tGW-700 Series

User Manual

Ver.2.0.0

Tiny Modbus/TCP to RTU/ASCII Gateway

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, service.icpdas@gmail.com

SUPPORT

This manual relates to the following modules:

tGW-712, tGW-722, tGW-732

tGW-715, tGW-725, tGW735

tGW-718, tGW-724, tGW-734

tGW-712i, tGW-722i, tGW-732i

tGW-715i, tGW-725i, tGW735i

tGW-718i, tGW-724i, tGW-734i

TABLE OF CONTENTS

Packing List

The shipping package includes the following items:

tGW-700/700i Series

Quick Start

Software CD

CA-002 Cable

Note	*If any of these items are missing or damaged, please contact the local distributor for more information. Save the shipping materials and cartons in case you need to ship the module in the future.*

More Information

Documentation

CD:\NAPDOS\tGW-700\Document

http://ftp.icpdas.com/pub/cd/tinymodules/napdos/tgw-700/document/

Firmware

CD:\NAPDOS\tGW-700\Firmware

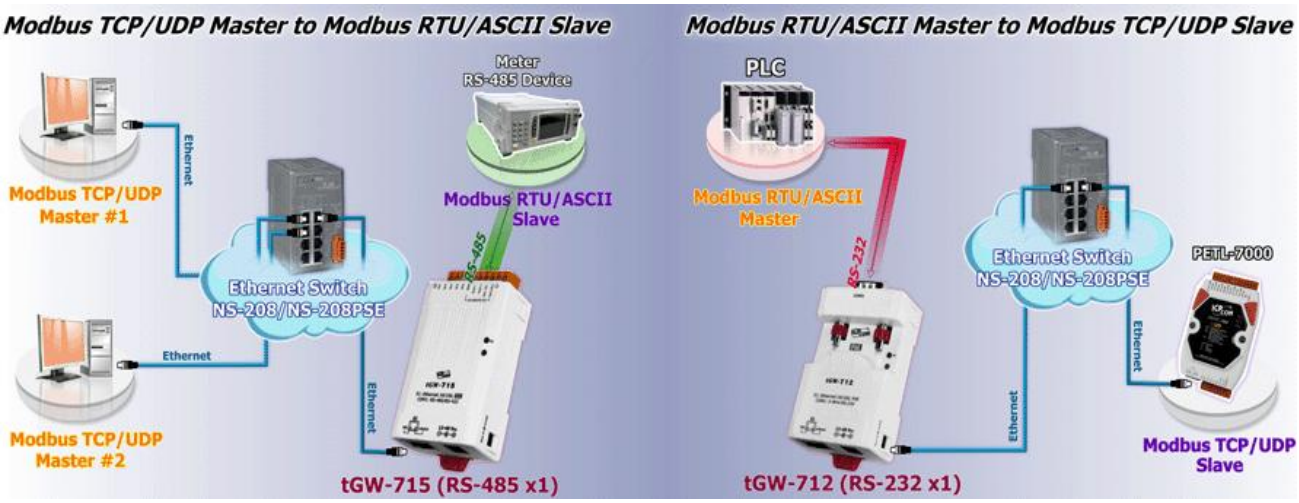
http://ftp.icpdas.com/pub/cd/tinymodules/napdos/tgw-700/firmware/

Software

CD:\NAPDOS\Software

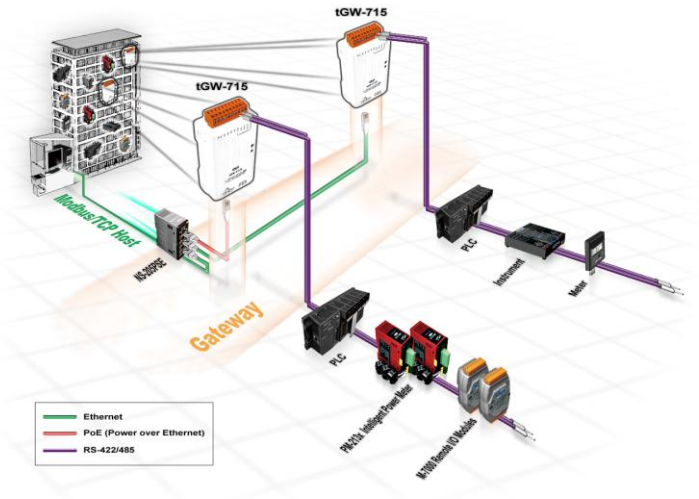
http://ftp.icpdas.com/pub/cd/tinymodules/napdos/software/

1. Introduction



Modbus has become the de facto standard protocol for industrial communication, and is now the most commonly available means of connecting industrial electronic devices. Modbus allows for communication between many devices connected to the same RS-485 network, for example, a system that measures temperature and humidity and communicates the results to a computer. Modbus is often used to connect a supervisory computer with a remote terminal unit (RTU) in supervisory control and data acquisition (SCADA) systems.

The tGW-700 module is a Modbus TCP/UDP to RTU/ASCII gateway that enables a Modbus/TCP 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. The module can be used to create pair-connection applications (as well as serial-bridge or serial-tunnel applications), and can then route data over a TCP/IP connection between two serial Modbus RTU/ASCII devices, which is useful when connecting mainframe computers, servers or other serial devices that use Modbus RTU/ASCII protocols and do not themselves have Ethernet capability.



In harsh industrial environments, the tGW-700 series (for i version) also adds 2500 V_{DC} and +/- 4 kV ESD protection component that diverts the potentially damaging charge away from sensitive circuit to protects the module and equipment from the sudden and momentary electric current.

The tGW-700 module features a powerful 32-bit MCU that enables efficient handling of network traffic, and also has a built-in web server that provides an intuitive web management interface that allows users to modify the configuration of the module, including the DHCP/Static IP, the gateway/mask settings and the serial port settings

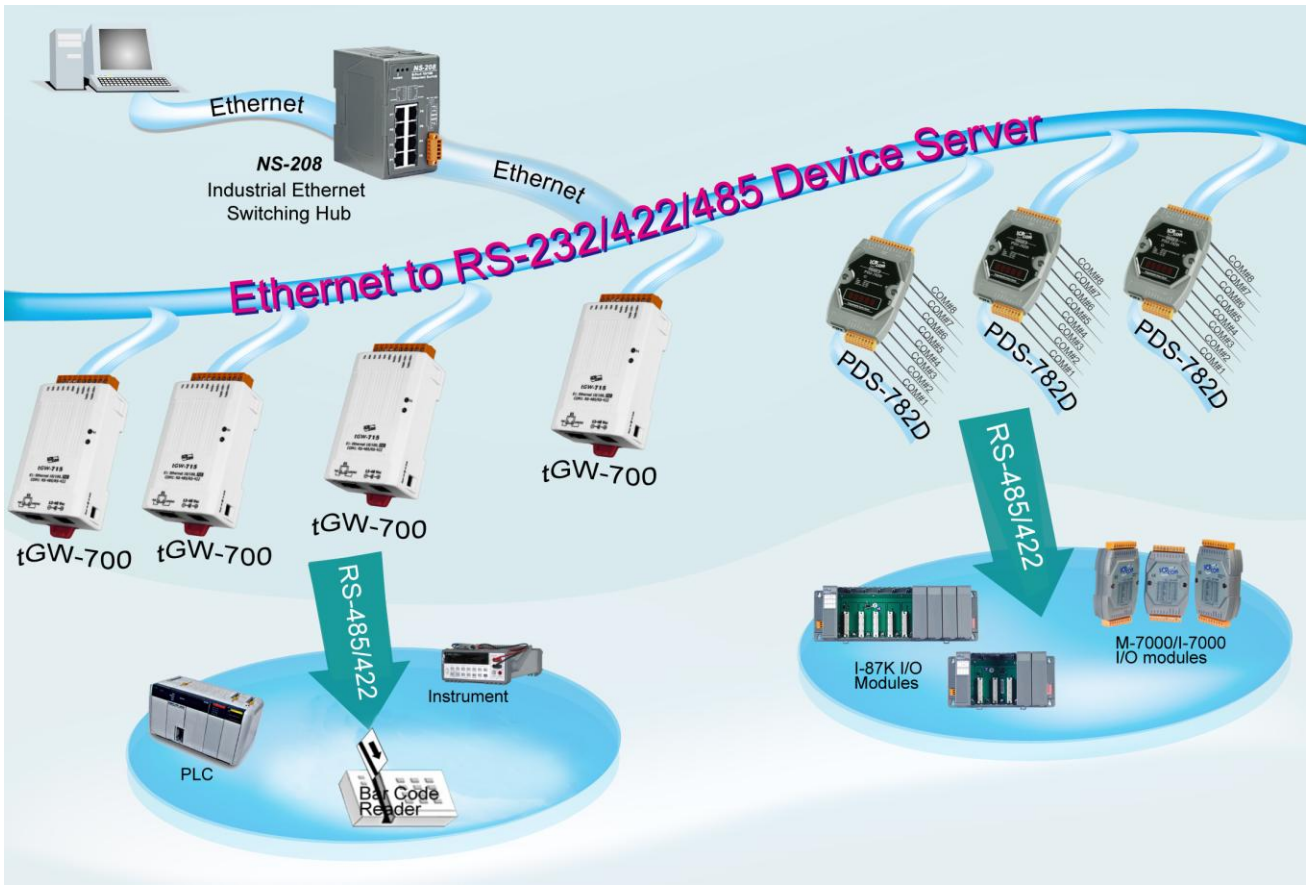
The tGW-700 module offers true IEEE 802.3af-compliant (classification, Class 1) Power over Ethernet (PoE) functionality using a standard Category 5 Ethernet cable to receive power from a PoE switch, such as the NS-205PSE. If there is no PoE switch on site, the module will also accept power input from a DC adapter. The tGW-700 module is designed for ultra-low power consumption, reducing the hidden costs resulting from increasing fuel and electricity prices, especially when a large number of modules are installed. Reducing the amount of electricity consumed by choosing energy-efficient equipment can also have a positive impact on maintaining a green environment.

➢ Comparison of Device Servers:

Series / Features	PPDS	PDS	DS	tDS	tGW
Virtual COM	Yes	Yes	Yes	Yes	-
Programmable	Yes	Yes	-	-	-
PoE	Yes	-	-	Yes	Yes
Modbus Gateway	Yes	-	-	-	Yes
Multi-client	About 20 Sockets			1 Sockets/Port	10 Sockets/Port
Remarks	Professional	Powerful	Isolation for DS-715	Cost-effective, Entry-level	Cost-effective, Entry-level

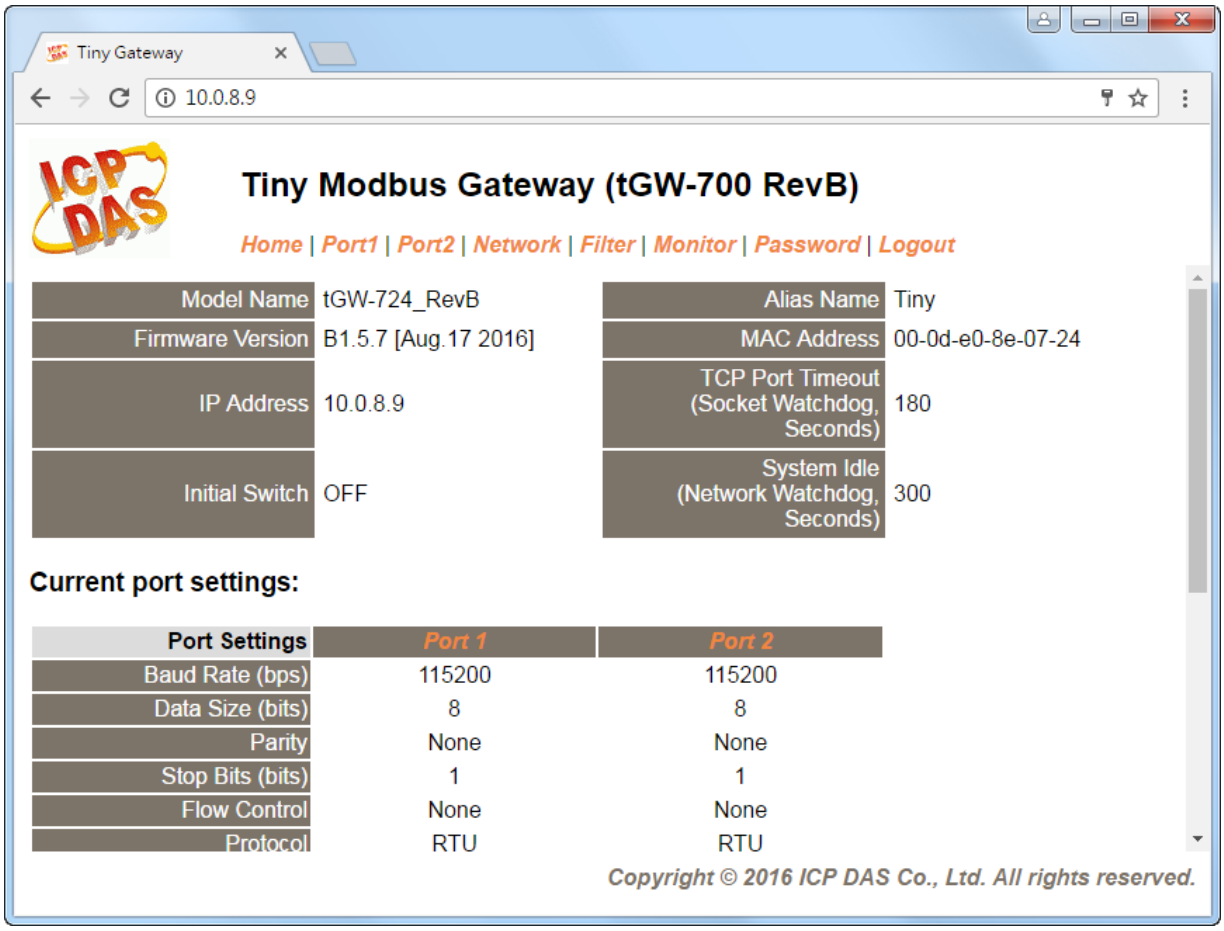
1.1 Ethernet Solutions

Nowadays, the Ethernet protocol has become the foremost standard for local area networks. Connectivity via the Internet is now common in many of the latest applications from home appliances, to vending machines, to testing equipment, to UPS, etc. An Ethernet network can link office automation and industrial control networks, access remote systems and share data and information between machines from multiple vendors, and also provides a cost-effective solution for industrial control networks.



1.2 Web Server Technology

Web server technology enables the tGW-700 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 tGW-700 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 tGW-700 series modules.

2.1 Specifications

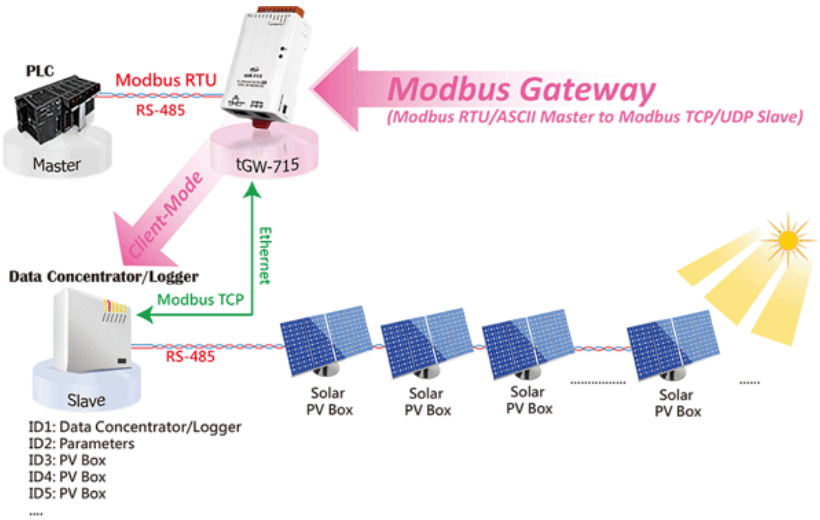
Model	tGW-712 tGW-712i	tGW-722 tGW-722i	tGW-732 tGW-732i	tGW-715 tGW-715i	tGW-725 tGW-725i	tGW-735 tGW-735i	tGW-718 tGW-718i	tGW-724 tGW-724i	tGW-734 tGW-734i
System									
CPU	32-bit ARM								
Communication Interface									
Ethernet	10/100 Base-TX, 8-pin RJ-45 x 1, (Auto-negotiating, Auto-MDI/MDIX, LED indicator) PoE (IEEE 802.3af, Class 1)								
COM1	5-wire RS-232	5-wire RS-232	3-wire RS-232	2-wire RS-485 4-wire RS-422	2-wire RS-485	2-wire RS-485	3-wire RS-232 2-wire RS-485 4-wire RS-422	2-wire RS-485	2-wire RS-485
COM2	-	5-wire RS-232	3-wire RS-232	-	2-wire RS-485	2-wire RS-485	-	5-wire RS-232	3-wire RS-232
COM3	-	-	3-wire RS-232	-	-	2-wire RS-485	-	-	3-wire RS-232
Self-Tuner	-			Yes, automatic RS-485 direction control					
RS-485 Bias Resistor	-			Yes, 1 KΩ					
RS-485 Node	-			254 (max.)					
UART	16c550 or compatible								
Isolation	2500 V_{DC} for only tGW-712i/722i/732i/715i/725i/735i/718i/724i/734i								
ESD Protection	+/-4 kV for only tGW-712i/722i/732i/715i/725i/735i/718i/724i/734i								
COM Port Format									
Baud Rate	115200 bps Max.								
Data Bit	5, 6, 7, 8								
Parity	None, Odd, Even, Mark, Space								
Stop Bit	1, 2								
Power									
Power Input	PoE: IEEE 802.3af, Class 1 DC jack: +12 ~ 48 V_{DC}								
Power Consumption	0.07 A @ 24 V_{DC}								
Mechanism									
Connector	Male DB-9 x1	10-Pin Removable Terminal Block x 1							
Mounting	DIN-Rail								
Flammability	Fire Retardant Materials (UL94-V0 Level)								
Environment									
Operating Temperature	-25 ~ +75 °C								
Storage Temperature	-30 ~ +80 °C								
Humidity	10 ~ 90% RH, non-condensing								
Note: COM1/COM2/COM3 = TCP Port 502/503/504									

2.2 Features

- Supports Modbus TCP/UDP master and slave
- Supports Modbus RTU/ASCII master and slave
- Max. connections (masters) per serial port: 32 (tGW-71x), 16 (tGW-72x) or 10 (tGW-73x)
- Read-cache ensures faster Modbus TCP/UDP response
- Supports UDP responder for device discovery (UDP Search)
- Static IP or DHCP network configuration
- Easy firmware update via the Ethernet (BOOTP, TFTP)
- Tiny Web server for configuration (HTTP)
- Contains a 32-bit MCU that efficiently handles network traffic
- 10/100 Base-TX Ethernet, RJ-45 x1 (Auto-negotiating, auto MDI/MDIX, LED Indicators)
- Includes redundant power inputs: PoE (IEEE 802.3af, Class 1) and DC jack
- Allows automatic RS-485 direction control
- 2500 V_{DC} isolation and +/- 4 kV ESD protection for i versions
- Male DB-9 or terminal block connector for easy wiring
- Tiny form-factor and low power consumption
- RoHS compliant with no Halogen
- Cost-effective Modbus Gateway

2.3 Applications

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



2.4 Selection Guide

Model		CPU	Ethernet	Baud Rate	COM1	COM2	COM3
Non-Isolated	Isolated						
tGW-712	tGW-712i	32-bit MCU	10/100 Base-TX, PoE	115200 bps	5-wire RS-232	-	-
tGW-722	tGW-722i				5-wire RS-232	5-wire RS-232	-
tGW-732	tGW-732i				3-wire RS-232	3-wire RS-232	3-wire RS-232
tGW-715	tGW-715i				2-wire RS-485 4-wire RS-422	-	-
tGW-725	tGW-725i				2-wire RS-485	2-wire RS-485	-
tGW-735	tGW-735i				2-wire RS-485	2-wire RS-485	2-wire RS-485
tGW-718	tGW-718i				3-wire RS-232 2-wire RS-485 4-wire RS-422	-	-
tGW-724	tGW-724i				2-wire RS-485	5-wire RS-232	-
tGW-734	tGW-734i				2-wire RS-485	3-wire RS-232	3-wire RS-232

3-Wire RS-232: RxD, TxD, GND

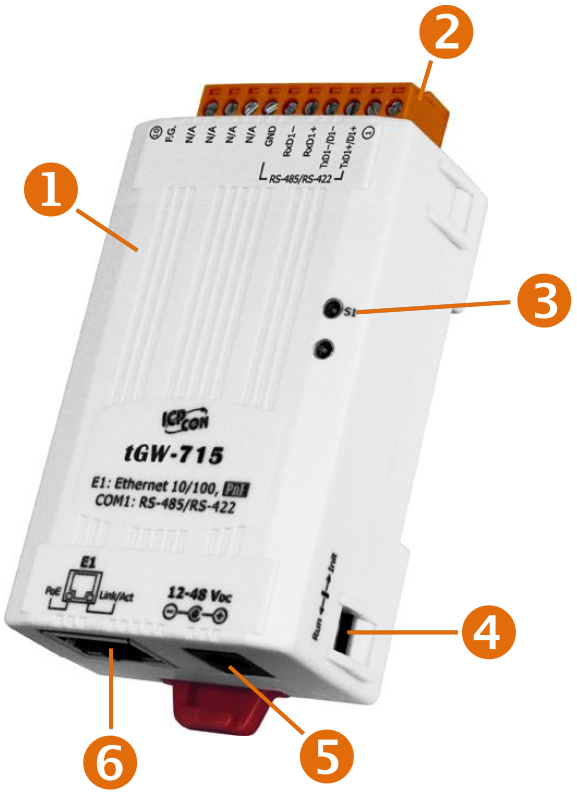
5-Wire RS-232: RxD, TxD, CTS, RTS, GND

2-Wire RS-485: DATA+, DATA-, GND

4-Wire RS-422: TxD+, TxD-, RxD+, RxD-, GND

2.5 Appearance

Front View



1. Robust Insulated and Fire-retardant Case

2. Serial COM Ports

The number of serial COM Ports available depends on the type of tGW-700 module. For more detailed information regarding the pin assignments for the Serial COM ports, refer to Section 2.7 Pin Assignments.

3. S1: System LED indicator

Once power is supplied to the tGW-700 module, the system LED indicator will illuminate. An overview of the LED functions is given below:

Function	System LED Behavior
Running Firmware	Steady ON
Network Ready	Slow flashing – Once every 3 seconds
Serial Port Busy	Rapid flashing – Once every 0.2 seconds

4. Operating Mode Switch

Init Mode: Configuration mode

Run Mode: Firmware operation mode

For tGW-700 series modules, the operating mode switch is set to the **Run** position by default. In order to update the firmware for the tGW-700 module, the switch must be moved from the **Run** position to the **Init** position. The switch must be returned to the Run position after the update is complete.

5. +12 to +48 V_{DC} Jack:

The tGW-700 is equipped with a +12 V_{DC} to +48 V_{DC} jack 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 tGW-700 module.

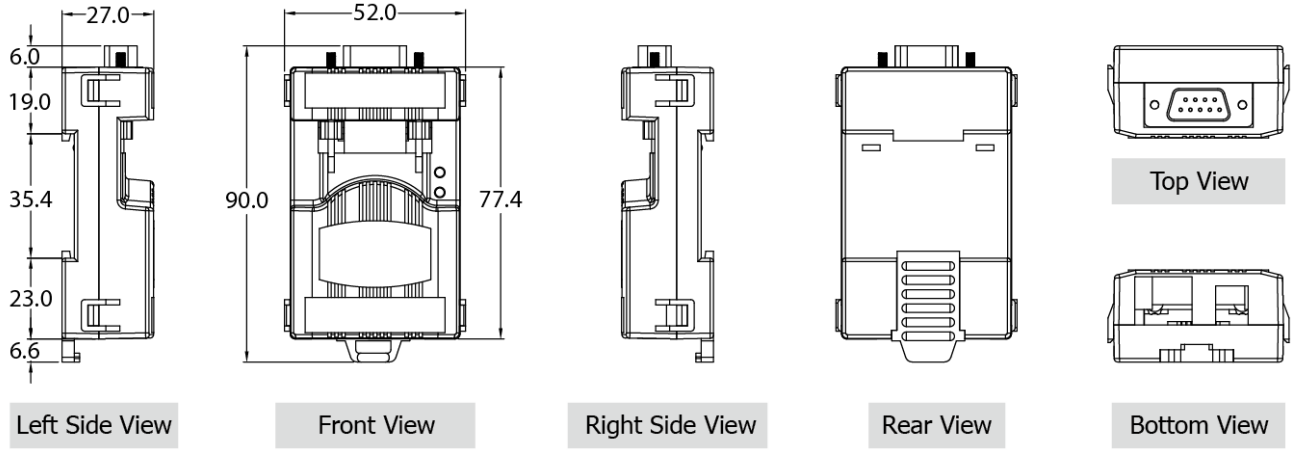
6. PoE and Ethernet RJ-45 Jack

The tGW-700 module is equipped with an RJ-45 jack that is used as the 10/100 Base-TX Ethernet port and features networking capabilities. 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 (Green)** indicator will be illuminated.

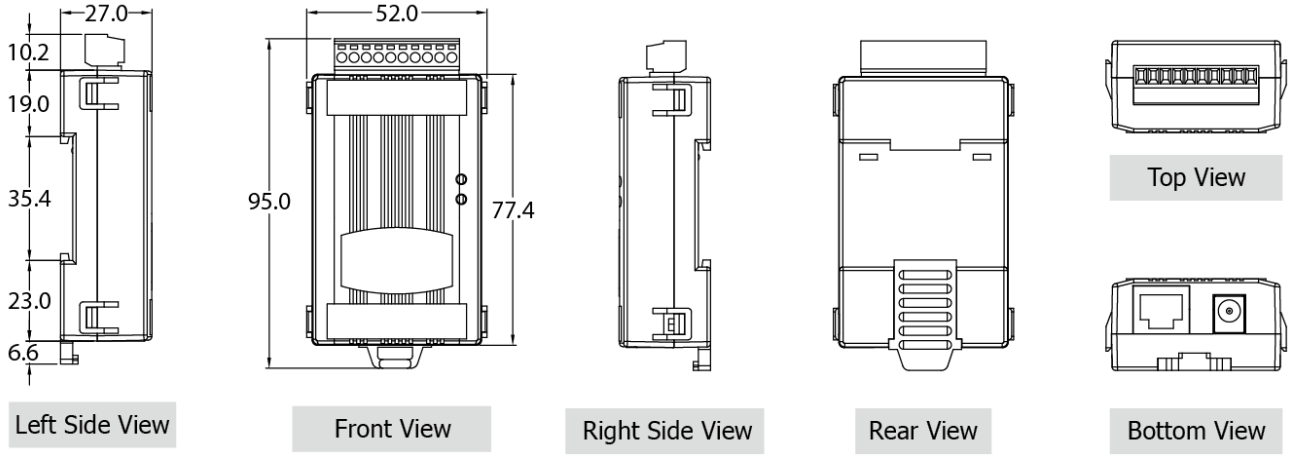
2.6 Dimensions

The following diagrams provide the dimensions of the tGW-700 series module and can be used as a reference when defining the specifications for any custom enclosures. All dimensions are in millimeters.

➢ **tGW-712/712i:**

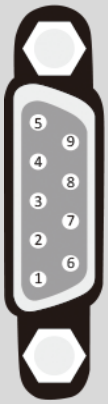


➢ **tGW-722/722i/732/732i/715/715i/725/725i/735/735i/718/718i/724/724i/734/734i:**



2.7 Pin Assignments

2.7.1 tGW-712/tGW-712i

		tGW-712	tGW-712i
Terminal No.		Pin Assignment	
COM1 	09	N/A	N/A
	08	CTS1	CTS1
	07	RTS1	RTS1
	06	N/A	N/A
	05	GND	ISO.GND
	04	N/A	N/A
	03	TxD1	TxD1
	02	RxD1	RxD1
	01	N/A	N/A

Note: The CTS and RTS pins are reserved and have no function.

2.7.2 tGW-722/tGW-722i

		tGW-722	tGW-722i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
COM2	09	CTS2	CTS2
	08	RTS2	RTS2
	07	RxD2	RxD2
	06	TxD2	TxD2
COM1	05	GND	ISO.GND
	04	CTS1	CTS1
	03	RTS1	RTS1
	02	RxD1	RxD1
	01	TxD1	TxD1

Note: The CTS and RTS pins are reserved and have no function.

2.7.3 tGW-732/tGW-732i

		tGW-732	tGW-732i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
COM3	09	GND	ISO.GND
	08	RxD3	RxD3
	07	TxD3	TxD3
COM2	06	GND	ISO.GND
	05	RxD2	RxD2
	04	TxD2	TxD2
COM1	03	GND	ISO.GND
	02	RxD1	RxD1
	01	TxD1	TxD1

2.7.4 tGW-715/tGW-715i

		tGW-715	tGW-715i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
	09	N/A	N/A
	08	N/A	N/A
	07	N/A	N/A
	06	N/A	N/A
RS-485/RS-422	05	GND	ISO.GND
	04	RxD1-	RxD1-
	03	RxD1+	RxD1+
	02	TxD1-/D1-	TxD1-/D1-
	01	TxD1+/D1+	TxD1+/D1+

2.7.5 tGW-725/tGW-725i

		tGW-725	tGW-725i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
	09	N/A	N/A
	08	N/A	N/A
	07	N/A	N/A
COM2	06	GND	ISO.GND
	05	D2-	D2-
	04	D2+	D2+
COM1	03	GND	ISO.GND
	02	D1-	D1-
	01	D1+	D1+

2.7.6 tGW-735/tGW-735i

		tGW-735	tGW-735i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
COM3	09	GND	ISO.GND
	08	D3-	D3-
	07	D3+	D3+
COM2	06	GND	ISO.GND
	05	D2-	D2-
	04	D2+	D2+
COM1	03	GND	ISO.GND
	02	D1-	D1-
	01	D1+	D1+

2.7.7 tGW-718/tGW-718i

		tGW-718	tGW-718i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
	09	N/A	N/A
RS-232	08	GND	ISO.GND
	07	RxD1	RxD1
	06	TxD1	TxD1
RS-485/RS-422	05	GND	ISO.GND
	04	RxD1-	RxD1-
	03	RxD1+	RxD1+
	02	TxD1-/D1-	TxD1-/D1-
	01	TxD1/D1+	TxD1/D1+

2.7.8 tGW-724/tW-724i

		tGW-724	tGW-724i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
	09	GND	ISO.GND
COM2	08	CTS2	CTS2
	07	RTS2	RTS2
	06	GND	ISO.GND
	05	RxD2	RxD2
	04	TxD2	TxD2
COM1	03	GND	ISO.GND
	02	D1-	D1-
	01	D1+	D1+

Note: The CTS and RTS pins are reserved and have no function.

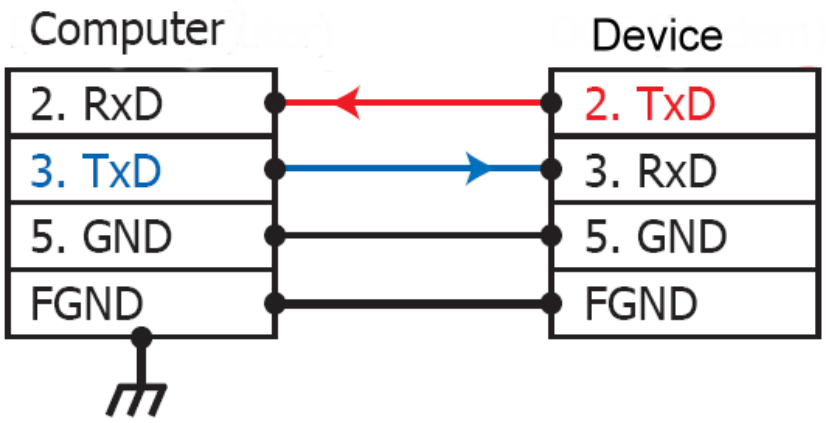
2.7.9 tGW-734/tGW-734i

		tGW-734	tGW-734i
Terminal No.		Pin Assignment	
	10	F.G.	F.G.
COM3	09	GND	ISO.GND
	08	RxD3	RxD3
	07	TxD3	TxD3
COM2	06	GND	ISO.GND
	05	RxD2	RxD2
	04	TxD2	TxD2
COM1	03	GND	ISO.GND
	02	D1-	D1-
	01	D1+	D1+

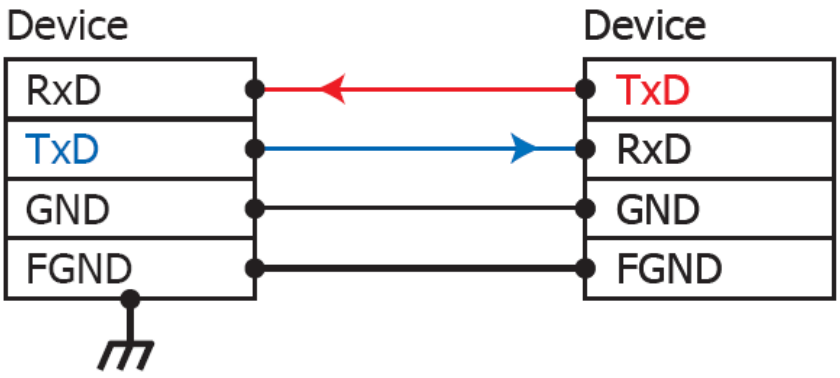
2.8 Wiring Notes for RS-232/485/422 Interfaces

2.8.1 RS-232 Wiring

5-wire RS-232 Connection (DB-9)

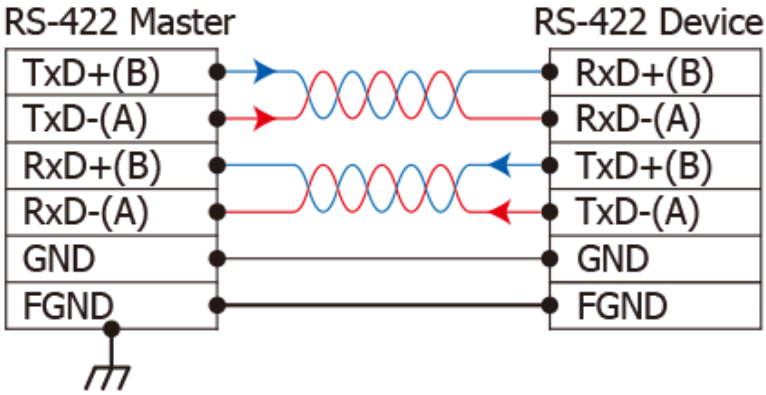


3-wire RS-232 Connection

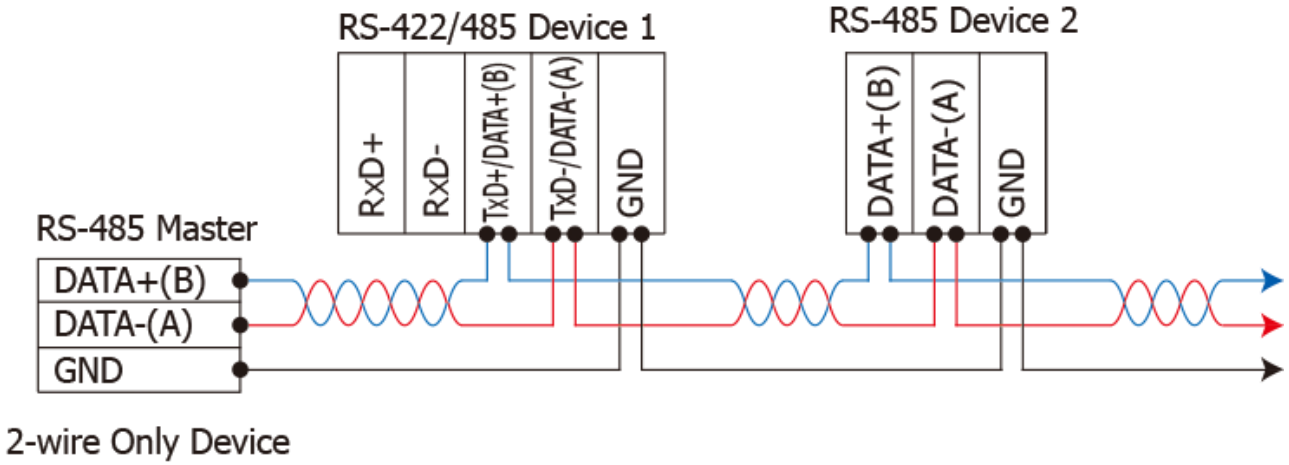


Note	*FGND is the frame ground that is soldered to the metal shield on the DB-9 cable.*

2.8.2 RS-422 Wiring



2.8.3 RS-485 Wiring



Note	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 DATA+/- wires.* 3. *Both two ends of the cable may require a termination resistor connected across the two wires (DATA+ and DATA-). Typically 120 Ω resisters are used.* 4. *The Data+ and B pins are positive-voltage pins, and Data- 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.*

3. Setting up the tGW-700 Module

This chapter provides detailed information about the "Self-Test" process, which is used to confirm that the tGW-700 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:

Step 1: Connect the Power Supply and the 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. Check that the Init/Run switch is in the **"Run"** position.

Figure 3-1

4. Connect both the tGW-700 and the Host computer to the same sub-network or the same Ethernet Switch, and then power on the tGW-700. Refer to **Figures 3-2 and 3-3** for illustrations of how to do this.

+12 to +48 V_{DC} jack Power Supply (Non-PoE)

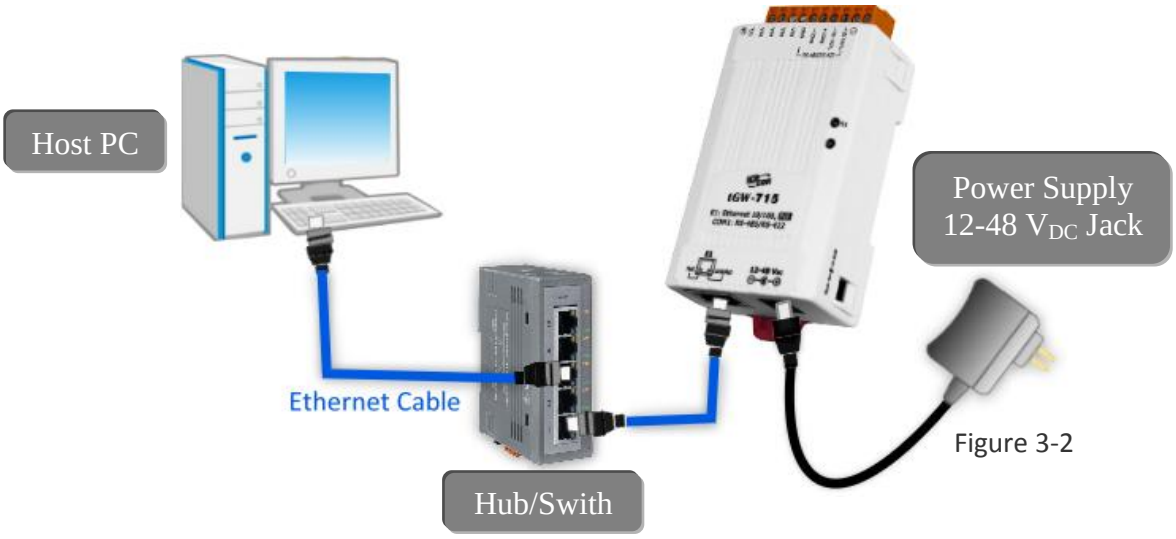


Figure 3-2

PoE Power Supply

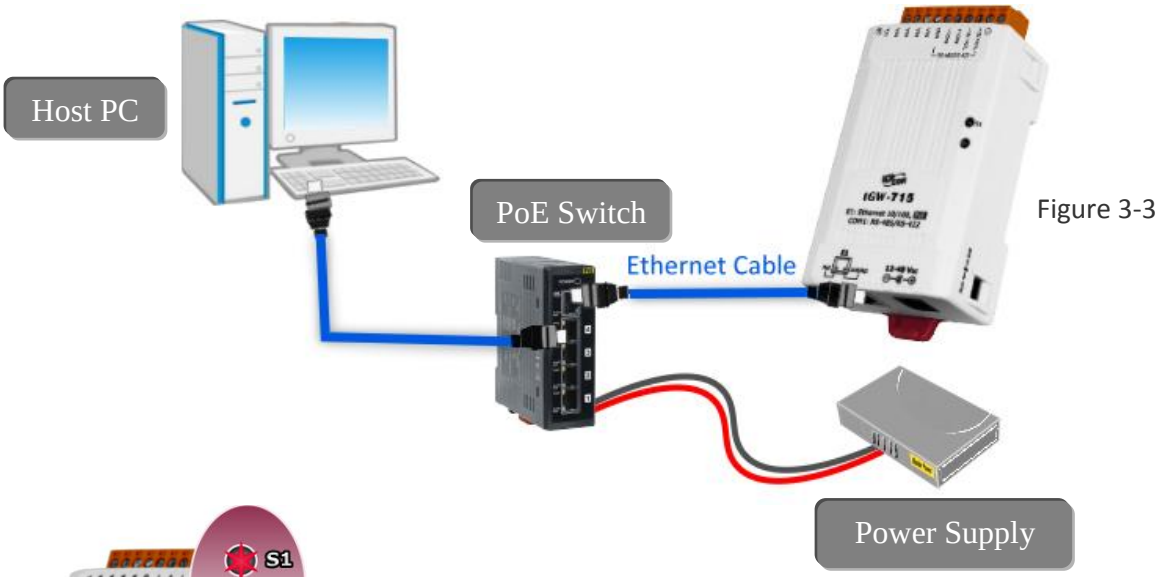


Figure 3-3

Figure 3-4

5. Verify that the System LED indicator is flashing.

Step 2: Install the the eSearch Utility

The eSearch Utility can be obtained from either the companion CD-ROM or the ICP DAS web site. The location of the install files on the CD and the download addresses are shown below:

CD:\\ NAPDOS\Software\eSearch\

http://ftp.icpdas.com/pub/cd/tinymodules/napdos/software/esearch/

Step 3: Search for the tGW-700 series module on the Ethernet network

1. Open the eSearch Utility and then click the **"Search Servers"** button to search for the tGW-700 module.
2. Once the search process is complete, double-click the name of the tGW-700 module to open the **"Configure Server"** dialog box.

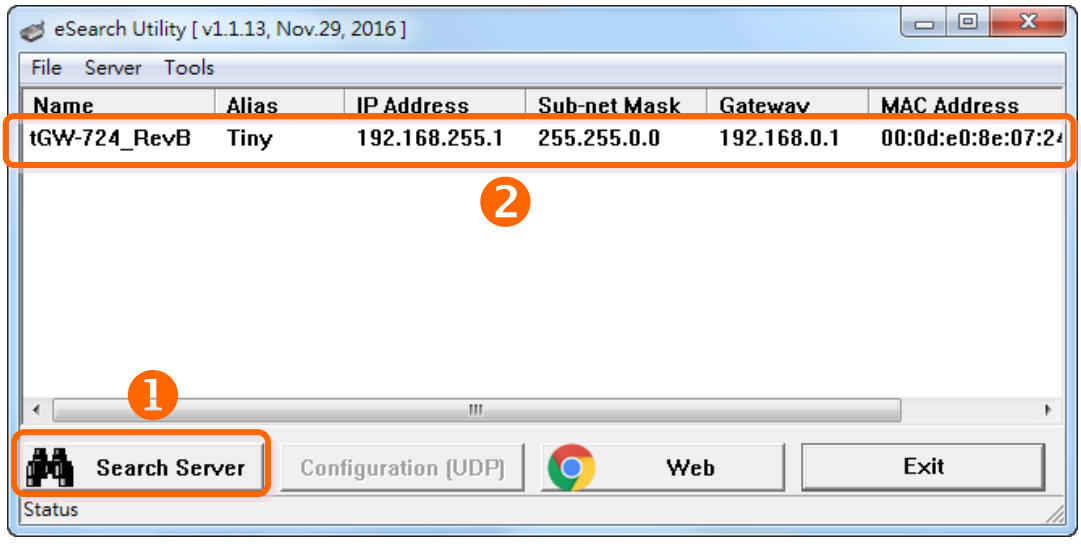


Figure 3-5

3. Enter the network settings information, including the **IP, Mask and Gateway addresses**, and then click **"OK"** button. The new settings for the tGW-700 will take effect within 2 seconds. If you don't know the correct network configuration information, contact your Network Administrator to obtain the details.

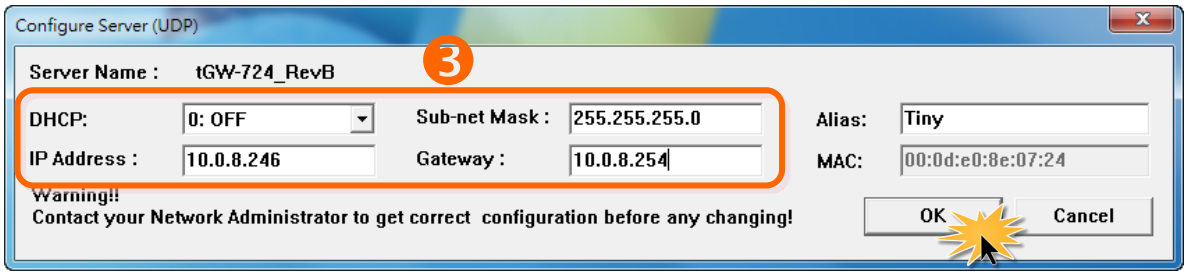


Figure 3-6

Step 4: Testing the tGW-700 series module

1. Connect the tGW-700 series module with Modbus device (e.g., M-7055D, optional) using the RS-485 bus. For more detailed information related to wiring options for RS-232/422/485, refer to Section 2.8 Wiring Notes for RS-232/485/422 Interfaces.

➢ **RS-232 Wiring**

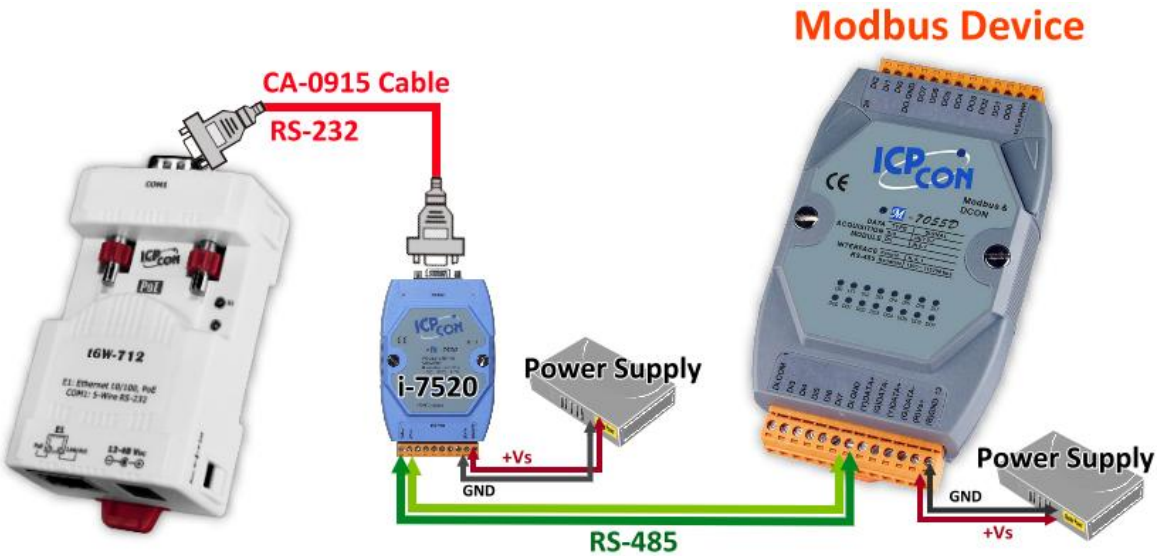


Figure 3-7

➢ **RS-485 Wiring**

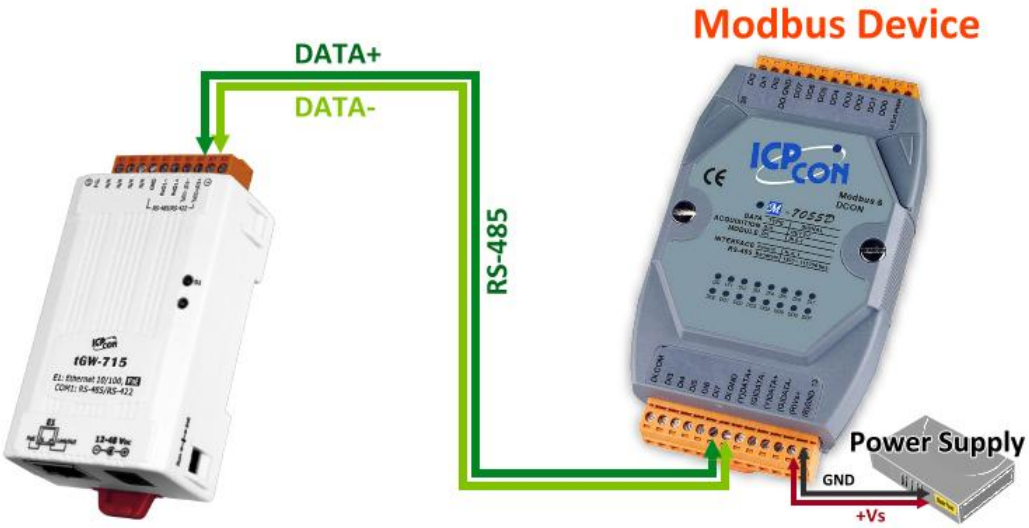
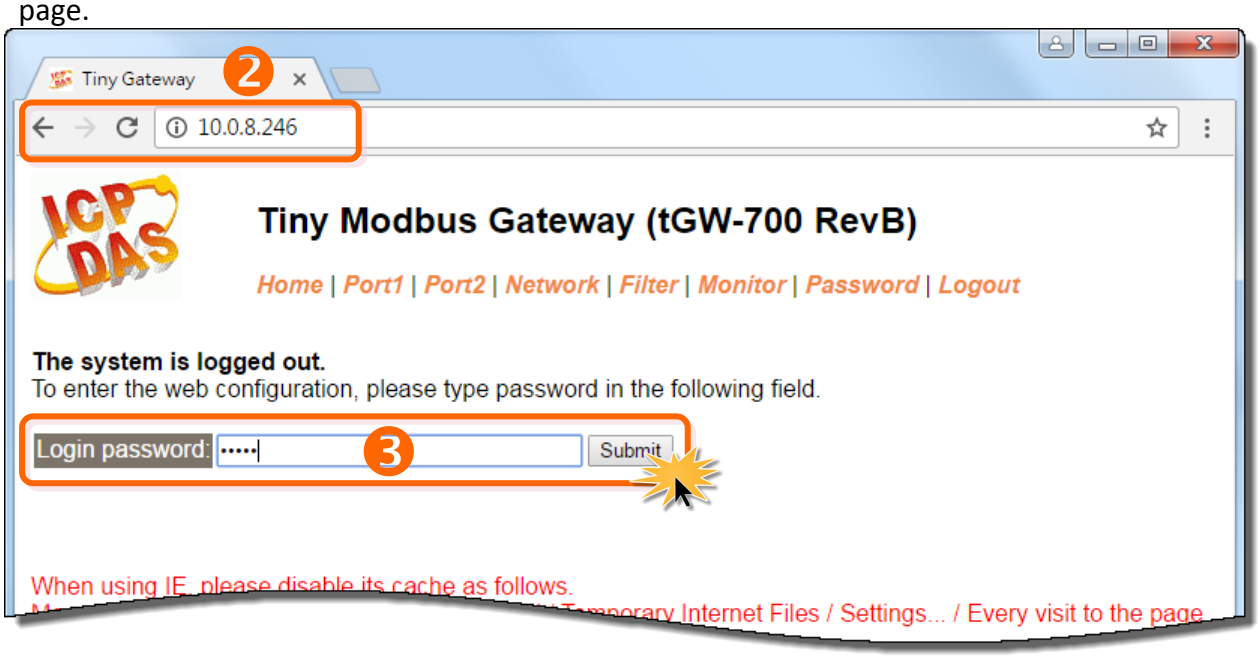


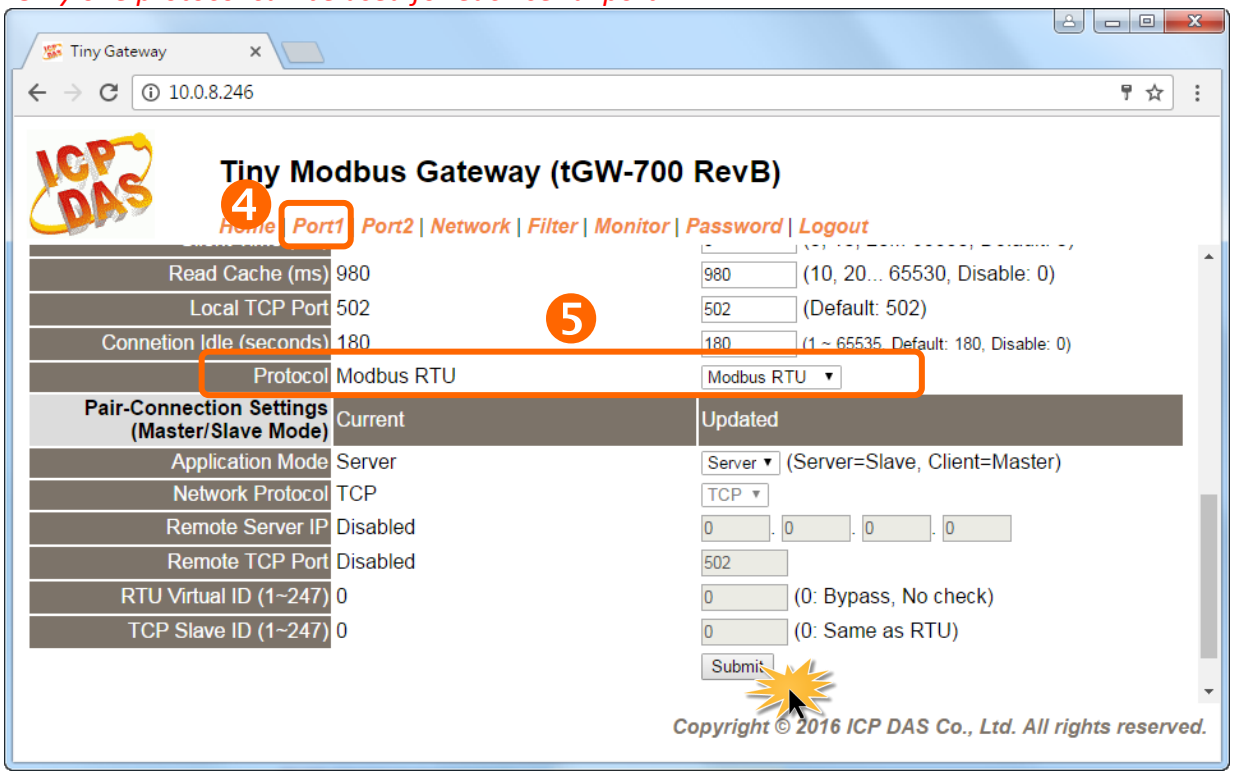
Figure 3-8

2. Open a web browser, such as Google Chrome, Internet Explorer, or Firefox, and enter the URL for the tGW-700 series module in the address bar of the browser, or click the button in the eSearch Utility.

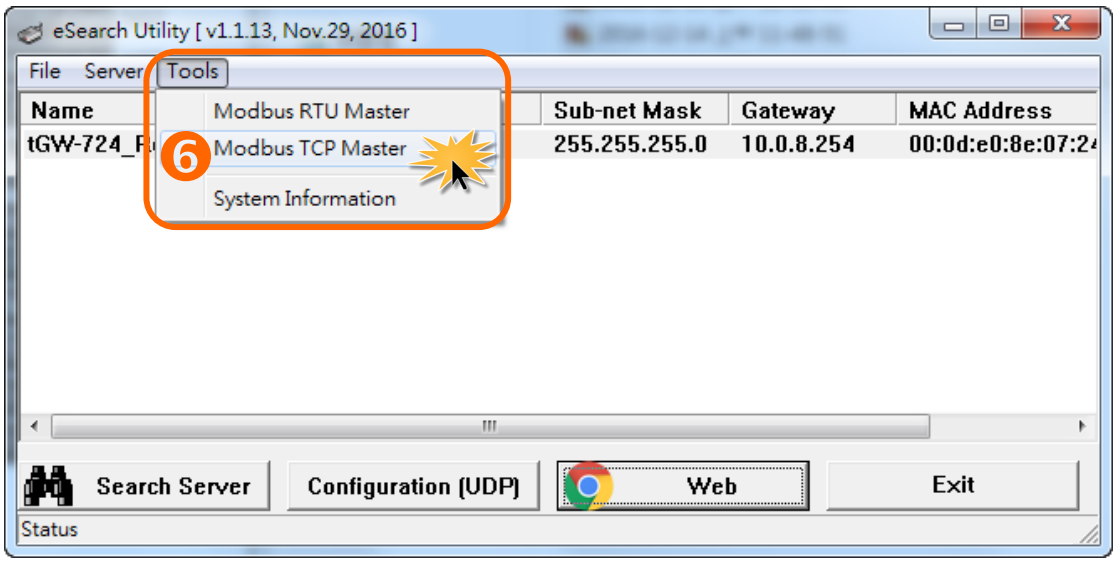
3. When the login screen is displayed, enter the password (use the default password: **admin**) in the login password field, and then click the **"Submit"** button to enter the configuration web page.



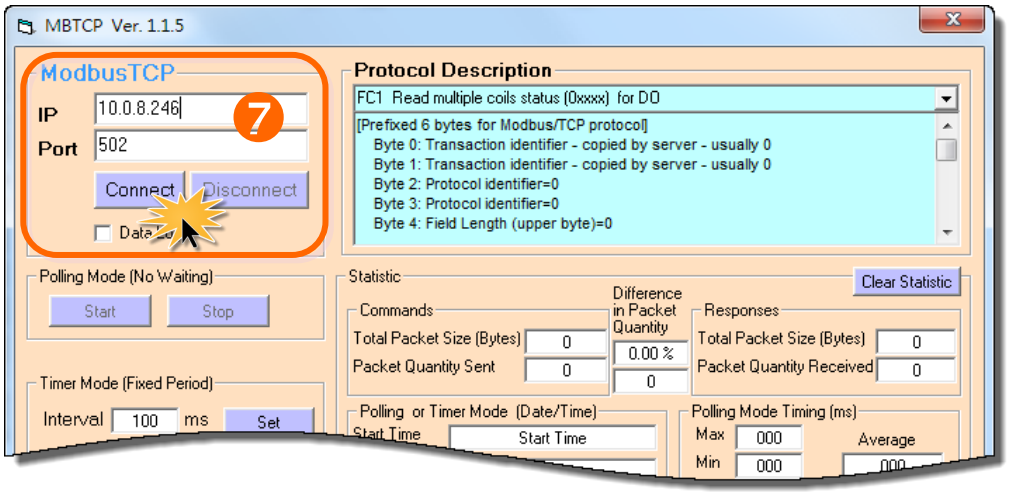
4. Click the **"Port1"** tab to display the Port1 Settings page.
5. Form the **"Protocol"** drop-down menu, select the protocol, either **"Modbus RTU or Modbus ASCII"** that is used by the attached Modbus devices, and then click the **"Submit"** button. *Note: Only one protocol can be used for each serial port.*



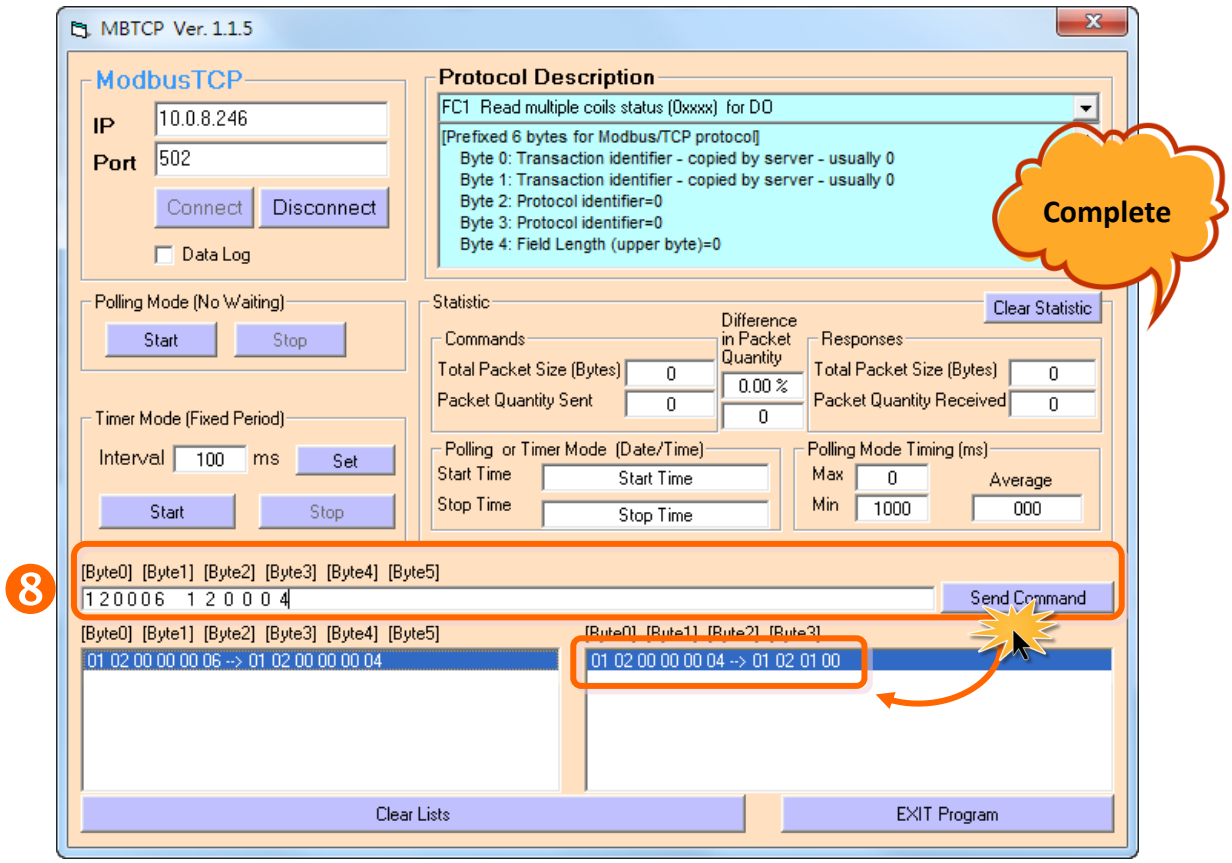
6. In the eSearch Utility, select the **"Modbus TCP Master"** item from the **"Tools"** menu to open the Modbus TCP Master Utility.



7. Enter the **IP address** and **TCP Port** information for the tGW-700 series module in the **"Modbus TCP"** section, and then click the **"Connect"** button to connect to the tGW-700.



8. Refer to the **"Protocol Description"** field in the top right-hand section of the Modbus Utility windows. You can send a request command and confirm that the response is correct. For example, if the Modbus device ID for the M-7000 is 1, send the command "1 2 0 0 0 6 1 2 0 0 0 4" to read the D/I value, the response will be "1 2 0 0 0 4 1 2 1 0".



4. Web Configuration

Once the tGW-700 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.

4.1 Logging in to the tGW-700 Web Server

The embedded tGW-700 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, Firefox or Internet Explorer, which are reliable and popular Internet browsers that can be used to configure tGW-700 series module.

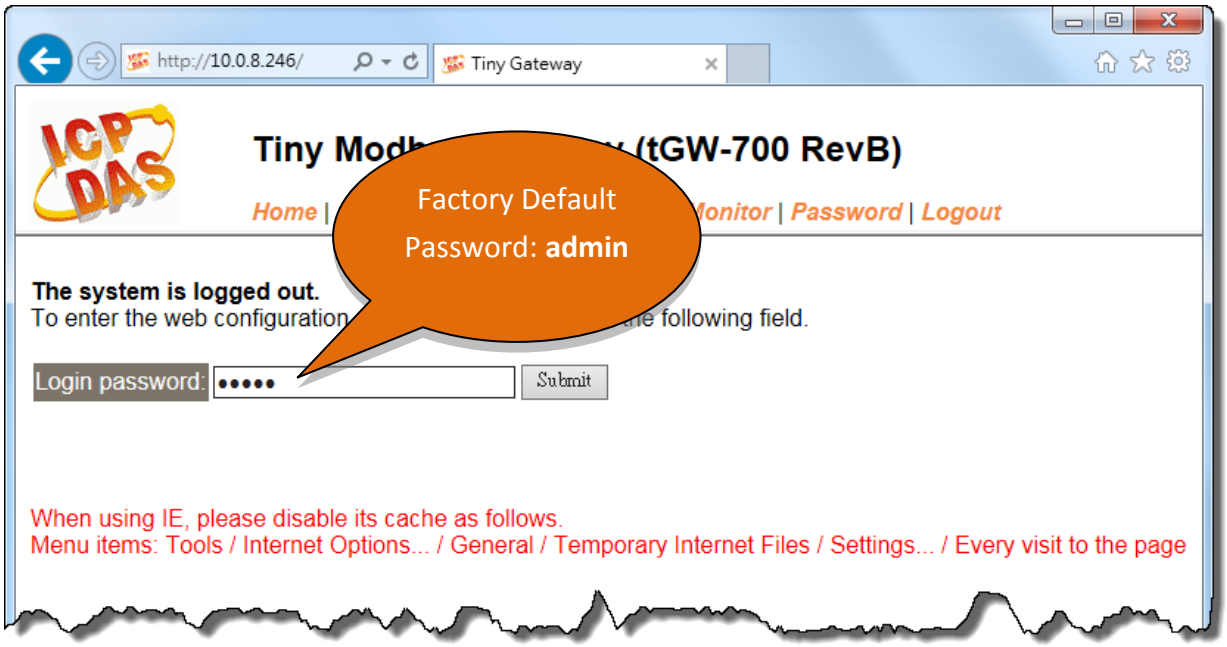
Note that if you intend to use Internet Explorer, ensure that the cache function is disabled in order to prevent browser access errors. Detailed instructions for how to do this can be found in "FAQ: How to avoid a browser access error that causes a blank page to be displayed when using Internet Explorer".

Step 2: Enter the URL for the tGW-700 web server

Ensure that you have correctly configured the network settings for the tGW-700 series module (refer to Chapter 3 Setting up the tGW-700 module for detailed instructions), and then enter the URL for the tGW-700 web server in the address bar of the browser.

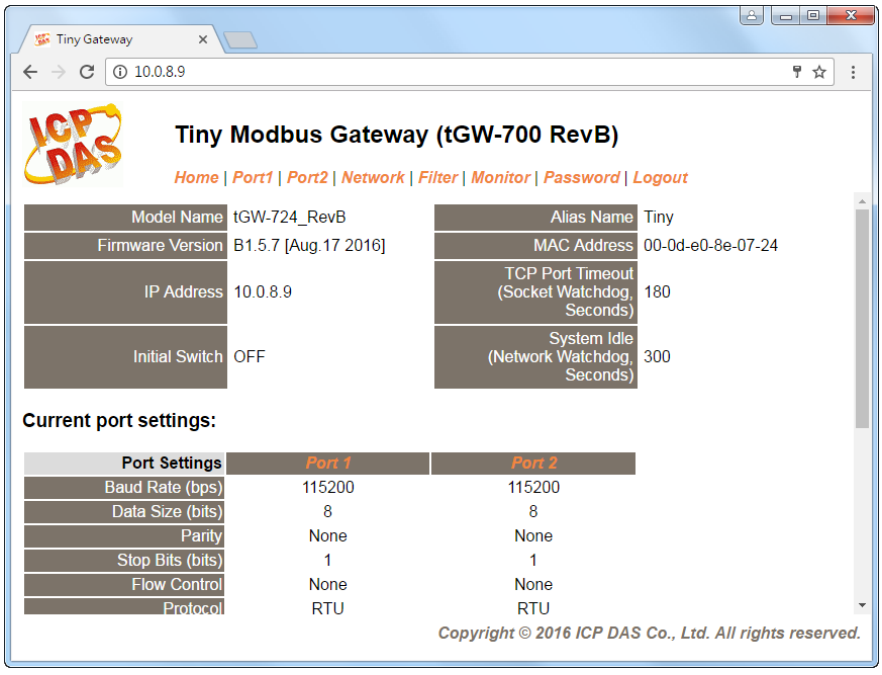
Step 3: Enter the Password

After the main login page is displayed, enter a password (the factory default password is "**admin**"), and then click the "**Submit**" button to continue.



Step 4: Log in to the tGW-700 Web Server

After logging into the tGW-700 web server, the main page will be displayed.



4.2 Home Page

The Home link connects to the main page, which contains two parts.

The first part of this page provides basic information about the tGW-700 hardware and software.

Model Name	tGW-724_RevB	Alias Name	Tiny
Firmware Version	B1.5.7 [Aug.17 2016]	MAC Address	00-0d-e0-8e-07-24
IP Address	10.0.8.246	TCP Port Timeout (Socket Watchdog, Seconds)	180
Initial Switch	OFF	System Idle (Network Watchdog, Seconds)	300

The software and hardware information section includes information related to the Model Name, the current Firmware version, the IP Address, the current position of the Initial Switch, the Alias, the MAC Address, and the TCP Port, and the System Timeout values. **If you update the firmware for the tGW-700 module, this page can be used to check the version information** of the tGW-700 software.

The lower section provides information related to the port settings and the pair-connection settings.

Current port settings:

Port Settings	*Port 1*	*Port 2*
Baud Rate (bps)	115200	115200
Data Size (bits)	8	8
Parity	None	None
Stop Bits (bits)	1	1
Flow Control	None	None
Protocol	RTU	RTU
Slave Timeout (ms)	300	300
Char Timeout (bytes)	4	4
Silent Time (ms)	0	0
Read Cache (ms)	980	980
Local TCP Port	502	503
Connetion Idle (Seconds)	180	180
Pair-Connection Settings (Master/Slave Mode)	Port 1	Port 2
Application Mode	Server	Server
Remote Server IP	-	-
Remote TCP Port	-	-
RTU Virtual ID	-	-
TCP Slave ID	-	-

4.3 Network Page

After clicking the **Network** tab, the **Network** page will be displayed, allowing you to verify the current settings, configure the IP Address, and the general parameters, and restore the default settings for the tGW-700 module, each of which will be described in more detail below.

4.3.1 IP Address Selection

The **Address Type, Static IP 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 tGW-700 module will not operate correctly. If the settings are changed while the module is operating, any connection currently in use will be lost and an error will occur.

IP Address Selection

Address Type	Static IP
Static IP Address	10 . 0 . 8 . 246
Subnet Mask	255 . 255 . 255 . 0
Default Gateway	10 . 0 . 8 . 254
MAC Address	00-0d-e0-8e-07-24 (Format: FF-FF-FF-FF-FF-FF)
Gateway Net ID	255 (Default: 255) Note: This is reserved for gateway, NOT for slave devices.
	Update Settings

The following is an overview of the parameters contained in the IP Address configuration 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 4.3.1.1 Manual Configuration for more details.
	DHCP: The Dynamic Host Configuration Protocol (DHCP) is a network application protocol that automatically assigns an IP address to each device. Refer to Section 4.3.1.2 Dynamic Configuration for more details.
Static IP Address	Each tGW-700 connected to the network must have its own unique IP address. This parameter is used to assign a specific IP address.
Subnet Mask	This parameter is used to assign the subnet mask for the tGW-700. The subnet mask indicates which portion of the IP address is used to identify the local network or subnet.
Default Gateway	This parameter is used to assign the IP Address of the Gateway to be used by the tGW-700. 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.
Modbus Net ID (for Gateway)	This parameter is used to set the device ID to be used by the tGW-700. The default value is 255.
Update Settings	Click this button to save the revised settings to the tGW-700.

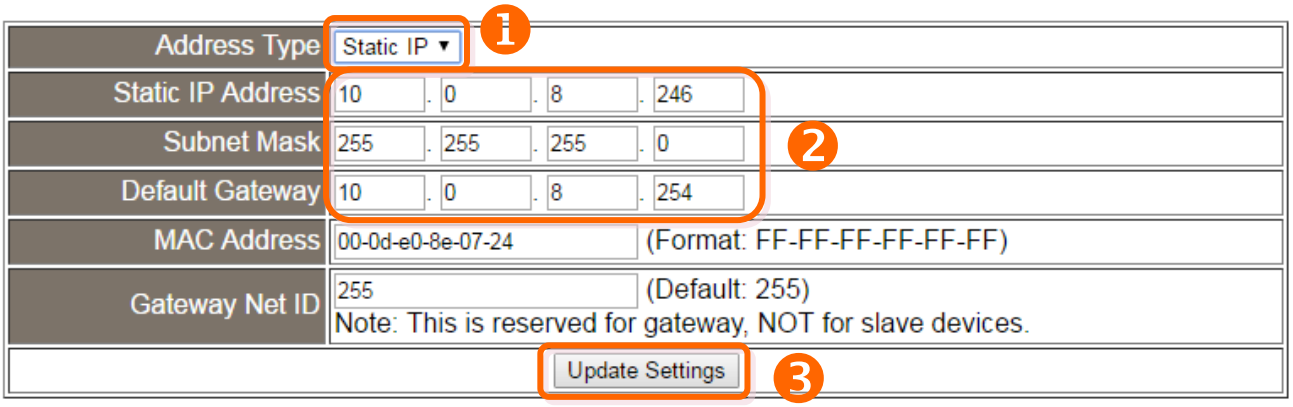
4.3.1.1 Manual Configuration

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

Step 1: Select the **"Static IP"** option from the **"Address Type"** drop-down menu.

Step 2: Enter the relevant details in the respective **network settings** fields**.**

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

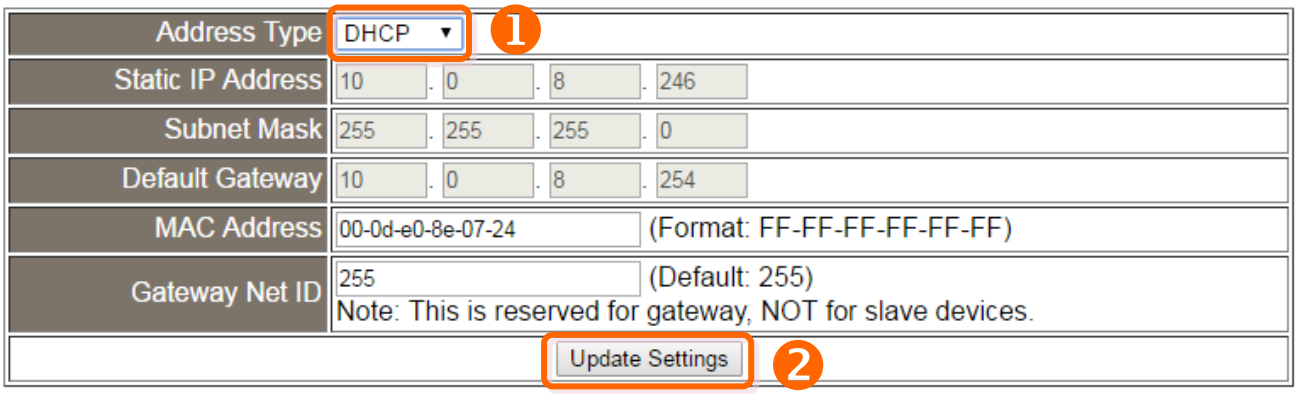


4.3.1.2 Dynamic Configuration

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

Step 1: Select the **"DHCP"** option from the **"Address Type"** drop-down menu.

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



4.3.2 General Settings

General Settings

Ethernet Speed	Auto (Auto=10/100 Mbps Auto-negotiation)
Alias Name	Tiny (Max. 18 chars)
System Timeout (Network Watchdog)	300 (30 ~ 65535 seconds, Default: 300, Disable: 0)
Web Auto-logout	10 (1 ~ 65535 minutes, Default: 10, Disable: 0)
UDP Configuration:	Enable (Enable/Disable the UDP Configuration, Enable=default.)
Protocol Exception	1 (Default: 1, Disable: 0, Enable: 1) Reports exception 0x41 when slave response is invalid Modbus message.
CRC Exception	1 (Default: 1) 0: Gateway returns raw data including CRC when CRC error. 1: Gateway reports exception 0x43 when CRC error. 2: Gateway drops packet when CRC error.
Timeout Exception	1 (Default: 1, Disable: 0, Enable: 1) Gateway reports exception 0x0B for slave no response, and 0x4B for data timeout.
Busy Exception	1 (Default: 1, Disable: 0, Enable: 1) Gateway reports exception 0x06 when queued requests are full.
Check TCP Header	1 (Default: 1, Disable: 0, Enable: 1) Drops packet when Modbus TCP header (protocol ID, length) is wrong.
Update Settings	

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

Item	Description	Default
Ethernet Speed	This parameter is used to set the Ethernet speed. The default value is Auto (Auto = 10/100 Mbps Auto-negotiation).	Auto
Alias Name	This parameter is used to assign an alias for each tGW-700 device to assist with easy identification.	Tiny
System Timeout (Network Watchdog)	This parameter is used to configure the system timeout value. If there is no activity on the network for a specific period of time, the system will be rebooted based on the configured system timeout value. Timeout value range: 30 to 65535 (seconds); Disable = 0;	0
Web Auto-logout	This parameter is used to configure the automatic logout value. If there is no activity on the web server for a certain period of time, the current user account will be automatically logged out. Range: 1 to 65535 (minutes); Disable = 0.	10

Item	Description	Default
UDP Configuration	This parameter is used to enable or disable UDP configuration function.	1
Protocol Exception	This parameter is used to enable or disable whether the slave response is checked for compatibility with the Modbus RTU format. If the slave response is an invalid Modbus message, a 0x41 exception code will be reported. Enable =1; Disable = 0.	1
CRC Exception	This parameter is used to enable or disable whether the validity of the RTU/ASCII CRC of the slave response is checked. 0 = Returns the raw data, including the CRC, if a CRC error occurs; 1 = Reports a 0x43 exception code if a CRC error occurs; 2 = Drops the packet if a CRC error occurs.	1
Timeout Exception	This parameter is used to enable or disable whether a slave/data timeout exception error is reported by the Gateway. If There is no response from a slave device, a 0x0B exception error will be reported. If serial data is being received, a 0x4B exception will be reported. Enable =1; Disable = 0.	1
Busy Exception	This parameter is used to enable or disable whether a busy exception code (0x06) is reported if the Gateway request queue is full. Enable =1; Disable = 0.	1
Check TCP Header	This parameter is used to enable or disable the drop-packet function when the Modbus TCP header is invalid. Enable = 1; Disable = 0.	1
Update Settings	Click this button to save the revised settings to the tGW-700 device.	

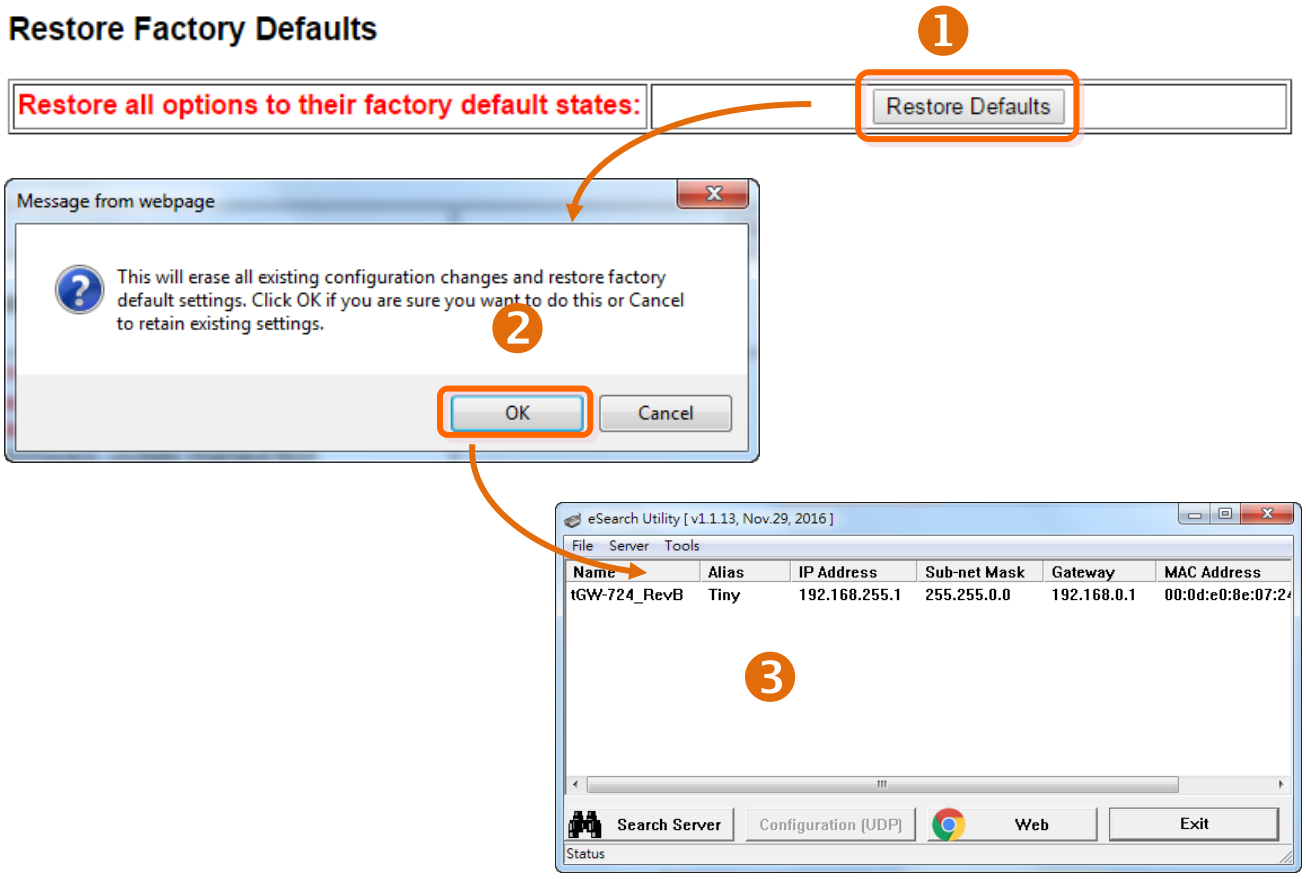
4.3.3 Restore Factory Defaults

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

Step 1: Click the **"Restore Defaults"** button to reset the configuration.

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

Step 3: Check whether the module has been reset to the original factory default settings for use with the eSearch Utility. Refer to Chapter 3 Setting up the tGW-700 Module for more details.



The following is an overview of the factory default settings:

Factory Default Settings			
Network Settings		**Basic Settings**	
IP Address	192.168.255.1	Alias	Tiny
Gateway Address	192.168.0.1		
Subnet Mask	255.255.0.0		
DHCP	Disabled		

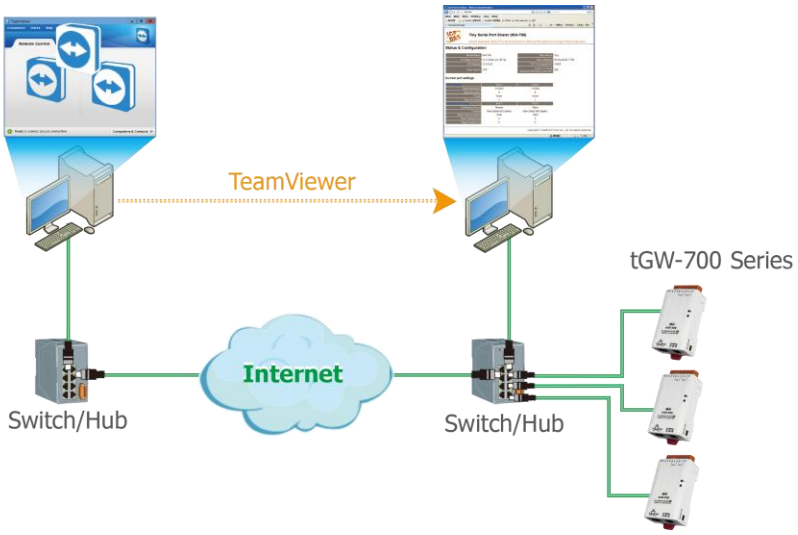
The **Forced Reboot** function: can be used to force the tGW-700 to reboot or to remotely reboot the device. After the tGW-700 module has rebooted, the original login screen will be displayed requesting that you enter your Login Password before continuing.

4.3.4 Update by Ethernet

Update by Ethernet

If the remote firmware update is failed, then the traditional firmware update (on-site) is required to make the module working again. Step 1: Refer to firmware update manaul first. Step 2: Run eSearch Utility to prepare and wait for update. Step 3: Click the [**Update**] button to **reboot** the module and start update. Step 4: Configure the module again.	Update

Firmware update requires initialization and local network operations. Traditional firmware update requires adjusting the Init/Run Switch and reboots the module manually for the initialization of firmware update, while new firmware 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 via TeamViewer.



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

For detailed information regarding how to use this function to update the Firmware for your tGW-700 series module, refer to the **tGW_Firmware_v146_and_later_Update_vxxx_en.pdf.** The location of the user manual on the CD and the download address are shown below:

	CD:\\ NAPDOS\tGW-700\Firmware\
	http://ftp.icpdas.com/pub/cd/tinymodules/napdos/tgw-700/firmware/

4.4 Serial Port Page

After clicking the **Port1** tab, the serial port settings page will be displayed, allowing you to configure the settings for the tGW-700, including the Baud Rate, Data Format, Slave Timeout, Char Timeout, Silent Time, Read Cache, TCP Timeout, Modbus Protocol and Pair-connection parameters, etc., each of which will be described in more detail below.

4.4.1 Settings (Port Settings)

Settings:

Port 1 Settings

Port Settings	Current	Updated
Baud Rate (bps)	115200	115200 ▾ bits/S
Data Size (bits)	8	8 ▾ bits/character
Parity	None	None ▾
Stop Bits (bits)	1	1 ▾
Flow Control	None	None ▾ (for tGW-712/722)
Slave Timeout (ms)	300	300 (Default: 300)
Char Timeout (bytes)	4	4 (4 ~ 15, Default: 4)
Silent Time (ms)	0	0 (0, 10, 20... 65530, Default: 0)
Read Cache (ms)	980	980 (10, 20... 65530, Disable: 0)
Local TCP Port	502	502 (Default: 502)
Connetion Idle (seconds)	180	180 (1 ~ 65535, Default: 180, Disable: 0)
Protocol	Modbus RTU	Modbus RTU ▾
Pair-Connection Settings (Master/Slave Mode)	Current	Updated
Application Mode	Server	Server ▾ (Server=Slave, Client=Master)
Network Protocol	TCP	TCP ▾
Remote Server IP	Disabled	0 . 0 . 0 . 0
Remote TCP Port	Disabled	502
RTU Virtual ID (1~247)	0	0 (0: Bypass, No check)
TCP Slave ID (1~247)	0	0 (0: Same as RTU)
		Submit

The following is an overview of the parameters contained in the Settings – Port Settings section:

Item	Description	Default
Port Settings		
Baud Rate (bps)	This parameter is used to set the Baud Rate for the COM ports.	115200
Data Size (bits)	This parameter is used to set the Data Size for the COM ports.	8
Parity	This parameter is used to set the Parity for the COM ports.	None
Stop Bits (bits)	This parameter is used to set the Stop Bits for the COM ports.	1
Flow Control	This parameter is used to enable or disable hardware flow control (CTS/RTS) function for the tGW-712, tGW-722 and tGW-724 only.	None
Slave Timeout (ms)	This parameter is used to set the waiting time that should elapse after last character of the request sent to the device before the timeout is activated. The tGW-700 will report a 0x0B exception code if there is no response from the slave device before the timeout period expires, or will report a 0x4B exception code if the slave device is still sending data when the timeout is activated. The Slave Timeout value must be set to less than the response timeout configured in the client software.	300
Char Timeout (bytes)	Modbus RTU requires 3.5 char time between messages. This parameter is used to set the waiting time (based on bytes) that should elapse after last byte of data of the response is received from the slave device is activated. If no more data is received before the timeout period expires, then the transmission of this packet is deemed to have been completed and the tGW-700 begins processing the packet. Valid range: 4 to 15 (bytes);	4

Item	Description	Default
Silent Time (ms)	This parameter is used to set the idle time that should elapse before sending each request to the serial port. This causes the serial bus to be "silent" for the specified period, and allows slower slave devices more time to process previous requests and responses, thereby reducing communication problems. Valid range: 10, 20 to 65530 (ms);	0
Read Cache (ms)	When sharing Modbus RTU/ASCII device/data between several master devices, the read-cache function can be used to reduce the loading on the serial communication and ensure faster TCP responses. Valid range: 10, 20 to 65530 (ms); Disable = 0.	980
Local TCP Port	This parameter is used to configure the Modbus TCP port. *Note: The default COM1/COM2/COM3 = TCP Ports 502/503/504.*	502
Connection Idle (seconds)	If Modbus TCP communication is idle for a specified period of time, the system will automatically terminate the connection. Valid range: 1 to 65535 (seconds); Disable = 0;	180
Protocol	This parameter is used to configure the serial port that is used by the Modbus RTU or Modbus ASCII protocol.	Modbus RTU

4.4.2 Settings (Pair-Connection Settings)

The following is an overview of the parameters contained in the Settings – Pair-Connection Settings (Master/Slave Mode) section:

Pair-Connection Settings (Master/Slave Mode)		
Application Mode	Server	Client
Network Protocol	-	Select the Modbus protocol (Modbus TCP or UDP) for the remote device
Remote Server IP	Disabled	The IP address for the remote device
Remote TCP Port	Disabled	The TCP Port number for the remote device
RTU Virtual ID (1~247)	Disabled	The Modbus RTU Slave ID of the tGW-700
TCP Slave ID (1~247)	Disabled	The Modbus TCP Slave ID of the remote device
Update Settings	Click this button to save the revised settings to the tGW-700.	

Note

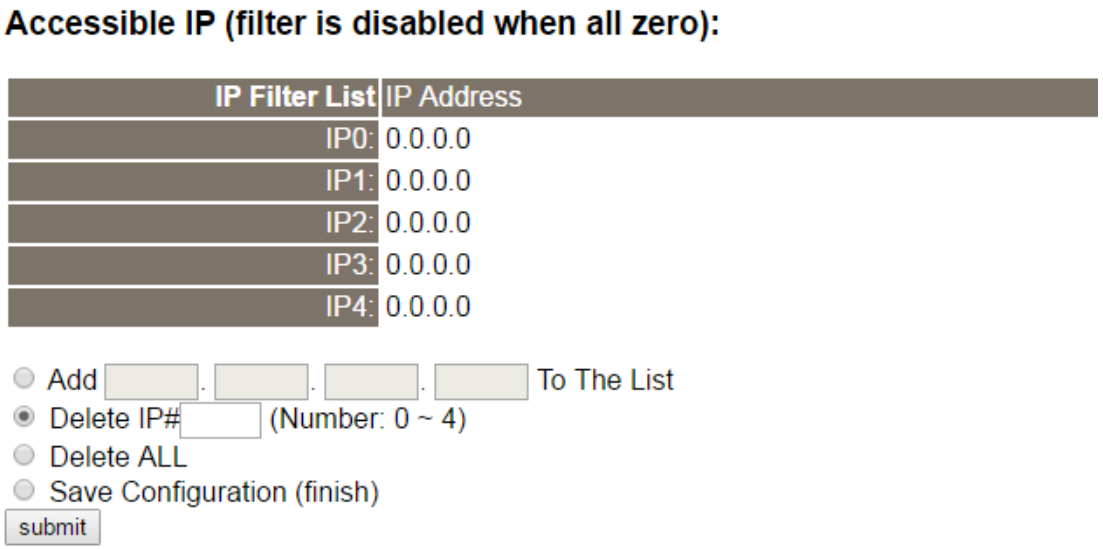
1. For more detailed information regarding pair-connection applications settings, refer to Section 5.3 Pair-Connection Applications.

2. For more detailed information regarding the mapping configuration for the Modbus RTU Slave ID and the TCP Slave ID, refer to "FAQ: How to access multiple Modbus TCP slave devices from a single Modbus RTU/ASCII master device."

4.5 Filter Page

4.5.1 Accessible IP (filter is disabled when all zero)

The Accessible IP Settings 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 one or more IP address are saved to the IP Filter table, only clients whose IP is specified in the IP Filter List can access the tGW-700.



The following is an overview of the parameters contained in the Accessible IP section:

Item	Description
Add "IP" to the list	Add an IP address to the IP Filter List.
Delete IP# "Number"	Delete a specific IP# address from the IP Filter List. (Number = 0 to 4)
Delete All	Delete all items from the IP Filter List.
Save Configuration (finish)	Save a new IP Filter List to the Flash memory.
Submit	Click this button to save the revised settings to the tGW-700.

4.6 Monitor Page

Tiny Modbus Gateway (tGW-700 RevB)

Home | Port1 | Port2 | Network | Filter | Monitor | Password | Logout

After clicking the **Monitor** tab, the Current Connection Status page will be displayed showing detailed information regarding the current status of the serial port connection settings for the tGW-700 module.

Current Connection Status:

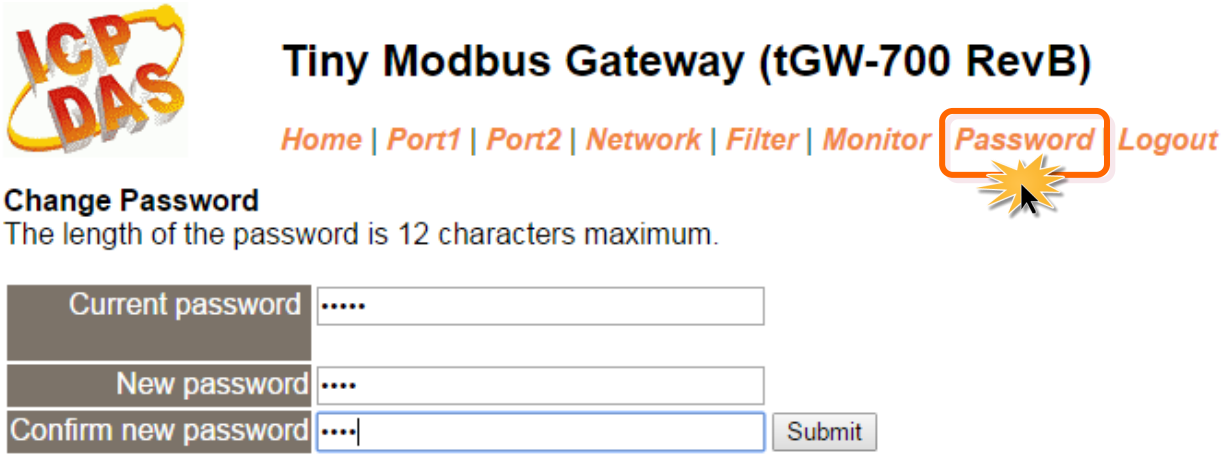
Port Number	Port 1	Port 2
Application Mode	Server	Server
Connected IP1:	0.0.0.0	0.0.0.0
IP2:	0.0.0.0	0.0.0.0
IP3:	0.0.0.0	0.0.0.0
IP4:	0.0.0.0	0.0.0.0
Available Connections:	32	32
Queued MB Requests:	0	0
Busy Error:	-	-
Last Error:	0,0,0	0,0,0

Clear Last Error

Note:
1. *CLICK HERE* for error codes and descriptions.
2. The "**Busy Error**" can happen when too many Modbus requests are queued and waiting for process. Set a larger timeout and scan-time value on all master software (clients) for fixing this problem.

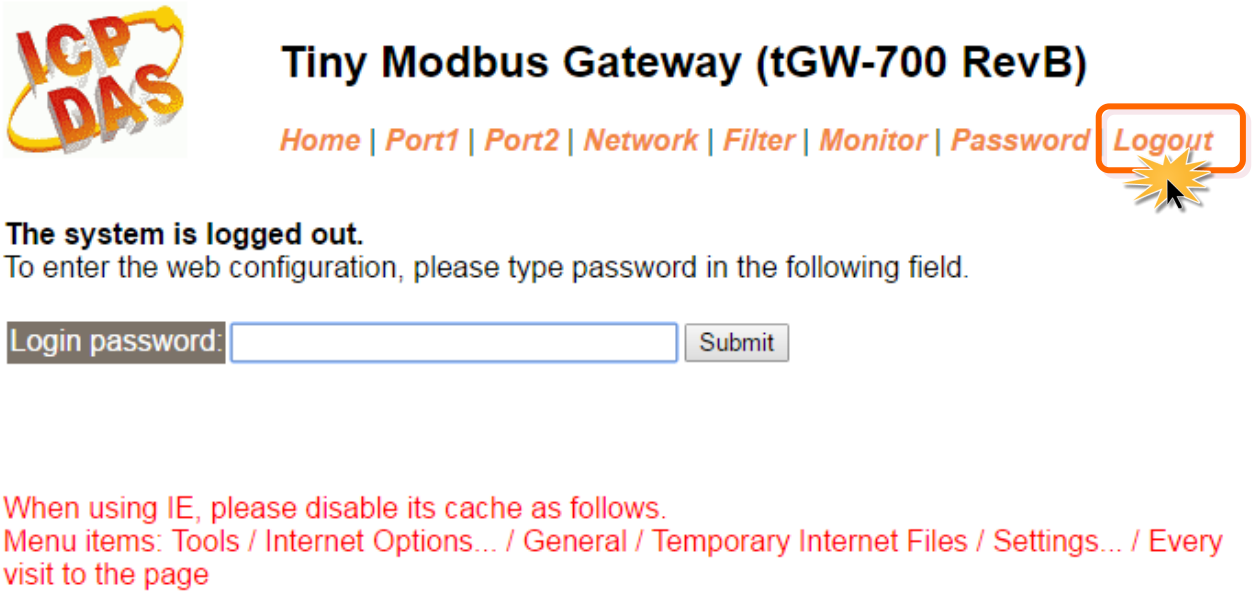
4.7 Password Page

After clicking the **Password** tab, the **Change Password** page will be displayed. To change a password, first enter the old password in the **"Current password"** field (use the default password **"admin"**) and then enter a new password in the **"New password"** field. Re-enter the new password in the **"Confirm new password"** field, and then click the **"Submit"** button to update the password.



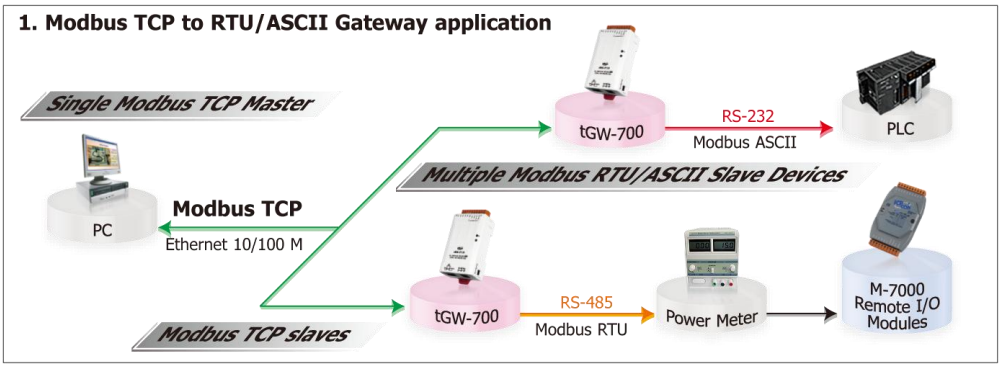
4.8 Logout Page

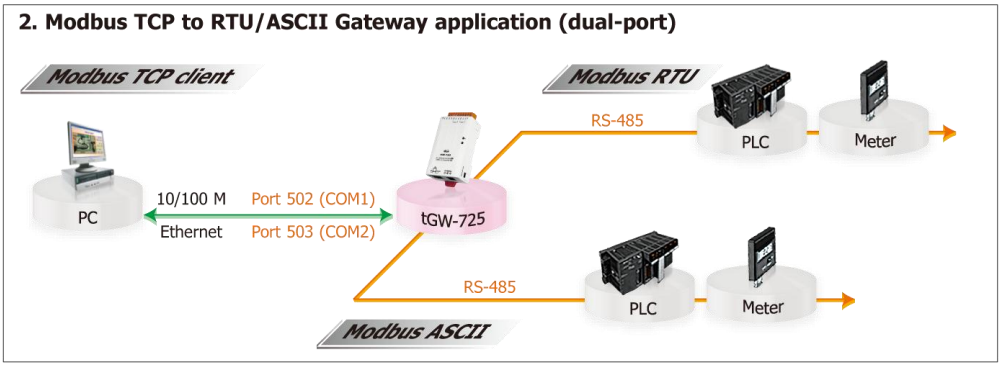
After clicking the **Logout** tab, you will be immediately logged out from the system and be returned to the login page.

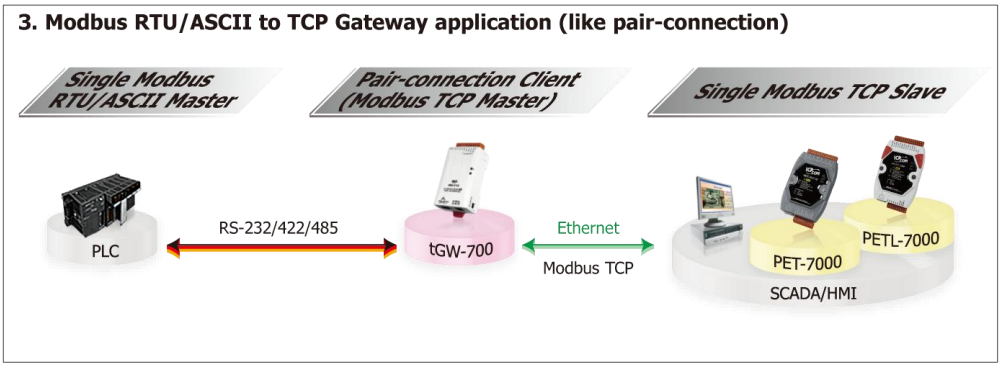


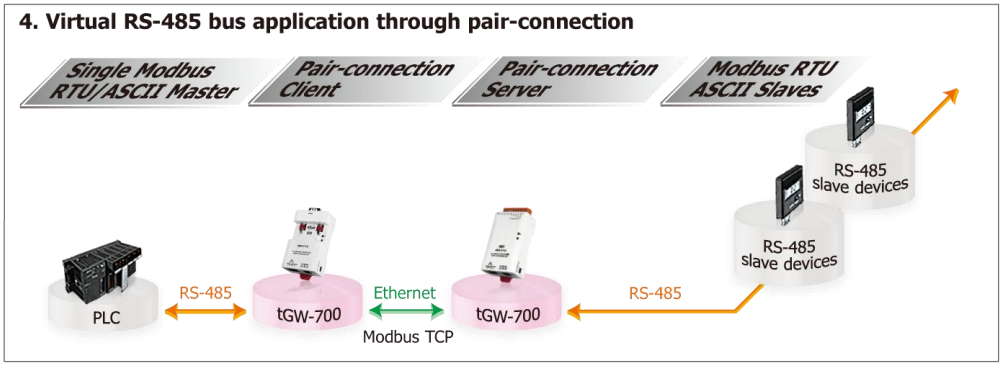
5. Typical Applications

This chapter provides some examples of typical scenarios for the tGW-700 series module, including applications focused on the Modbus Gateway, Modbus Net ID, Pair-connection and TCP Client Mode, etc..



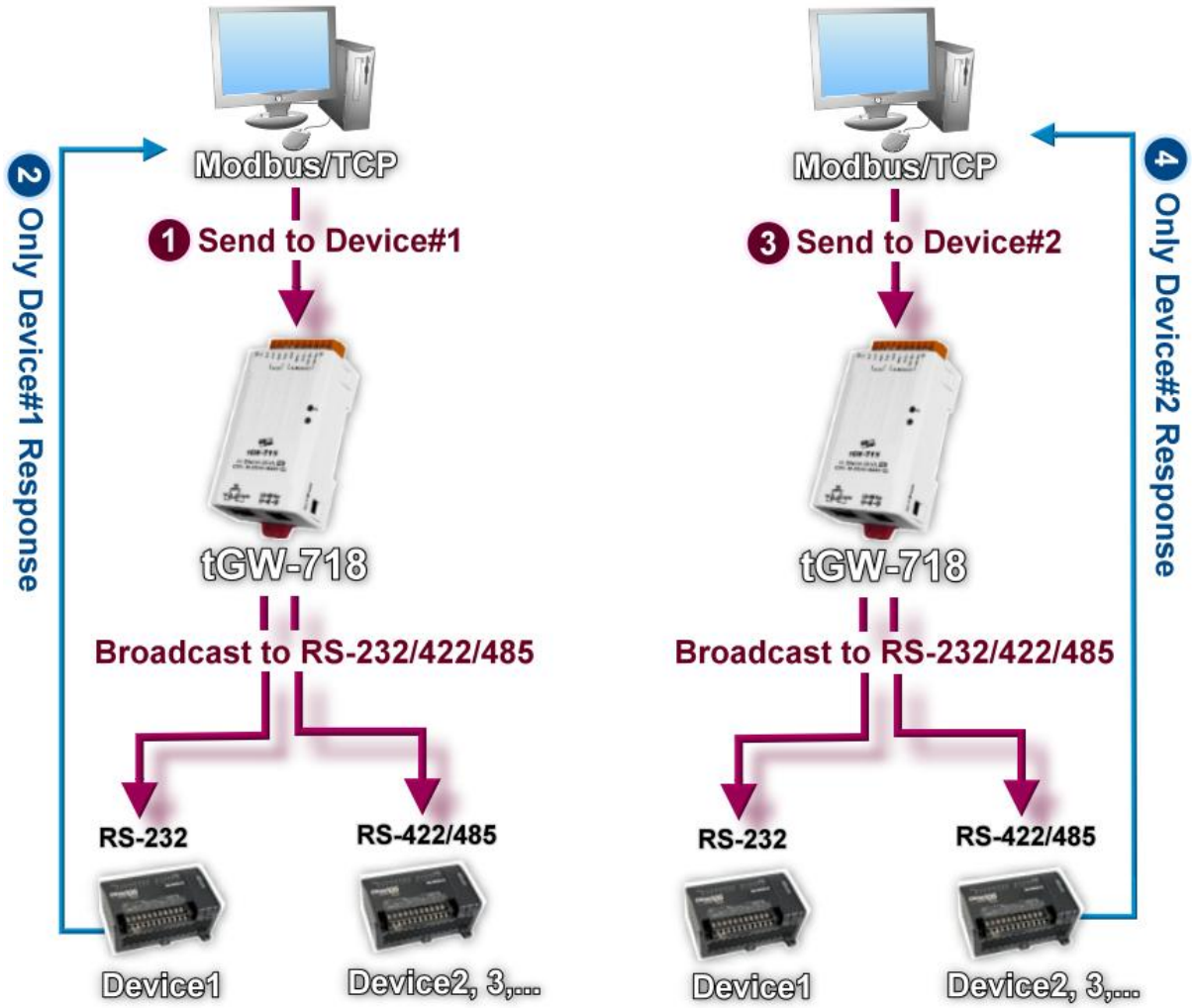






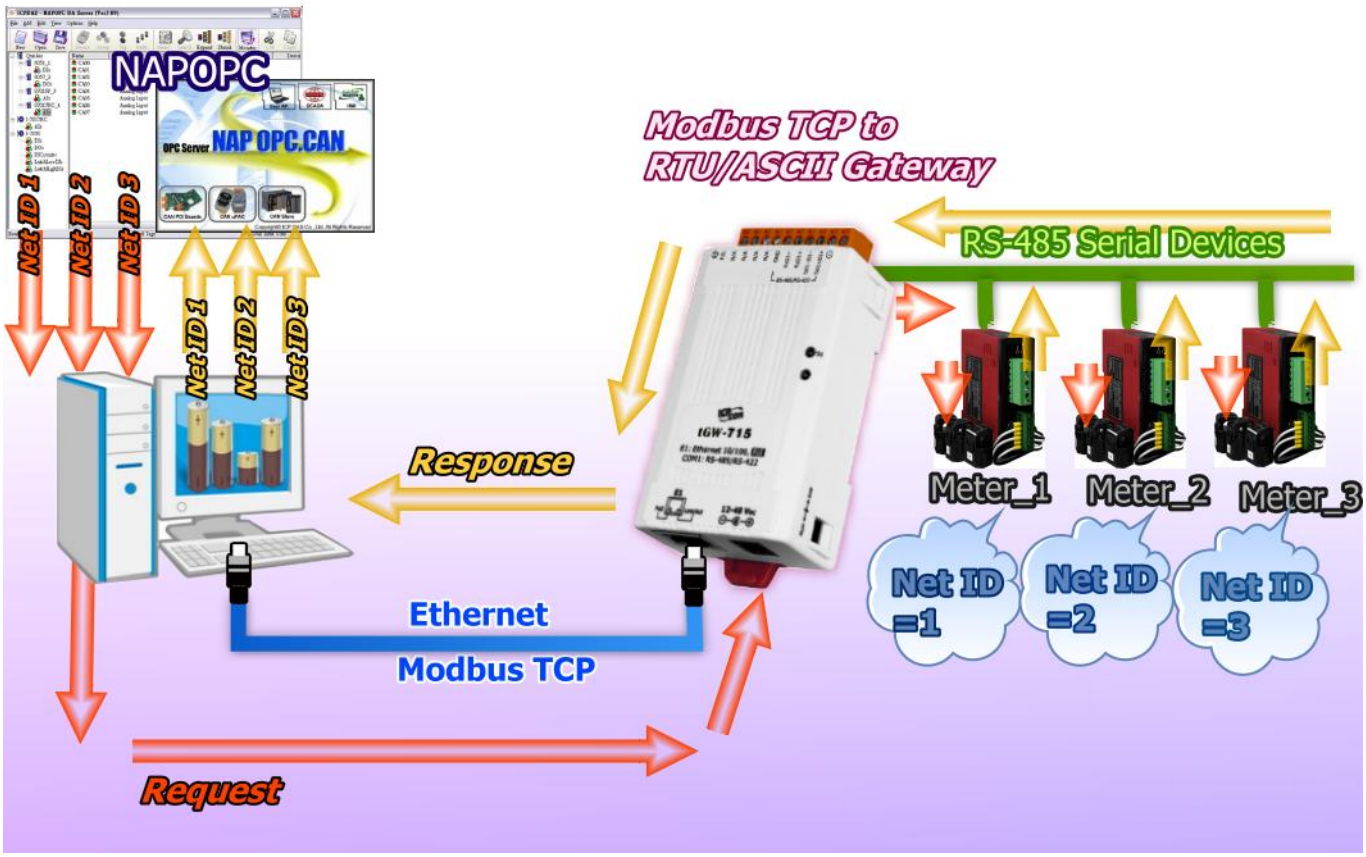
5.1 Modbus Gateway

The tGW-700 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.



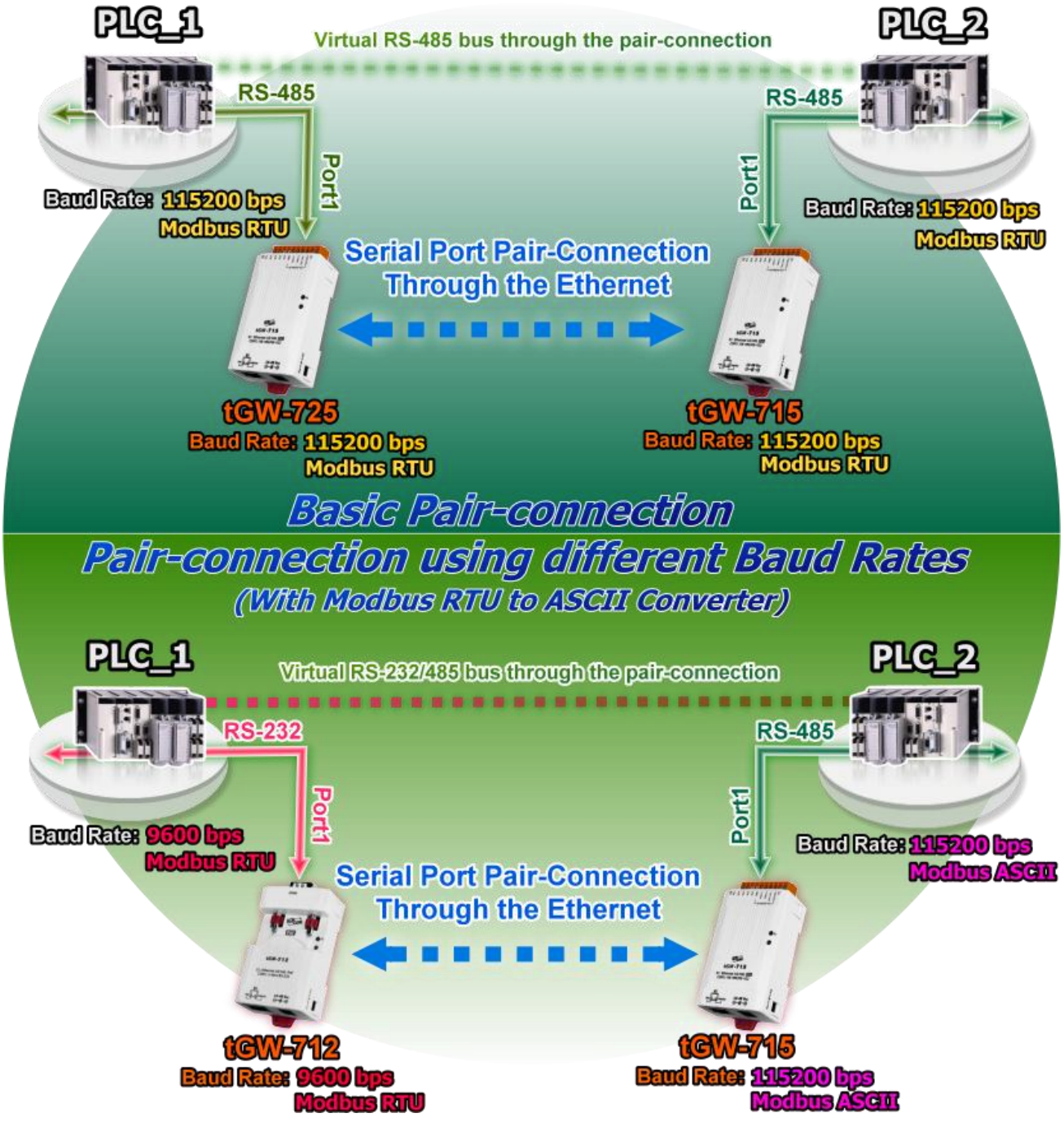
5.2 Modbus Net ID

The tGW-700 series 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 tGW-700 gateway by specifying correct NetID of the intended slave device in each Modbus TCP request. *Note that the NetID of the tGW-700 gateway is reserved for specific control purposes, and is not used to access slave devices.*



5.3 Pair-connection Applications

The tGW-700 Modbus gateway can be used to create a pair-connection applications (as well as serial-bridge or serial-tunnel), and then route Modbus messages between two serial devices via TCP/IP, which is useful when connecting Modbus RTU/ASCII devices that do not themselves have Ethernet capability.



The following are examples of pair-connection tests:

Pair-connection Settings:

Model	Port Settings (default)		Pair-connection Settings		
	Baud Rate	Data Format	Server Mode	Remote Server IP	Remote TCP Port (default)
tGW-700 #1	115200	8N1	Client	IP Address of tGW-700 #2	502
tGW-700 #2	115200	8N1	Server	-	-

Note

The Baud Rate and Data Format settings of the client and server (tGW-700 #1 and #2) depend on the COM ports of the PC (or the connected device). The serial port settings between tGW-700 #1 and tGW-700#2 can be different.

Follow the procedure described below:

Step 1: Connecting to a network, PC and Power

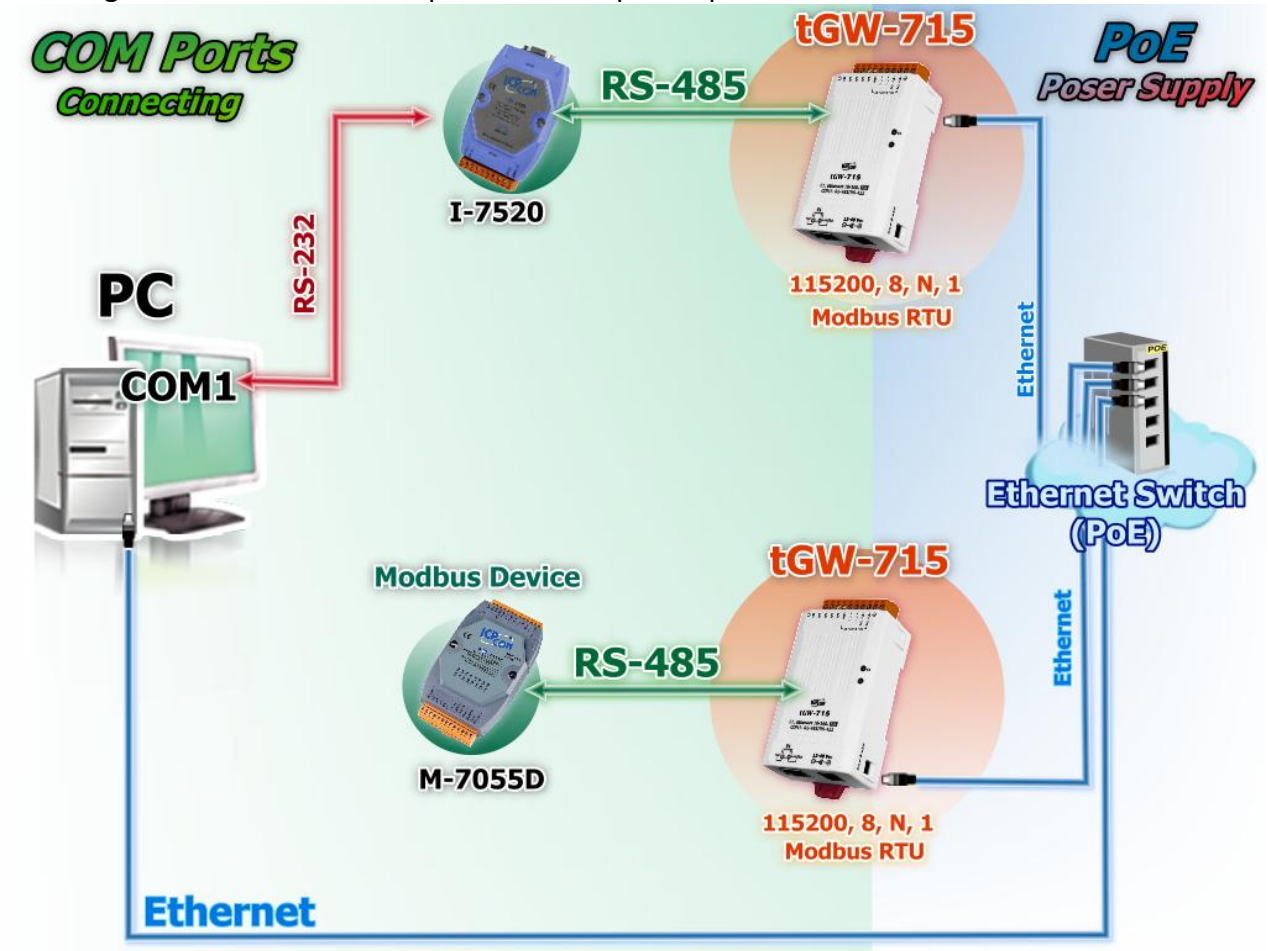
1. Confirm that the tGW-700 modules are functioning correctly. For detailed information regarding how to install, configure and operate your tGW-700 series module, refer to Chapter 3 Setting up the tGW-700 Module.

2. Use an I-7520 module to connect COM1 on the PC to COM1 on tGW-700 #1. For detailed information regarding RS-422/485 wiring, refer to Section 2.8 Wiring Notes for RS-232/422/485. (For detailed information related to the I-7520 module, refer to: http://www.icpdas.com/products/Remote_IO/i-7000/i-7520.htm)

3. Connect the Modbus device (e.g., M-7055D) to COM1 of tGW-700 #2. (For detailed information related to the M-7055D module, refer to: http://www.icpdas.com/products/Remote_IO/m-7000/m-7055d.htm)

※ Refer to Figure 5-1 for an illustration of how to perform Steps 1-1 to 1-3 of the procedure described above.

The image below shows an example of the setup for a pair-connection test: Figure 5-1



Step 2: Configuring the Ethernet Settings

Contact your Network Administrator to obtain the correct and functioning network configuration for the tGW-700 modules (including the **IP Address, Mask and Gateway details)**. Also refer to Chapter 3 Setting up the tGW-700 Module for more details.

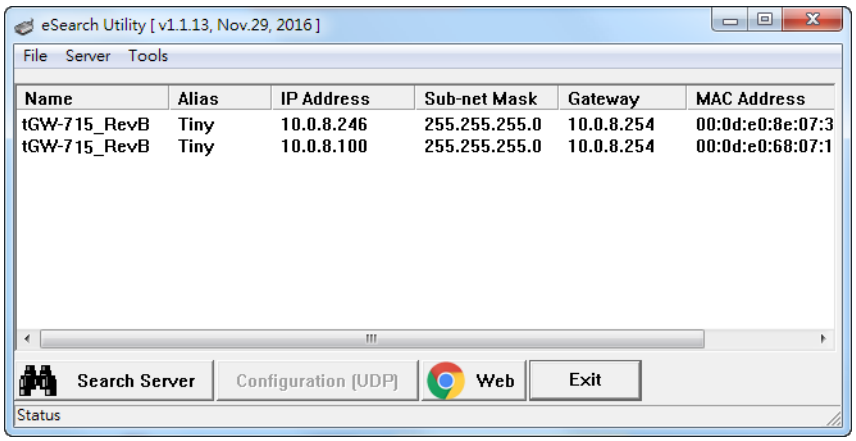


Figure 5-2

Step 3: Configuring the Pair-connection (Client Mode) on the Web Server for tGW-700 #1

1. Open the eSearch Utility to search for the tGW-700 modules connected to the network. Click the name of the first tGW-715 module (tGW-700 #1) to select it, and then click the **"Web"** button to launch a browser window to connect to the web server on the tGW-700 #1 module. Alternatively, you can enter the URL for tGW-700 #1 in the address bar of the browser.
2. When the login screen is displayed, enter the password **(use the default password "admin")** in the Password field, and then click the **"Submit"** button to display the configuration page.

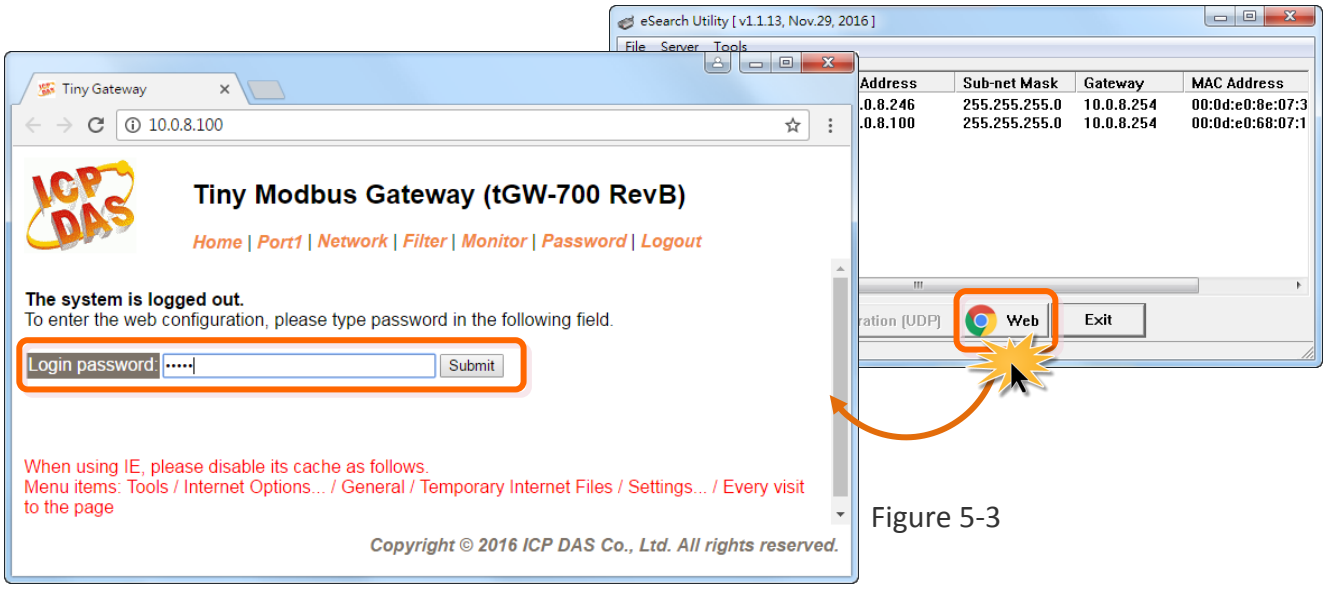


Figure 5-3

3. Click the **"Port1"** tab to display to the **Port1 Settings** page.

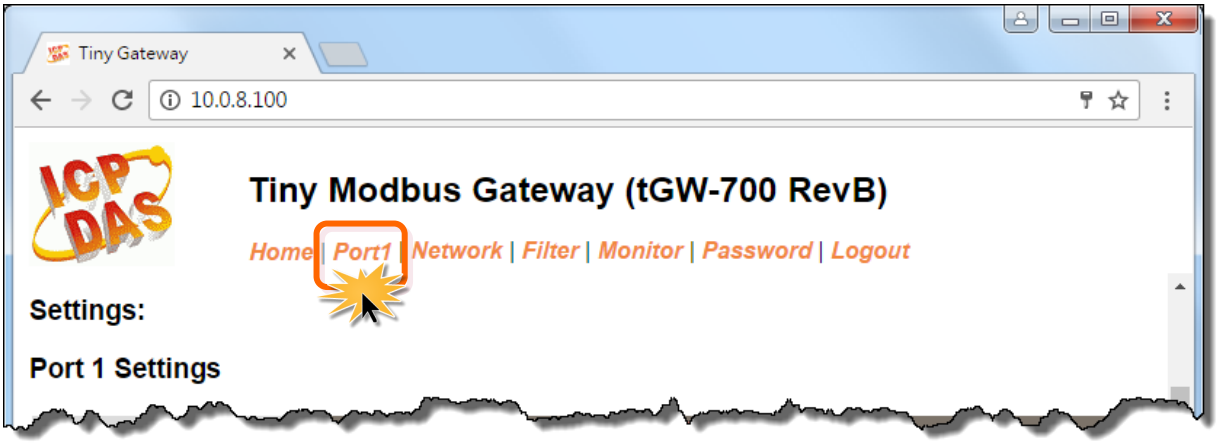


Figure 5-4

4. Select the appropriate **Baud Rate, Data Format and Modbus Protocol** settings from the relevant drop down options. The following is an example: Baud Rate (bps) **"115200"**, Data Bits (bits) **"8",** Parity **"None"**, Stop Bits (bits) **"1"** and Modbus Protocol **"Modbus RTU"**.

Figure 5-5

Port 1 Settings

Port Settings	Current	Updated
Baud Rate (bps)	115200	115200 ▾ bits/S
Data Size (bits)	8	8 ▾ bits/character
Parity	None	None ▾
Stop Bits (bits)	1	1 ▾
Flow Control	None	None ▾ (for tGW-712/722)
Slave Timeout (ms)	300	300 (Default: 300)
Char Timeout (bytes)	4	4 (4 ~ 15, Default: 4)
Silent Time (ms)	0	0 (0, 10, 20... 65530, Default: 0)
Read Cache (ms)	980	980 (10, 20... 65530, Disable: 0)
Local TCP Port	502	502 (Default: 502)
Connetion Idle (seconds)	180	180 (1 ~ 65535, Default: 180, Disable: 0)
Protocol	Modbus RTU	Modbus RTU ▾

5. In the **Pair-connection Settings** area for Port1, check that the configuration details are the same as those shown below.

Field	Server Mode	Modbus Protocol	Remote Server IP	Remote TCP Port	TCP Slave ID (1~247)	RTU Slave ID (1~247)
Pair-connection Settings	**Client**	**TCP**	**10.0.8.246**	**502**	**0**	**0**
		Modbus Protocol, IP address and TCP port for tGW-700 #2				

6. Amend and details as required and then click the **"Submit"** button to complete the configuration.

Figure 5-6

Pair-Connection Settings (Master/Slave Mode)	Current	Updated
Application Mode	Server	Client ▾ (Server=Slave, Client=Master)
Network Protocol	TCP	TCP ▾
Remote Server IP	Disabled	10 . 0 . 8 . 246
Remote TCP Port	Disabled	502
RTU Virtual ID (1~247)	0	0 (0: Bypass, No check)
TCP Slave ID (1~247)	0	0 (0: Same as RTU)
		Submit

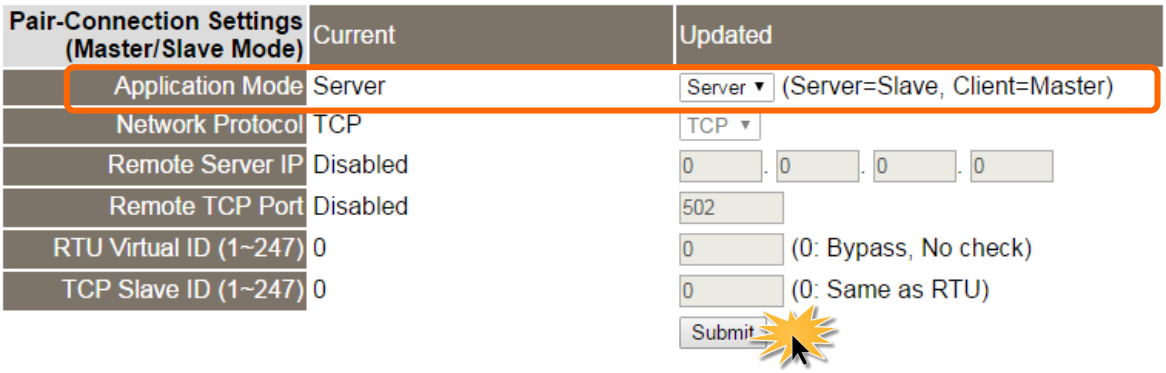
Step 4: Configuring the Pair-connection (Server Mode) on the Web Server for tGW-700 #2

1. In the eSearch Utility, click the name of the second tGW-715 module (tGW-700 #2) to select it, and then click the **"Web"** button to launch a browser window to connect to the web server on the tGW-700 #2 module. Alternatively, you can enter the URL for tGW-700 #2 in the address bar of the browser.
2. When the login screen is displayed, enter the password (use the default password **"admin"**) in the Password field, and then click the **"Submit"** button to display the configuration page.
3. Click the **"Port1"** tab to display the **Port1 Settings** page.
4. Select the appropriate **Baud Rate, Data Format and Modbus Protocol** settings from the relevant drop down options. The following is an example: Baud Rate (bps) **"9600"**, Data Bits (bits) **"8",** Parity **"None"**, Stop Bits (bits) **"1"** and Modbus Protocol **"Modbus RTU"**.

※ Refer to Figures 5-3 to 5-5 for an illustration of how to perform Steps 4-1 to 4-4 of the procedure described above.

5. In the **Pair-connection Settings** area for Port1, select "Server" from the "Server Mode" drop down options.
6. Amend any details as required and then click the **"Submit"** button to complete the configuration.

Figure 5-7



Step 5: Testing the Pair-connection Functions

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

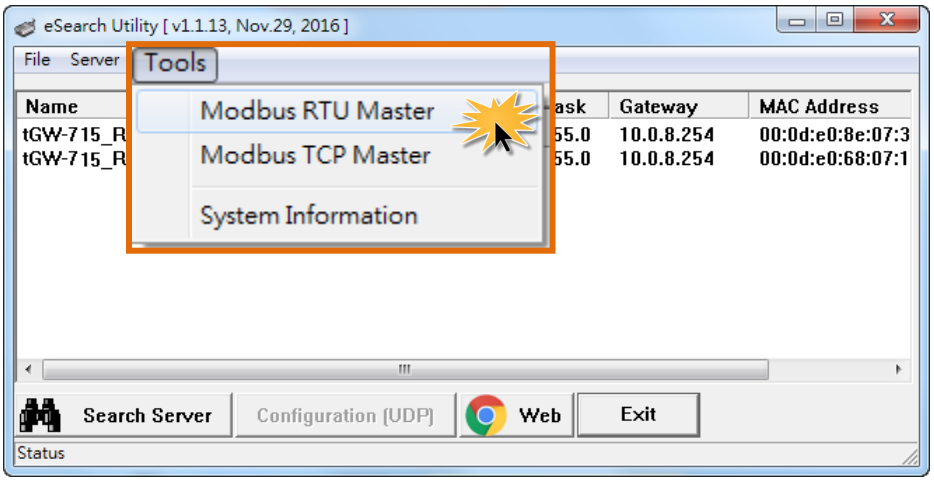


Figure 5-8

2. Select the appropriate COM port, Baud Rate and Data Format (e.g., COM1, 115200, N,8,1) settings for the tGW-700, and then click the "**Open**" button.

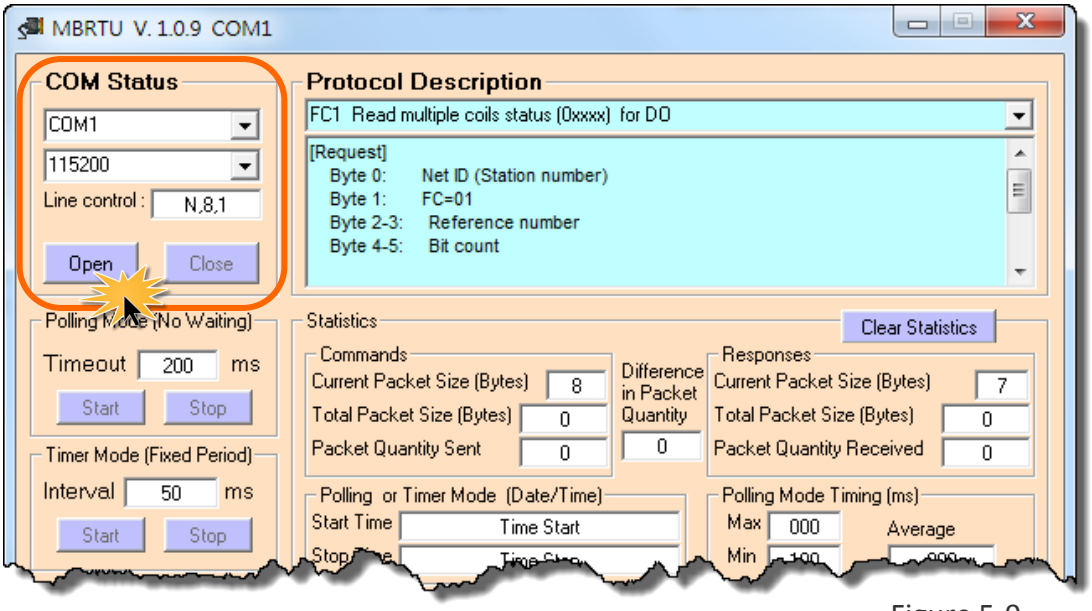


Figure 5-9

3. Refer to the **"Protocol Description"** field in the top right-hand section of the Modbus Utility window. You can send a request command and confirm the response is correct.

   (For example, if the remote Modbus device ID for the Modbus device (M-7055D) is 1, send the command “1 2 0 0 0 4” to reading D/I value, and the response shown in the “Responses” section will be “1 2 1 0 A1 88”. (Note that “A1 88” is the CRC value.)

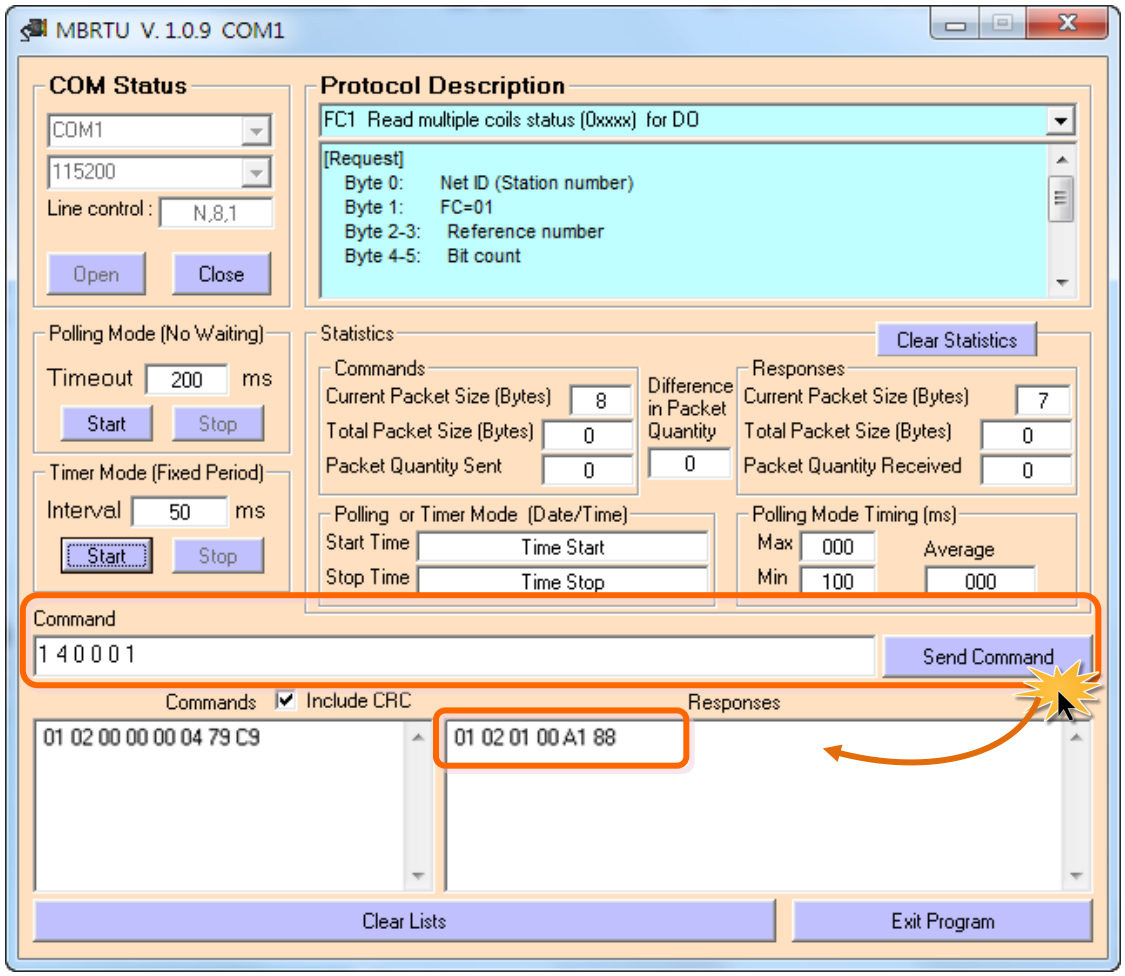
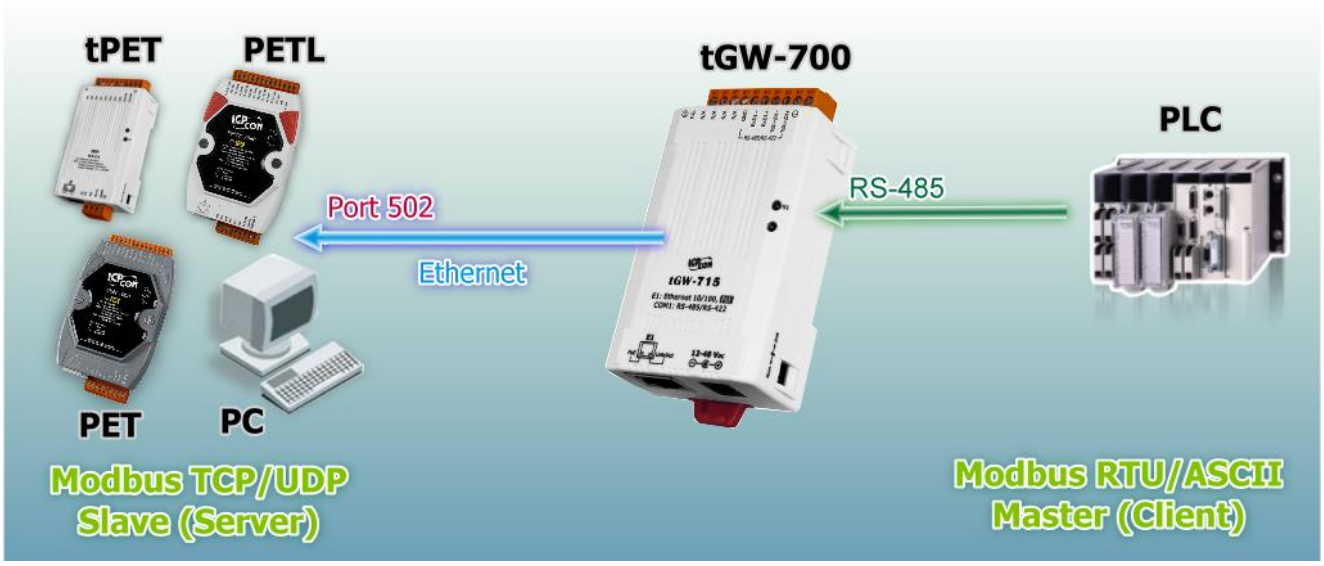


Figure 5-10

Note	*1. The response will depend on which Modbus is device connected.* *2. The Baud Rate and Data Format values depend on the serial port settings configured for the web configuration described above.*

5.4 TCP Client Mode Applications

In TCP Client Mode, the tGW-700 can actively establish a TCP connection to a specific Modbus TCP slave device. An example of how the complete system should operate is shown below:



The following are examples of TCP Client Mode tests:

TCP Client Mode Settings:

Model	Port Settings (default)		Pair-connection Settings		
	Baud Rate	Data Format	Server Mode	Remote Server IP	Remote TCP Port
tGW-700	115200	8, N, 1	Client	10.0.8.10	502
				IP address and TCP port for the tPET-P6 (Slave Device)	
tPET-P6 (Slave Device)	-	-	-	-	-

Follow the procedure described below:

Step 1: Connecting to a network, a PC and a Power Supply

1. Confirm that the tGW-700 device is functioning correctly. For detailed information regarding how to install, configure and operate your tGW-700 series module, refer to Chapter 3 Setting up the tGW-700 Module.
2. Connect both the tGW-700, the Slave Device (e.g. a tPET-P6 module) and your computer to the same sub network or the same Ethernet Switch. For detailed information regarding RS-232/RS-422/485 wiring, refer to Section 2.8 Wiring Notes for RS-232/422/485.

The wiring diagram is as follows:

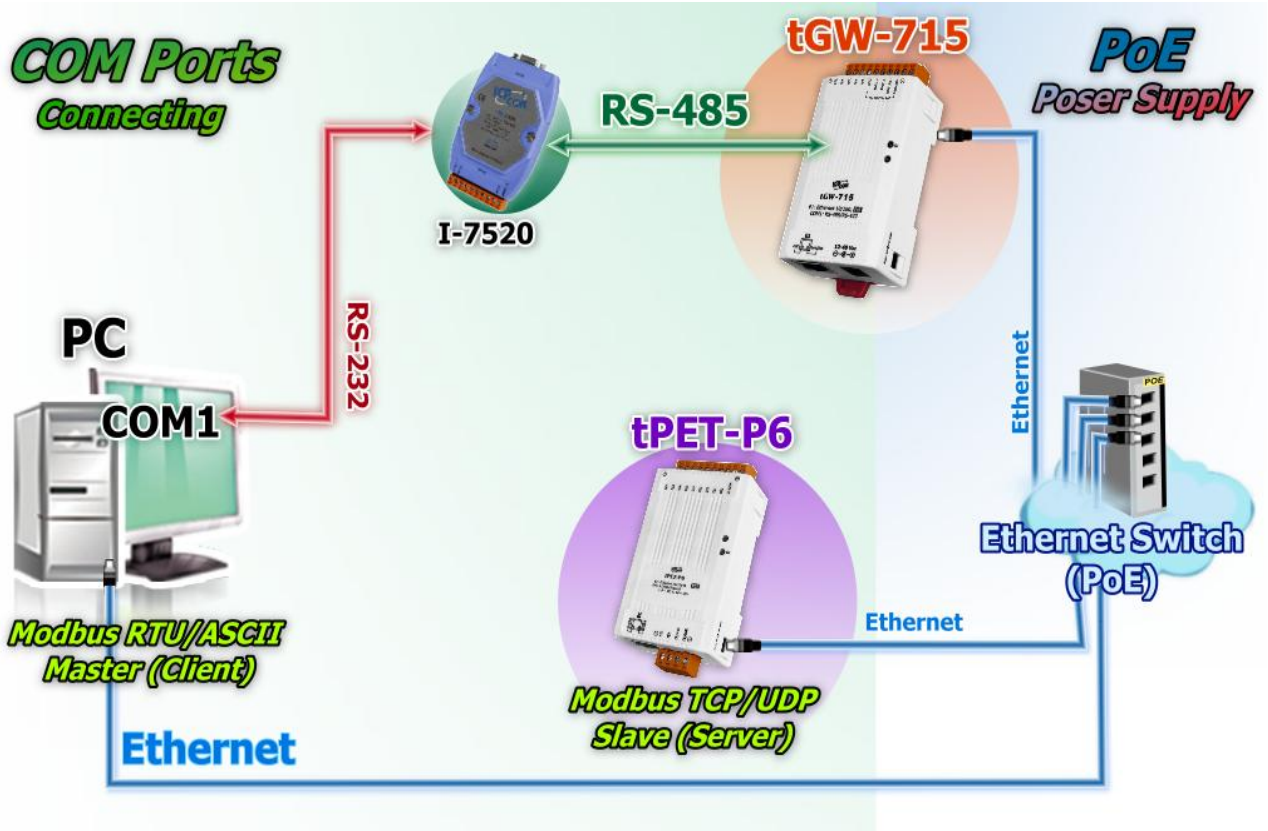


Figure 5-11

Step 2: Configuring the Ethernet Settings

Contact your Network Administrator to obtain a correct and functioning network configuration (including the **IP Address, Mask and Gateway details)** for the tGW-700 module. Also refer to Chapter 3 Setting up the tGW-700 Module for more details.

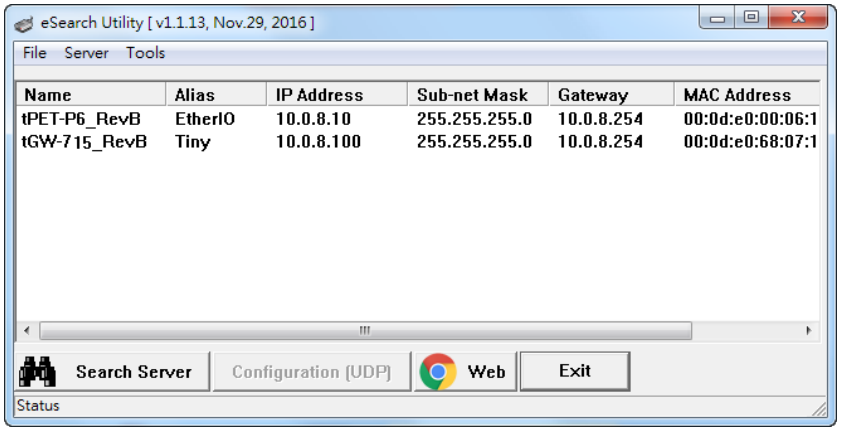


Figure 5-12

Step 3: Configuring Pair-connection (TCP Client Mode) on the Web Server for the tGW-700 module

1. Open the eSearch Utility to search for the tGW-700 modules connected to the network. Click the name of the first tGW-700 module to select it, and then click the **"Web"** button to launch a browser window to connect to the web server on the tGW-700 module. Alternatively, you can enter the URL for tGW-700 in the address bar of the browser.
2. When the login screen is displayed, enter the password **(use the default password "admin")** in the Password field, and then click the **"Submit"** button to display the configuration page.

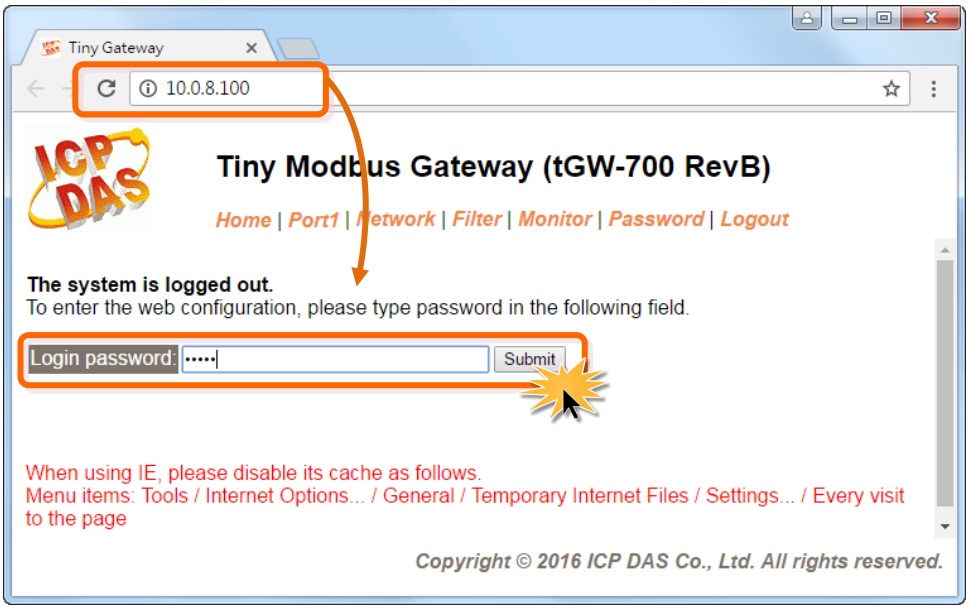


Figure 5-13

3. Click the **"Port1"** tab to display the **Port1 Settings** page.

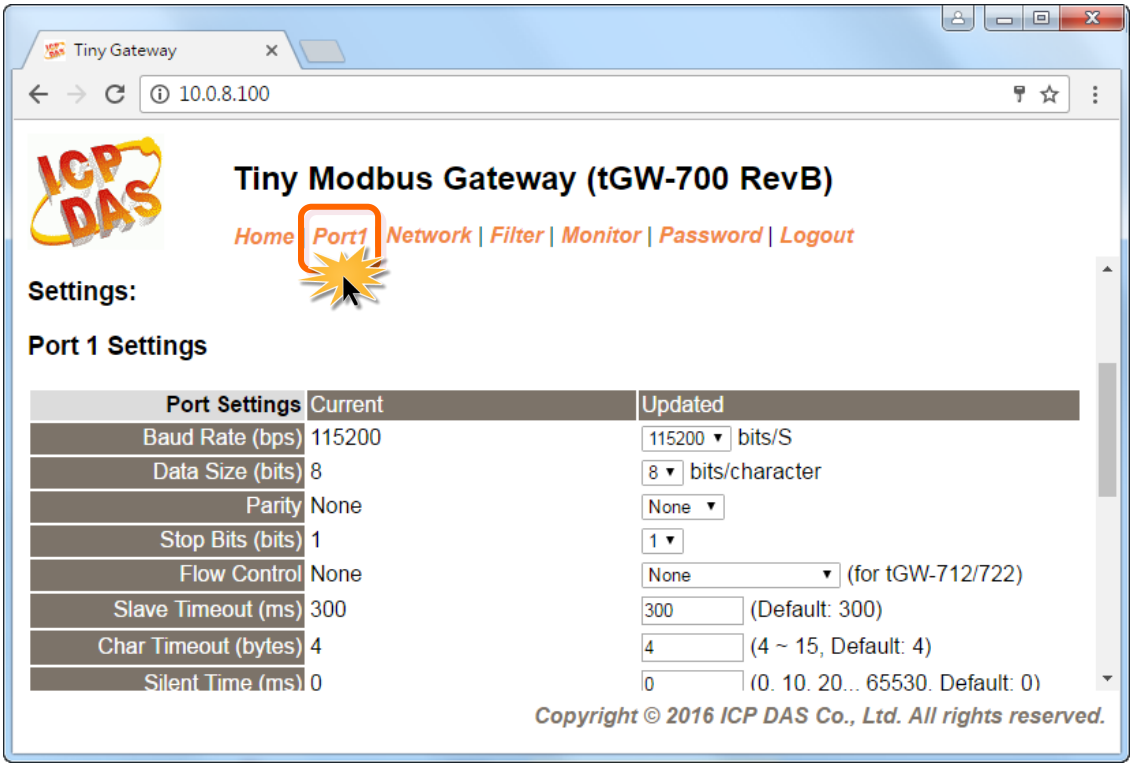


Figure 5-14

4. Select the appropriate Baud Rate, Data Format and Modbus Protocol settings from the relevant drop down options. The following is an example: Baud Rate (bps) **"115200"**, Data Bits (bits) **"8"**, Parity **"None"**, Stop Bits (bits) **"1"** and Modbus Protocol **"Modbus RTU"**.

Port 1 Settings

Figure 5-15

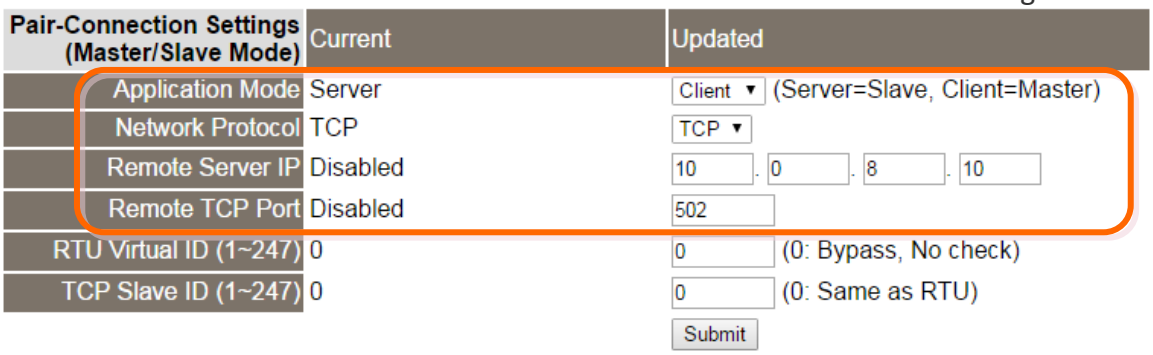
Port Settings	Current	Updated
Baud Rate (bps)	115200	115200 bits/S
Data Size (bits)	8	8 bits/character
Parity	None	None
Stop Bits (bits)	1	1
Flow Control	None	None (for tGW-712/722)
Slave Timeout (ms)	300	300 (Default: 300)
Char Timeout (bytes)	4	4 (4 ~ 15, Default: 4)
Silent Time (ms)	0	0 (0, 10, 20... 65530, Default: 0)
Read Cache (ms)	980	980 (10, 20... 65530, Disable: 0)
Local TCP Port	502	502 (Default: 502)
Connetion Idle (seconds)	180	180 (1 ~ 65535, Default: 180, Disable: 0)
Protocol	Modbus RTU	Modbus RTU

5. In the **Pair-connection Settings** area of the Port1 Settings page, check that the configuration details are the same as those shown below.

Field	Server Mode	Modbus Protocol	Remote Server IP	Remote TCP Port	TCP Slave ID (1~247)	RTU Slave ID (1~247)
Pair-Connection Settings	**Client**	**TCP**	**10.0.8.10**	**502**	**0**	**0**
		Modbus Protocol, IP address and TCP port of the slave device (e.g., tPET-P6)				

6. Amend any details as required and then click the **"Submit"** button to complete the configuration.

Figure 5-16



※ Refer to Figures 5-16 for an illustration of how to perform the procedure described above.

Step 4: Testing the Pair-connection (TCP Client Mode) Functions

※ For more detailed information regarding the testing procedure, refer to Step 5 (Figures 5-8 to 5-10) in the Section 5.3 Pair-Connection Applications.

6. 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.

6.1 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: specifies the address of the receiver (Modbus/TCP slave).

Function Code: specifies the message type.

Data Field: is the data block.

Net ID (Station Number)

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

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 be 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

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 16 bits 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.

6.1.1 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

6.1.2 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

6.1.3 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

6.1.4 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.

6.1.5 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.

6.1.6 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.

6.1.7 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)	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	1to 247
01	Function Code	1 Byte	0x8F
02	Exception Code	1 Byte	Refer to the Modbus Standard Specifications for more details.

6.1.8 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.

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 actions and 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

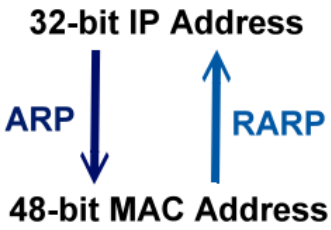
A packet is the unit of data sent across a physical network. It consists of a series of bits containing data and control information, including the source and the destination node (host) address, and is formatted for transmission from one node to another.

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) are standard network protocols that are almost always implemented and used together in a formation is 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 an other, 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
50	50	0.00%
110	109.92	0.07%
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: 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 tGW-700:

Code	Name and Description
0x0B	GATEWAY TARGET DEVICE FAILED TO RESPOND Timeout. The slave device does not respond within the timeout value, the tGW-700 will return this code.
0x4B	GATEWAY TARGET DATA FAILED TO RESPOND Timeout. The slave device is still sending data when timed out, the tGW-700 will return this code. Please use larger Slave Timeout value for the serial port of the tGW-700 module.
0x41	MODBUS PROTOCOL FORMAT ERROR The tGW-700 will return this code when slave response is invalid Modbus message.
0x42	WRONG DATA LENGTH The tGW-700 will return this code when tGW-700 received wrong data length. Please use larger Slave Timeout value for the serial port of the tGW-700 module.
0x43 0x52	CRC ERROR