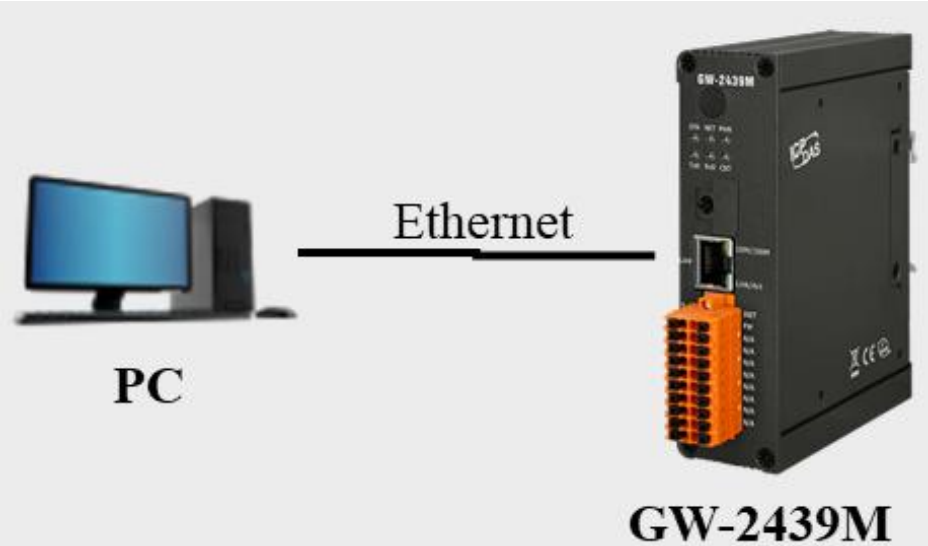
The【FW】pin is primarily used to update the firmware of the GW-2439M module. If a new version of the firmware is released on the official website, users can update the module themselves to gain access to the latest product features.

To begin, the GW-2439M supports firmware updates via a software tool provided by ICP DAS (Windows version) and a standard network cable. Please download the latest firmware and the update utility (FW_Update_Tool) from the GW-2439M product webpage, and save them to your computer. Download link for the update tool:

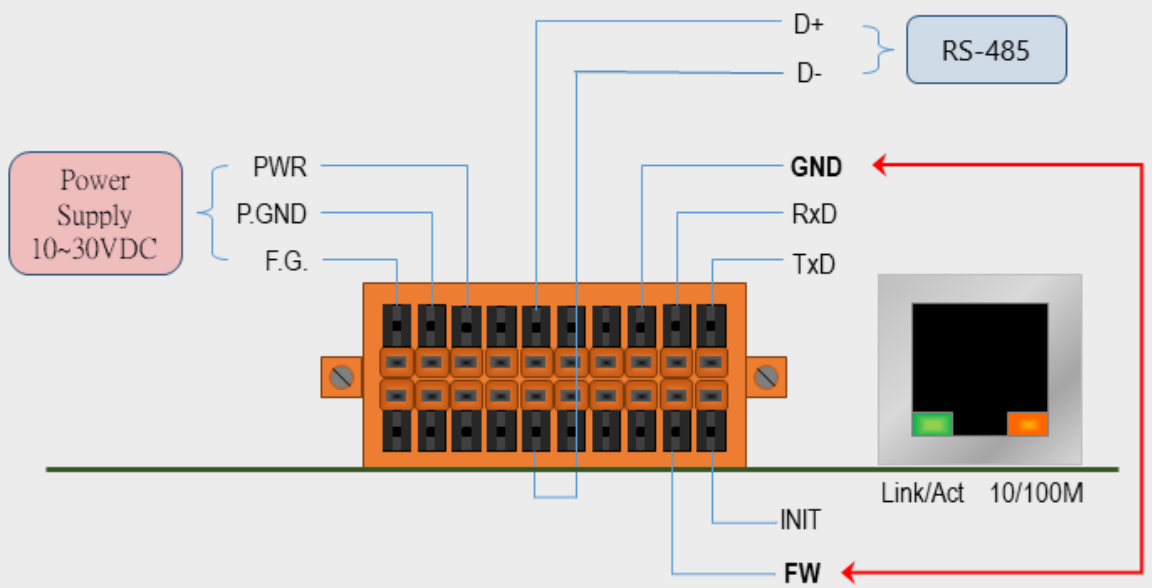
https://www.icpdas.com/en/download/show.php?num=9351

The operation steps are as follows:

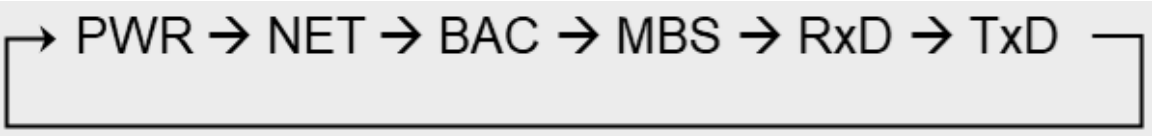
1. Connect a standard RJ-45 Ethernet cable directly between the GW-2439M and the computer's network port, then turn off the module's power. See illustration below.



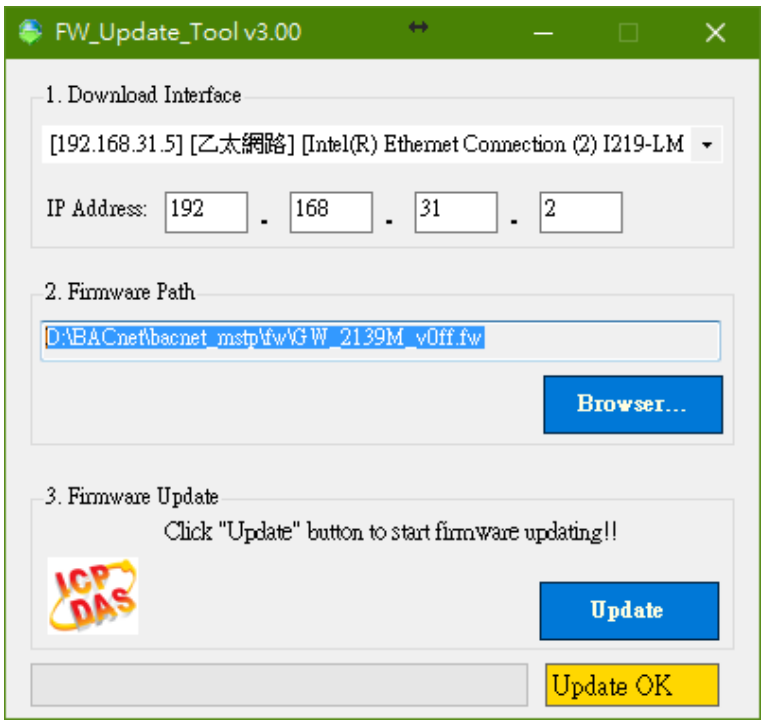
2. Connect the【FW】pin of the GW-2439M to the【GND】signal pin, as shown in the illustration.



3. Power on the GW-2439M module. Once it enters firmware update mode, all six LEDs will blink in sequence to indicate the module is ready for firmware flashing. See illustration below.



4. Launch the FW_Update_Tool.exe utility, as shown in the illustration below.



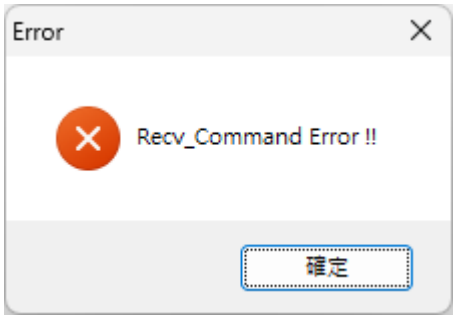
Follow the steps below in sequence:

(1). In the【Download Interface】dropdown menu, select the network port connected to the GW-2439M.

(2).The【IP Address】field is not used—no configuration is required.

(3).In the【Firmware Path】field, select the latest firmware file (e.g., GW_2439M_vxxx.fw).

(4).Click the【Firmware Update】button to start the firmware writing process. Please try to click this button within 10 seconds after the GW-2439M enters firmware update mode. If the dialog box shown below appears, power-cycle the GW-2439M and click the Update button again.



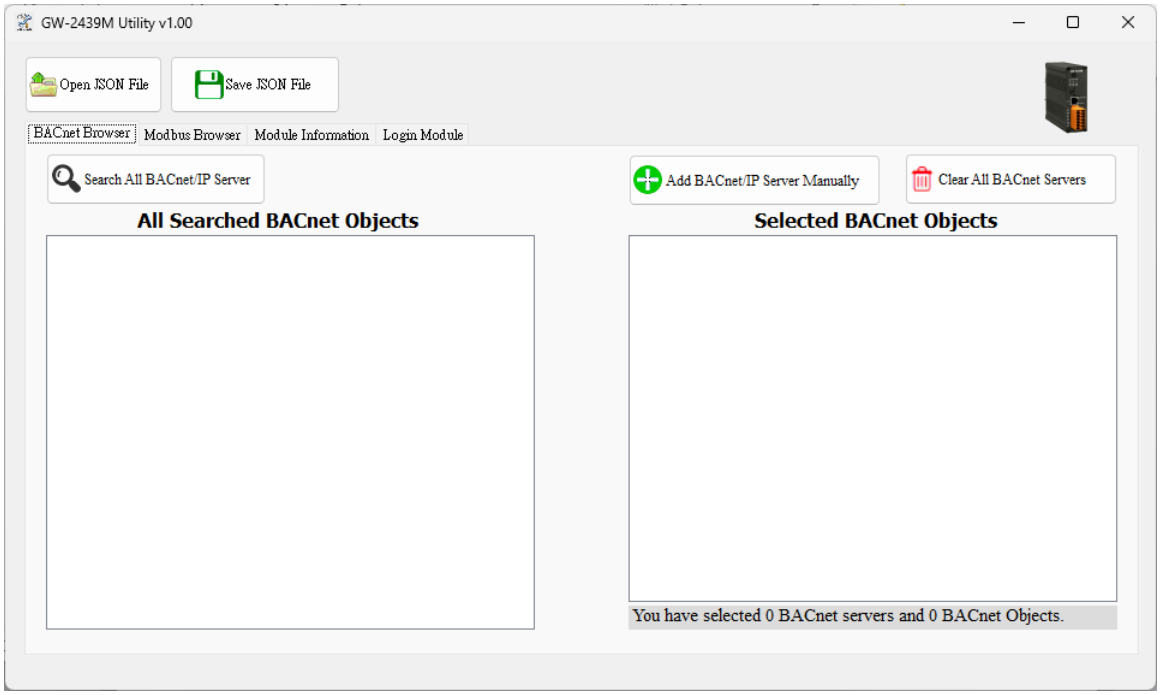
(5). After the firmware update has been completed, the FW_Update_Tool will display “Update OK”, indicating that the flashing process has finished successfully.At this point, please disconnect the connection between the【FW】and【GND】pins, then restart the power to the module. After rebooting, the user can use either eSearch or the GW-2439M Utility to verify the firmware version.

3. GW-2439M Software Utility Introduction

GW-2439M_Utility is a user-friendly configuration tool specifically designed to assist users during the setup process. It offers multiple convenient features that significantly simplify the integration of BACnet and Modbus systems. Its core functionality is the ability to automatically search for BACnet/IP Server devices, eliminating the need to manually input objects one by one. This reduces human error and saves configuration time.

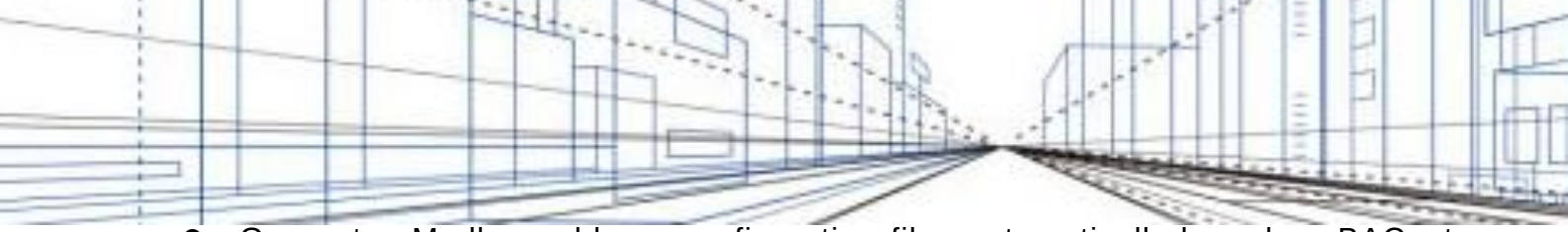
Additionally, based on the discovered BACnet objects, the tool can automatically generate a corresponding Modbus address mapping file. Users can export this address table separately for easy integration and reference in other Modbus management systems.

GW-2439M_Utility also supports saving and importing configuration files, allowing users to quickly apply existing settings to other GW-2439M modules. This enhances deployment efficiency and improves overall system management.



Features :

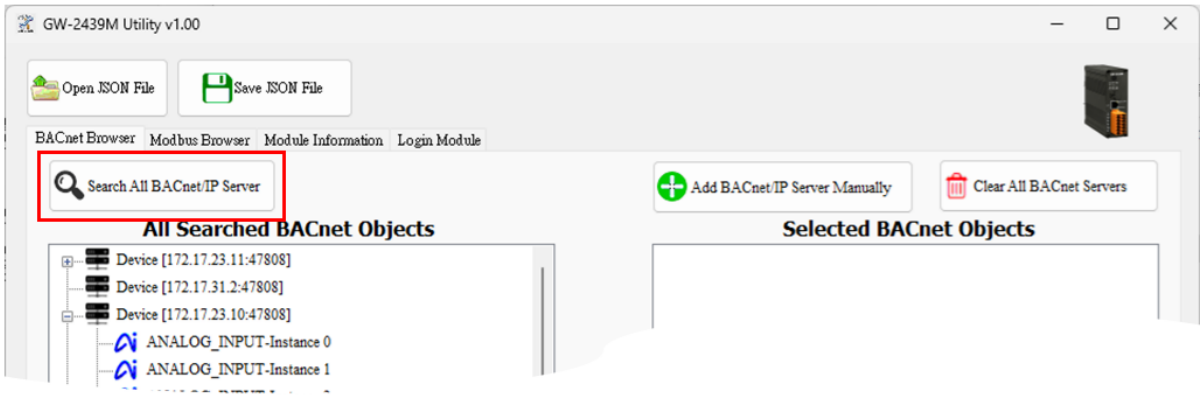
- Automatically searches for BACnet/IP Servers, eliminating the tedious process of manually entering objects.



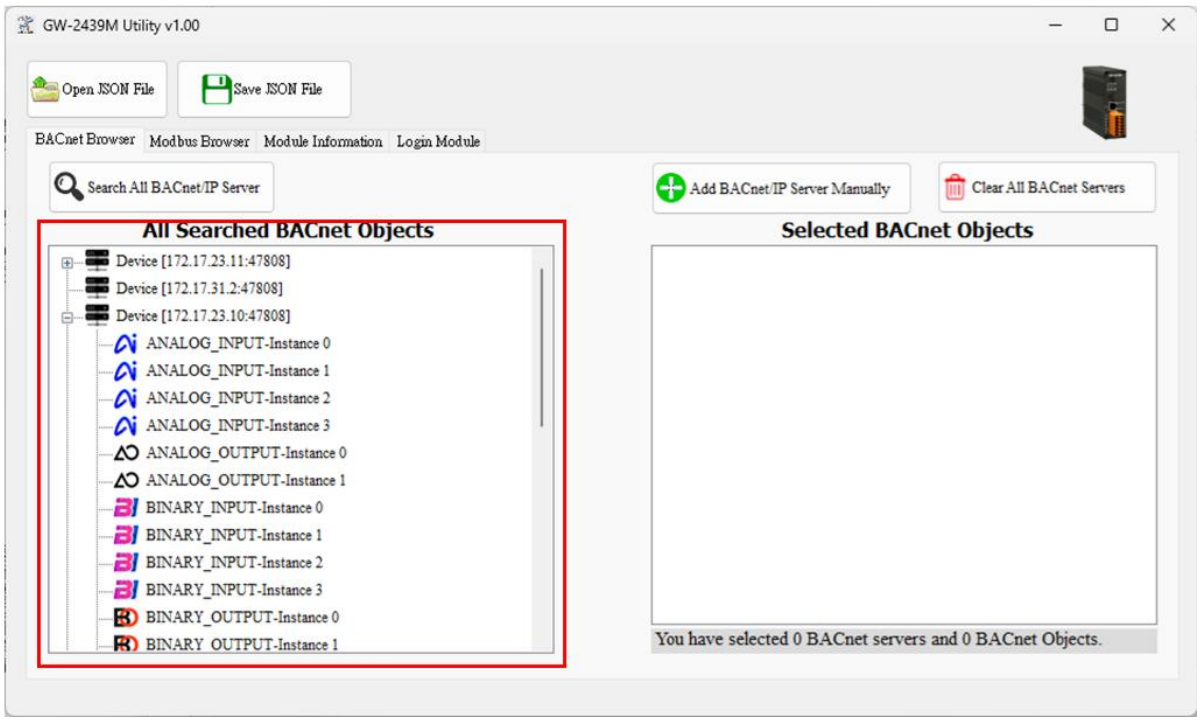
- Generates Modbus address configuration files automatically based on BACnet objects, saving configuration time.
- Allows independent export of Modbus address tables for use in other systems.
- Supports exporting and importing configuration files, making it easy to apply settings to other GW-2439M devices.

3.1 Search BACnet/IP Server

GW-2439M_Utility provides a convenient search function that allows users to automatically scan and detect remote BACnet/IP Servers over the Ethernet network. This feature quickly lists all available BACnet/IP Servers and their associated objects, helping users manage and configure devices more efficiently. See illustration below.



To use this feature, simply connect the computer to the same local area network (LAN) as the BACnet/IP Servers, then click the【Search All BACnet/IP Server】 button to begin scanning. The system will automatically search for all servers on the network and display them in a tree structure, showing each server and its associated objects for easy browsing and management, as shown in the illustration below.

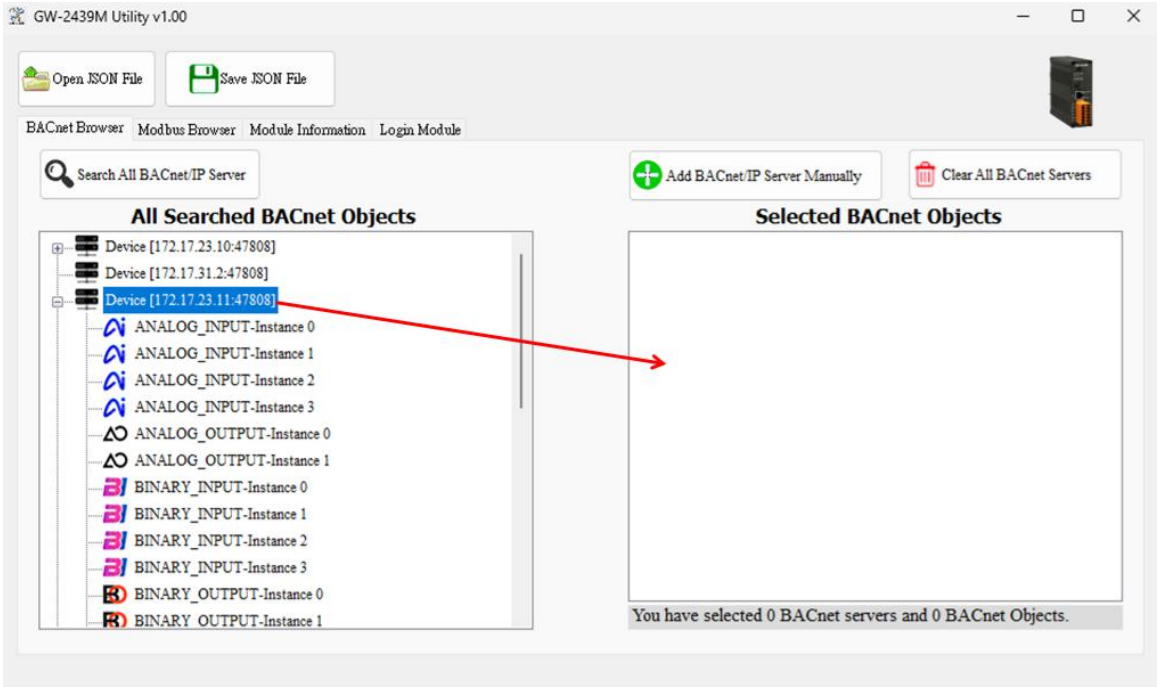


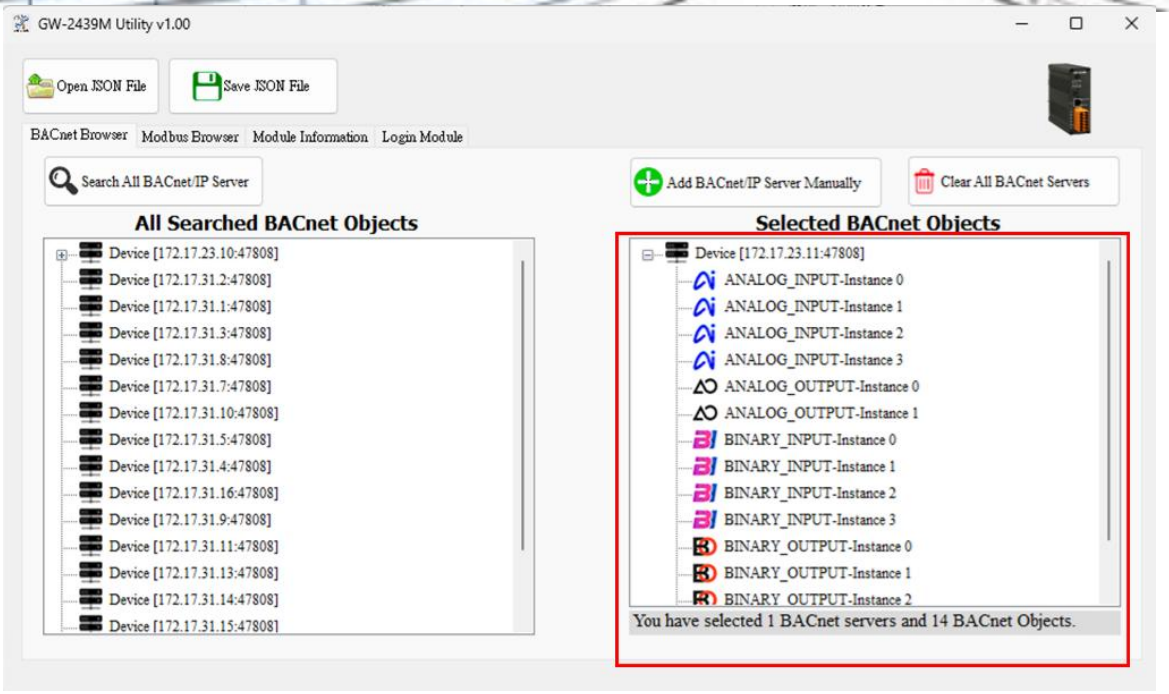
In the GW-2439M_Utility interface, the【All Searched BACnet Objects】panel on the left displays all the BACnet/IP Servers and their associated objects discovered during the search. Each BACnet/IP Server is listed with a “Device” label, followed by its IP address and UDP port in parentheses (e.g., [172.17.23.10:47808]).

Under each Device node, the various BACnet objects contained in that server—such as Analog Input, Binary Output, etc.—are listed and categorized accordingly.

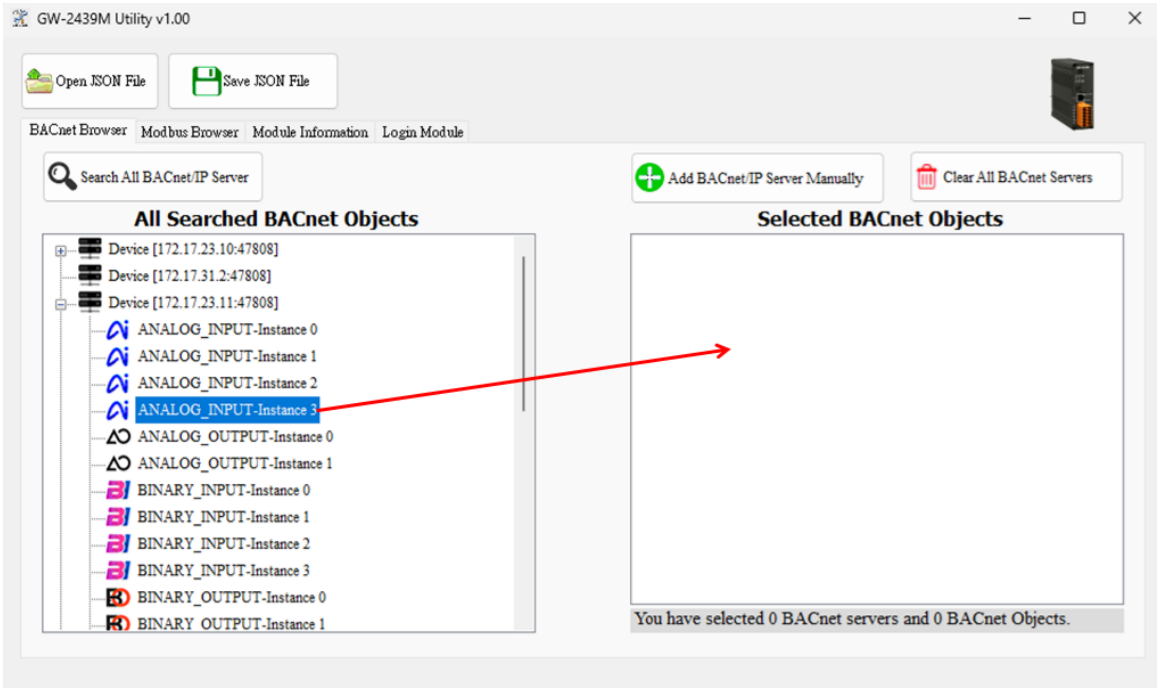
3.2 Configuring BACnet/IP Servers and Objects

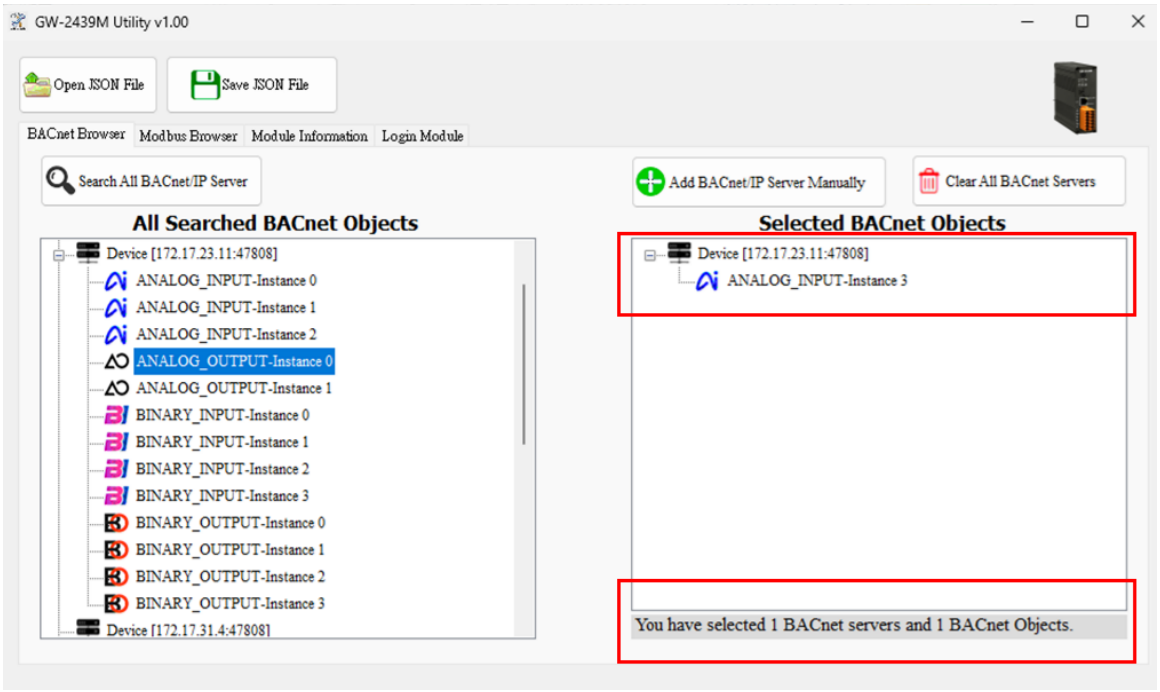
After completing the server search, the 【All Searched BACnet Objects】 panel on the left side of the GW-2439M_Utility interface will display all discovered BACnet/IP Servers and their associated objects. Users can use drag-and-drop operations with the mouse to move either an entire BACnet/IP Server or individual BACnet objects from the【All Searched BACnet Objects】section. By dragging a Device node (representing a BACnet/IP Server) into the【Selected BACnet Objects】panel on the right, users can quickly select and configure the desired objects for integration. See illustration below.





Alternatively, you can drag and drop an individual object node into the 【Selected BACnet Objects】 panel on the right to complete the selection and configuration of that specific object. See illustration below.



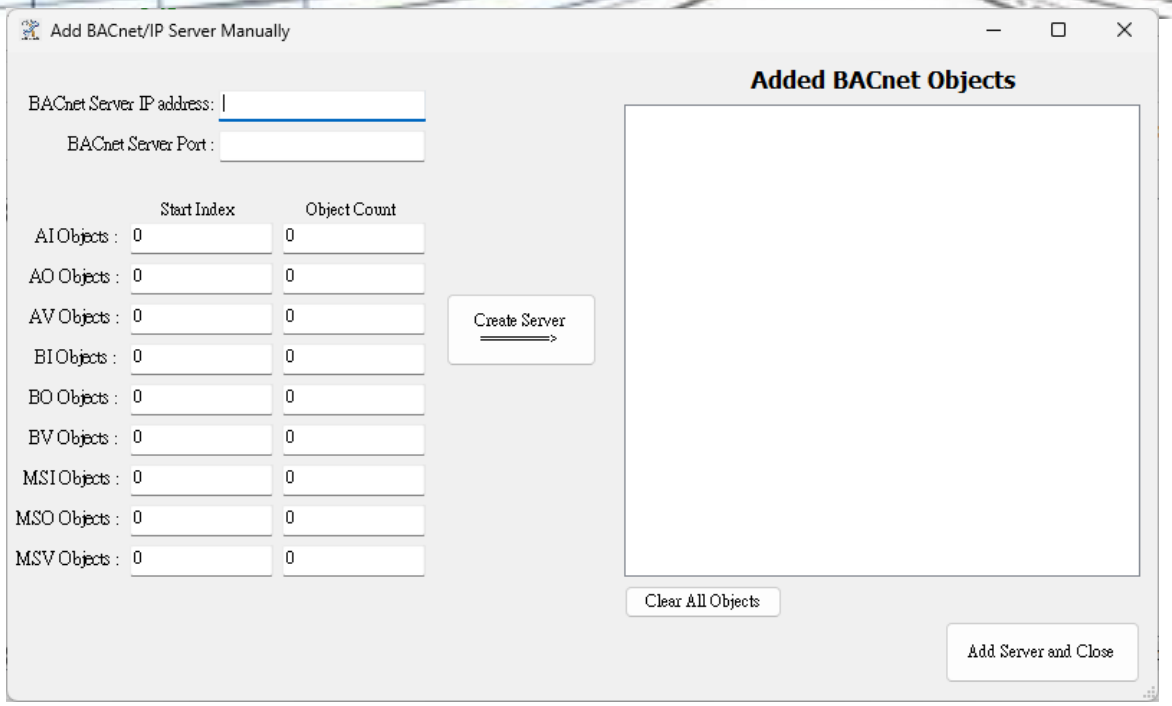


After each object has been dragged and dropped, a summary of the selected items will appear in the lower-right corner of the screen. This includes the number of selected BACnet/IP Servers and the total number of selected BACnet objects.

Manually Adding BACnet/IP Servers and Objects

f the user is unable to automatically discover BACnet/IP Servers over the network (e.g., if they are not on the same local network), the utility provides a manual add function to create a BACnet/IP Server and define its corresponding BACnet objects. Please follow the steps below:

(1). Click the【Add BACnet/IP Server Manually】button. A dialog box will appear, as shown in the figure below.



(2). Enter the IP address and port of the BACnet/IP Server.In the [BACnet Server IP address] field, input the IP address of the device. In the [BACnet Server Port] field, enter its UDP port number (default is 47808; if uncertain, please consult the device provider).

(3). Configure the number and starting index of each BACnet object type. In the object list on the left (AI, AO, AV, BI, BO, BV, MSI, MSO, MSV), input the [Start Index] and [Object Count] for each type. The dialog box will generate a continuous range of objects based on the provided quantity.

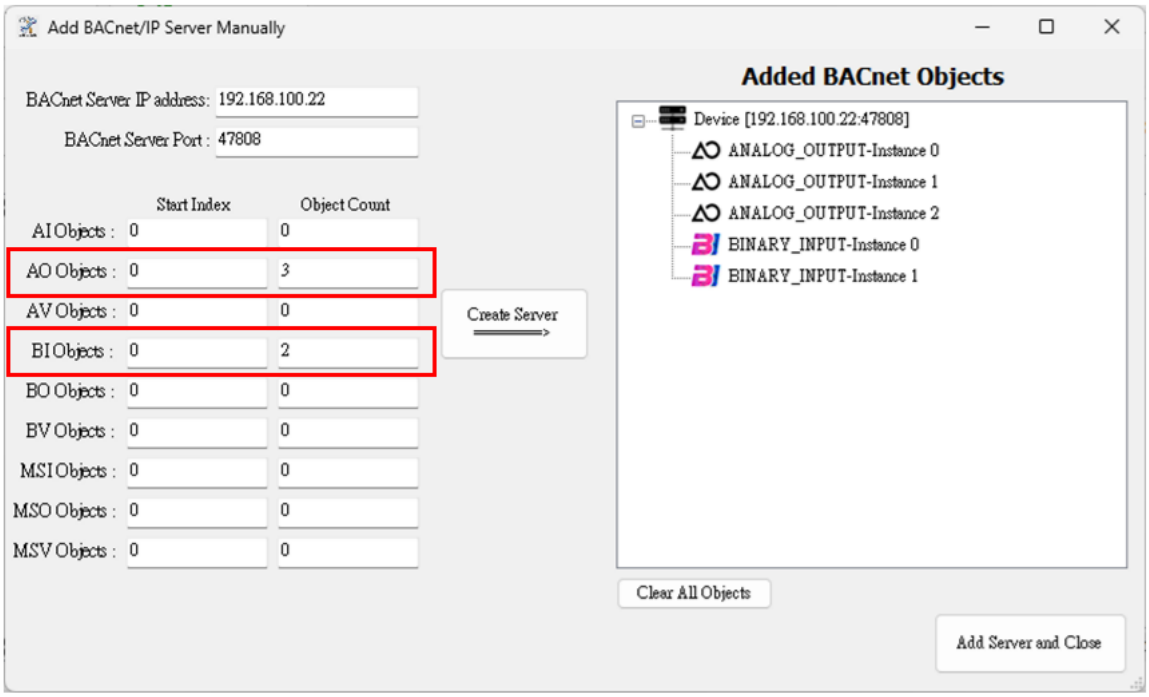
(4). Create the BACnet/IP Server.

After entering all required information, click the 【Create Server】 button.

The system will display the configured data in the【Added BACnet Objects】 panel on the right for user confirmation.

Below are some input configuration samples:

[Example 1]



BACnet Server IP is 192.168.100.22, and the Port is the default value 47808.

In the AO Objects (Analog Output) field, the [Start Index] is set to 0 and the [Object Count] to 3, so the system automatically creates three consecutive AO objects:

ANALOG_OUTPUT - Instance 0

ANALOG_OUTPUT - Instance 1

ANALOG_OUTPUT - Instance 2

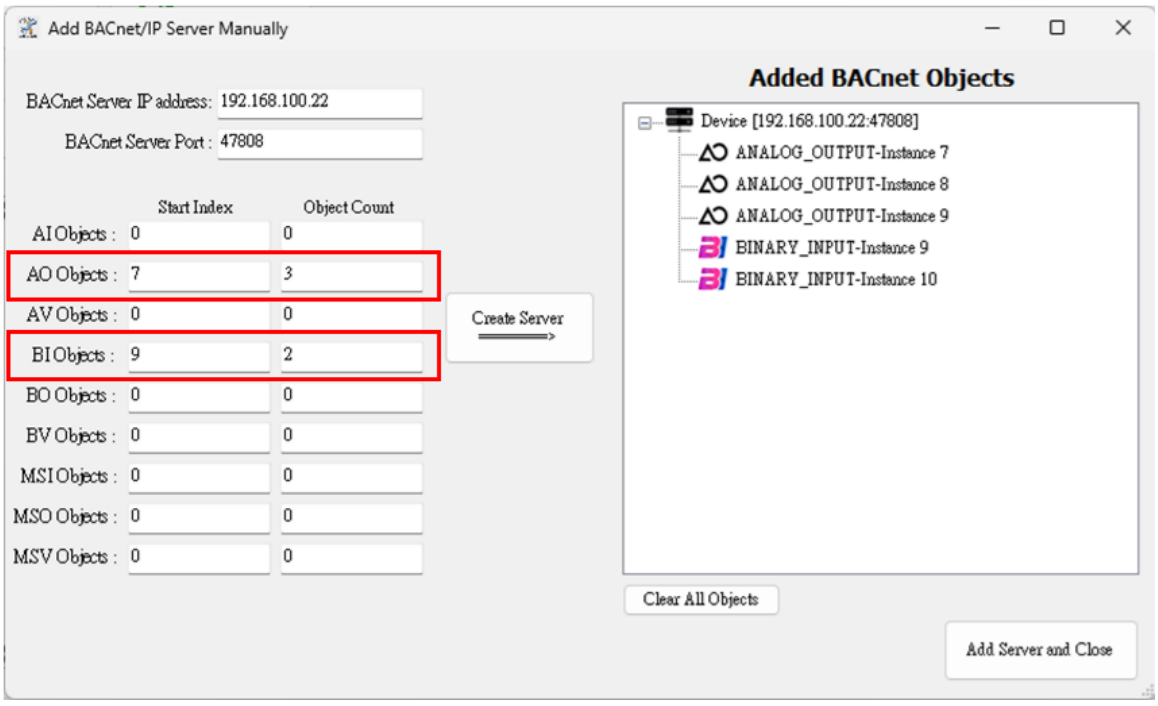
In the BI Objects (Binary Input) field, the [Start Index] is set to 0 and the [Object Count] to 2, resulting in the creation of two BI objects: :

BINARY_INPUT - Instance 0

BINARY_INPUT - Instance 1

These objects are displayed in a tree structure on the right side, mounted under the corresponding Device node (in this example: Device [192.168.100.22:47808]). Users can add multiple BACnet/IP Servers. When done, click the【Add Server and Close】button to insert the configuration into the Selected BACnet Objects area and close the dialog box.

[Example 2]



The entered BACnet Server IP is 192.168.100.22, and the Port is set to the default

value 47808.

In the AO Objects (Analog Output) field, the [Start Index] is set to 7 and the [Object Count] to 3, so the system automatically creates three consecutive AO objects：

ANALOG_OUTPUT - Instance 7

ANALOG_OUTPUT - Instance 8

ANALOG_OUTPUT - Instance 9

In the BI Objects (Binary Input) field, the [Start Index] is set to 9 and the [Object Count] to 2, resulting in the creation of two BI objects:

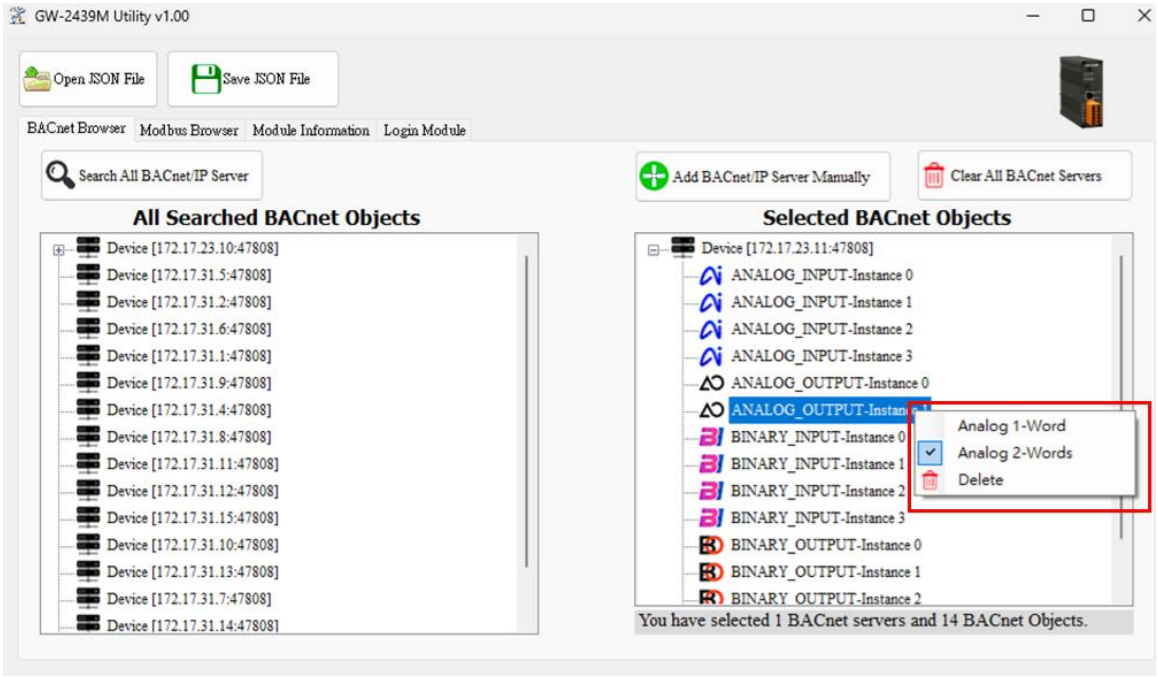
BINARY_INPUT - Instance 9

BINARY_INPUT - Instance 10

These objects will be displayed in a tree structure on the right side, mounted under the corresponding Device node (in this example: Device [192.168.100.22:47808]). Users can add multiple BACnet/IP Servers, and once completed, click the 【Add Server and Close】 button to insert them into the Selected BACnet Objects area and close the dialog box.

3.3 Modbus Register Length for BACnet Objects

In GW-2439M Utility, once BACnet objects are added to the【Selected BACnet Objects】panel on the right, their data must be mapped to Modbus registers. Users can specify the number of Modbus register addresses each object occupies according to actual application needs. After selecting the desired object, right-click to open the context menu, as shown in the figure below.



This context menu is available only for BACnet objects of the Analog and Multi-State categories. The six supported object types are:

Analog_Input、Analog_Output、Analog_Value、

Multi-State_Input、Multi-State _Output、Multi-State _Value。

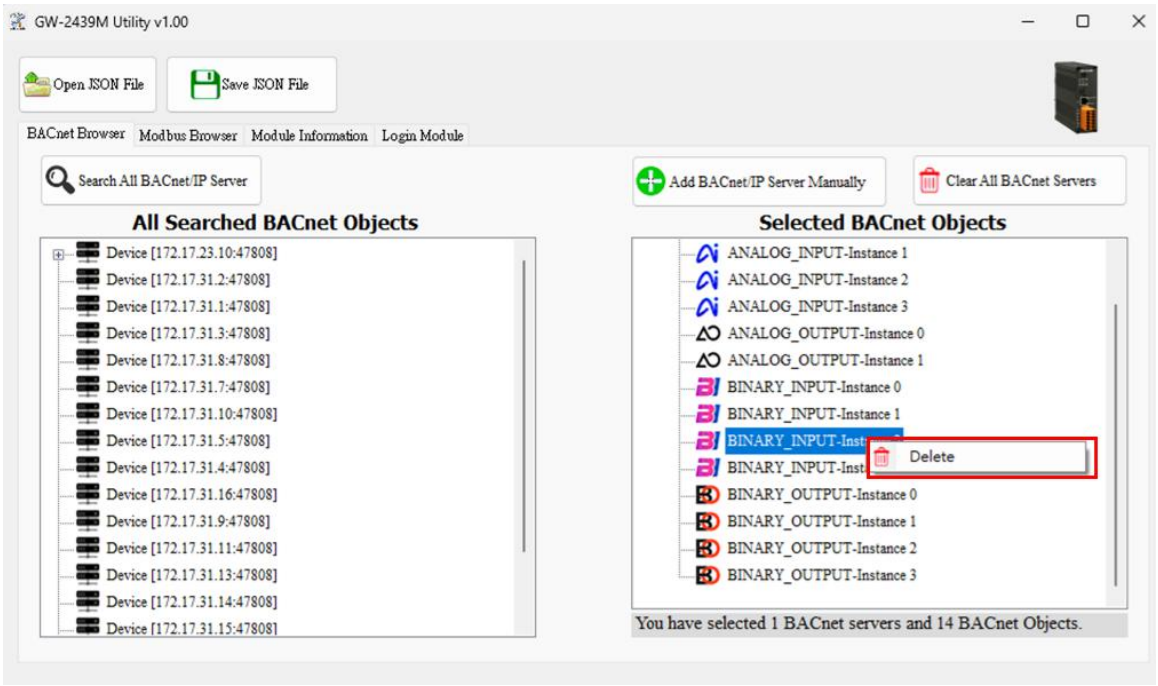
The options in the right-click context menu are as follows:

- Analog 1-Word：Assigns the object to occupy one 16-bit Modbus register address.

- Analog 2-Words (Default): Assigns the object to occupy two consecutive 16-bit Modbus register addresses (means 32-bit).
- Delete : Removes the selected object from the list.

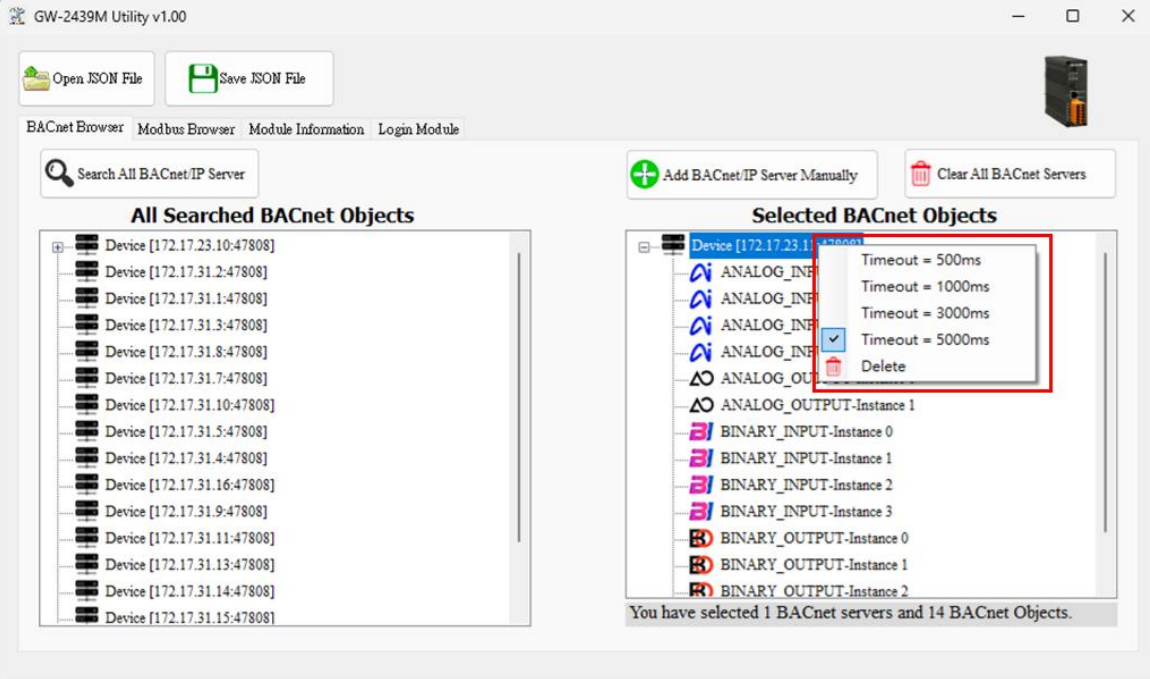
This setting directly affects the [Start Address] configuration in the Modbus mapping table. If [2-Words] is selected, the object will occupy two addresses in the Modbus register map, and the start address of subsequent objects will automatically shift forward. If [1-Word] is selected, the object will occupy only one address.

For BINARY_INPUT, BINARY_OUTPUT, and BINARY_VALUE objects, the right-click context menu will only show the [Delete] item, as shown in the figure.



3.4 Setting the Timeout for a BACnet Server

In GW-2439M Utility, once BACnet objects have been added to the【Selected BACnet Objects】panel on the right, how can users configure the communication timeout for a specific BACnet Server? This timeout defines the maximum amount of time that the GW-2439M will wait for a response from the BACnet Server during communication.To set this, select the desired BACnet Server, then right-click to open the context menu, as shown in the figure.



The options in the right-click context menu are as follows:

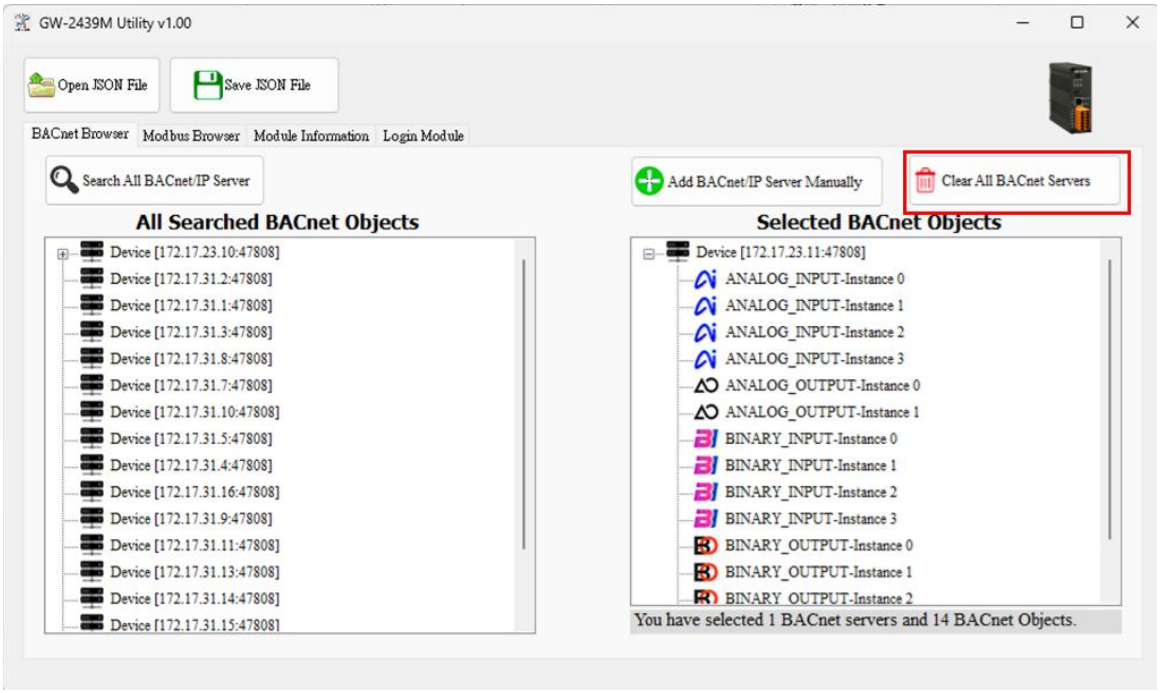
- Timeout = 500ms：If the server fails to respond within 500 milliseconds, it will be considered disconnected.
- Timeout = 1000ms：If the server fails to respond within 1000 milliseconds, it will be considered disconnected.
- Timeout = 3000ms：If the server fails to respond within 3000 milliseconds, it will

be considered disconnected.

- Timeout = 5000ms：If the server fails to respond within 5000 milliseconds, it will be considered disconnected.
- Delete：Removes the selected server and all of its associated objects.

3.5 Delete All BACnet Servers and Their Associated Objects

In the upper-right corner of the GW-2439M Utility interface, there is a【Clear All BACnet Servers】 button, as shown in the figure below. Clicking this button will remove all BACnet Servers and their associated objects from the【Selected BACnet Objects】panel on the right.



3.6 Viewing the Modbus Browser Page

After the user has finished selecting BACnet Servers—meaning the【Selected BACnet Objects】panel on the right side of the GW-2439M_Utility interface contains all the desired BACnet Servers and their associated objects—the user can switch to the [Modbus Browser] page.

The GW-2439M will automatically map the BACnet objects to Modbus register types according to the following table.

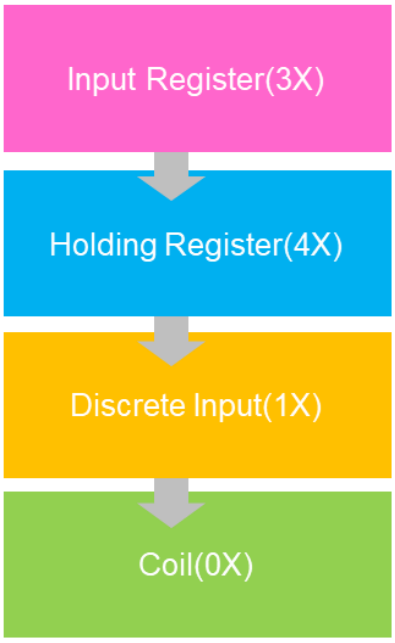
No.	BACnet Object	Modbus Register
1	ANALOG_INPUT	Input register(3X)
2	ANALOG_OUTPUT	Holding register(4X)
3	ANALOG_VALUE	Holding register(4X)
4	BINARY_INPUT	Discrete Input(1X)
5	BINARY_OUTPUT	Coil(0X)
6	BINARY_VALUE	Coil(0X)
7	MULTI_STATE_INPUT	Input register(3X)
8	MULTI_STATE _OUTPUT	Holding register(4X)
9	MULTI_STATE _VALUE	Holding register(4X)

GW-2439M_Utility reorganizes all the objects retrieved from the BACnet Server according to a specific set of rules in order to generate a complete Modbus register

mapping table. If users are interested in understanding how the entire sorting process works, the explanation below outlines the three-tier sorting procedure.

Level 1: Classification by Modbus Register Type

The system first categorizes BACnet objects based on their corresponding Modbus register type, in the following order:



Level 2: Sorting by BACnet Object Type

Within each Modbus register type category, the system further sorts the entries based on the BACnet object type, as illustrated in the figure below.

No.	Modbus Register	BACnet Object Types and Sorting Order
1	Input register(3X)	ANALOG_INPUT
		MULTI_STATE_INPUT

2	Holding register(4X)	ANALOG_OUTPUT
		ANALOG_VALUE
		MULTI_STATE _OUTPUT
		MULTI_STATE _VALUE
3	Discrete Input(1X)	BINARY_INPUT
4	Coil(0X)	BINARY_OUTPUT
		BINARY_VALUE

Level 3: Further Sorting by Server Information

For objects of the same type, an additional sorting step is applied based on their originating BACnet Server, in the following order:

(1) Sorted by Server IP address in ascending order

(2) If the IP addresses are the same, then sorted by object name (e.g., AO1, AO2, etc.)

For example, the following order：

ANALOG_OUTPUT1 of Server #1 (IP#1)

ANALOG_OUTPUT2 of Server #1 (IP#1)

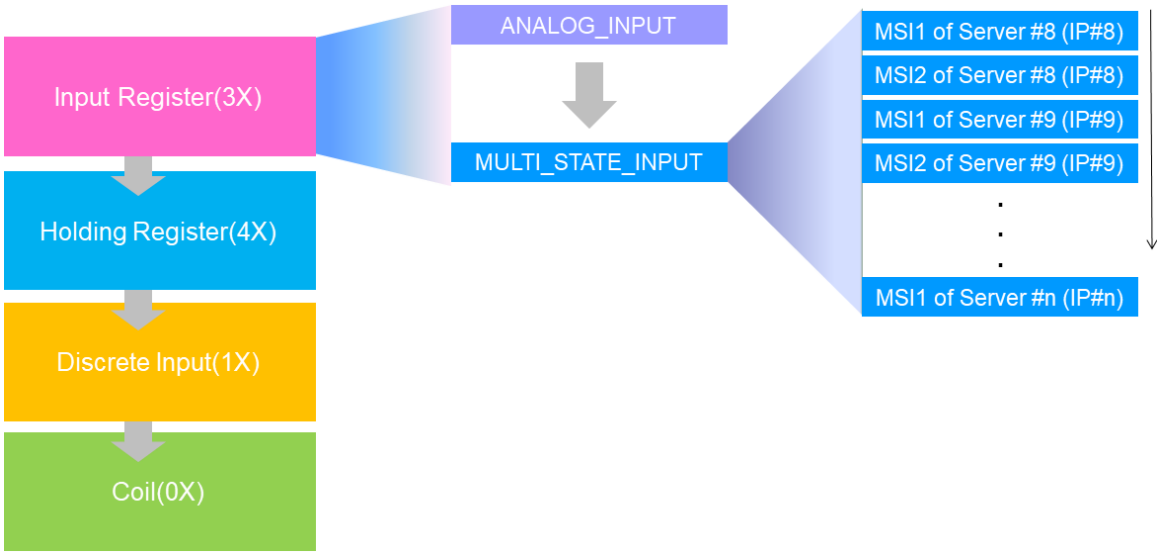
ANALOG_OUTPUT1 of Server #2 (IP#2)

.

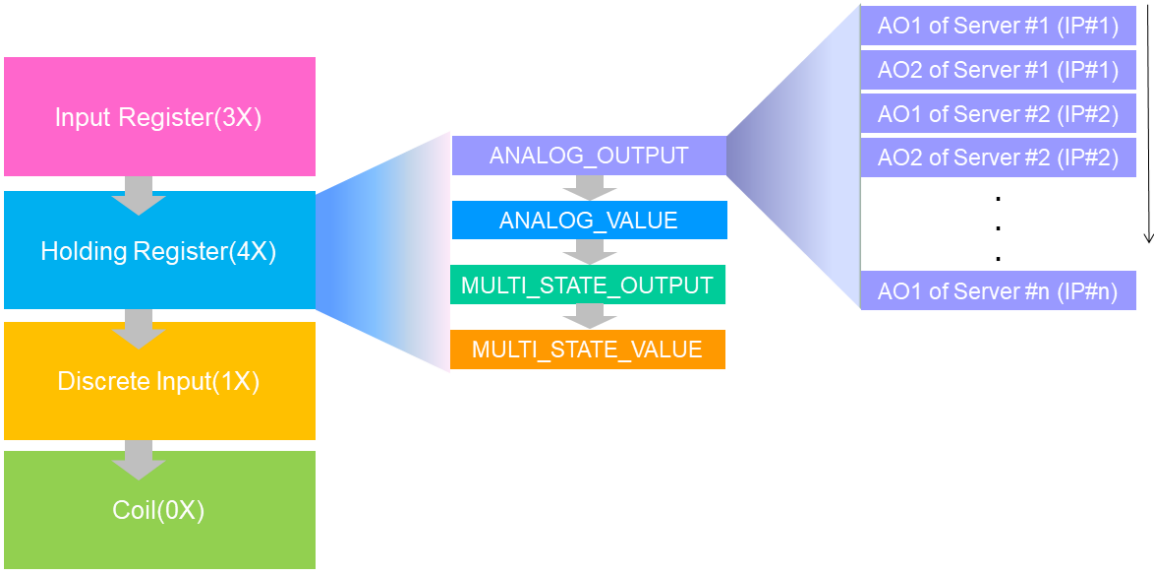
.

The following illustration shows the three-tier sorting process:

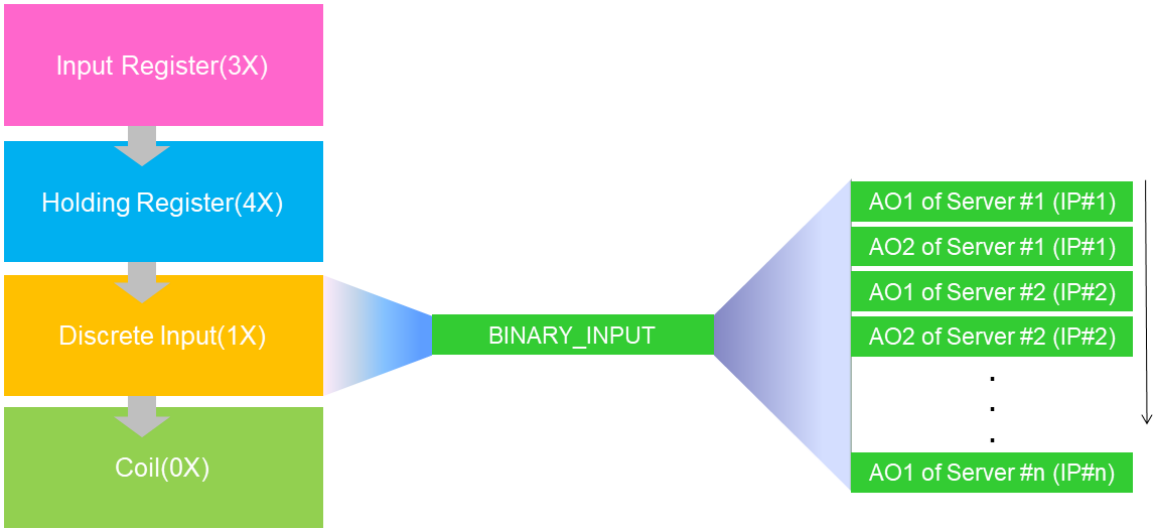
(1) Input Register(3X)



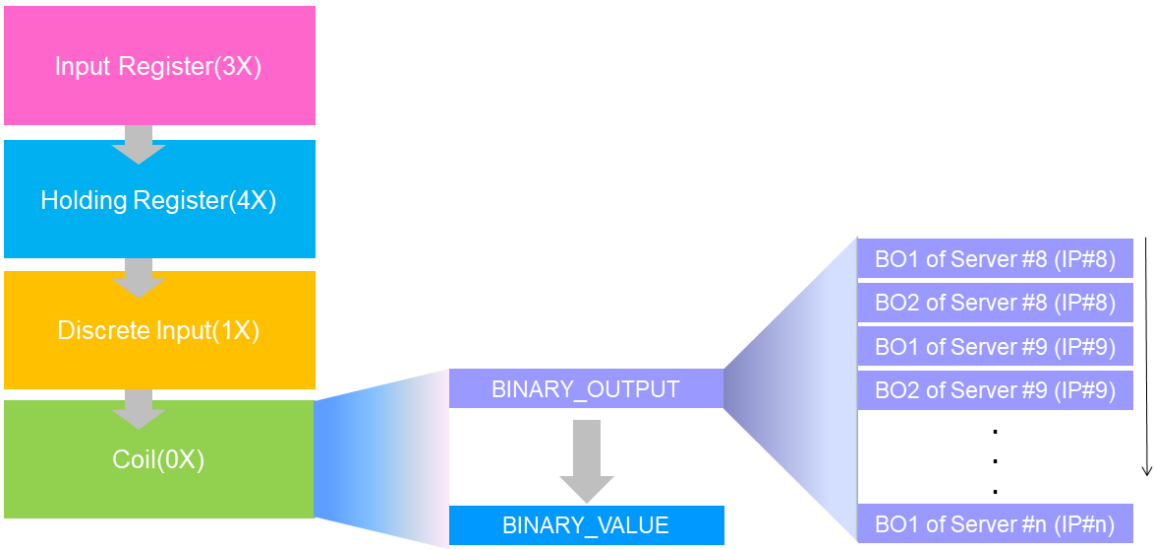
(2) Holding Register(4X)



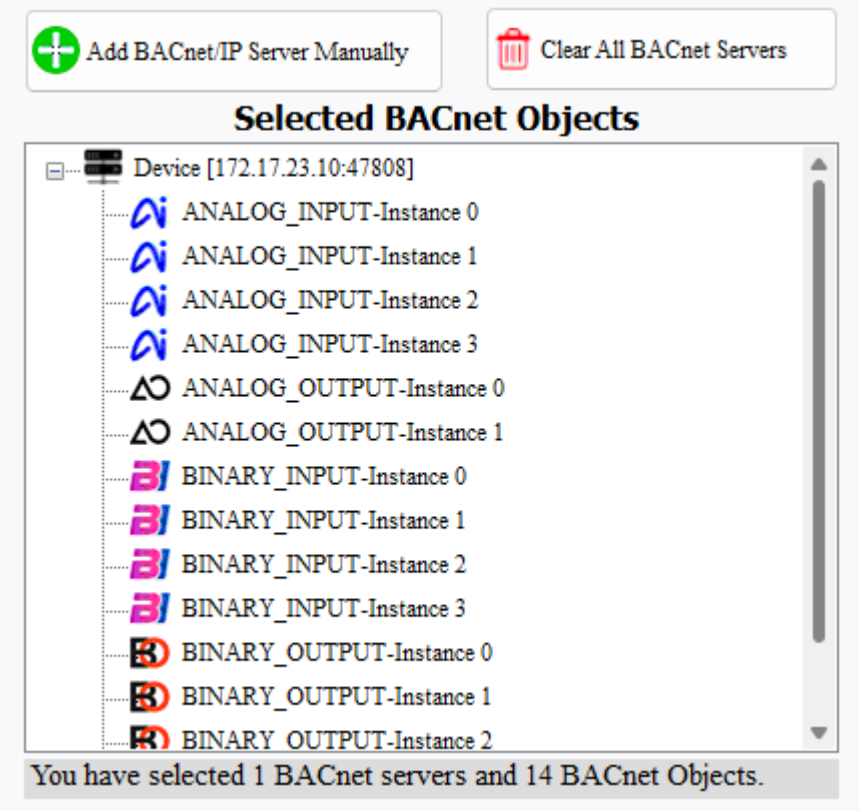
(3) Discrete Input(1X)



(4) Coil(0X)



Based on the sorting rules above, the following example shows how to arrange one server and its associated objects, as illustrated in the figure below:

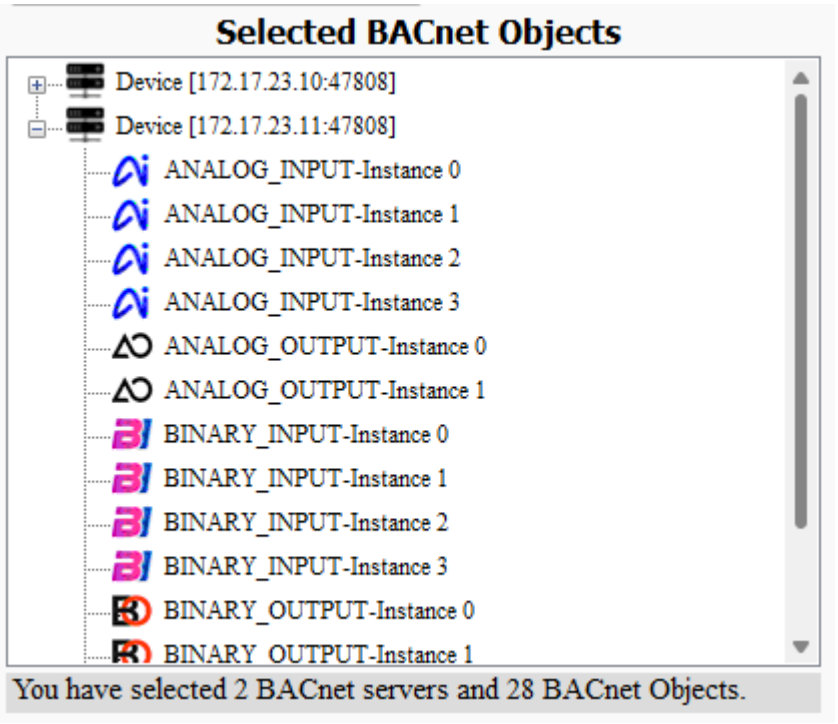


Users can see the automatically sorted Modbus mapping table, as shown below:

BACnet Browser | Modbus Browser | Module Information | Login Module

Server IP:Port	Object Type	Instance	Register Type	Register Len	StartAddress
172.17.23.10:47808	Analog_Input	0	Input register(3X)	32-bit	0
172.17.23.10:47808	Analog_Input	1	Input register(3X)	32-bit	2
172.17.23.10:47808	Analog_Input	2	Input register(3X)	32-bit	4
172.17.23.10:47808	Analog_Input	3	Input register(3X)	32-bit	6
172.17.23.10:47808	Analog_Output	0	Holding register(4X)	32-bit	0
172.17.23.10:47808	Analog_Output	1	Holding register(4X)	32-bit	2
172.17.23.10:47808	Binary_Input	0	Discrete Input(1X)	1-bit	0
172.17.23.10:47808	Binary_Input	1	Discrete Input(1X)	1-bit	1
172.17.23.10:47808	Binary_Input	2	Discrete Input(1X)	1-bit	2
172.17.23.10:47808	Binary_Input	3	Discrete Input(1X)	1-bit	3
172.17.23.10:47808	Binary_Output	0	Coil(0X)	1-bit	0
172.17.23.10:47808	Binary_Output	1	Coil(0X)	1-bit	1
172.17.23.10:47808	Binary_Output	2	Coil(0X)	1-bit	2
172.17.23.10:47808	Binary_Output	3	Coil(0X)	1-bit	3

The following example shows the arrangement based on two servers and their associated objects, as illustrated in the figure below:



Users can see the automatically sorted Modbus mapping table, as shown below:

BACnet Browser | Modbus Browser | Module Information | Login Module

Server IP:Port	Object Type	Instance	Register Type	Register Len	StartAddress
172.17.23.10:47808	Analog_Input	0	Input register(3X)	32-bit	0
172.17.23.10:47808	Analog_Input	1	Input register(3X)	32-bit	2
172.17.23.10:47808	Analog_Input	2	Input register(3X)	32-bit	4
172.17.23.10:47808	Analog_Input	3	Input register(3X)	32-bit	6
172.17.23.11:47808	Analog_Input	0	Input register(3X)	32-bit	8
172.17.23.11:47808	Analog_Input	1	Input register(3X)	32-bit	10
172.17.23.11:47808	Analog_Input	2	Input register(3X)	32-bit	12
172.17.23.11:47808	Analog_Input	3	Input register(3X)	32-bit	14
172.17.23.10:47808	Analog_Output	0	Holding register(4X)	32-bit	0
172.17.23.10:47808	Analog_Output	1	Holding register(4X)	32-bit	2
172.17.23.11:47808	Analog_Output	0	Holding register(4X)	32-bit	4
172.17.23.11:47808	Analog_Output	1	Holding register(4X)	32-bit	6
172.17.23.10:47808	Binary_Input	0	Discrete Input(1X)	1-bit	0
172.17.23.10:47808	Binary_Input	1	Discrete Input(1X)	1-bit	1
172.17.23.10:47808	Binary_Input	2	Discrete Input(1X)	1-bit	2
172.17.23.10:47808	Binary_Input	3	Discrete Input(1X)	1-bit	3
172.17.23.11:47808	Binary_Input	0	Discrete Input(1X)	1-bit	4
172.17.23.11:47808	Binary_Input	1	Discrete Input(1X)	1-bit	5
172.17.23.11:47808	Binary_Input	2	Discrete Input(1X)	1-bit	6
172.17.23.11:47808	Binary_Input	3	Discrete Input(1X)	1-bit	7
172.17.23.10:47808	Binary_Output	0	Coil(0X)	1-bit	0
172.17.23.10:47808	Binary_Output	1	Coil(0X)	1-bit	1
172.17.23.10:47808	Binary_Output	2	Coil(0X)	1-bit	2
172.17.23.10:47808	Binary_Output	3	Coil(0X)	1-bit	3
172.17.23.11:47808	Binary_Output	0	Coil(0X)	1-bit	4
172.17.23.11:47808	Binary_Output	1	Coil(0X)	1-bit	5
172.17.23.11:47808	Binary_Output	2	Coil(0X)	1-bit	6
172.17.23.11:47808	Binary_Output	3	Coil(0X)	1-bit	7

Modifying the StartAddress Parameter

To modify the StartAddress value for a specific data block, follow these steps:

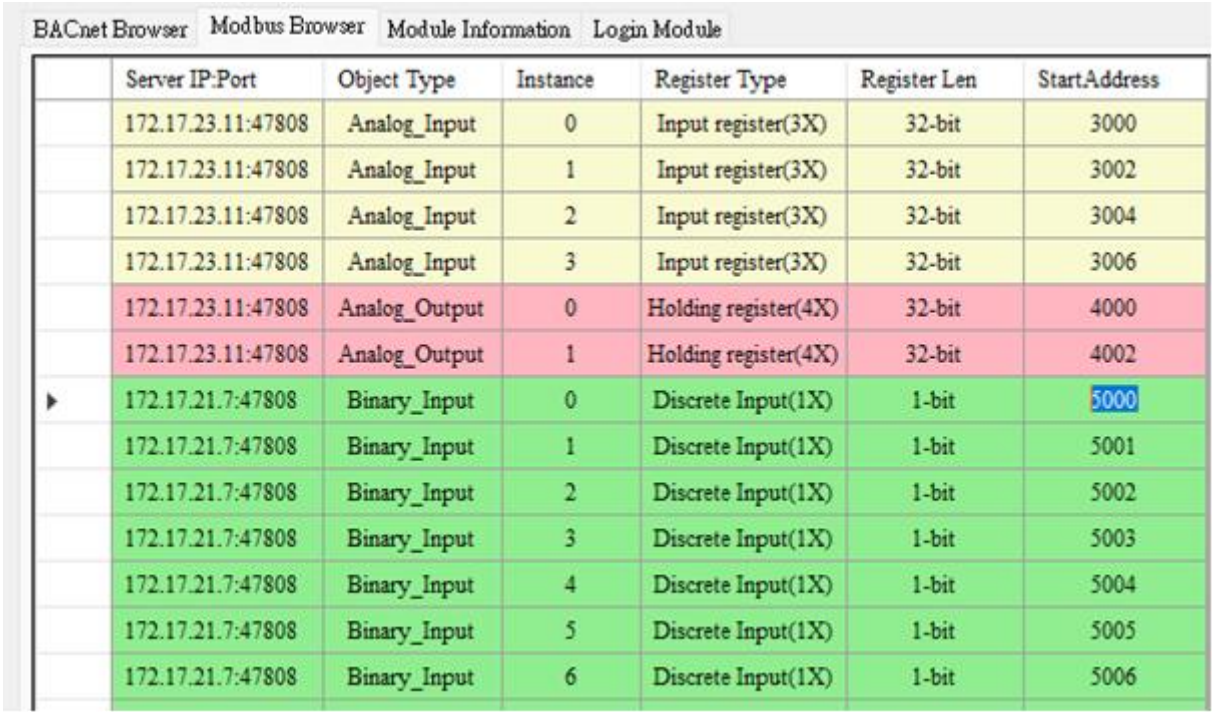
Double-click the target field with the left mouse button (e.g., a field currently set to 0).

Enter the new value directly (e.g., change it to 5000).

Press the [Enter] key to confirm and complete the modification.

Once a modification is made, the system automatically detects subsequent entries of the same data type and populates their addresses using a sequential increment method.

Example Scenario: If you change the StartAddress of the first Binary_Input entry to 5000, the system will automatically update the next entry to 5001, the following one to 5002, and so on.

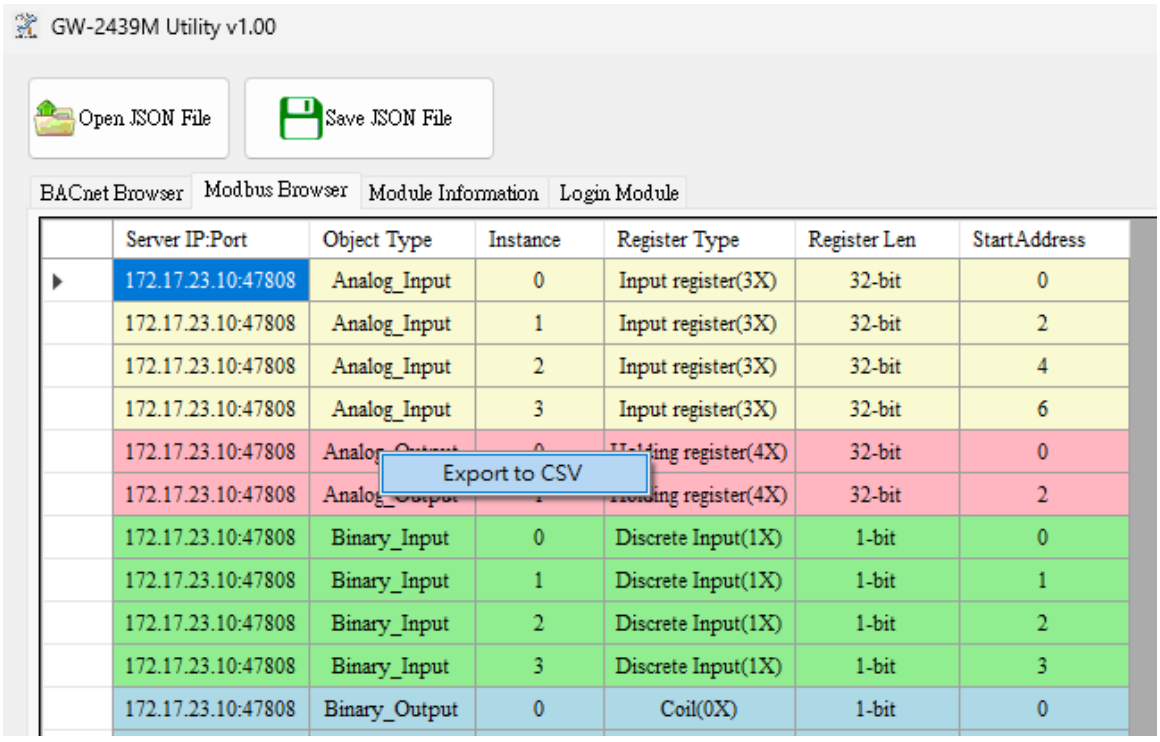


Server IP:Port	Object Type	Instance	Register Type	Register Len	StartAddress
172.17.23.11:47808	Analog_Input	0	Input register(3X)	32-bit	3000
172.17.23.11:47808	Analog_Input	1	Input register(3X)	32-bit	3002
172.17.23.11:47808	Analog_Input	2	Input register(3X)	32-bit	3004
172.17.23.11:47808	Analog_Input	3	Input register(3X)	32-bit	3006
172.17.23.11:47808	Analog_Output	0	Holding register(4X)	32-bit	4000
172.17.23.11:47808	Analog_Output	1	Holding register(4X)	32-bit	4002
172.17.21.7:47808	Binary_Input	0	Discrete Input(1X)	1-bit	5000
172.17.21.7:47808	Binary_Input	1	Discrete Input(1X)	1-bit	5001
172.17.21.7:47808	Binary_Input	2	Discrete Input(1X)	1-bit	5002
172.17.21.7:47808	Binary_Input	3	Discrete Input(1X)	1-bit	5003
172.17.21.7:47808	Binary_Input	4	Discrete Input(1X)	1-bit	5004
172.17.21.7:47808	Binary_Input	5	Discrete Input(1X)	1-bit	5005
172.17.21.7:47808	Binary_Input	6	Discrete Input(1X)	1-bit	5006

Key Benefits:

- Efficiency: Eliminates the need for manual, repetitive data entry for each row.
- Integrity: Ensures all addresses are continuous and non-overlapping.
- Error Reduction: Minimizes the risk of manual configuration errors and address conflicts.

The Modbus mapping table can be exported as a CSV file, making it convenient for users to review. To do this, simply right-click anywhere within the Modbus mapping table to bring up the context menu, as shown in the figure.



The contents of the exported CSV file are shown in the figure below.

	A	B	C	D	E	F
1	Server IP:Port	Object Type	Instance	Register Type	Register Len	StartAddress
2	172.17.23.10:47808	Analog_Input	0	Input register(3X)	32-bit	0
3	172.17.23.10:47808	Analog_Input	1	Input register(3X)	32-bit	2
4	172.17.23.10:47808	Analog_Input	2	Input register(3X)	32-bit	4
5	172.17.23.10:47808	Analog_Input	3	Input register(3X)	32-bit	6
6	172.17.23.10:47808	Analog_Output	0	Holding register(4X)	32-bit	0
7	172.17.23.10:47808	Analog_Output	1	Holding register(4X)	32-bit	2
8	172.17.23.10:47808	Binary_Input	0	Discrete Input(1X)	1-bit	0
9	172.17.23.10:47808	Binary_Input	1	Discrete Input(1X)	1-bit	1
10	172.17.23.10:47808	Binary_Input	2	Discrete Input(1X)	1-bit	2
11	172.17.23.10:47808	Binary_Input	3	Discrete Input(1X)	1-bit	3
12	172.17.23.10:47808	Binary_Output	0	Coil(0X)	1-bit	0
13	172.17.23.10:47808	Binary_Output	1	Coil(0X)	1-bit	1
14	172.17.23.10:47808	Binary_Output	2	Coil(0X)	1-bit	2
15	172.17.23.10:47808	Binary_Output	3	Coil(0X)	1-bit	3
16						

3.7 BACnet Object and Modbus Value Conversion

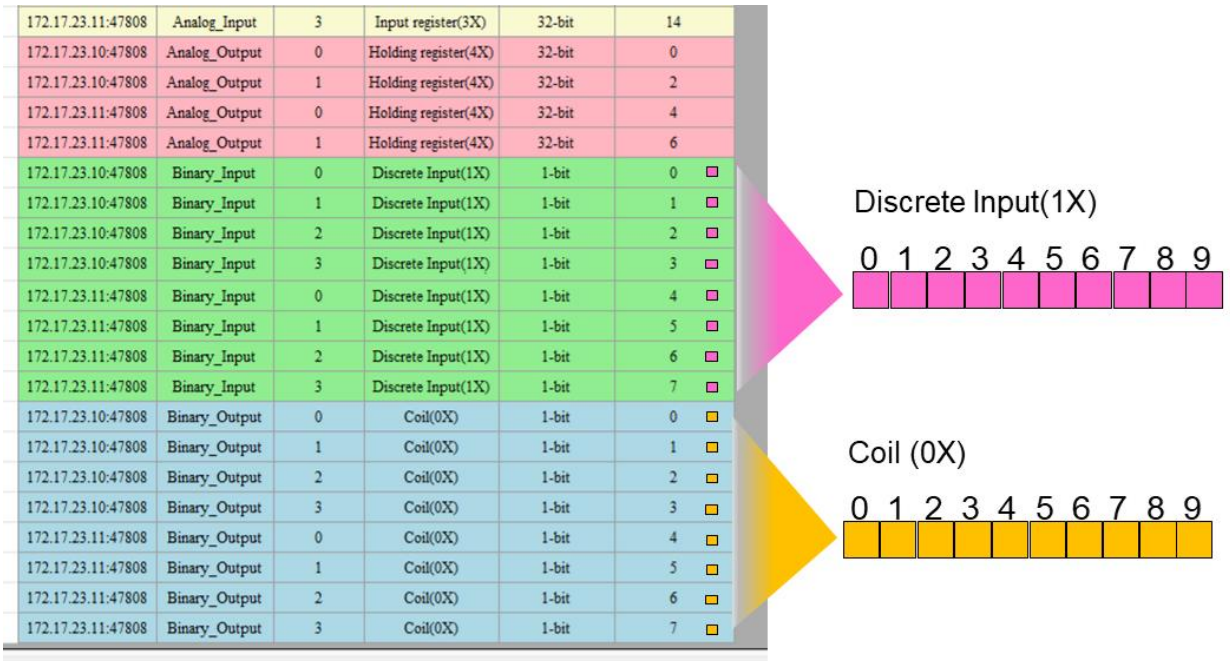
This chapter explains how BACnet objects in the GW-2439M are mapped to Modbus registers, ensuring that various BACnet data types can be seamlessly integrated into the Modbus protocol structure, enabling real-time data conversion.

3.7.1 Conversion of BI / BO / BV to Discrete Input and Coil

When acting as a BACnet Client, the GW-2439M can accurately map Binary Input (BI), Binary Output (BO), and Binary Value (BV) objects to the corresponding Modbus types: Discrete Input (1X) and Coil (0X) registers, enabling bit-level control conversion.

(1) BI → Discrete Input (1X)：Each BI object is mapped to a 1-bit Modbus Discrete Input address. In the example below, Instance 03 of device 172.17.23.11 is mapped to Modbus address 07.

(2) BO / BV → Coil (0X)：Each BO or BV object is mapped to a 1-bit Modbus Coil address. In the example below, Instance 02 of device 172.17.23.10 is mapped to Coil address 02, and so on.



This design allows logical states in the BACnet system to be accurately reflected in Modbus Coils, supporting applications such as switch status monitoring and device activation control. The data is 1-bit wide, requiring no special conversion, making the mapping straightforward and efficient.

3.7.2 Conversion of AI / AO / AV / MSI / MSO / MSV to Input/Holding Registers

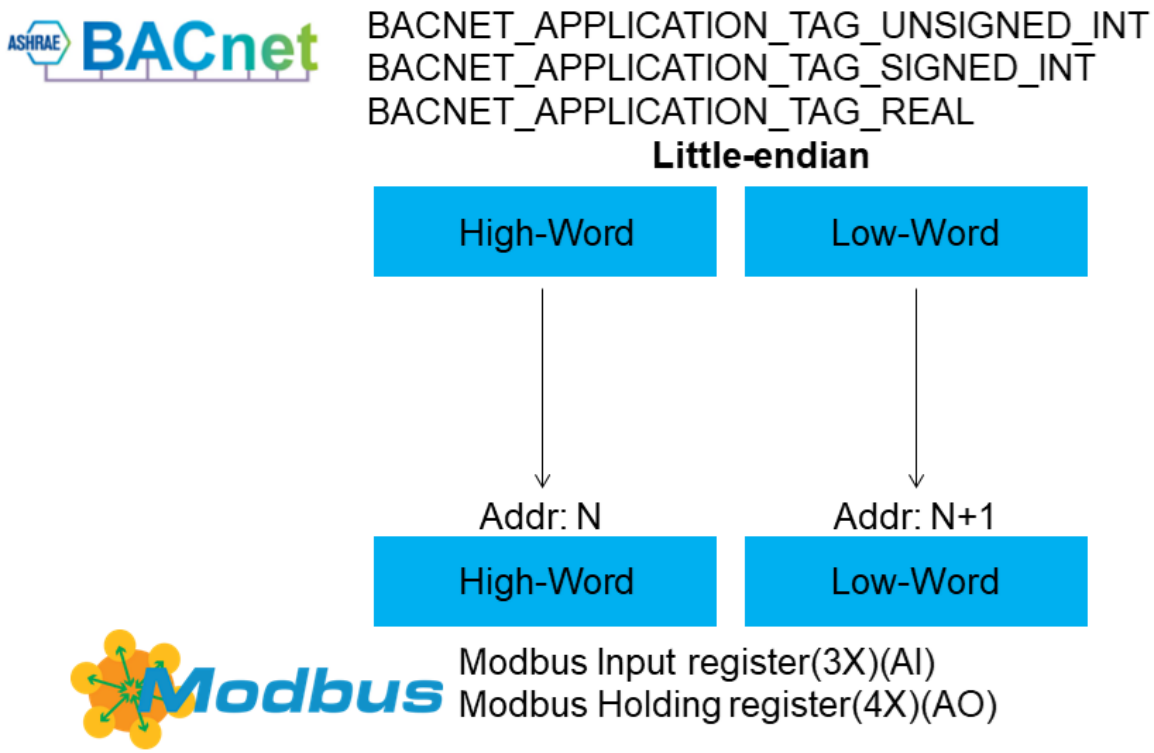
For BACnet Analog and Multi-State data handling, the GW-2439M converts:

- Sensor-type data such as AI (Analog Input) and MSI (Multi-State Input) into Input Registers (3X)
- Control-type data such as AO (Analog Output), AV, MSO, and MSV into Holding Registers (4X).

The conversion follows these rules:

Analog and Multi-State Data Type	Occupies Modbus 2-Word	Occupies Modbus 1-Word
Analog and Multi-State Data Type	Occupies Modbus 2-Word	Occupies Modbus 1-Word
BACNET_UNSIGNED_INT	Hi/Lo words are directly stored in Modbus registers.	Only the Lo word is stored in the Modbus register.
BACNET_SIGNED_INT	Hi/Lo words are directly stored in Modbus registers.	Only the Lo word is stored in the Modbus register.

The following illustration explains the conversion method when selecting the option to occupy 2 words:



Using the BACnet REAL type as an example, its float value is converted into IEEE754 format and then split into two 16-bit words for storage:

BACnet REAL = 10.0 → IEEE754：0x41200000

Modbus Register Address 0：0x4120

Modbus Register Address 1：0x0000

Using the BACnet SIGNED INT type as an example, its value is stored by converting it into two 16-bit words:

BACnet SIGNED INT = -123456 → Hex：0xFFFE1DC0

Modbus Register Address 0：0xFFFE

Modbus Register Address 1：0x1DC0

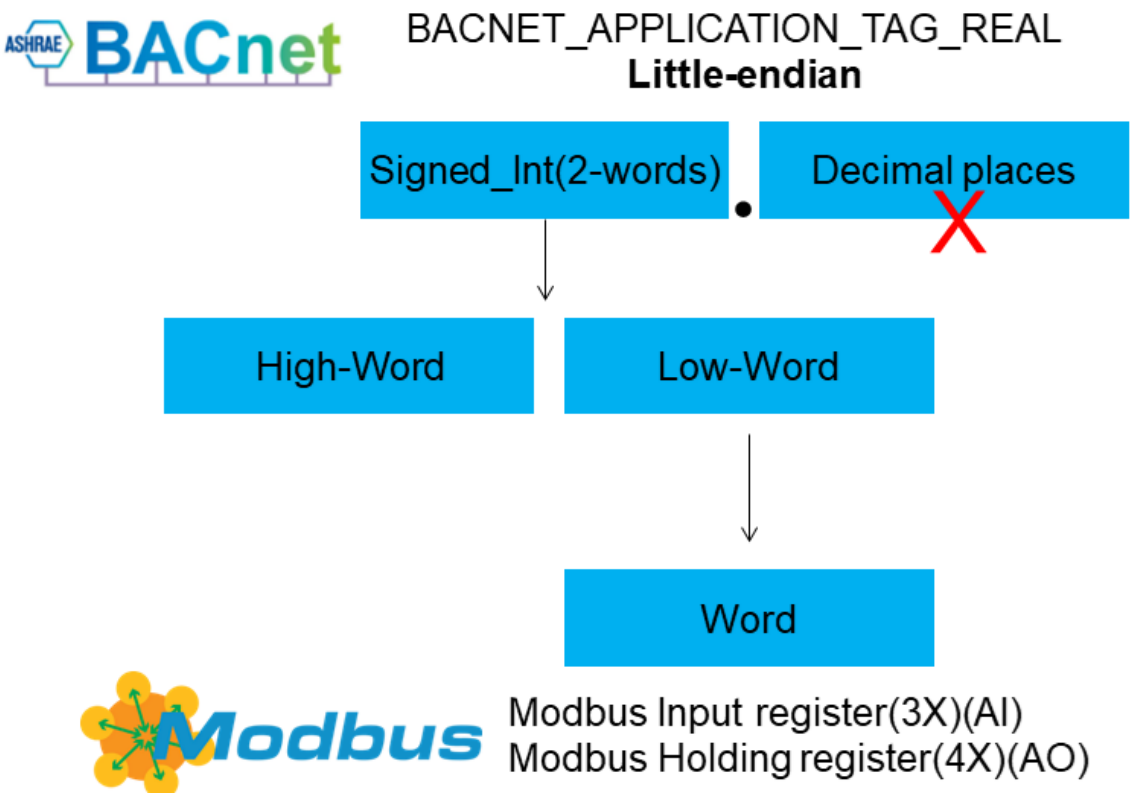
Using the BACnet UNSIGNED INT type as an example, its value is stored by converting it into two 16-bit words:

BACnet UNSIGNED INT = 123456 → Hex：0x0001E240

Modbus Register Address 0：0x0001

Modbus Register Address 1：0xE240

The following illustration explains the conversion method for BACnet REAL type when choosing to occupy only one 16-bit word:



Its float value is truncated without rounding to remove the decimal part, then converted into two 16-bit words. The Lo-word is then stored in the Modbus address.

Example 1:

BACnet REAL = 10.5 → Singed_Int: 10 → Hex：0x0000000A

Lo-word: 0x000A

Modbus Register Address 0：0x000A

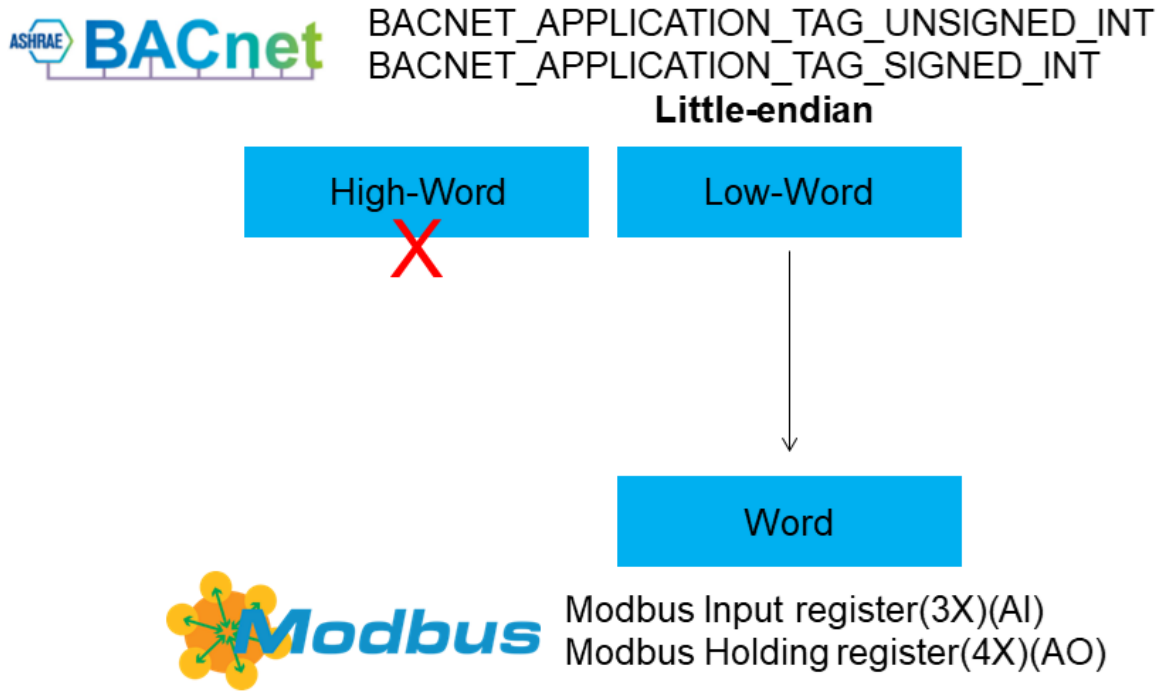
Example 2:

BACnet REAL = -10.5 → Singed_Int: -10 → Hex：0x FFFFFFF6

Lo-word: 0xFFF6

Modbus Register Address 0：0xFFF6

The following illustration explains the conversion method for BACnet SIGNED INT / UNSIGNED INT types when choosing to occupy only one 16-bit word:



- Using the BACnet SIGNED INT type as an example, the value is first converted into two 16-bit words, and then the Lo-word is stored in the Modbus address.

  BACnet SIGNED INT = -123456 → Hex：0xFFFE1DC0

  Modbus Register Address 0：0x1DC0

- Using the BACnet UNSIGNED INT type as an example, the value is first converted into two 16-bit words, and then the Lo-word is stored in the Modbus address.

  BACnet UNSIGNED INT = 123456 → Hex：0x0001E240

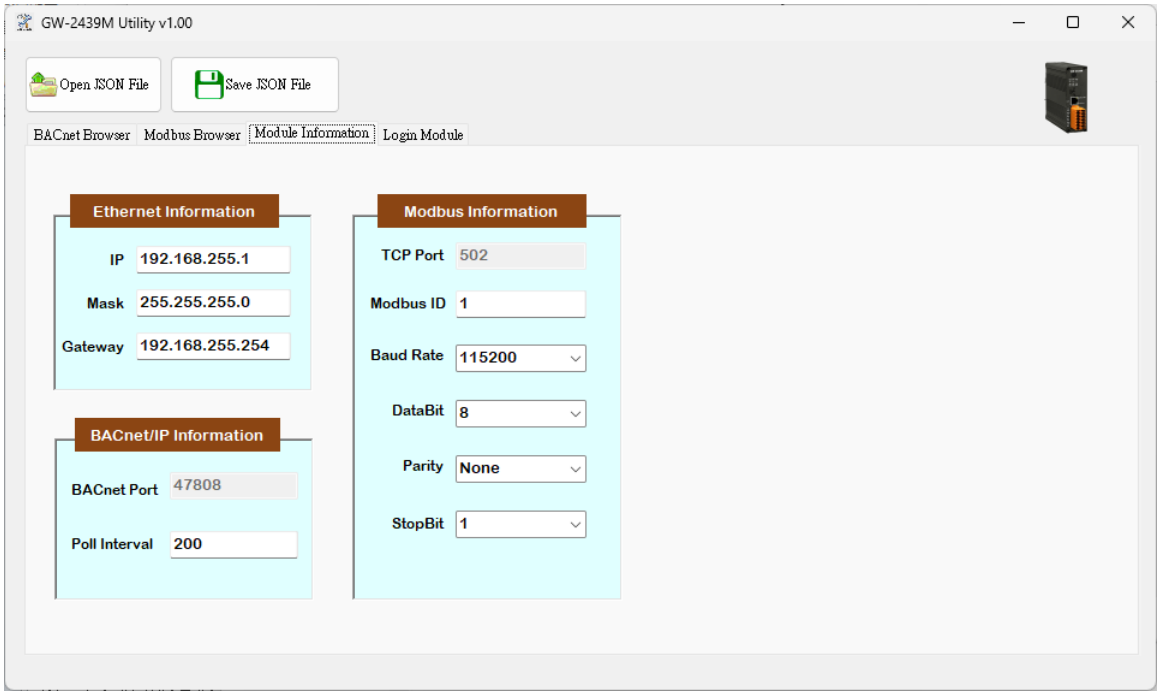
  Modbus Register Address 0：0xE240

3.8 Module Information Page

In the “Module Information” tab of the 《GW-2439M Utility》, users can configure the module’s network and communication parameters. This page is divided into three main sections:

- Ethernet Information
- BACnet/IP Information
- Modbus Information

Each section corresponds to settings related to the module’s network connectivity and communication protocols, as shown in the figure below.



1. Ethernet Information

This section configures the basic network parameters of the module, which are shared by both the BACnet/IP Client and Modbus TCP Server roles. It includes the

following fields:

- IP: Sets the module's static IP address.
- Mask: Sets the subnet mask.
- Gateway: Sets the default gateway to support cross-subnet communication.

2. BACnet/IP Information

This section contains configuration parameters for the module operating as a BACnet/IP Client:

- BACnet Port: The communication port is fixed at 47808, in compliance with the standard BACnet port specification. This setting cannot be changed.
- Poll Interval: Sets the time interval (in milliseconds) for the BACnet Client to actively poll target objects. The configurable range is 10 to 5000 ms.It is recommended to adjust this value according to the actual application scenario to achieve stable and real-time data updates.

3. Modbus Information

This section defines the RS-485 communication parameters when the GW-2439M module operates as a Modbus RTU Slave:

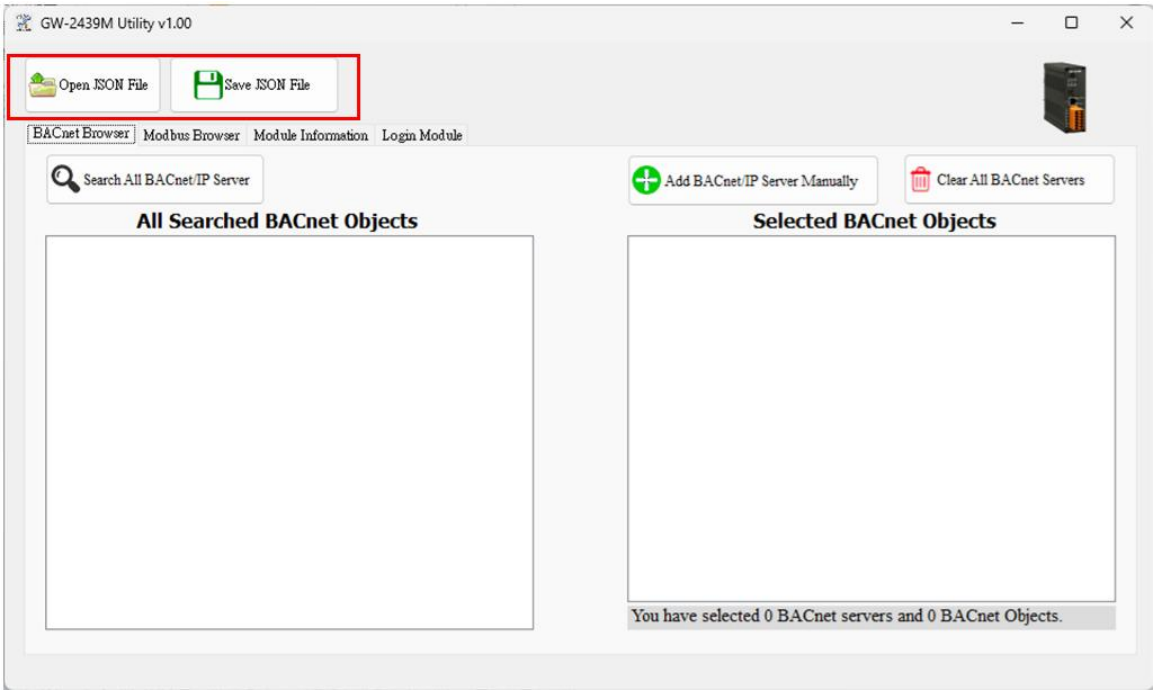
- TCP Port：This refers to the Modbus TCP port, which is fixed at 502 and cannot be changed in TCP mode.

- Modbus ID : Sets the station address (Slave ID) for the RTU device, allowing identification by the Modbus Master.
- Baud Rate : Supports standard baud rates (e.g., 9600, 19200, 115200, etc.) and must match the Master's configuration.
- DataBit / Parity / StopBit : Configures the data bit length, parity setting (None, Odd, Even), and stop bits to ensure protocol compatibility.

Correctly setting the above parameters ensures stable RS-485 communication between the module and the host system (such as a PLC or HMI).

3.9 Opening/Saving Configuration Files

The《GW-2439M Utility》interface provides a convenient configuration file management feature.Users can use the 【Open JSON File】and【Save JSON File】buttons located at the top of the interface (highlighted in red in the figure below) to open or save configuration data.



Feature Descriptions：

1.Open Configuration File (Open JSON File)

Users can load a previously saved configuration file (in JSON format). The system will automatically parse and import all related BACnet Server settings, object mappings, and communication parameters, allowing users to review and modify them as needed.

2.Save Configuration File (Save JSON File)

After completing the setup, users can save all current configurations as a JSON file. This file can serve as a backup, or be imported into other modules to quickly replicate the same settings across multiple GW-2439M devices effectively reducing human errors, minimizing repetitive work, and improving deployment efficiency and configuration consistency.

To improve project efficiency and maintainability, it is recommended that users:

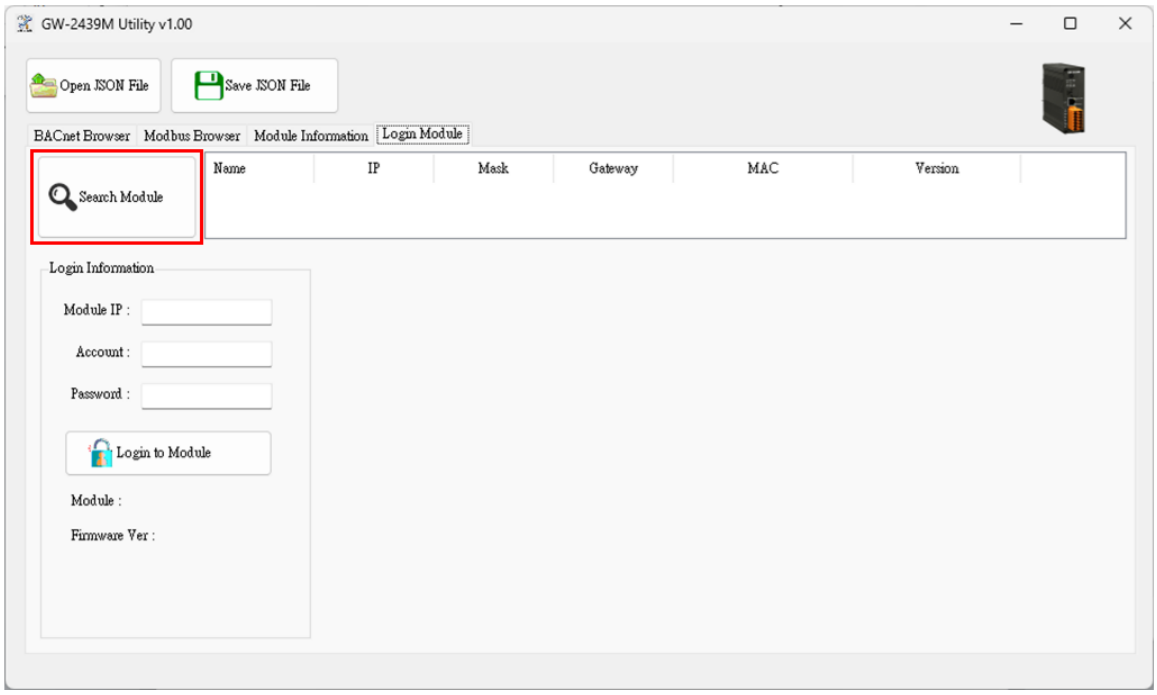
- Back up the original configuration file before making any changes.
- Create multiple configuration files for different application scenarios.
- Use version numbers or a naming convention to manage files for easy identification and tracking.

This design not only enhances operational flexibility and convenience, but also strengthens configuration consistency and project management efficiency.

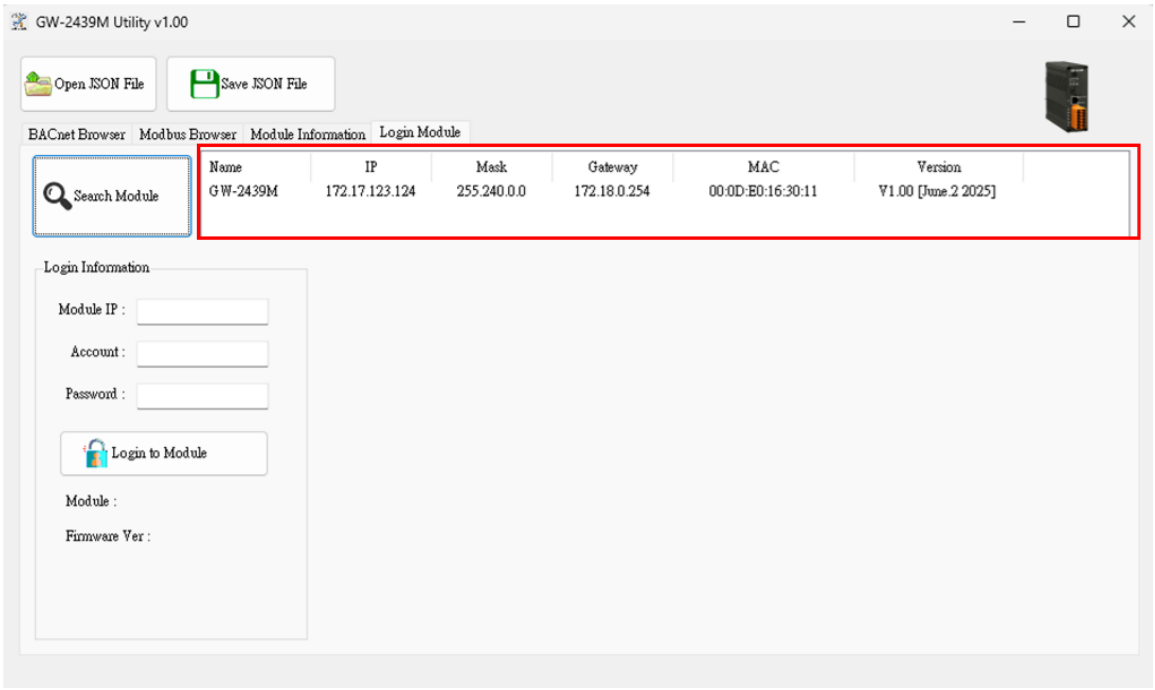
3.10 Logging In and Changing Account Password

To ensure the integrity of system configurations and prevent unauthorized modifications, the GW-2439M employs an access control mechanism. All configuration-related operations, such as modifying parameters, importing/exporting configuration files, require the user to log in to the module with a valid username and password before execution. This security feature effectively prevents unauthorized users from altering communication settings or data mapping logic, thereby safeguarding system stability and data security. It is recommended that users change passwords regularly and securely manage account credentials to maintain optimal operational security.

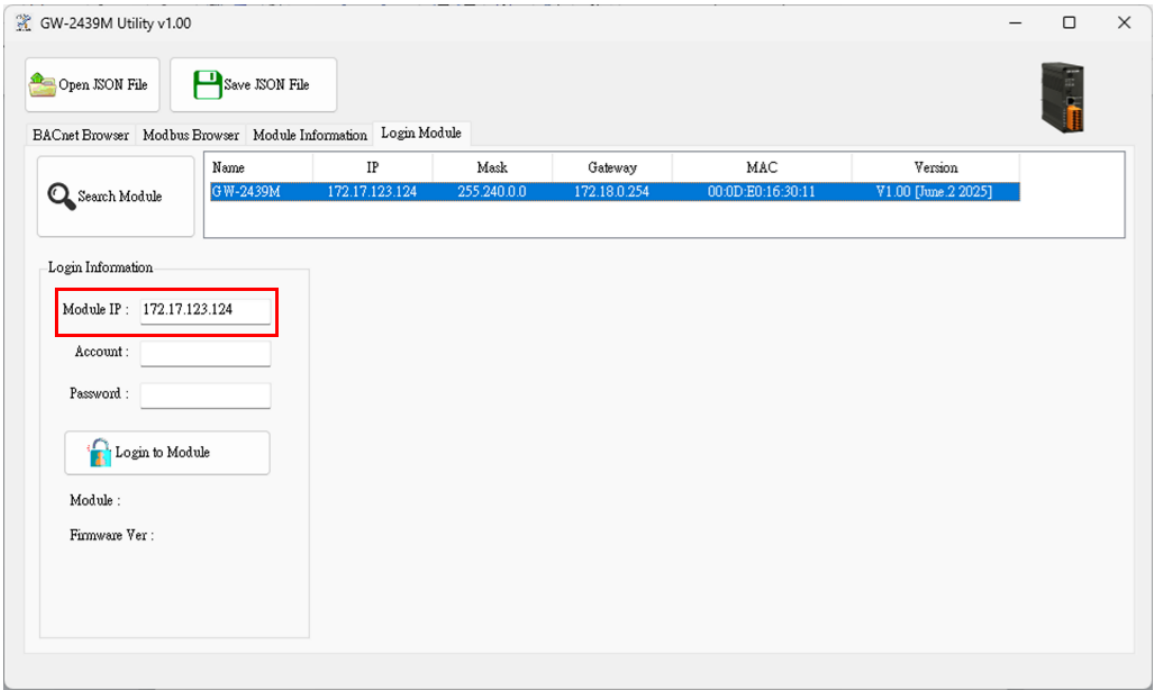
In the《GW-2439M Utility》, users can go to the “Login Module” tab and click the 【Search Module】button (highlighted in red in the figure below) to quickly scan and list all GW-2439M modules available on the local area network (LAN).



After clicking the 【Search Module】 button, the system will automatically scan the local area network and display the discovered modules in a list on the right side of the screen. Each entry includes detailed information such as the module name, IP address, subnet mask, default gateway, MAC address, and firmware version, as shown in the figure below.



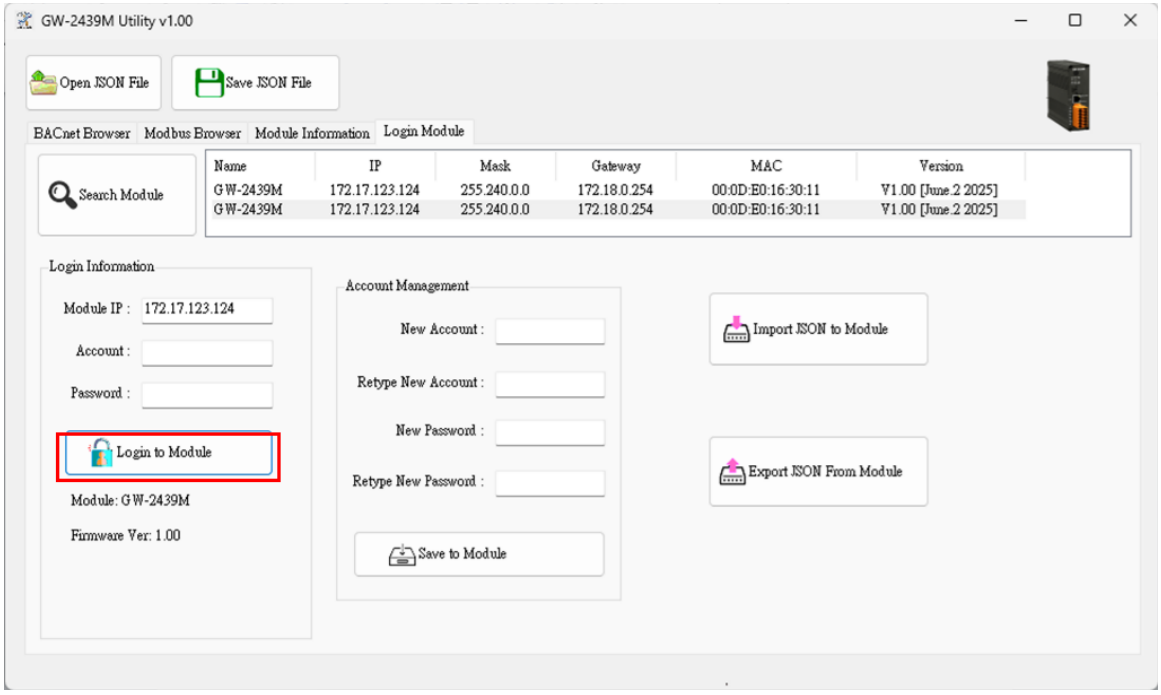
When the user selects any module from the list, the module's IP address will automatically populate the "Module IP" field in the "Login Information" section at the bottom of the screen.This eliminates the need for manual input and enhances operational convenience, as shown in the figure below.



After entering the correct Account and Password, the user can click the 【Login to Module】button to complete the login process. Upon successful login, the firmware will maintain the session for up to 30 minutes. To extend the session, the user can log in again. There is no limit to the number of logins.

- The default account is：ADMIN
- The default password is：ADMIN

Enter the correct credentials in the 「Account」 and 「Password」 fields, then click the【Login to Module】button (highlighted in red in the figure below). The GW-2439M will proceed to verify the login information.



Once login is successful, the module name and firmware version will be displayed in the lower-left corner of the screen. At the same time, the account management and configuration import/export features on the right side will be unlocked and made visible, allowing further operations.

Account and Password Management：

After logging into the module, users can update their account credentials via the 「Account Management」 section. The relevant rules are as follows：

- The account and password must be between 3 and 11 characters in length.
- The new account and new password must each be entered twice for confirmation, to prevent input errors.
- Click the【Save to Module】button to apply the changes. The new settings take effect immediately.

It is recommended that users change passwords regularly to enhance security and prevent unauthorized access.

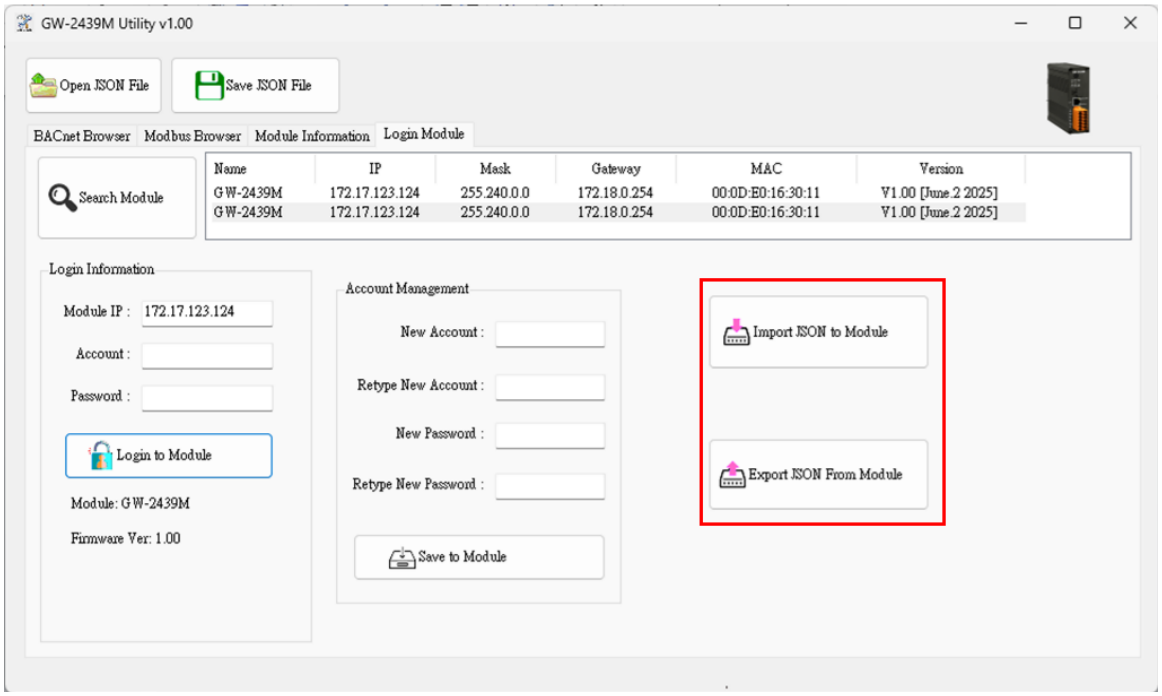
Note: Configuration-related functions (such as importing/exporting settings) are only accessible after a successful login. This ensures that all changes are properly authorized and protects the system from unintended modifications or tampering.

3.11 Importing and Exporting JSON Files

In the 「Login Module」 tab of the《GW-2439M Utility》, once the user has successfully logged into the module, two key functions will be enabled on the right side:

Import JSON to Module and Export JSON From Module (highlighted in red in the figure below).

These features are designed to help users efficiently manage module configurations, enhancing both operational efficiency and configuration consistency.



1. Export JSON From Module

Clicking this button allows you to export all current settings from the module—including network parameters, BACnet/IP and Modbus configurations, and object mappings—into a JSON file. This file can be used as a backup or applied to

other modules for quick configuration replication.

2. Import JSON to Module

Users can use this feature to import a previously exported JSON configuration file into the target module. After importing and powering the module back on, it will immediately apply all configuration settings contained in the file.

3. Export/Import Scope Description

The content that can be exported/imported includes:

- Ethernet settings.
- BACnet server information.
- Communication parameters.
- Object mapping tables.

The following content cannot be exported/imported:

- Account and password information.

To ensure information security and administrative flexibility, account and password details are not included in the JSON file.These must be configured manually by the administrator and cannot be automatically applied through import.

4. Monitoring BACnet Server Status via GW-2439M

The GW-2439M module supports simultaneous connections to multiple BACnet servers. The connection status information for each server is mapped to a fixed set of Modbus Input Registers (3X), starting at decimal address 50000. This architecture allows users to monitor the health and connectivity of all downstream servers simply by reading Modbus registers.

Each BACnet server is allocated a block of 5 contiguous registers.

Address(Dec.)	0	1	2	3	4
50000	Server IP#1 (MSB)	Server IP#1 (LSB)	Error Code	Error Count	Reserved
50005	Server IP#2 (MSB)	Server IP#2 (LSB)	Error Code	Error Count	Reserved

•

•

The addressing follows a sequential pattern:

- Server 1: Registers 50000 – 50004
- Server 2: Registers 50005 – 50009

  …
- Server N: Subsequent blocks follow the same 5-register offset.

【Server IP(MSB)】is the high-order word of the BACnet Server's IP address.

【Server IP(LSB)】is the low-order word of the BACnet Server's IP address.

【Error Code】shows the current communication status：

- 0 = Normal (No Error)
- 1 = Connection Timeout (Timeout)

【Error Count】shows the Cumulative count of timeout events since system startup. It is used to evaluate the link stability.

【Reserved】is reserved for future functional expansion.