



ICP DAS CO., LTD.

# UA-5200 Series User Manual

IloT Communication Server



ICP DAS CO., LTD. All Rights Reserved.

[www.icpdas.com](http://www.icpdas.com)

## Warranty

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

## Warning

ICP DAS CO., LTD. assumes no liability for damages consequent to the use of this product. ICP DAS CO., LTD. reserves the right to change this manual at any time without notice. The information furnished by ICP DAS CO., LTD. is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS CO., LTD. for its use, or for any infringements of patents or other rights of third parties resulting from its use.

## Copyright and Trademark Information

© Copyright 2015 by ICP DAS CO., LTD. All rights reserved worldwide.

## Trademark of Other Companies

The names used for identification only maybe registered trademarks of their respective companies.

## License

The user can use, modify and backup this software on a single machine. The user may not reproduce, transfer or distribute this software, or any copy, in whole or in part.

## Technical Output:

Please contact local agent or email problem-report to [output@icpdas.com](mailto:output@icpdas.com) .

For more product information, please refer to [www.icpdas.com](http://www.icpdas.com) .

Technique: Sun Chen & Tim Chen; Translation & Edition: Eva Li & Lisa Chen;  
R&D Dept., ICP DAS CO., LTD.  
Latest edited by: Eva Li; V.4.0, 12/2017

## Table of Contents

1.	UA-5200 IloT Communication Server.....	6
1.1.	Introduction.....	6
1.2.	Features .....	7
1.3.	Functions.....	8
1.4.	Hardware Specifications .....	9
1.5.	Software Specifications.....	11
1.6.	Appearance .....	12
1.7.	Dimensions.....	15
2.	Quick Start.....	16
2.1.	Hardware Connection .....	16
2.1.1.	Preparations for Devices.....	16
2.1.2.	Hardware Wiring .....	16
2.2.	Network Connection .....	17
2.2.1.	Connection By Factory Default Settings.....	17
2.2.2.	Connection By Utility Searching .....	19
2.3.	Project Setting Example.....	23
2.3.1.	A Quick Setup Project Example .....	24
3.	Web UI Login and Environment Overview.....	32
3.1.	Login The UA Web UI .....	32
3.2.	Web UI Environment Overview .....	34
3.3.	Setting Procedures and Steps .....	35
4.	Function Wizard.....	36
4.1.	Module Communication Conversion .....	37
4.1.1.	Modbus / OPC UA Conversion.....	38
4.1.2.	Modbus / MQTT Conversion .....	49
4.1.3.	Modbus / MQTT JSON Conversion.....	61
4.2.	Module Connecting to Azure.....	76

4.2.1. Modbus TCP / Azure Connecting .....	77
4.3. PID .....	88
4.3.1. PID Operation .....	89
5. System Setting.....	99
5.1. Controller Service Setting .....	100
5.2. Time Setting.....	101
5.3. Network Setting .....	103
5.4. Account Setting.....	105
5.5. Boot.....	106
5.6. COM Port Interface Setting.....	107
6. Module Setting.....	108
6.1. Modbus RTU (Master) .....	109
6.2. Modbus TCP (Master).....	114
6.3. Modbus ASCII (Master).....	119
7. IoT Platform Setting .....	124
7.1. MQTT Local Broker.....	125
7.2. MQTT Remote Broker.....	126
7.3. MQTT Group Connection.....	129
7.4. MQTT Connection - Microsoft Azure Platform .....	133
7.5. OPC UA Connection - Local Server .....	136
8. Convert Setting .....	137
8.1. OPC UA and Modbus RTU/ASCII Conversion .....	139
8.2. OPC UA and Modbus TCP Conversion.....	142
8.3. MQTT and Modbus RTU/ASCII Conversion.....	145
8.4. MQTT and Modbus TCP Conversion .....	149
8.5. MQTT JSON and Modbus RTU/ASCII Conversion .....	153
8.6. MQTT JSON and Modbus TCP Conversion.....	156
9. Advanced Setting.....	159
9.1. PID Operation.....	160

10.	File Setting.....	165
	10.1. Settings .....	166
11.	Factory Setting Recovering and Middleware Updating.....	167
	11.1. Recovering to Factory Setting (Rotary Switch: 8).....	167
	11.2. Updating Middleware via USB (Rotary Switch: 9) .....	168
12.	Security Certificate: Download/Upload/Update.....	169
	12.1. Download the Certificate of UA Controller .....	169
	12.2. Upload/Update the Certificate to UA Controller .....	170
	Appendix A. MQTT JSON Format of the UA Series.....	171
	Appendix B. Technical Reference Websites .....	172
	OPC UA.....	172
	MQTT .....	172
	Modbus.....	172

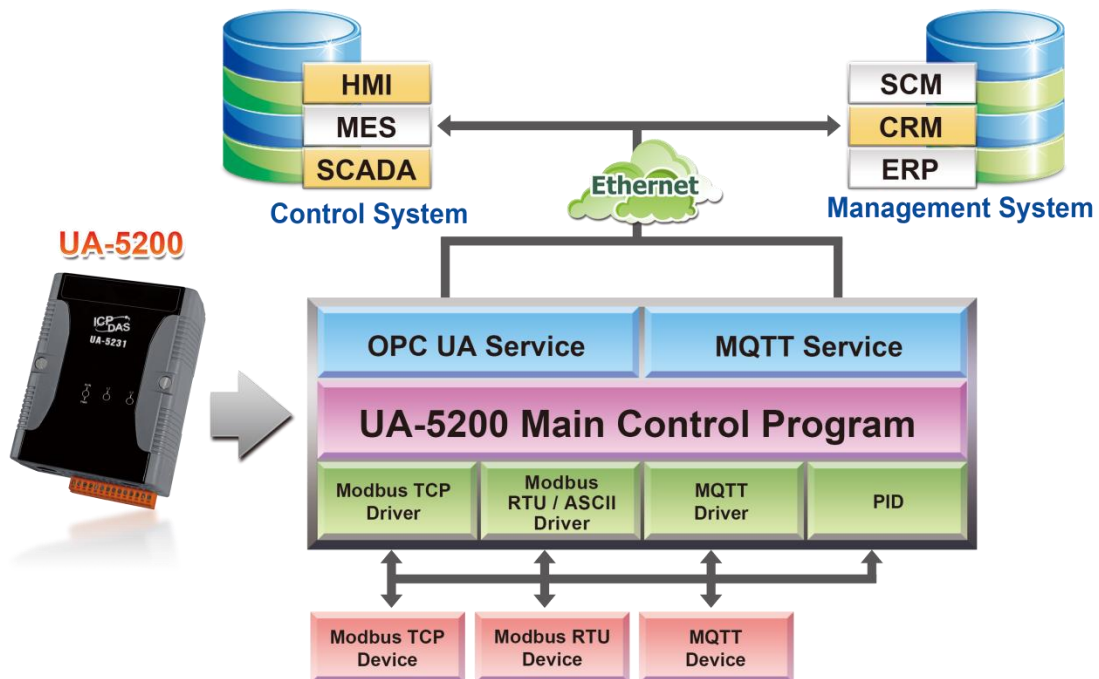
# 1. UA-5200 IIoT Communication Server

This chapter introduces the UA-5200 and its functions, software/hardware specifications...

## 1.1. Introduction

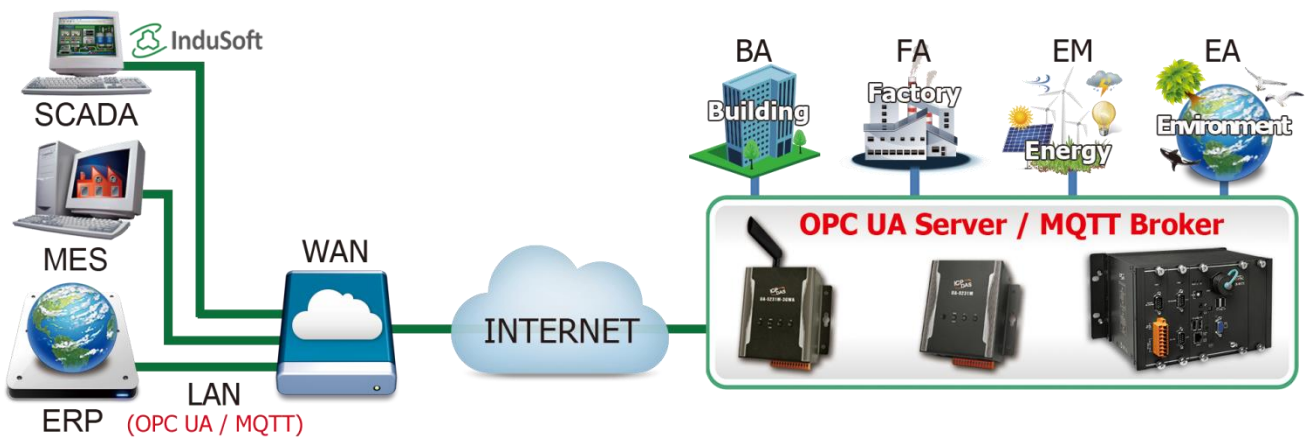
The **UA-5200** is a series of data acquisition controller and also an IIoT communication server by ICP DAS (IIoT: Industrial Internet of Things). The UA-5200 built-in **OPC UA Server** and **MQTT Client Service** support a variety of common industrial communication protocols. Its RISC-based CPU architecture has the advantages of small size and low power consumption that lets this series can be placed in a small space to fit a variety of rooms, equipment and case environment. In the hardware, it provides a variety of communication interfaces, such as Gigabit Ethernet, USB, RS-232 and RS-485... ports to connect diverse devices.

Applying the **OPC UA**, the UA-5200 can integrate the I/O products and the third-party devices, import their data to the back-end SCADA management system or the big-data analysis/decision system, to satisfy the reliability, interoperability and security needs of the Industrial 4.0 automation system. Using the **MQTT** active communications to bridge the Internet of Things (IoT) and transmit the statuses of various devices by the cloud-based interaction so that to meet the current trend of the IIoT and achieve the full smart automation system based on **Industry 4.0**.



## 1.2. Features

- OPC UA Server and MQTT Client Service
- MQTT Broker Inside
- ARM CPU, 1.0 GHz
- 512 MB RAM and 512 MB Flash
- Linux kernel 3.2.14 OS
- Real-Time Capability
- 64-bit Hardware Serial Number for Software Protection
- Support Redundancy (OPC UA) and PID
- 10/100/1000 Mbit/s Ethernet Port
- 4 Serial Ports (RS-232/RS-485)
- Operating Temperature: -25 ~ +75°C



## 1.3. Functions

### ■ Web-based UI

With the Web-based User Interface, users can log in and configure the controller via a normal web browser that only need a mobile device or computer with web browsing capabilities.

### ■ OPC UA Server: IEC 62541 Standard

The OPC UA Server certified by the OPC Foundation can assist the integration for the local-end devices, actively upload data to the application system, and support to across the multiple platforms.

### ■ PID Logic Operation

The PID function can dynamically combine the remote I/O devices for the PID logic control to provide temperature control and case field solutions.

### ■ Support Modbus TCP/RTU/ASCII Master

Through the controller's RS-485, RS-232 and Ethernet ports can connect to the Modbus TCP/RTU/ASCII Slave devices. Build systems with scalability and flexibility to meet the diverse application needs and expansion at any time.

### ■ MQTT Broker Inside

Compliance with MQTT v3.1.1 protocol. Support MQTT message distribution management. Users do not need to build Broker system when using MQTT communications.

### ■ Support MQTT Protocol

Support MQTT to allow the IoT devices communicating with the OPC UA system and the UA-5200 conducting the data acquisition and management; and also can convert and publish the devices' data under the UA-5200 to the IoT system.

UA-5200 Function Overview		
<b>Web-based UI</b>	Built-in Web-based User Interface	
<b>Flexible System Configuration</b>	Variable Table/Communication Task Dynamic Editor	
<b>OPC UA</b>	Compliance with IEC 62541 Standard Cross-platform Transmission Security SSL Encryption Active Transmission Support Redundancy Support Remote Function Call	
<b>MQTT Broker Inside</b>	Built-in MQTT Broker, Compliance with MQTT V.3.1.1 Protocol	
<b>PID Logic Operation</b>	Dynamic Combination of I/O Devices for PID Logic Control	
<b>Service (Output)</b> Up to Interact with the Host	Protocol	OPC UA Server MQTT Client
	Interface	Ethernet Data Transmission
<b>Driver (Input)</b> Down to Interact with the I/O Modules	Protocol	Modbus RTU/ASCII/TCP MQTT
	Interface	RS-232/RS-485 Ethernet Data Transmission



## 1.4. Hardware Specifications

(Available Soon)

Model	UA-5231	UA-5231M	UA-5231M-3GWA
<b>System Software</b>			
OS	Linux Kernel 3.2.14		
Embedded Service	SFTP server, Web server, SSH		
<b>CPU Module</b>			
CPU	ARM CPU, 1.0 GHz		
DDR3 SDRAM	512 MB		
Flash	512 MB		
FRAM	64 KB		
Expansion Flash Memory	microSD socket with one 4 GB microSD card (support up to 32 GB microSDHC card)		
RTC (Real Time Clock)	Provide second, minute, hour, date, day of week, month, year		
64-bit Hardware Serial Number	Yes, for Software Copy Protection		
Dual Watchdog Timers	Yes		
LED Indicators	4 LEDs (Power, Running and 2 user defined LEDs)		
Rotary Switch	Yes (0 ~ 9)		
<b>VGA &amp; Communication Ports</b>			
VGA & Communication Ports	Yes, resolution: 640 × 480, 800 × 600, 1024 × 768, 1280 × 720		
Ethernet	RJ-45 x 1; 10/100/1000 Based-TX ( Auto-negotiating, Auto MDI/MDI-X, LED indicators)		
USB 2.0 (host)	1		
Console Port	RS-232 (Rx, Tx and GND); Non-isolated		
ttyO2	RS-485 (Data+, Data-); Non-isolated		
ttyO4	RS-232 (Rx, Tx and GND); Non-isolated		
ttyO5	RS-485 (Data+, Data-); 2500 VDC isolated		
<b>Mechanical</b>			
Dimensions (W x L x H)	91 mm x 132 mm x 52 mm	117 mm x 126 mm x 58 mm	
Installation	DIN-Rail Mounting		
<b>Environmental</b>			
Operating Temperature	-25 ~ +75°C		
Storage Temperature	-40 ~ +80°C		
Ambient Relative Humidity	10 ~ 90% RH (non-condensing)		
<b>Power</b>			
Input Range	+12 ~ +48 VDC		

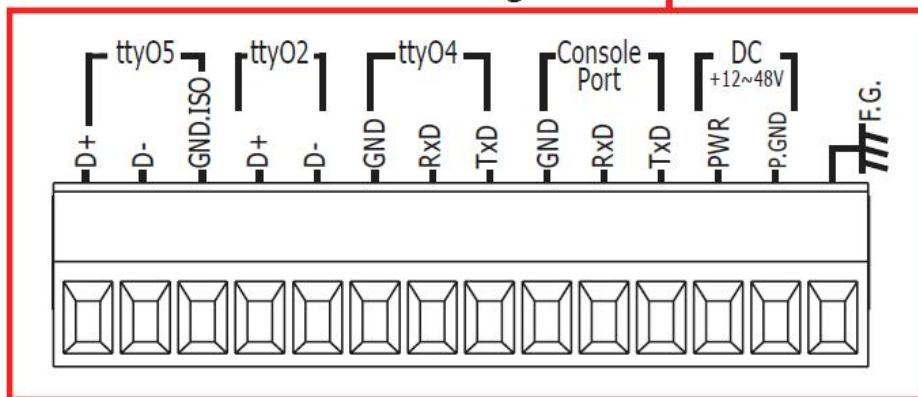
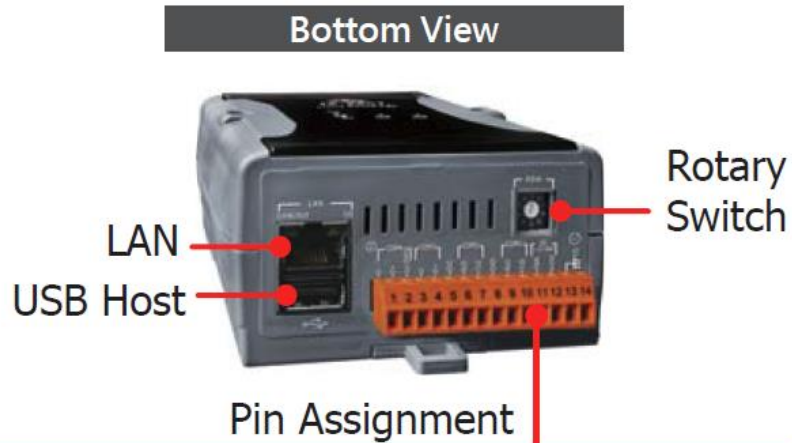
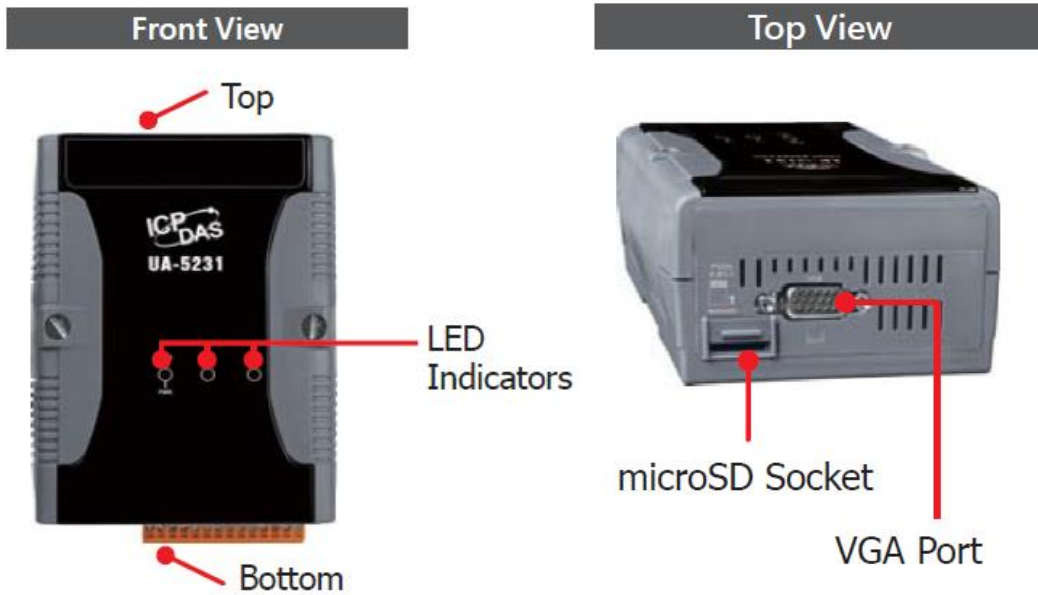
<b>Model</b>	<b>UA-5231</b>	<b>UA-5231M</b>	<b>UA-5231M-3GWA</b>
Consumption	4.8 W		6.5 W
<b>GSM System</b>			
Frequency Band	-		GSM: 850/900/1800/1900 MHz
GPRS Connectivity	-		GPRS class 12/10; GPRS station class B
Data GPRS	-		Downlink transfer: Max. 85.6 kbps;  Uplink transfer: Max 42.8kbps
<b>3G System</b>			
Frequency Band	-		WCDMA 850/900/1900/2100 MHz
Data Transmission	-		WCDMA / HSPA+  Download: Max. 14.4 Mbps;  Upload: Max. 5.76Mbps

## 1.5. Software Specifications

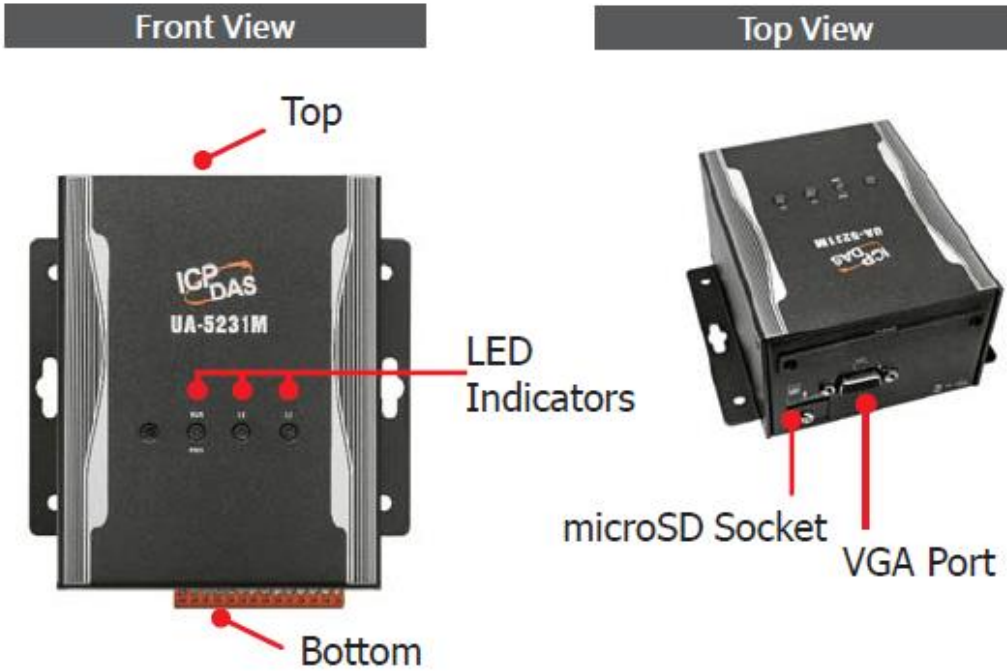
Model	UA-5200Series
<b>OPC UA</b>	
OPC UA Server	<ul style="list-style-type: none"> <li>● OPC Unified Architecture: 1.02</li> <li>● Core Server Facet</li> <li>● Data Access Server Facet</li> <li>● Method Server Facet</li> <li>● Client Redundancy Facet</li> <li>● UA-TCP UA-SC UA Binary</li> <li>● User Token User Name Password &amp; X509 Certificate</li> <li>● Security Policy                             <ul style="list-style-type: none"> <li>◦None</li> <li>◦Basic128Rsa15                                     <ul style="list-style-type: none"> <li>• Sign</li> <li>• Sign &amp; Encrypt</li> </ul> </li> <li>◦Basic256                                     <ul style="list-style-type: none"> <li>• Sign</li> <li>• Sign &amp; Encrypt</li> </ul> </li> </ul> </li> </ul> <p>Recommend to keep the maximum number of sessions within 20 connections.</p>
<b>Modbus Master</b>	
Modbus TCP	<p>To read or control the devices that support standard Modbus TCP Slave protocol.                      Recommend to keep the maximum number of devices within 100 connections.</p>
Modbus RTU/ASCII	<p>A max. of 3 ports: ttyO2, ttyO4, ttyO5 to connect other Modbus RTU Slave devices (e.g. M-7000).                      Recommend no more than 32 devices per port for better communication quality.</p>
<b>MQTT</b>	
MQTT Client	<p>Connect the MQTT Broker to read/control the devices supporting the MQTT protocol.</p>
MQTT Service	<p>Connect the MQTT Broker to externally read/control the devices supporting other protocols that linking with the UA-5200 series.</p>
MQTT Broker	<p>Compliance with MQTT v3.1.1 protocol. Support MQTT message distribution management.                      Recommend to keep the connection number of Client within 400.</p>
<b>Virtual Device</b>	
PID Function	<p>Combine the remote I/O devices for the PID logic control system.</p>

# 1.6. Appearance

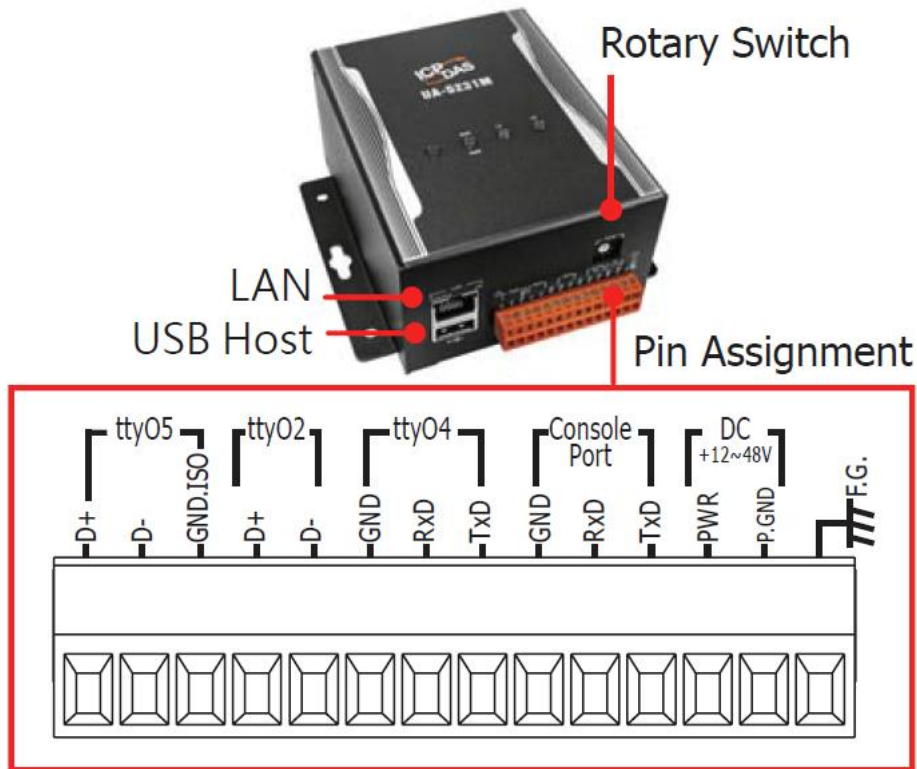
## UA-5231



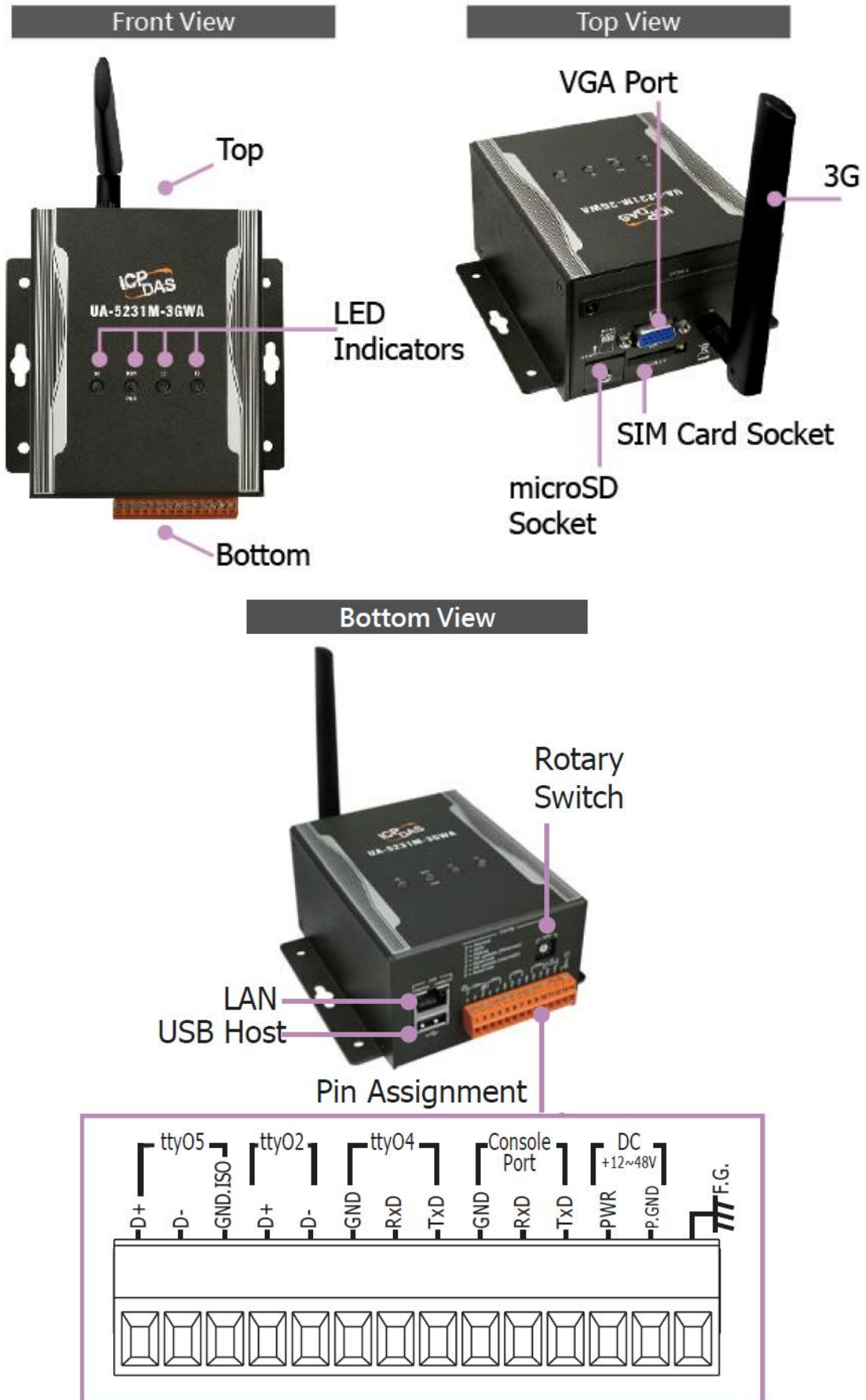
# UA-5231M



## Bottom View



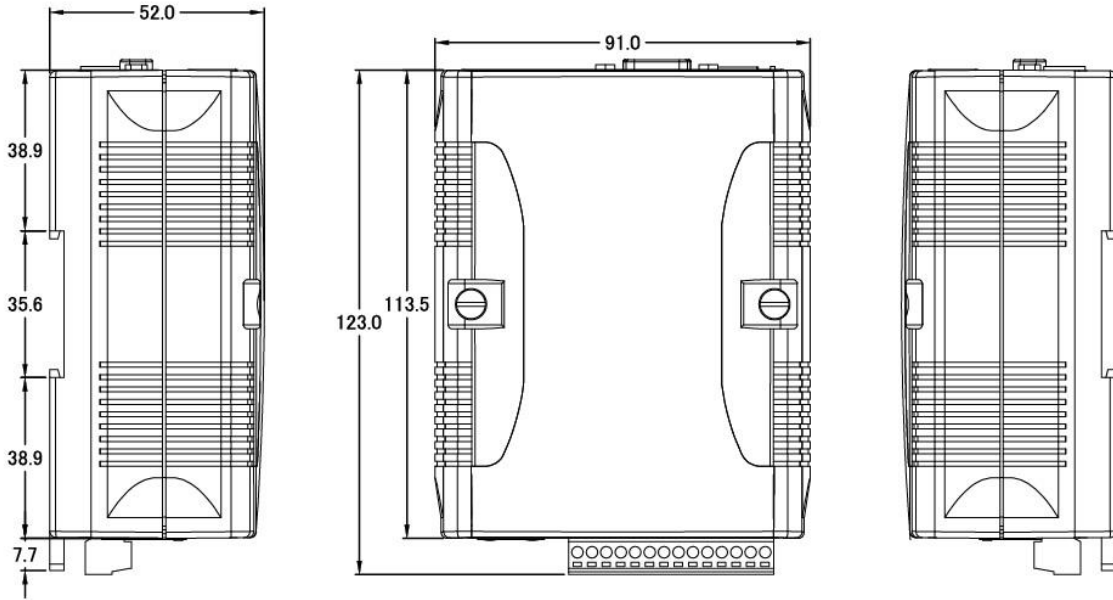
**UA-5231M-3GWA** (Available Soon)



## 1.7. Dimensions

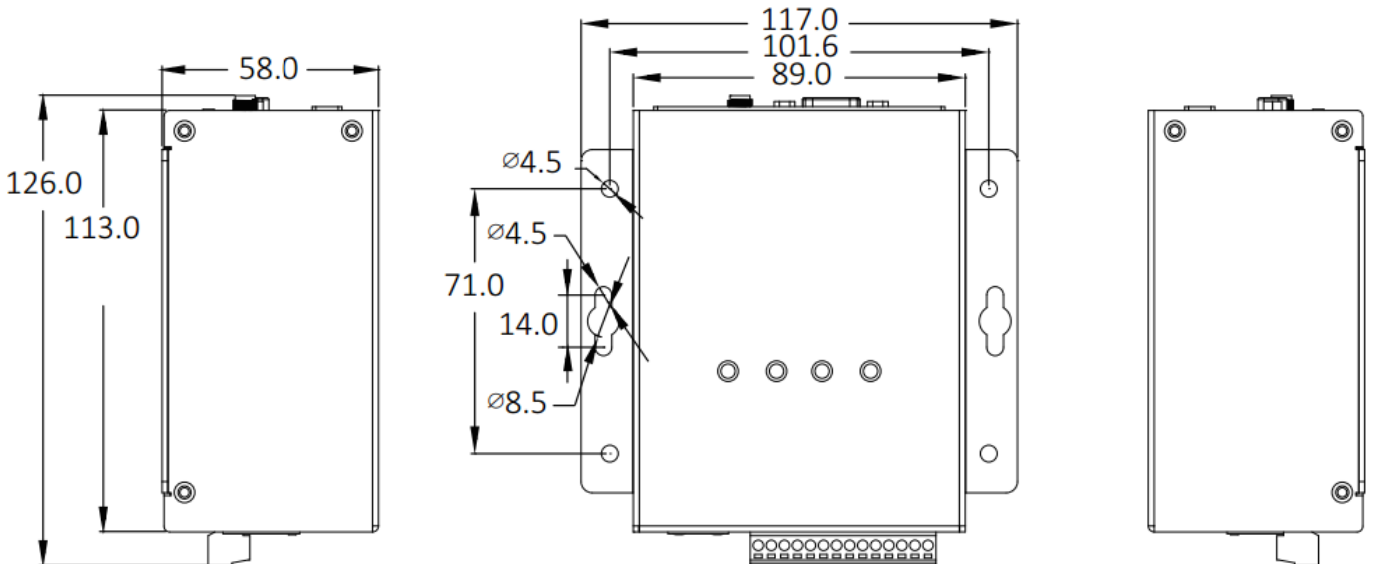
### UA-5231

Unit: mm



### UA-5231M/UA-5231M-3GWA

Unit: mm



## 2. Quick Start

This chapter describes the devices hardware connection, network connection and quick setting for the UA-5200 Controller, and how to connect to the UA controller web-based UI via a browser, set web functions step-by-step, and complete an example project.

### 2.1. Hardware Connection

This section describes the hardware wiring and connection for the UA-5200 Controller.

#### 2.1.1. Preparations for Devices

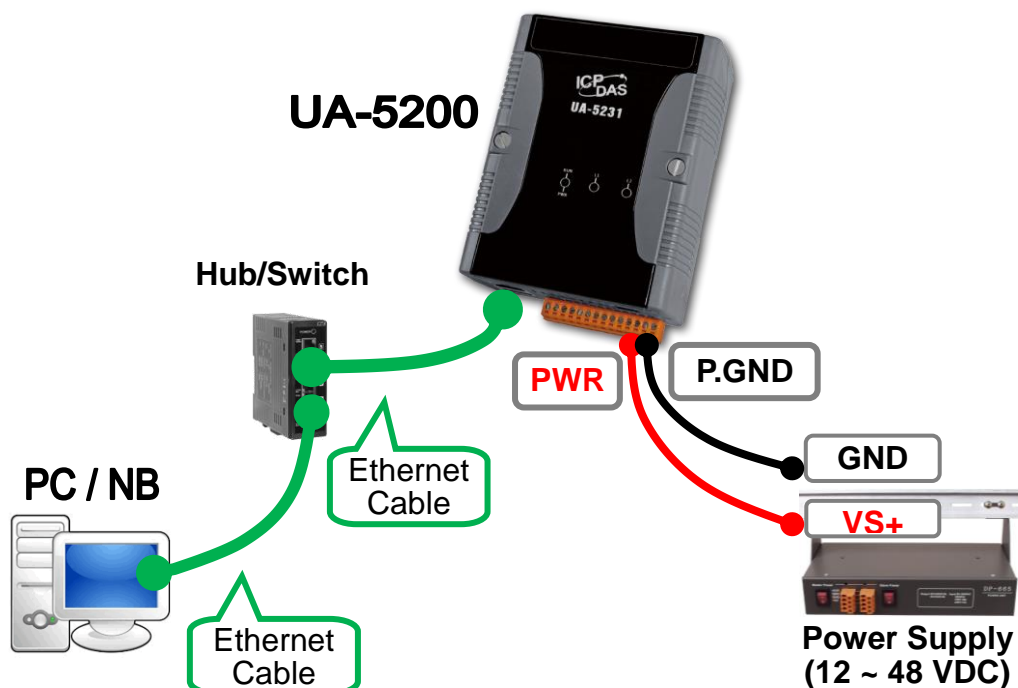
In addition to the UA-5200 series controllers (Ex: UA-5231), please prepare the following:

- Power Supply: +12 ~ +48 VDC (Ex: DP-665)
- 2. Ethernet Hub or Switch (Ex: NS-205)
- 3. PC/NB: Can connect to the network and set the network

#### 2.1.2. Hardware Wiring

Connect the UA-5200 with the RJ-45 Ethernet port to an Ethernet hub/switch and PC. You can also link directly the UA-5200 to PC with an Ethernet cable.

After power is connected, please [ wait 1 minute ] for UA-5200 start-up procedure. When the "RUN/PWR" light starts flashing, it represents the boot is complete.





## 2.2. Network Connection

This section introduces how to connect to the UA-5200 Web User Interface (UA Web UI). The new user or setting the new UA controller is recommended to follow the method in the first session. (The same method as the “UA-5200 Quick Start” manual)

### 2.2.1. Connection By Factory Default Settings

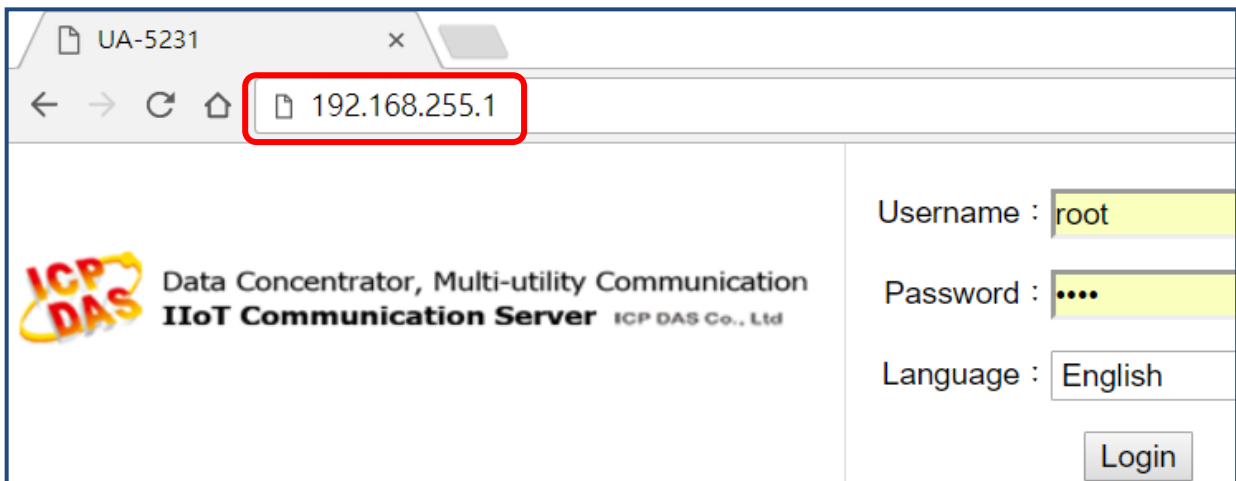
The factory default settings of the UA-5200 are as the following table:

Factory Default Settings of UA-5200			
Network	IP	192.168.255.1	Assign UA-5200 a new IP setting according to your case.
	Netmask	255.255.0.0	
	Gateway	192.168.1.1	
OS Account	Username	root	After login, change your password ASAP. (Refer <a href="#">User Manual</a> )
	Password	icpdas	
Web UI Account	Username	root	
	Password	root	

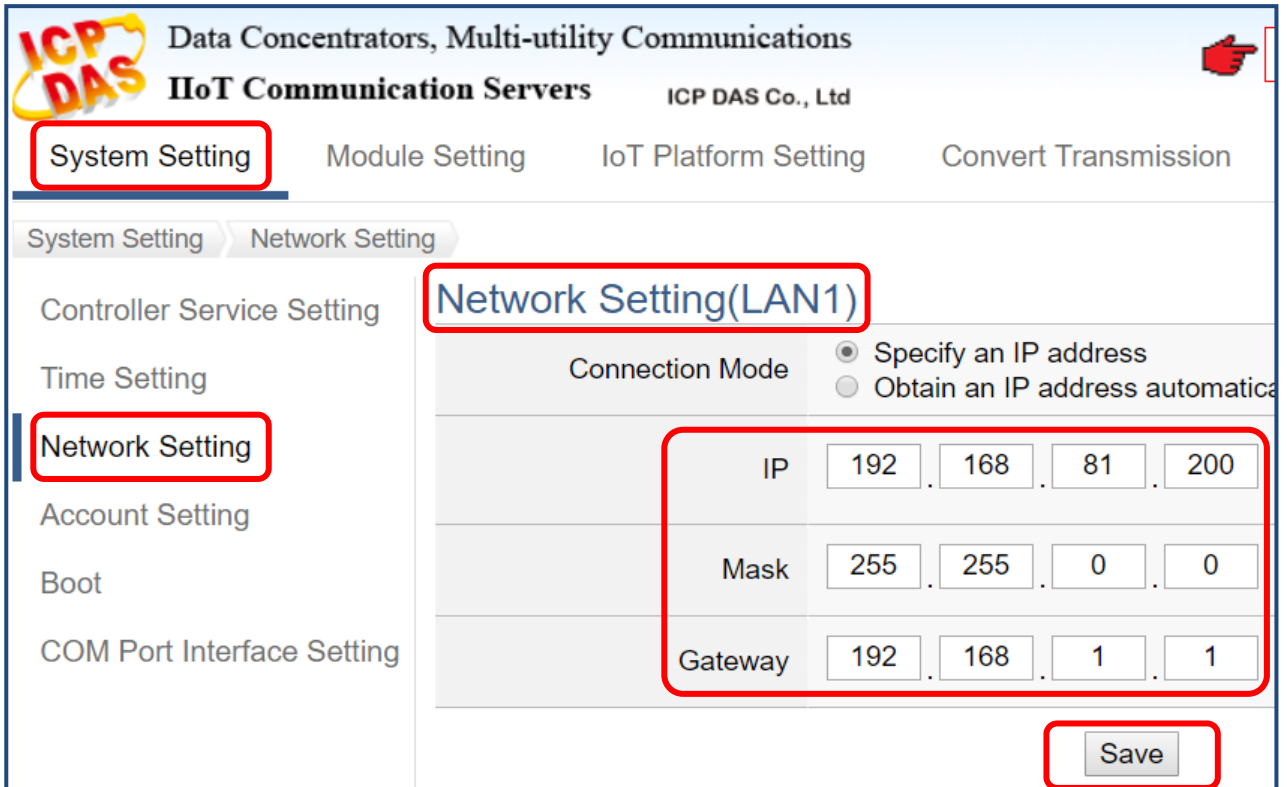
1. Change the PC's IP setting as following. (Write down the PC original network settings before modify.)

IP	192.168.255.10
Subnet mask	255.255.0.0
Gateway address	192.168.1.1

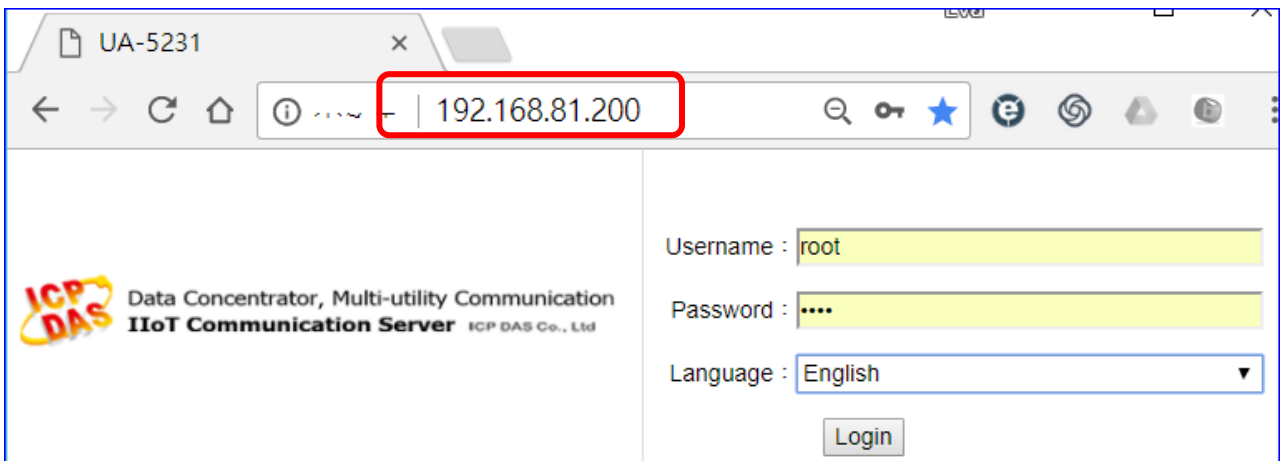
2. Make sure the PC and UA-5200 is connecting through Ethernet. And then open a PC side browser (Ex: Chrome, IE...). Type **http://192.168.255.1** in the URL address. Use default Web UI username/password **root/root** to login the system.



- Click **【System Setting】** → **【Network Setting】** → **【Network Setting(LAN1)】** to change the IP setting by user network.



- Save the IP setting, restore the PC original IP settings, and type the new IP in the browser as step-2 to login the Web UI of UA-5200. And then configure user's UA project.



## 2.2.2. Connection By Utility Searching

The method of using the UA-5200's factory default settings has described in the [Section 2.2.1](#).

If the UA-5200 has a fixed IP and in the same domain as the PC, users can directly enter the IP in the address bar of a web browser and log in to the Web UI of the UA-5200.

In addition to the above 2 methods, users also can use the UA Utility to search Network. This method is suitable for connecting multiple UA series controllers to the Internet, but the IP addresses of UA-5200 are unknown or need to quickly modify the UA controller.

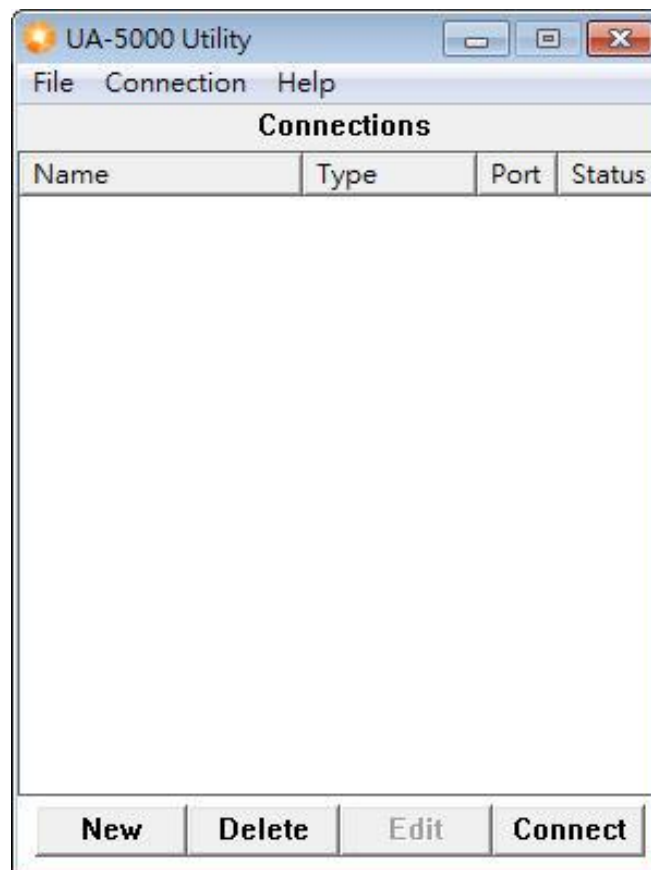
**UA Utility** is UA's free tool software to quickly search each UA-5200 controller on the network and connect to the UA-5200's Web UI for setting UA series controller and project.

In the PC, install the **UA-5000 Utility** (named "UA-5000utility.exe") at the path of the companion CD (i.e., **CD:/UA-5000/Utility/**). Please copy this file to your PC, and then run it to connect the device. Or download the utility program from the website:

<http://ftp.icpdas.com.tw/pub/cd/ua-5000/utility/>

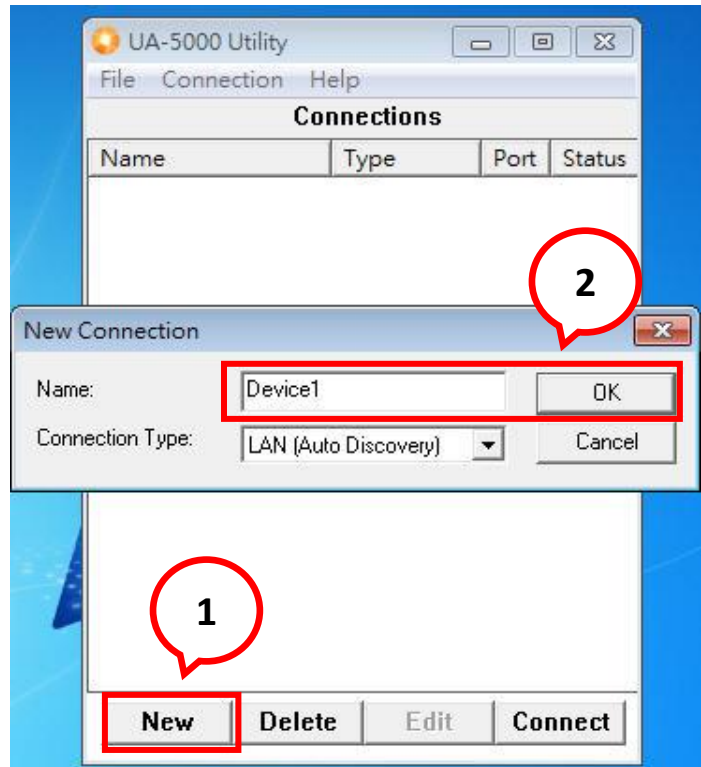
### 1. Install and execute the Utility

Run the UA-5000 Utility (file name: **UA-5000utility.exe**) to install the Utility program.



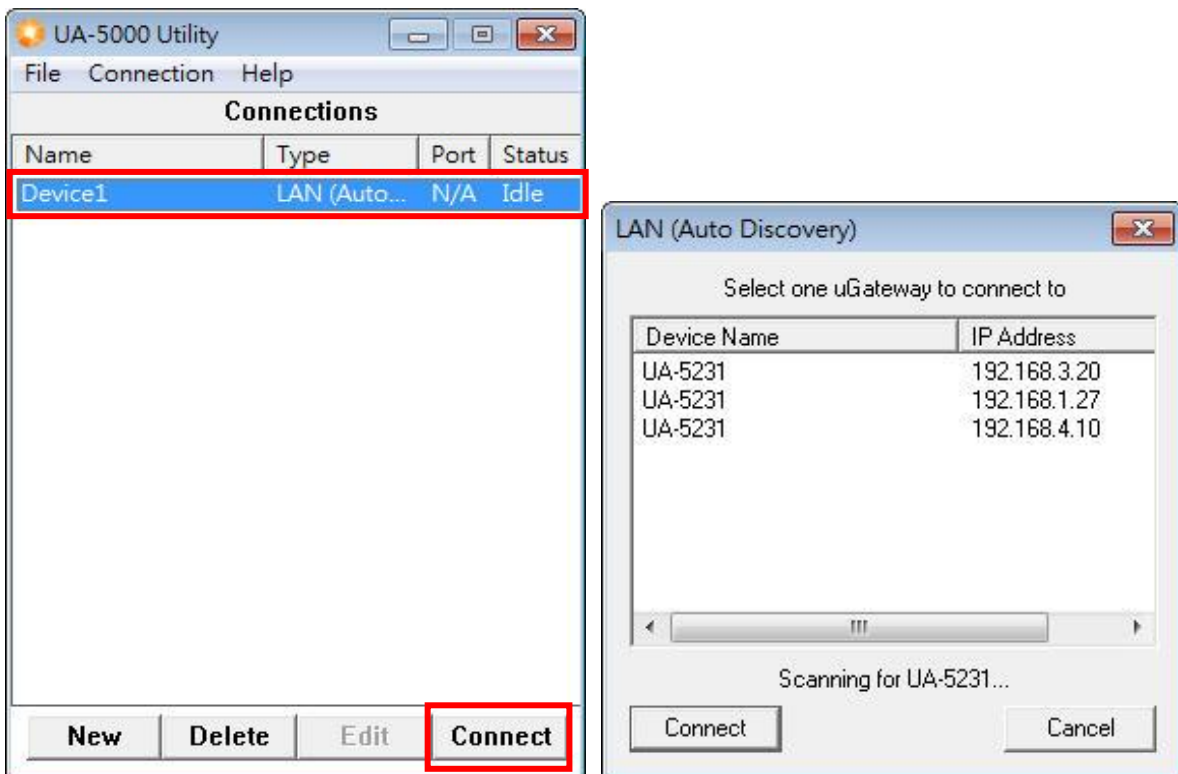
**2. Create a new connection**

Click “New” to add a connection item and give a name for it.



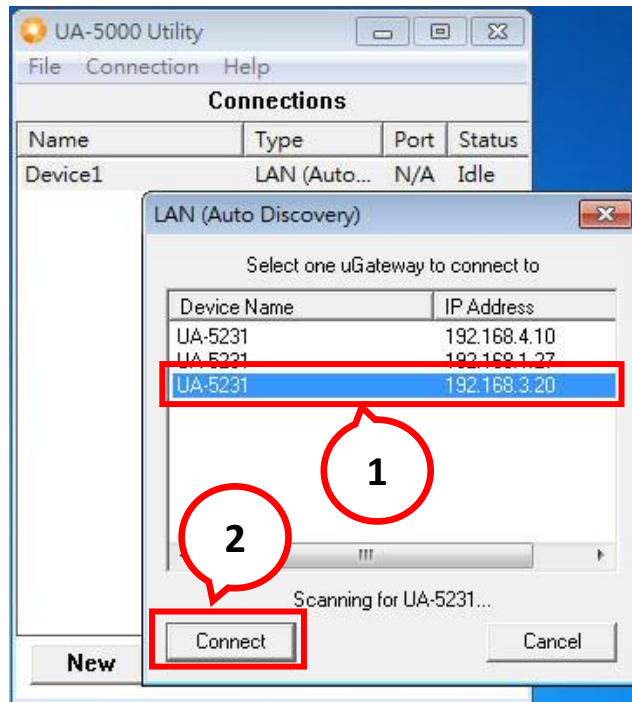
**3. Search the UA-5200 controller**

Mouse double-click on the name you created (or single-click and then click the “Connect” button), this utility will scan and list all UA-5200 devices over the network.



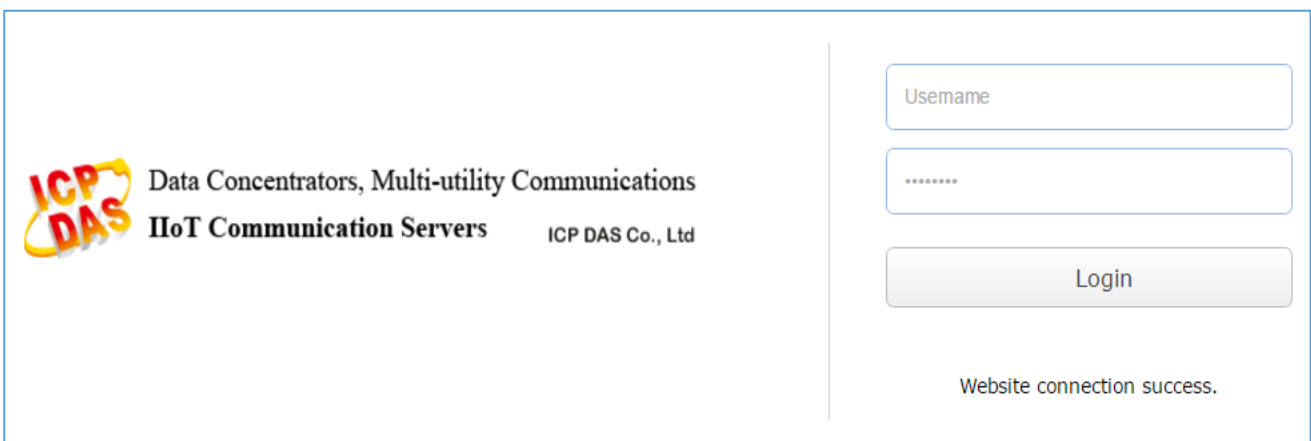
**4. Connect to the UA-5200 controller**

Click the device name you want to connect to, and then click the “Connect” button. It will connect to the UA-5200 webpage via the default Web browser (IE/Chrome...).



**5. Connect to the UA-5200 controller**

The default web browser will be run and direct go to the UA-5200 login web site. Please enter the username and password to login the UA series Web UI. The factory default user name: **root**. The factory default password: **root**.



## 6. Login the Web UI of the UA-5200 controller

When login into the web interface, the UA-5200 default home page (the main configuration screen) will be displayed as below, and will automatically read setting of that UA-5200 to the webpage.

The screenshot shows a web browser window at the address 192.168.81.200/html/Main/Main.html. The page header includes the ICP DAS logo and the text "Data Concentrators, Multi-utility Communications IIoT Communication Servers ICP DAS Co., Ltd". A "Function Wizard" dropdown menu is visible. The main navigation bar contains "System Setting", "Module Setting", "IoT Platform Setting", "Convert Setting", "Advanced Setting", and "File Setting". The "System Setting" section is active, displaying a sidebar with "Controller Service Setting", "Time Setting", "Network Setting", "Account Setting", "Boot", and "COM Port Interface Setting". The main content area is divided into two sections: "Version Information" and "System Setting".

Version Information	
Middleware Version	Version 1.0.0.10
Main Program	Version 1.0.1.0
Web Interface	Version : 2.0.0 Date : 2017/10/31

System Setting	
Controller Service Setting	Controller Service Setting provides the function to display and set the running status of the controller service about the project, MQTT broker and DDNS.
Time Setting	Time Setting provides the function to display and set the date, time and time zone of the controller. (Include manually, synchronization, etc.)
Network Setting	Network Setting provides the function to display and set the network settings. (Include IP, host controller, DDNS, etc.)
Account Setting	Account Setting provides the function to set the username and password of the web UI.
Boot	Boot function provides the function to reboot the controller, and enable the function to run the project, MQTT broker or DDNS at startup.
COM Port Interface Setting	COM Port Interface Setting allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication.

© ICP DAS Co., Ltd. All Rights Reserved

## 2.3. Project Setting Example

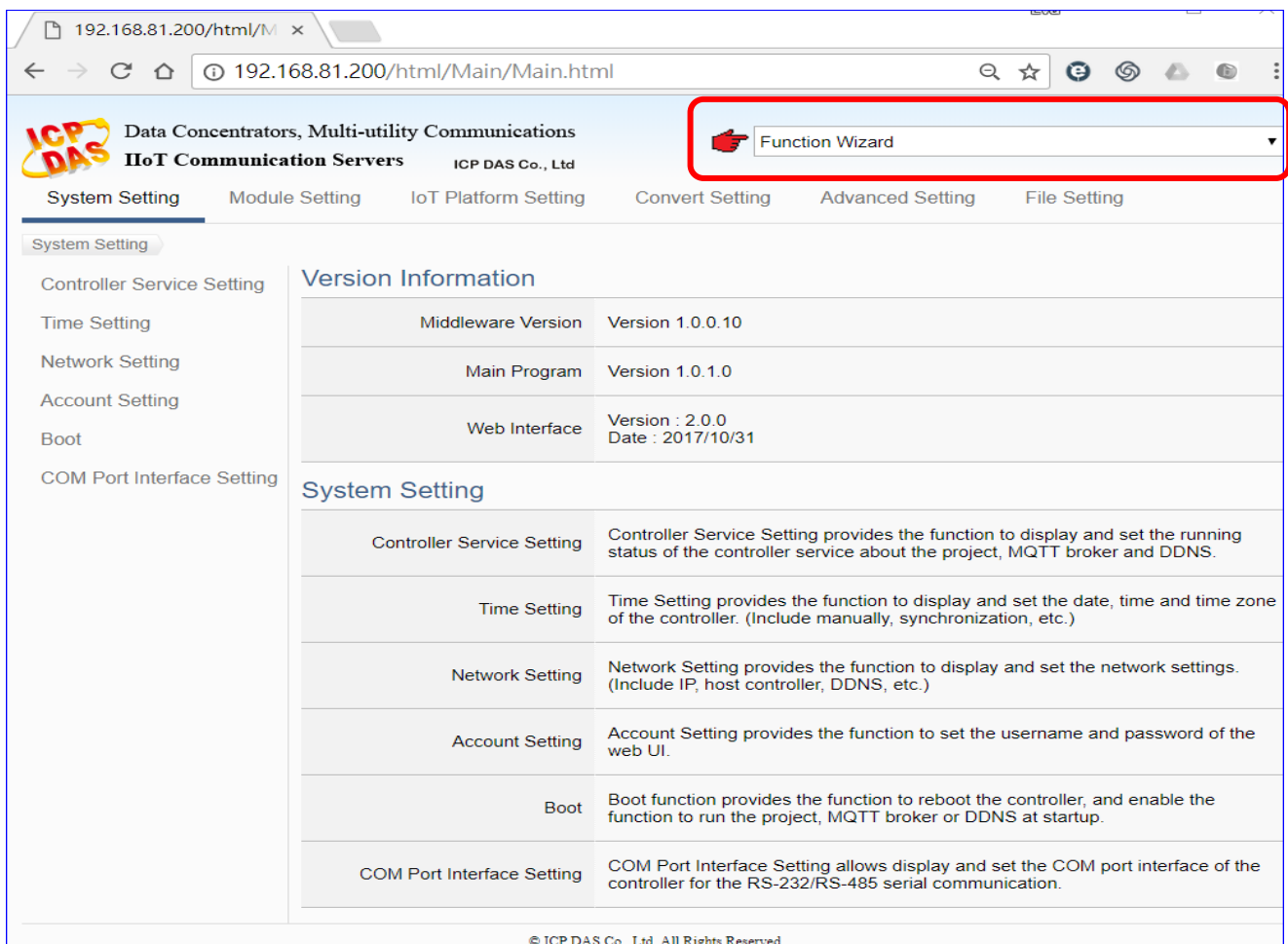
The screen view after login the UA-5200 Web UI (Web User Interface) is as the following picture. Then can start to setup the UA series controller.

If your UA-5200 controller has not connected to the Web UI, please refer to [Section 2.1 Hardware Connection](#) and [Section 2.2 Network Connection](#).

This section will introduce a quick setup method to complete a simple project example to allow users to learn about the project setting flow and steps.

The quick setup method is to use the [ **Function Wizard** ] that at the up-right corner of the Web UI since Version V2.0.0. The Function Wizard provides several items for quick setting the projects or functions step by step like the Wizard guide. The users just follow the “step box” and then can complete the project quickly and well. For more detail information of the Function Wizard, please refer to [Chapter 4](#).

The user can also select the main menu function of the Web UI to setup the project. The complete detail description of the menu functions, please see [Chapter 5](#) ~ Chapter 10.



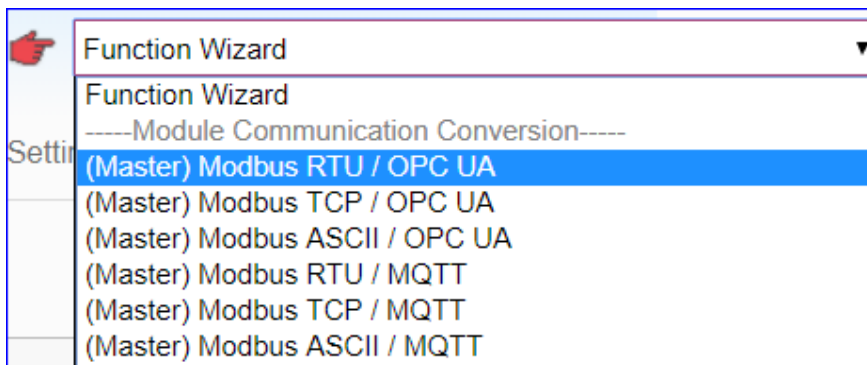
### 2.3.1. A Quick Setup Project Example

This example will setup a project for OPC UA and Modbus RTU (Master) communication protocol conversion using the Function Wizard. The devices include a UA-5231 controller and an M-7055D module that wired with RS-485 interface to read/write the Modbus RTU I/O data and need the convert setting. The connection is show as the picture below.

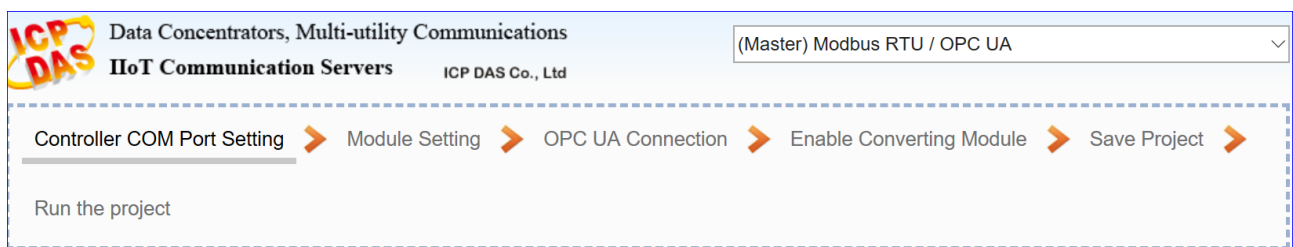


**Note:** 【Function Wizard】 at the up-right corner of the Web UI is a quick setup area. The hardware/network connection methods please see the [Sec. 2.1](#) and [Sec. 2.2](#).

This sample uses the conversion function of the Function Wizard to convert the Modbus RTU / OPC UA, so first click the “(Master) Modbus RTU / OPC UA” item of the Function Wizard.

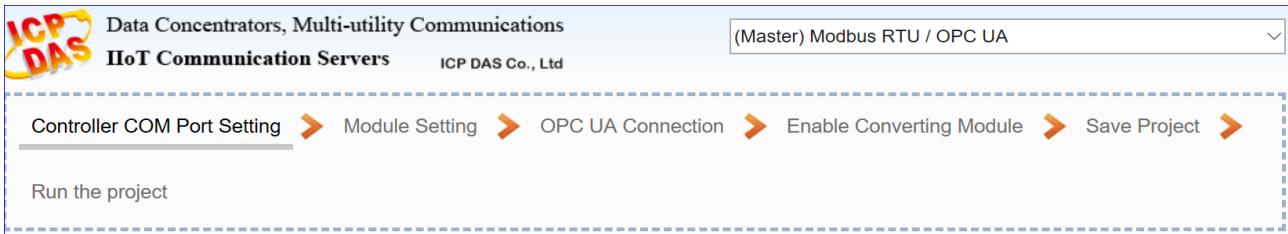


The Web UI will enable a Wizard guide mode and show a “Step Box” (as below picture). The user just needs to follow the “Step Box” step by step and then can complete the project quickly and correctly.





After click the **[(Master) Modbus RTU / OPC UA]** , follow the “**Step Box**” to complete the 6 steps: (The step with a bold underline means it is the current step.)



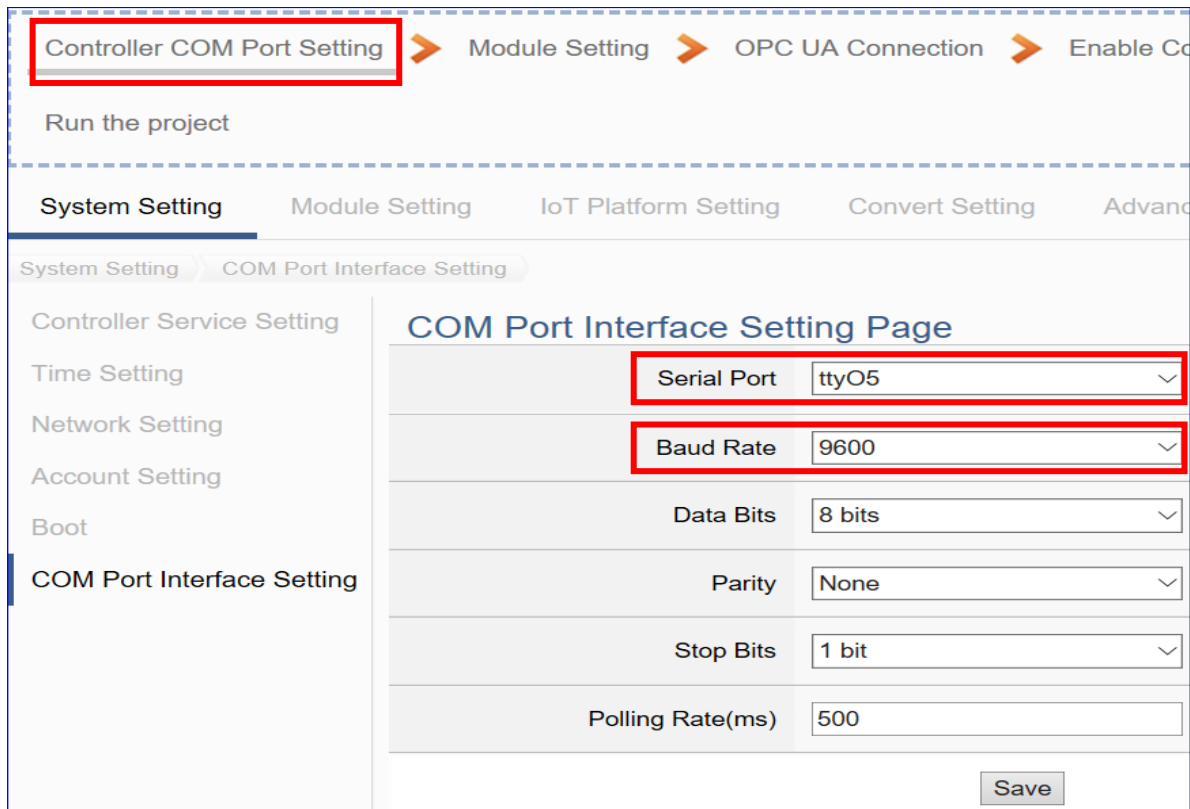
### Step 1. Controller COM Port Setting

This step sets up the COM port of the UA series controller to connect with the module and the communication setting.

#### <This Example>

The UA-5231 uses the ttyO5 port to connect with the M-7055D, so set the **Serial Port: ttyO5**. The M-7055D module default setting is “9600, 8, N, 1”, so set **Baud Rate: 9600**, **others need not to change**. After setting, click [Save] button to save the current settings. (The user also can save the whole project until the step 5 of “Save Project”.)

**Note:** If user uses other port to link other module, or the module is not in the default state, please set this step according to your case. The M-7055D default state can be found in the Module CD or the [Product Web Site](#) .



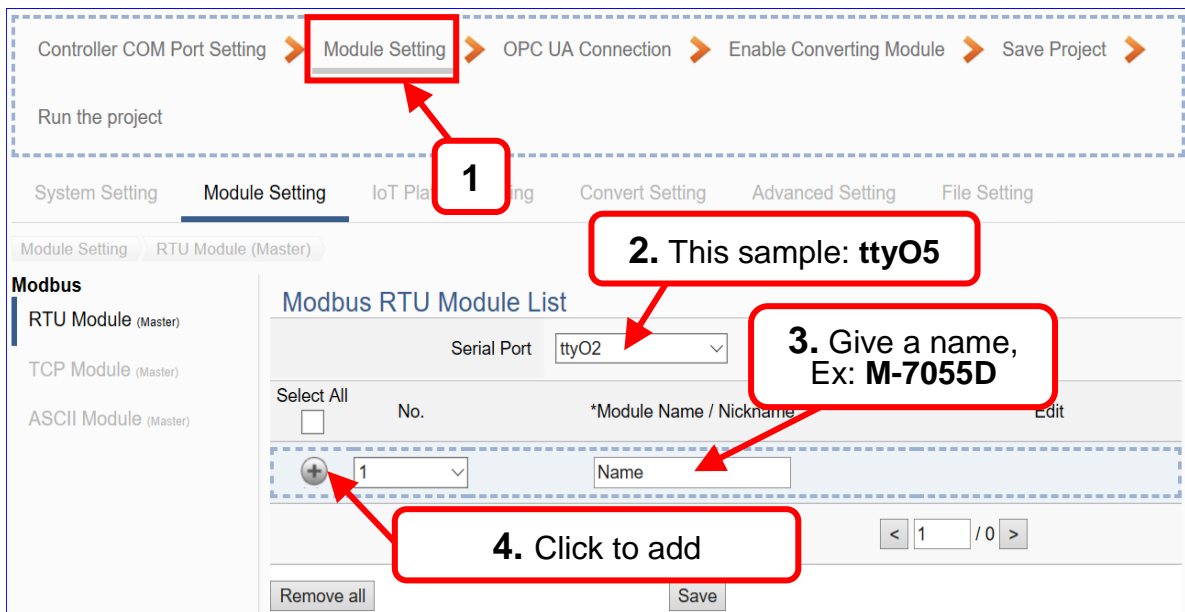
## Step 2. Module Setting

Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting.

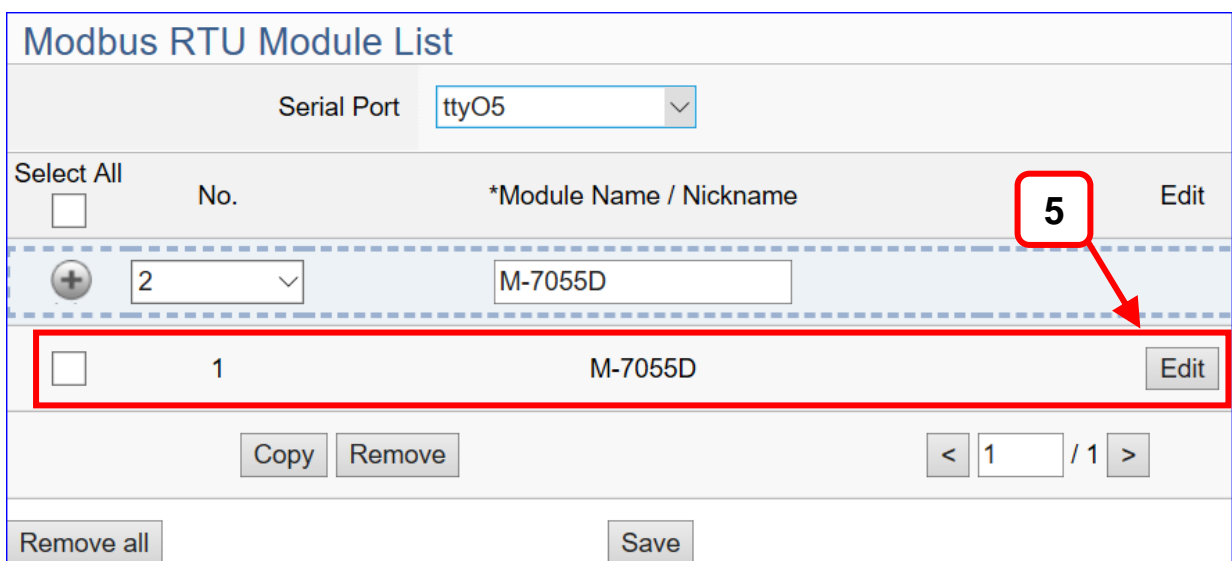
This step is for setting the connected modules. The user can set each module a name (Default name: Name), click [ + ] button to create a new module, and click [Edit] button to configure the module content setting and Modbus mapping table.

### <This Example>

First select the **Serial Port: ttyO5**, give the **Module Name: M-7055D**, and then click the button [ + ] to add a Module List.



Add a module M-7055D as below, and then click [Edit] button to enter the “Module Content Setting” page.



[**Module Content Setting**] page can set up the module and the Modbus address mapping table:

The screenshot shows a web interface for configuring a module. The top section is titled "Module Content Setting" and contains the following fields:

- No.: 1
- Module Name: M-7055D
- Slave ID: 1
- Timeout: 500

The bottom section is titled "Modbus Mapping Table Setting" and is highlighted with a red border. It contains the following fields:

- Data Model: 01 Coil Status(0x)
- Start Address: 0
- Data Number: 1
- Create Tables: Add

Below this is a "Modbus Mapping Table" section with tabs for "Address Setting" and "Nickname". The table has the following columns:

Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)
-----------------	------------------	-----------------------	---------------------

At the bottom of the interface are "OK" and "Cancel" buttons.

Please set up the addresses mapping with the module I/O channels in the [**Modbus Mapping Table Setting**]. The system provides 4 Modbus data models (as below) “01” to “04” for mapping to the **DO, DI, AO and AI** channels.

- 01 Coil Status(0x)
- 02 Input Status(1x)
- 03 Holding Registers(4x)
- 04 Input Registers(3x)

Please note that, the start address of UA series controller is start from address “0”. Although some modules are start from address 1, but here users must to set the start address from 1, and set enough Data number for mapping to the I/O channels of the linking module.

In this example, the M-7055D has DO and 8 DI channels, please create the table as following pictures of the [Modbus Mapping Table Setting]. After complete the setting, the DO and DI Modbus address settings will show in the [Modbus Mapping Table].

M-7055D 8 DO setting and the [Coil Status(0x)] table after setting are as below:

**DO mapping 01** → Data Model: 01 Coil Status(0x)

**UA start address: 0** → Start Address: 0

**DO x 8** → Data Number: 8

**Click [Add]** → Create Tables: Add

Coil Status(0x)	
Address	0
Number	8
Type	Bool
<input type="button" value="Edit"/>	

M-7055D 8 DI setting and the [Input Status(1x)] table after setting are as below:

**DI mapping 02** → Data Model: 02 Input Status(1x)

**UA start address: 0** → Start Address: 0

**DI x 8** → Data Number: 8

**Click [Add]** → Create Tables: Add Success.

Input Status(1x)	
Address	0
Number	8
Type	Bool
<input type="button" value="Edit"/>	

After setting, the Modbus Mapping table is showing as below. Click [OK] to save and exit.

Coil Status(0x)		Input Status(1x)		Holding Registers(4x)		Input Registers(3x)	
Address	0	Address	0				
Number	8	Number	8				
Type	Bool	Type	Bool				
<input type="button" value="Edit"/>		<input type="button" value="Edit"/>					

For more setting item description, please refer to chapter [6. Module Setting](#).

### Step 3. OPC UA Connection

Click the next step, and enter the **Step 3 [OPC UA Connection]** of the UI setting

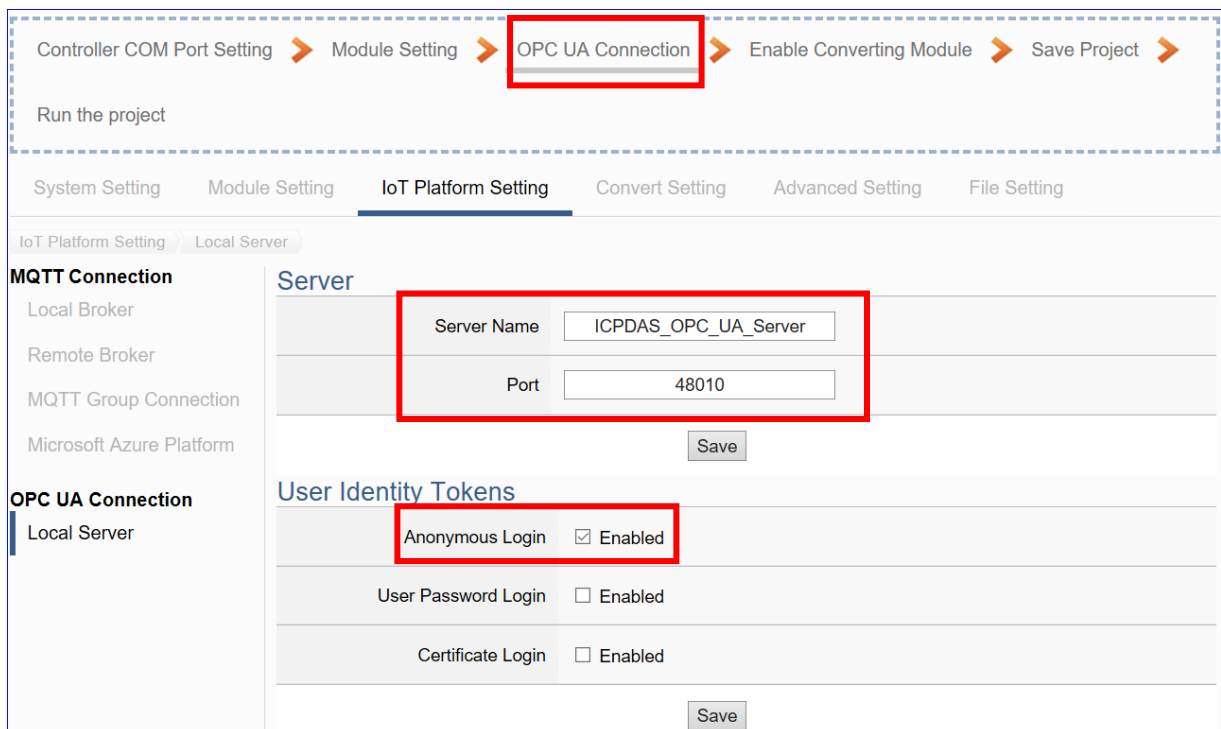
This step is for setting the IoT platform and the OPC UA connection, e.g. the server name, port, login identity information, etc.

We select the “Modbus RTU / OPC UA” conversion at the beginning, so this step will auto enter the **[OPC UA Connection > Local Server]** page of IoT Platform Setting. The “Step Box” will prevent the user from selecting the wrong platform.

#### <This Example>

The server name and port of **[OPC UA Connection]** will auto show up, user needs not to change in this example, but can change the port if needs.

The Anonymous Login default enables, you need not to change in this example. At last click [Save] button.



About other login methods will be found in the [OPC UA Connection] of the [Chapter 7 IoT Platform Setting](#) .

### Step 4. Enable Converting Module

Click the next step, and enter the **Step 4 [Enable Converting Module]** UI setting

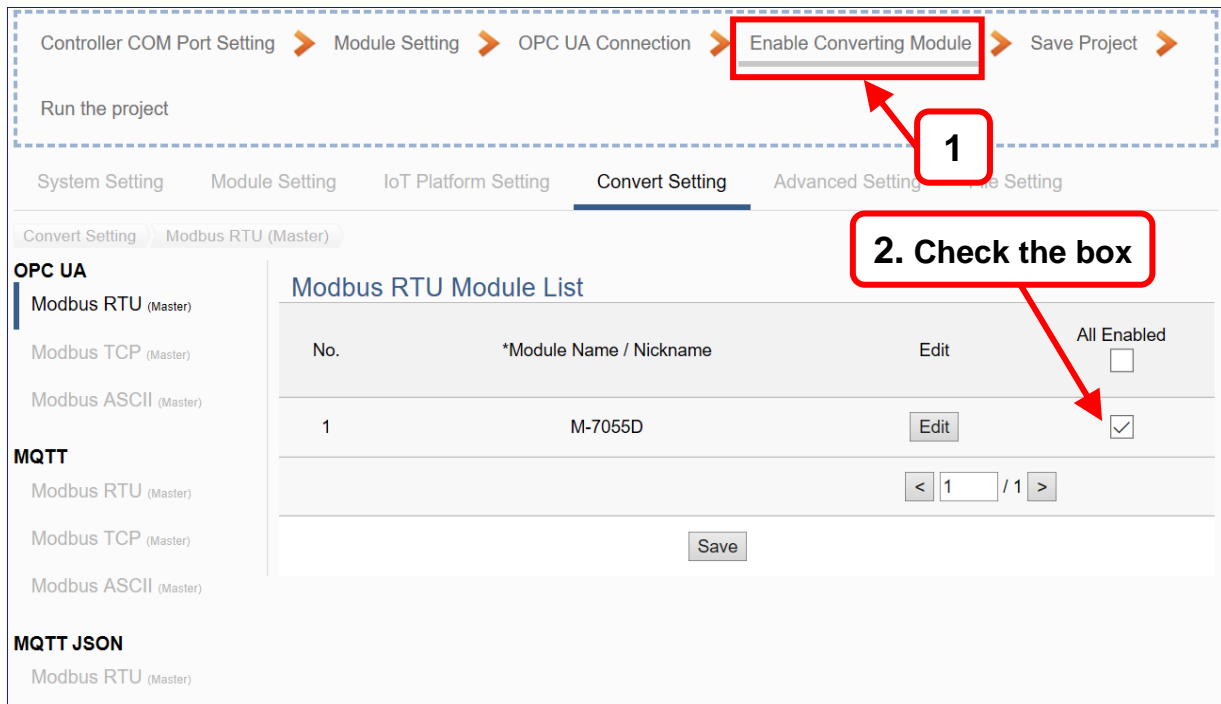
This step is for enabling the Modbus RTU / OPC UA conversion.

We select the “Modbus RTU / OPC UA” conversion at the beginning, so this step will auto enter the **[OPC UA > Modbus RTU (Master)]** page of Conversion setting. The “Step Box” will prevent the user from selecting the wrong platform.

#### <This Example>

In this setting page, please check the enable box of the module **M-7055D** we set up in the previous steps. And click [Save] button.

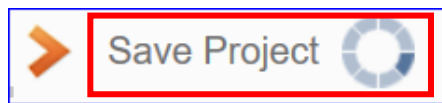
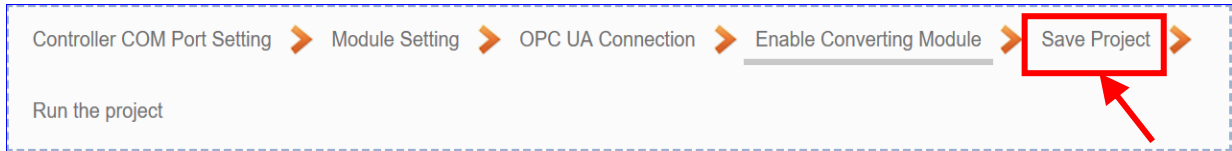
The above action will enable all I/O channels of the M-7055D for communication conversion. If users need to enable some channels only, please click [Edit] to enable individual channels. (Refer to [Chapter 8](#))



### Step 5. Save Project

The setting of this example is finished now, and then to save the whole project and run the project. So the last two steps will not show setting pages, but show some displays.

Click the next step [**Save Project**], the Step Box will show an animation as below picture, that means the project is saving. When the animation vanished, the project is saved completely.

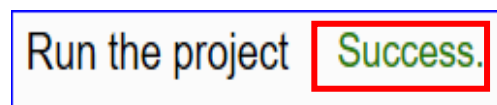
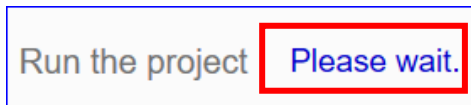


### Step 6. Run the Project

The project, after saving, needs to be executed. Click the next step [**Run the Project**].



The Step Box will show the words “**Please wait**” (as below), that means the system is deleting the old project in the UA controller, and will upload the new project into the UA series and run the new project. When the words “**Please wait**” disappears, the new words “**Success**” appears (as below), that means the UA controller is running new project successfully.



And then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

This example now completes the setting, uploading and running in the UA-5231 controller that connected with the M-7055D and can convert the OPC UA and Modbus RTU protocol communication.

For more and detail setting descriptions of the Web UI, please refer to the following chapters.

### 3. Web UI Login and Environment Overview

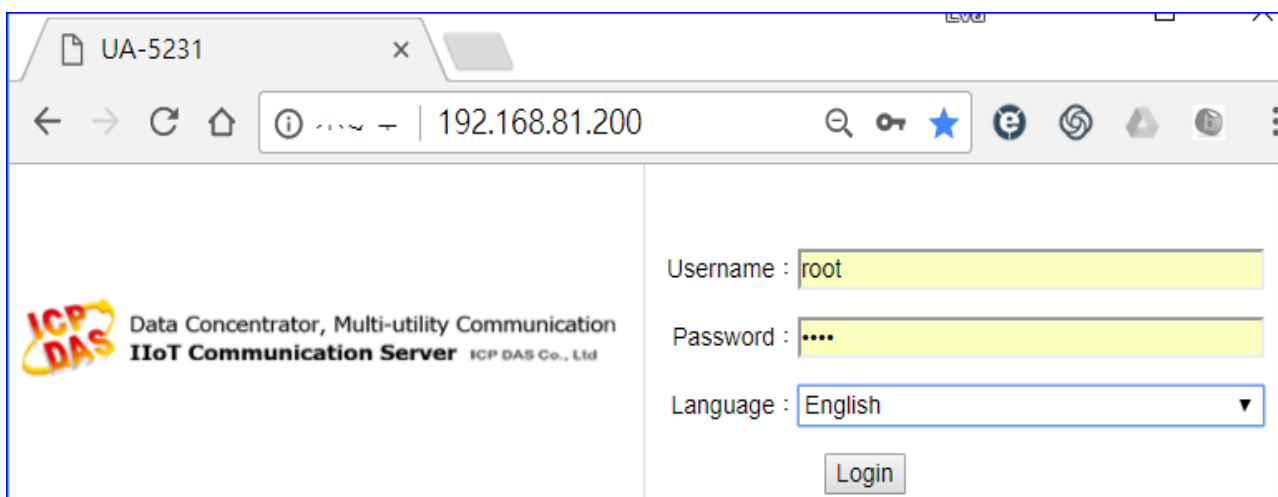
This chapter introduces the ways to login the UA Web User Interface (UI) and the environment of the Web UI of the UA series (IIoT Communication Server), including the version display, system information, function areas, etc. The detail information of the menus, functions, parameters, etc. will be introduced in the next chapters.

#### 3.1. Login The UA Web UI

The methods to login the UA series Web UI:

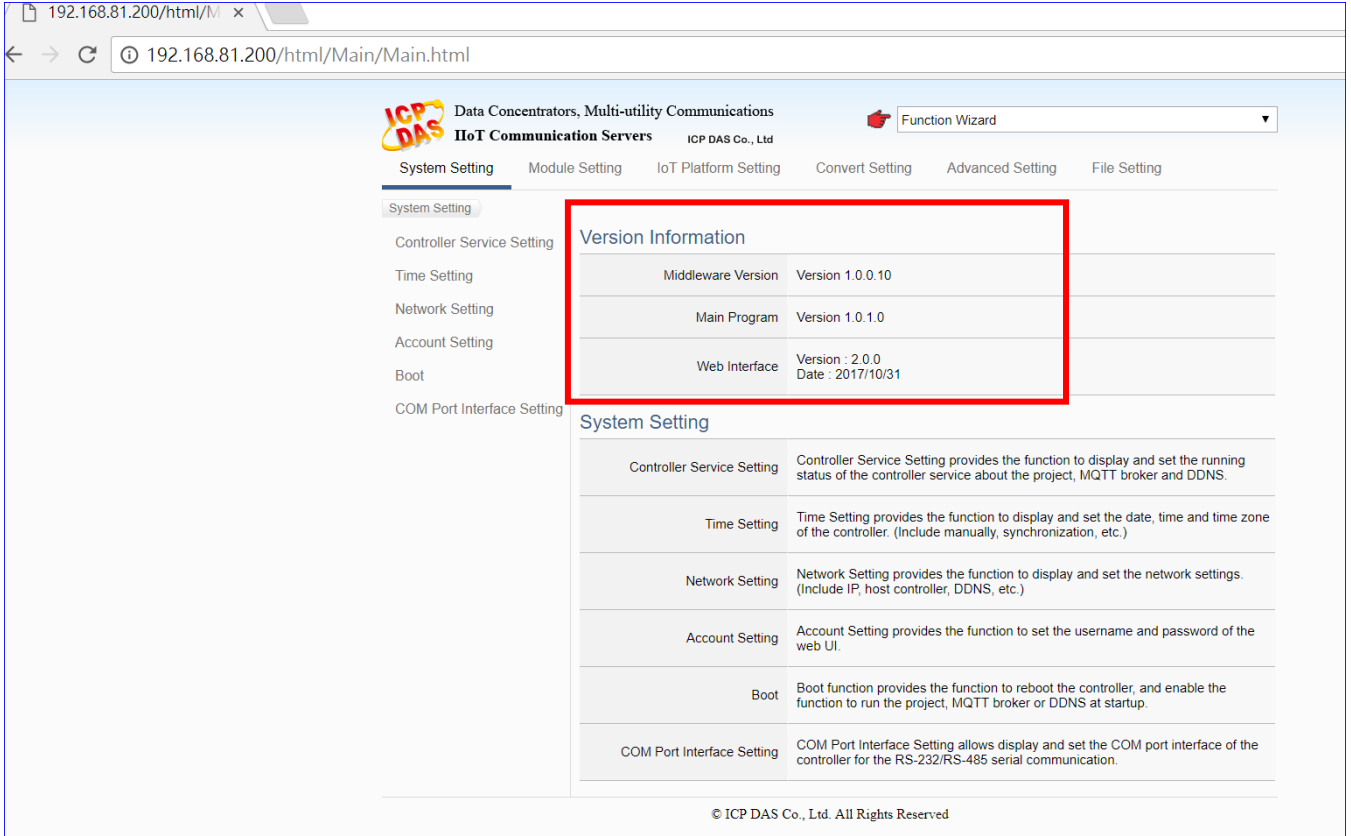
- A. Using Factory Default Setting:** Suitable for the UA new user, setting the new UA controller, or the controller network IP is unknown. This method changes the PC network IP to be the same domain with the UA factory default network IP to login the Web UI. (Refer [Section 2.2.1](#))
- B. Using Software Utility:** UA Series provides a free software utility for auto searching the UA controllers in the network and can quick jump to the login web page of the UA controller. It's very suitable for quick setting when many UA controllers in the network but the IP are unknown. (Refer [Section 2.2.2](#))
- C. Using IP Address:** If the UA-5200 has a fixed IP and in the same domain as the PC, users can directly enter the IP in the address bar of a web browser and log in to the Web UI of the UA-5200. It's suitable for the users how familiar the series controllers.

The login web page for the Web UI of the UA series is as below. Enter the username and the password can log in to set up the UA controller. (Default username/password: root/root)





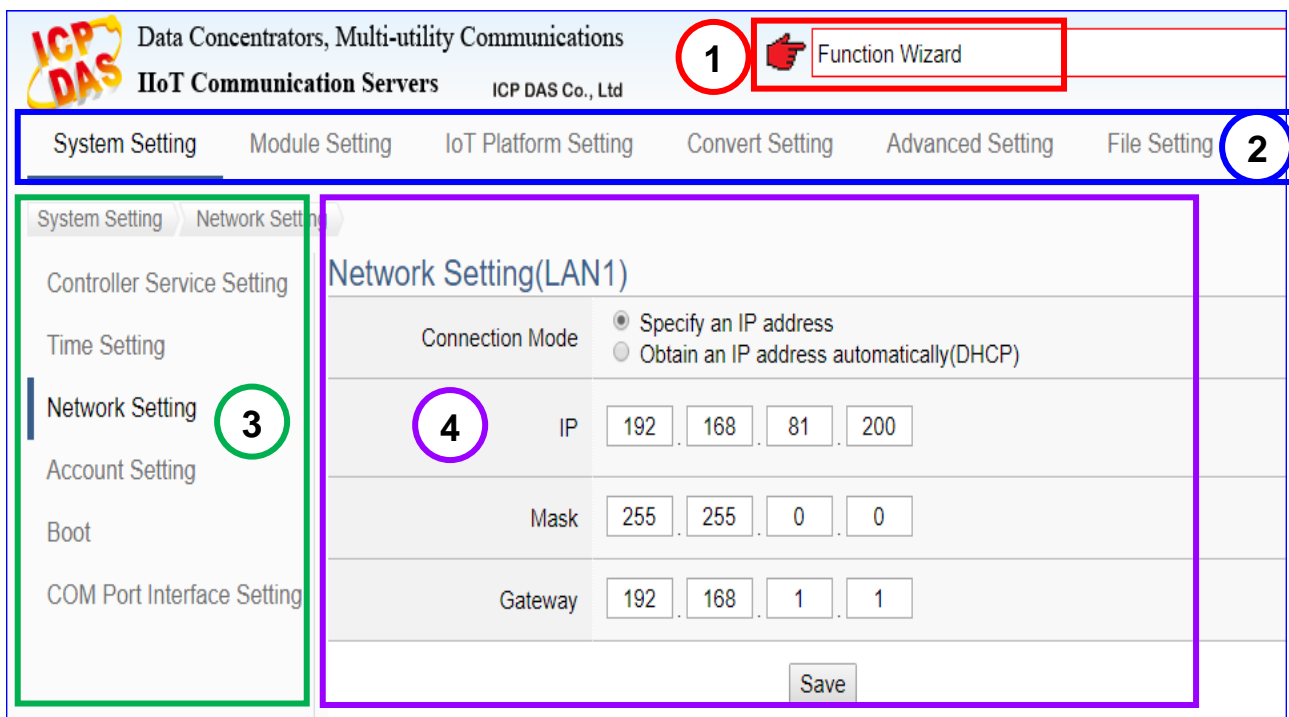
After log in the Web UI, the version information is first displayed on the screen. It includes: the version of the install Middleware program, main program and Web Interface (and date). The following picture shows the screen view of the Web UI since Version 2.0.0.



## 3.2. Web UI Environment Overview

The function setting of the Web UI is including the following areas. The next seven chapters will introduce the settings of the functions and parameters. Here will overview these areas.

1. **Function Wizard**: A quick setup area for commonly used projects or functions suits the new users. The Web UI will enable a Wizard mode and show a “Step Box”. The user just follows the “Step Box” step by step and then can complete the project quickly and rightly. (Refer [Chapter 4](#))
2. **Main Menu Area**: The main menu contains all the setting functions that classified into six categories. Click the main menu item, the sub-menu will appear on the bottom left of the page, and the function descriptions will appear under the main menu area. (Refer [Chapter 5](#) ~ [Chapter 10](#))
3. **Sub-Menu Area**: The sub-menu will display detailed functions under the selected main menu. The user could setup or review detailed function options in the setting area. (Refer from [Chapter 5](#) ~ [Chapter 10](#))
4. **Setting Area**: The setting area is for reviewing and setting the functions and parameters of UA series controller. The content of this area will be varied according to the selected main menu and sub-menu. (Refer [Chapter 5](#) ~ [Chapter 10](#))



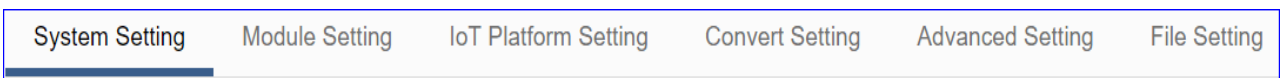
### 3.3. Setting Procedures and Steps

The function setting procedure for the UA series controllers is to set up from the left to the right of the main menu functions. The “Function Wizard” even provides the “**Step Box**” for new users to follow the steps and prevent from selecting the wrong function, e.g. the steps of the commonly used project about the Modbus communication and conversion with the OPC UA protocol are listed as follows:

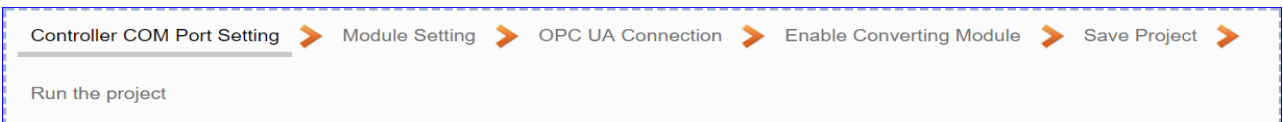
- **Function/Project Procedures:**

Controller Setting > Module Setting > Connecting OPC UA (in IoT Platform) > Conversion > File Setting > Execution

**Main Menu:**



**Step Box of the Function Wizard:**



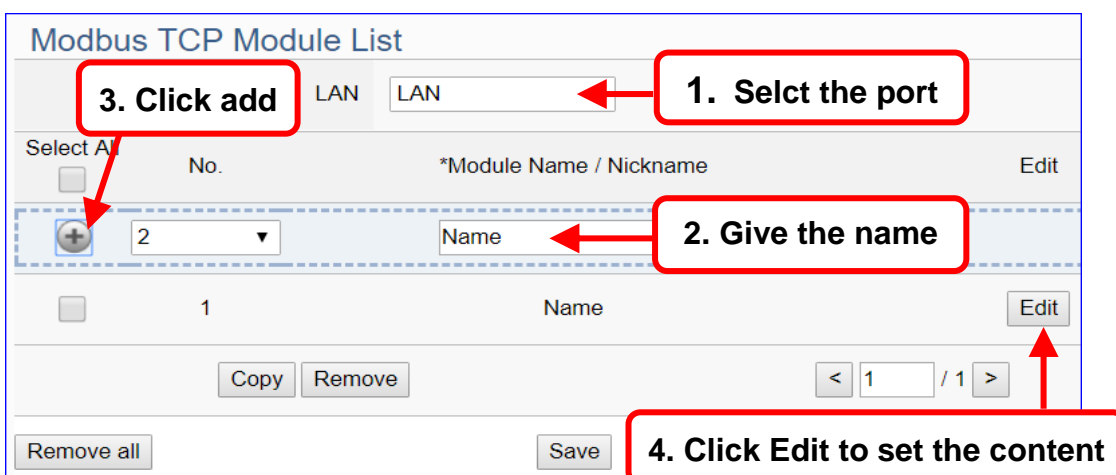
The [chapter 2](#) provides an example for user to quickly know the setting procedure, and the [chapter 4](#) provides various commonly used projects and functions for user to apply.

- **List Setting Steps:**

About the List setting of module, connection..., they have the similar setps as below:

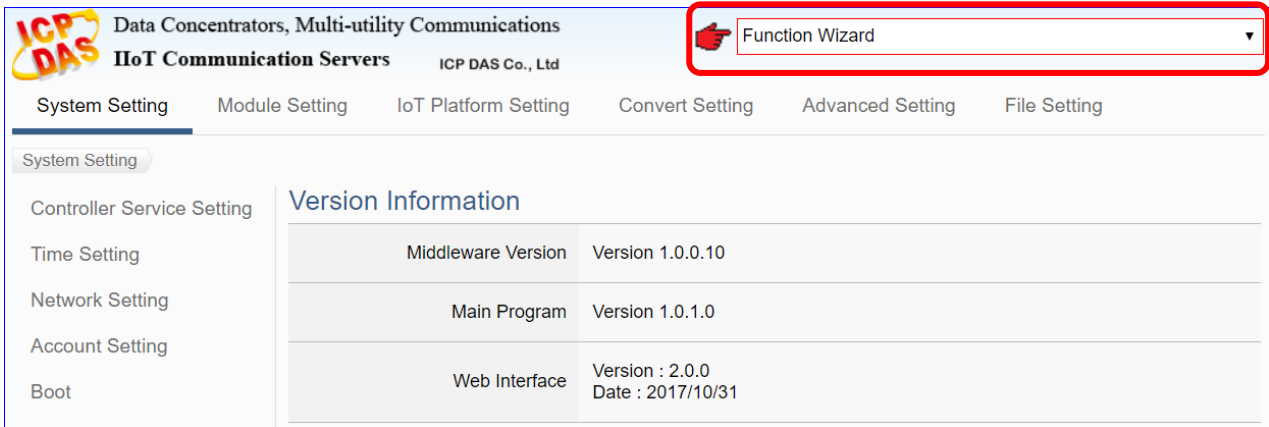
1. Select the prot number, name... for the list (module, connect...)
2. Give a name or nickname, normally default name: Name
3. Click the button [ + ] to add a list of module, connect...
4. Click the button [Edit] to enter the Content Setting page
5. Set up the list content, and then click [Save] to back to the list page.

EX: Modbus TCP Module List:

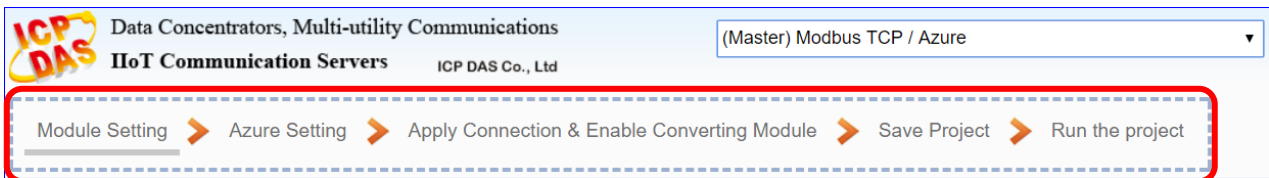


## 4. Function Wizard

[Function Wizard] at the up-right corner of the Web UI since the version V2.0.0 provides a quick setting “Step Box” suitable for new users to set up the projects or functions.

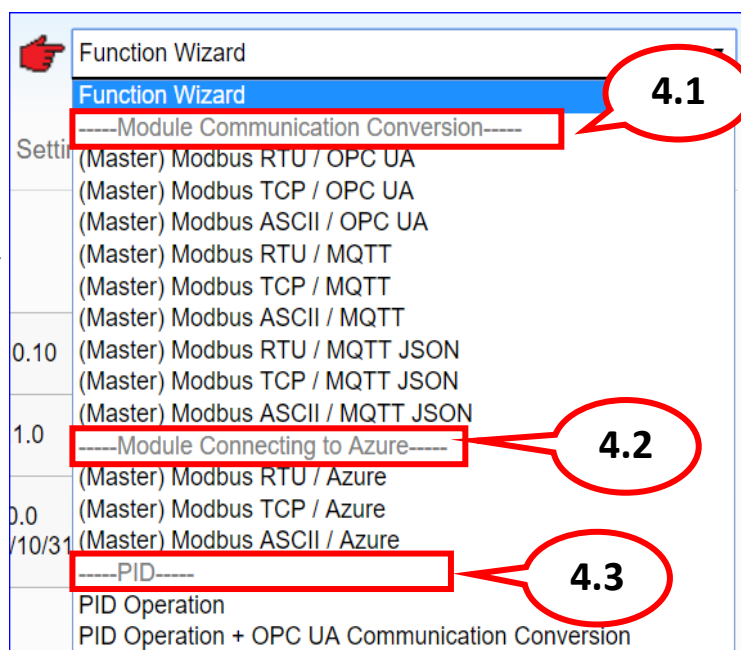


[Step Box] (As below picture) is a Wizzrd-like step guide. When the user selects a function item of the Function Wizard, the Web UI will enable a Wizard mode and show a “Step Box”. The user just needs to follow the “Step Box” step by step and then can complete the project or function quickly and rightly.



This chapter will focus on the steps and function settings. About the real module using, refer to Section 2.3.1, there is a project using M-7055D and UA-5231, and converting Modbus RTU with OPC UA protocol. The user could see that chapter and this chapter to know more procedure concept and setting tips.

Function Wizard will develop more functions or projects, but now there are 13 items in 3 major categories, this chapter will introduce them in three sections.

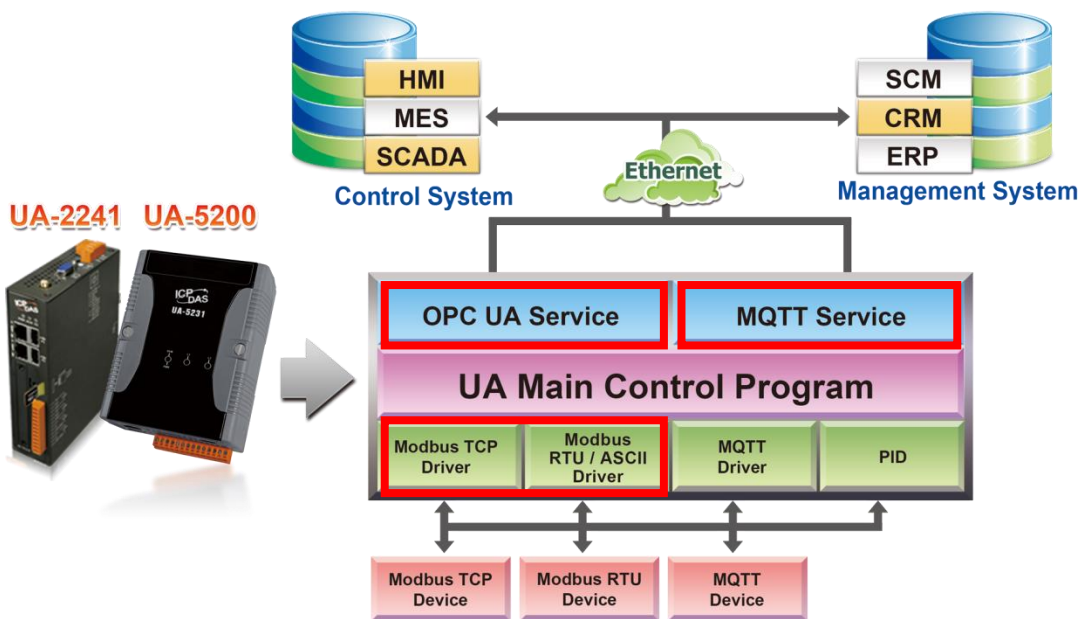


## 4.1. Module Communication Conversion

“Module Communication Conversion” of UA series, a very commonly used function, can effectly communicate the IoT devices or systems (e.g. cloud, database...) with I/O data of the module (e.g. Modbus module). This section will introduce the setting steps and the function parameters of the “Module Communication Conversion”. There are 9 items in this category that can be devided into 3 protocol types and introduced in 3 sub-sections: OPC UA, MQTT, MQTT JSON.

- Module Communication Conversion-----
- (Master) Modbus RTU / OPC UA
  - (Master) Modbus TCP / OPC UA
  - (Master) Modbus ASCII / OPC UA
  - (Master) Modbus RTU / MQTT
  - (Master) Modbus TCP / MQTT
  - (Master) Modbus ASCII / MQTT
  - (Master) Modbus RTU / MQTT JSON
  - (Master) Modbus TCP / MQTT JSON
  - (Master) Modbus ASCII / MQTT JSON

<b>Modbus / OPC UA Conversion</b>	Using the OPC UA Service function to convert with Modbus RTU/TCP/ ASCII protocols. <b>(Section 4.1.1)</b>
<b>Modbus / MQTT Conversion</b>	Using the MQTT Service function to convert with Modbus RTU/TCP/ ASCII protocols. <b>(Section 4.1.2)</b>
<b>Modbus / MQTT JSON Conversion</b>	Using the MQTT Service function in group of JSON format to convert with Modbus RTU/TCP/ ASCII protocols. <b>(Section 4.1.3)</b>

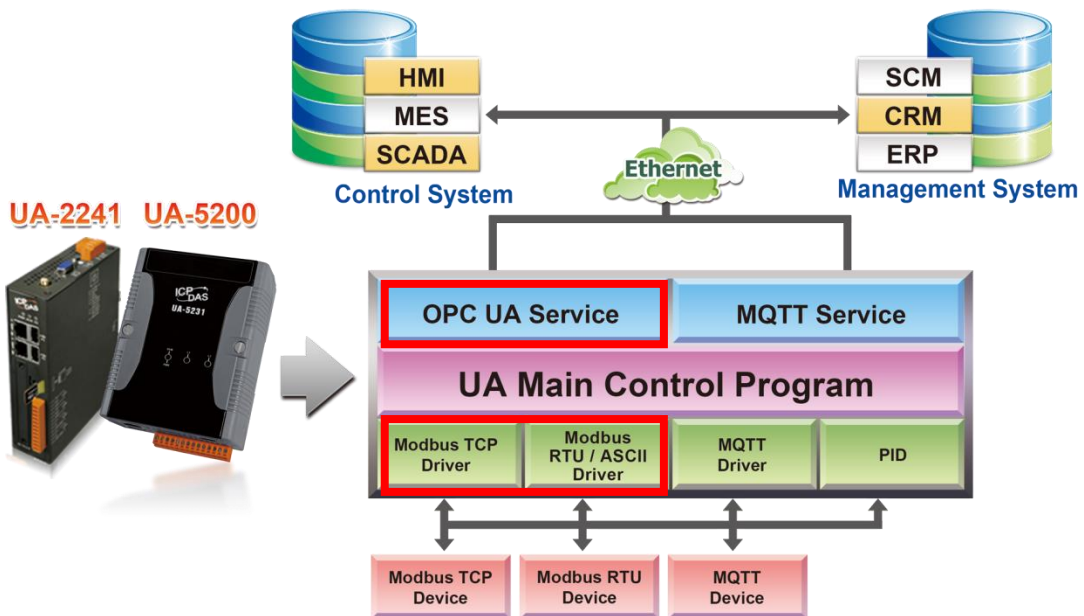


### 4.1.1. Modbus / OPC UA Conversion

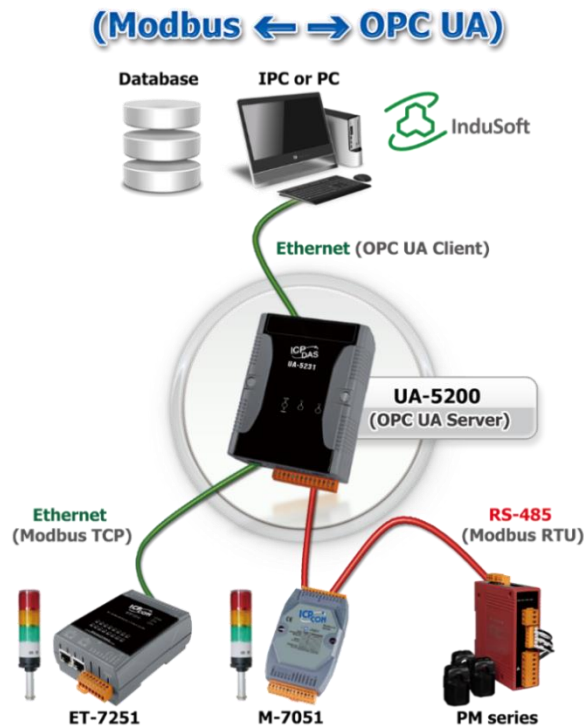
Modbus / OPC UA Conversion include the conversion of OPC UA and Modbus RTU / TCP / ASCII three protocols. With the OPC UA Service function, the OPC UA Server can read and write the Modbus RTU/TCP/ASCII devices that connected to the controller.

The settings of Modbus RTU/ASCII are the same. Here will introduce them together for a setting sample.

#### Modbus / OPC UA Function Diagram:

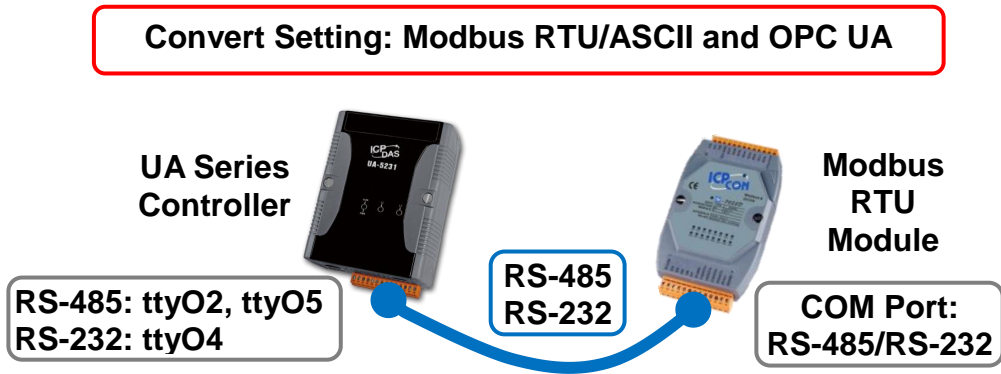


#### Application Solution:



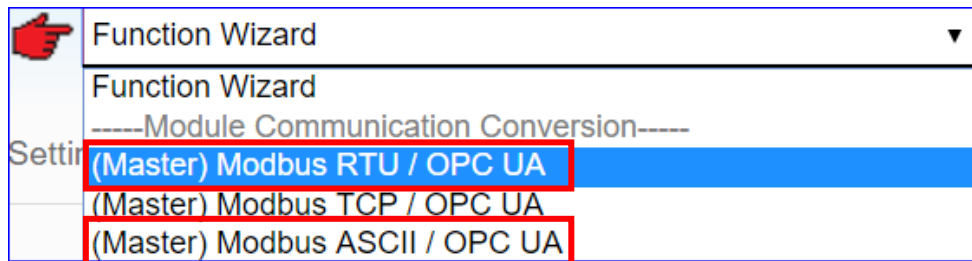
The settings of Modbus RTU/ASCII are the same. Here will introduce them together as a setting sample for Modbus / OPC UA conversion.

- **Convert Setting: Modbus RTU/ASCII and OPC UA**



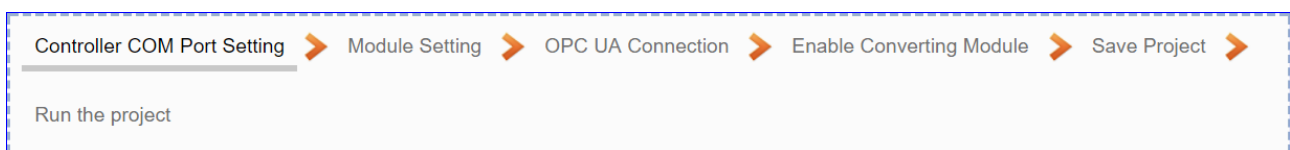
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus RTU or ASCII module (via RS-485 / RS-232, as the picture) and read/write the Modbus I/O by OPC UA Server, user can choose the item [**Modbus RTU / OPC UA**] or [**Modbus ASCII / OPC UA**] of the “Module Communication Conversion” in the Function Wizard.



**[Step Box]:**

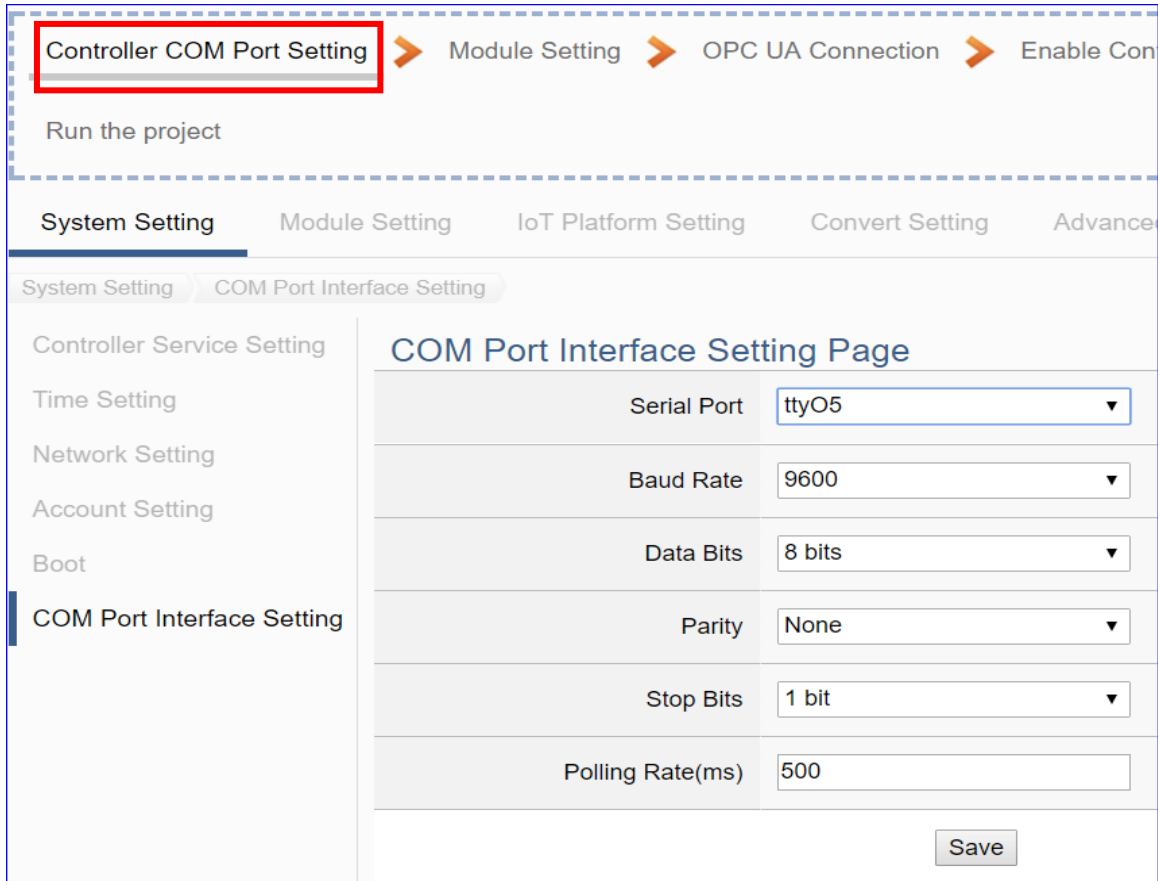
The Step Box of the [**Modbus RTU / OPC UA**] and [**Modbus ASCII / OPC UA**] has the same 6 steps, here will introduce them together. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step by step and then can complete the project quickly and rightly.



### Step 1. Controller COM Port Setting

This page allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication.

The user can find the the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).



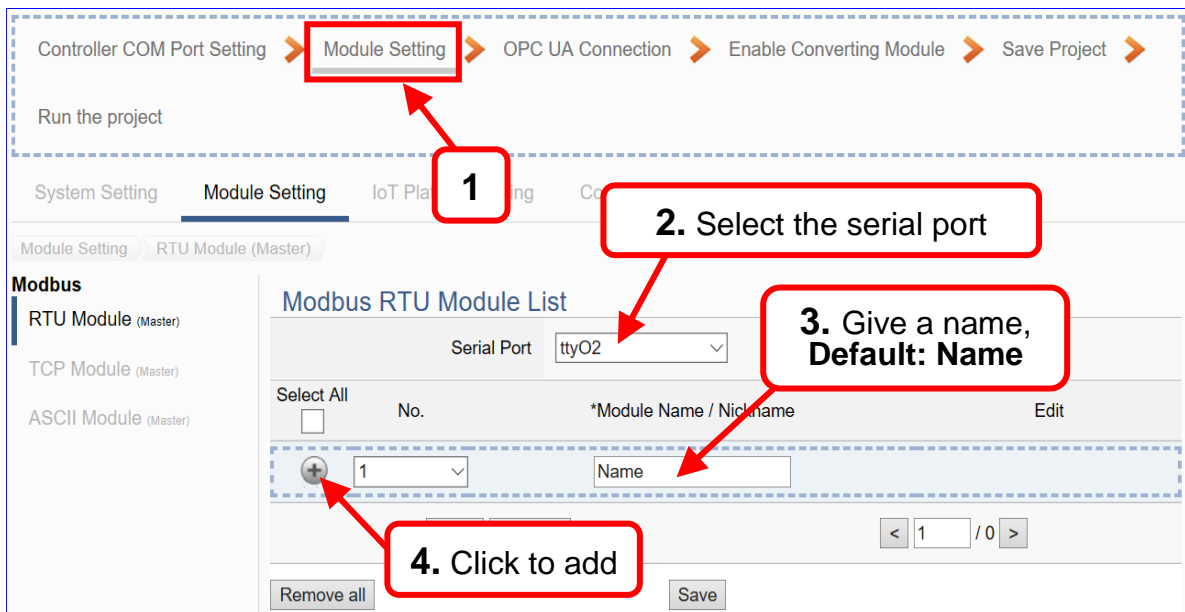
COM Port Interface Setting Page	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485 ; ttyO4: RS-232 ; ttyO5: RS-485
Baud Rate	Choose a baud rate to communicate with the module: 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. The UA controller and the I/O module need have the same baud rate.
Data Bits	The number of bits used to represent one byte of data: 7 bits or 8 bits. Default: 8 Bits.
Parity	Choose one way for the parity checking. Options: None, Even, and Odd. Default: None.
Stop Bits	Choose the number of stop bit: 1 bit or 2 bits. Default: 1.
Polling Rate(ms)	Set a time interval for the command. Default: 500 ms
Save	Click [Save] button could save the settings of this page.



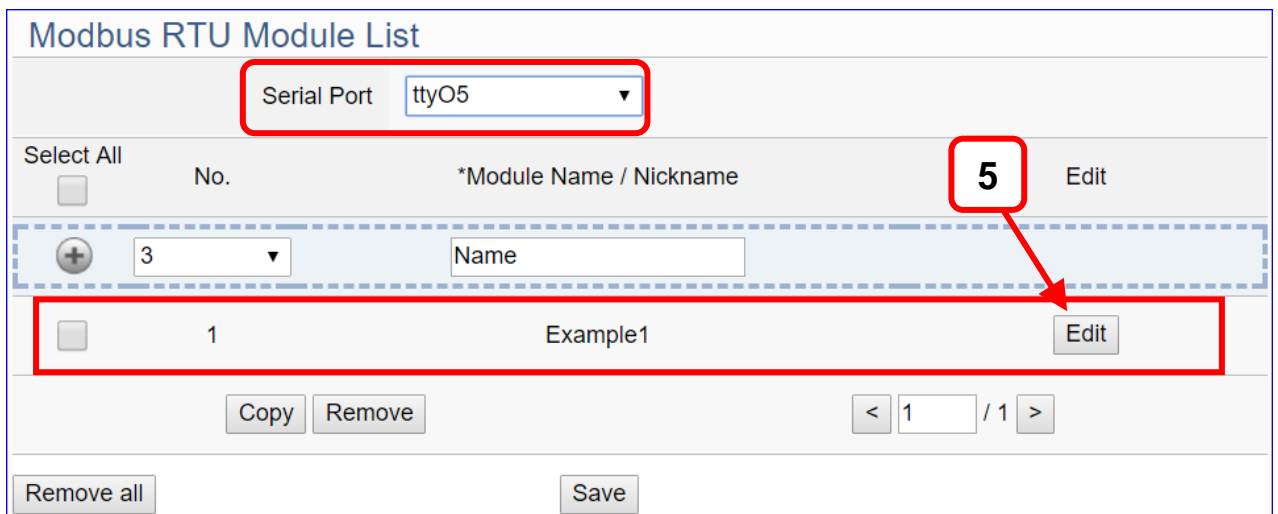
## Step 2. Module Setting

Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First choose the serial port that connected with the module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.



Add a module (No.: 1, Name: Example1) as below, and then click [Edit] button to enter the “Module Content Setting” page.



If set up a wrong module, user can click the box in the left side of the module number and click the [Remove] button to delete the module.

[**Module Content Setting**] page can set up the module and the Modbus address mapping table:

Module Content Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Name1"/>
Slave ID	<input type="text" value="1"/>
Timeout	<input type="text" value="500"/>
Modbus Mapping Table Setting	
Data Model	<input type="text" value="03 Holding Registers(4x)"/>
Start Address	<input type="text" value="0"/>
Data Number	<input type="text" value="1"/>
Type	<input type="text" value="16-bit Short"/>
Create Tables	<input type="button" value="Add"/>

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
Slave ID	Set the module Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models "01" ~ "04" for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI) <div style="border: 1px solid black; padding: 2px; margin-top: 5px;"> <span style="background-color: #007bff; color: white; padding: 2px;">01 Coil Status(0x)</span>  <span style="padding: 2px;">02 Input Status(1x)</span>  <span style="padding: 2px;">03 Holding Registers(4x)</span>  <span style="padding: 2px;">04 Input Registers(3x)</span> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.

The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting	Nickname Setting																																				
Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)																																				
<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>2</td></tr> <tr><td>Type</td><td>Bool</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	2	Type	Bool	<input type="button" value="Edit"/>		<table border="1"> <tr><td>Address</td><td><input type="text" value="0"/></td></tr> <tr><td>Number</td><td><input type="text" value="1"/></td></tr> <tr><td>Type</td><td>Bool</td></tr> <tr><td colspan="2"><input type="button" value="Delete"/></td></tr> <tr><td colspan="2"><input type="button" value="Save"/></td></tr> <tr><td colspan="2"><input type="button" value="Cancel"/></td></tr> </table>	Address	<input type="text" value="0"/>	Number	<input type="text" value="1"/>	Type	Bool	<input type="button" value="Delete"/>		<input type="button" value="Save"/>		<input type="button" value="Cancel"/>		<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>1</td></tr> <tr><td>Type</td><td>Short</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	1	Type	Short	<input type="button" value="Edit"/>		<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>1</td></tr> <tr><td>Type</td><td>Float</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	1	Type	Float	<input type="button" value="Edit"/>	
Address	0																																						
Number	2																																						
Type	Bool																																						
<input type="button" value="Edit"/>																																							
Address	<input type="text" value="0"/>																																						
Number	<input type="text" value="1"/>																																						
Type	Bool																																						
<input type="button" value="Delete"/>																																							
<input type="button" value="Save"/>																																							
<input type="button" value="Cancel"/>																																							
Address	0																																						
Number	1																																						
Type	Short																																						
<input type="button" value="Edit"/>																																							
Address	0																																						
Number	1																																						
Type	Float																																						
<input type="button" value="Edit"/>																																							
Press Save to finish editing.																																							
<input type="button" value="OK"/> <input type="button" value="Cancel"/>																																							

Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

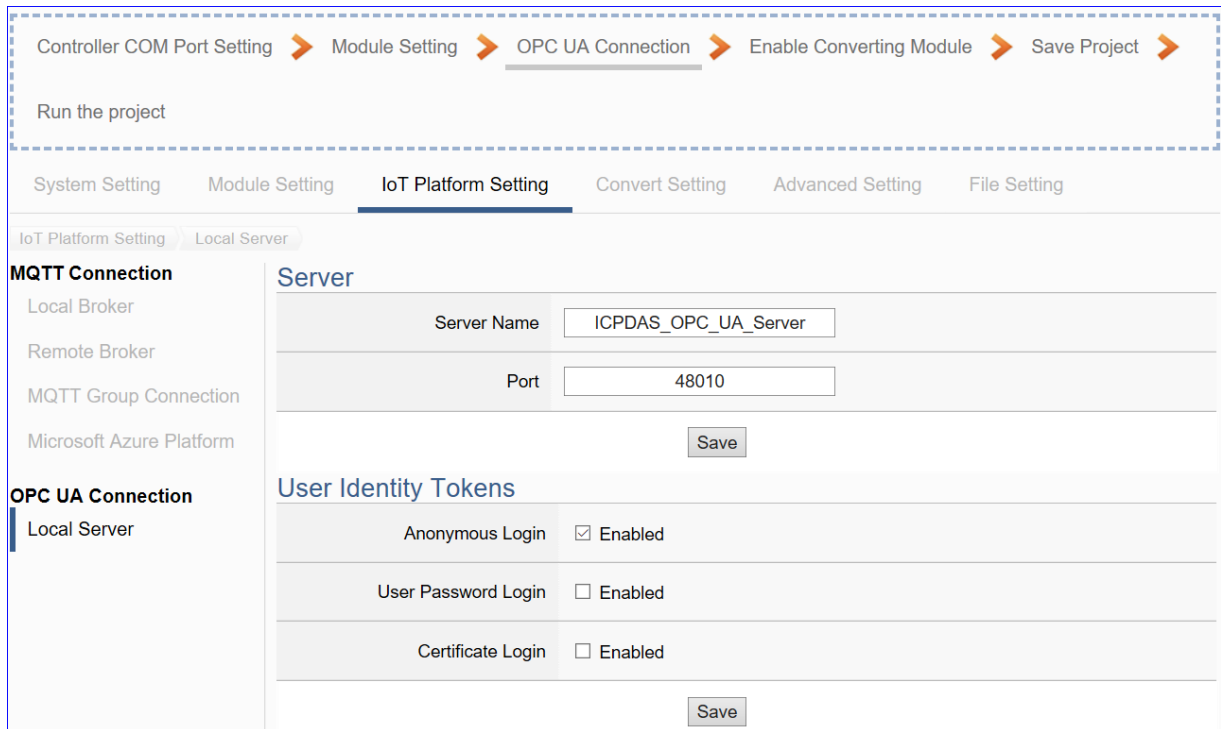
Modbus Mapping Table		Address Setting	Nickname Setting	
<b>01 Coil Status(0x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
1	<input type="text" value="Tag1"/>	Bool	<input type="text"/>	
<b>02 Input Status(1x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
<b>03 Holding Registers(4x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input type="text"/>
<b>04 Input Registers(3x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input type="text"/>
		<input type="button" value="OK"/>	<input type="button" value="Cancel"/>	

Modbus Mapping Table – Nickname Setting	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

### Step 3. OPC UA Connection

Click the next step, and enter the **Step 3 [OPC UA Connection]** of the UI setting. This page is for setting the IoT platform and the OPC UA connection, e.g. the server name, port, login identity information, etc.

We select the “Modbus RTU / OPC UA” conversion at the beginning, so this step will auto enter the **[OPC UA Connection > Local Server]** page of IoT Platform Setting. The “Step Box” will prevent the user from selecting the wrong platform.

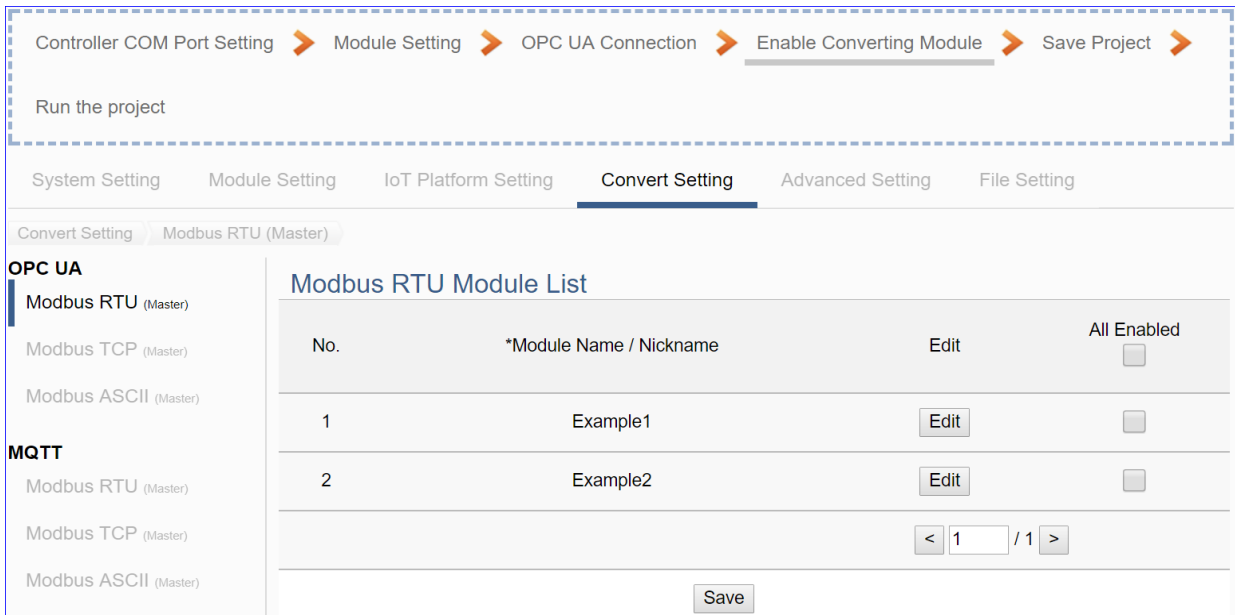



OPC UA Connection > Local Server Setting –Server	
Server Name	Display the active OPC UA Server name. Not editable. System valus: ICPDAS_OPC_UA_Server
Port	The communication port number of the OPC UA Server. System Default: 48010.
Save	Click to save the settings of this item.
OPC UA Connection > Local Server Setting –User Identity Tokens	
Anonymous Login	Check to enable the anonymous login of clients. Default: check.
User Password Login	Check to enable the user password login of clients. Default: uncheck.
Certificate Login	Check to enable the certificate login of clients. Default: uncheck.
Save	Click to save the settings of this item.

### Step 4. Enable Converting Module

Click the next step, and enter the **Step 4 [Enable Converting Module]** UI setting  
 This step is for enabling the Modbus RTU (or ASCII) / OPC UA conversion.

We select the “Modbus RTU (or ASCII) / OPC UA” conversion at the beginning, so this step will auto enter the **[OPC UA > Modbus RTU/ASCII (Master)]** page of Conversion setting. The “Step Box” will prevent the user from selecting the wrong platform.



Convert Setting > OPC UA > Modbus RTU (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	If user wants to enable some I/O channels for conversion, click [Edit] of that module to enter the “Variable Tale” setting. It is normal to set all channels as enabled, and the conversion will not affect the unconnected channels.
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “Module Content Setting” page:

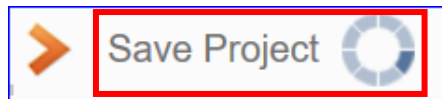
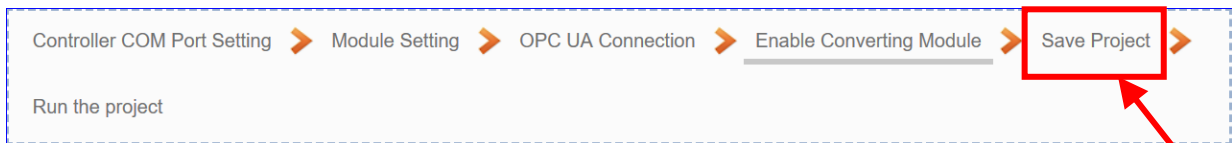
Module Content Setting			
No.	<input type="text" value="1"/>		
Module Name	<input type="text" value="Example1"/>		
Variable Table			
Name	Attribute	Data Type	Enabled <input type="checkbox"/>
Tag0	<input type="text" value="Read"/>	Float	<input type="checkbox"/>
Tag0	<input type="text" value="Read / Write"/>	Short	<input checked="" type="checkbox"/>
Tag0	<input type="text" value="Read"/>	Bool	<input checked="" type="checkbox"/>
Tag1	<input type="text" value="Read"/>	Bool	<input type="checkbox"/>
Tag0	<input type="text" value="Read / Write"/>	Bool	<input checked="" type="checkbox"/>
Tag1	<input type="text" value="Read / Write"/>	Bool	<input type="checkbox"/>
<input type="button" value="OK"/> <input type="button" value="Cancel"/>			

Convert Setting > OPC UA > Modbus RTU (Master) – Module Content	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
Convert Setting > OPC UA > Modbus RTU (Master) – Variable Table	
Name	Display the variable name that set in the Modbus Address Mapping Table page (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

### Step 5. Save Project

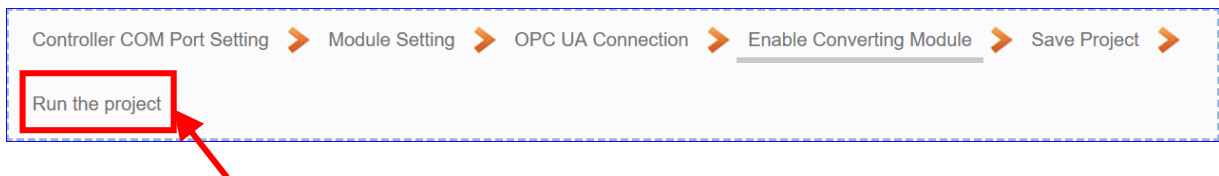
The setting of this example is finished now, and then to save the whole project and run the project. So the last two steps will not show setting pages, but show some displays.

Click the next step [**Save Project**], the Step Box will show an animation as below picture, that means the project is saving. When the animation vanished, the project is saved completely.

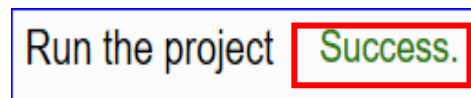


### Step 6. Run the Project

The project, after saving, needs to be executed. Click the next step [**Run the Project**].



The Step Box will show the words "**Please wait**" (as below), that means the system is deleting the old project in the UA controller, and will upload the new project into the UA series and run the new project. When the words "**Please wait**" disappears, the new words "**Success**" appears (as below), that means the UA controller is running new project successfully.



And then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

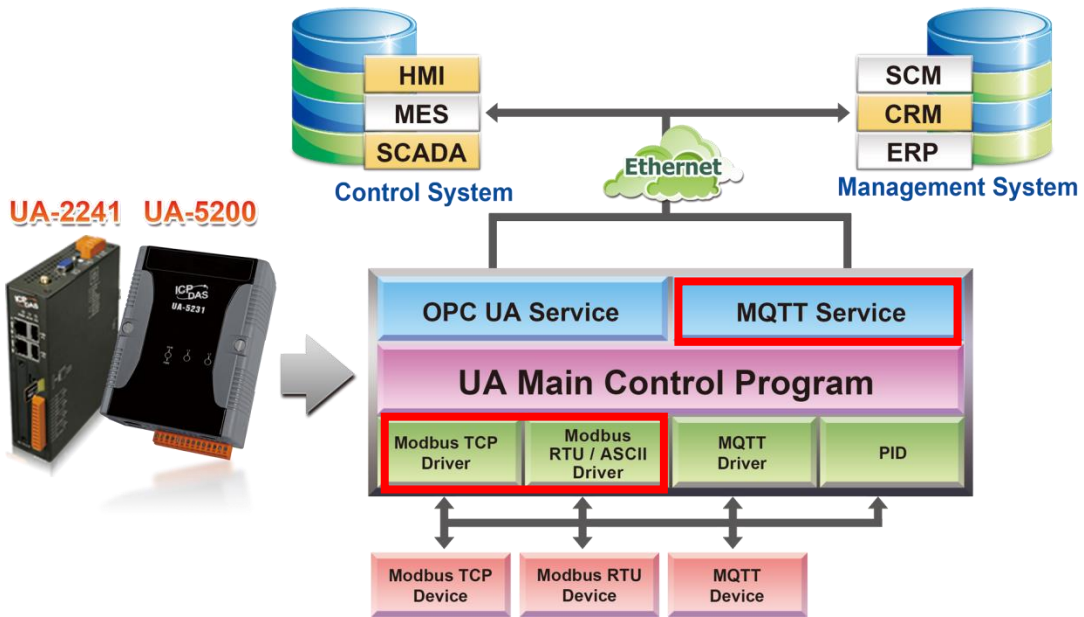
The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication.



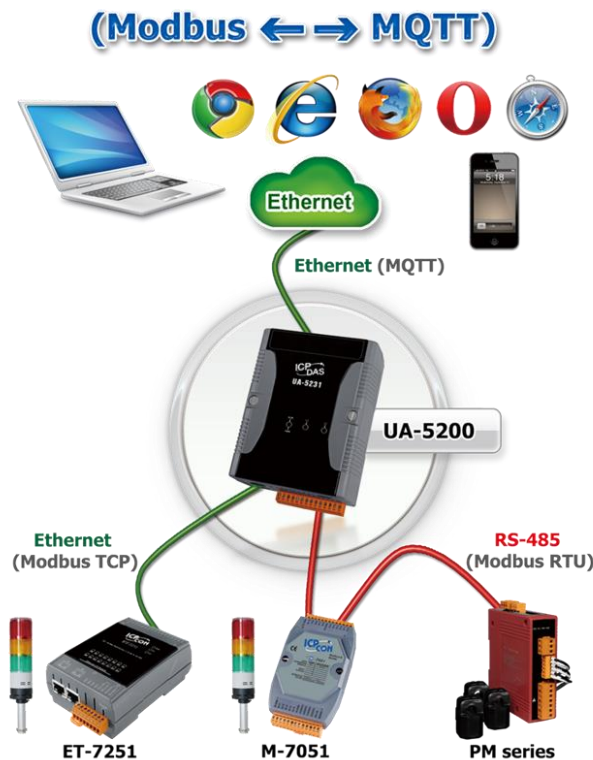
### 4.1.2. Modbus / MQTT Conversion

Modbus / MQTT Conversion include the conversion of MQTT and Modbus RTU / TCP / ASCII three protocols. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus device that connected to the controller.

#### Modbus / MQTT Function Diagram:

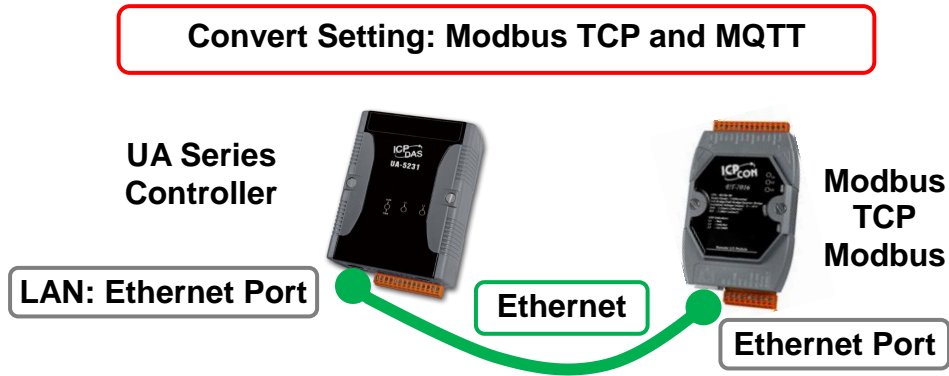


#### Application Solution:



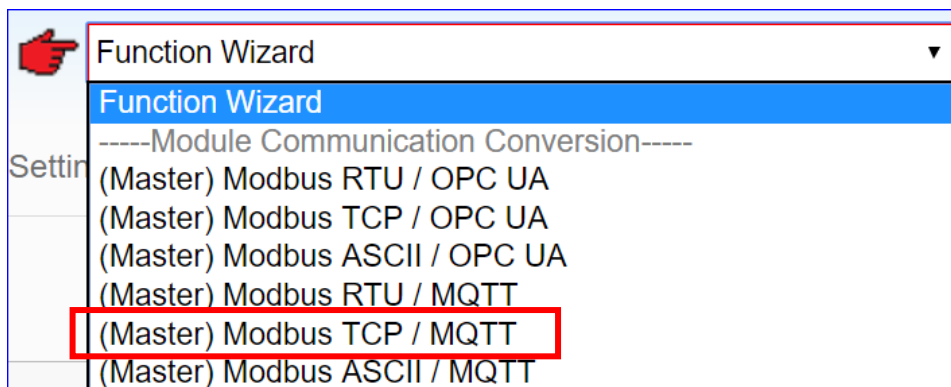
This section introduces the Modbus / MQTT conversion through the conversion of Modbus TCP and MQTT protocol.

- **Convert Setting: Modbus TCP and MQTT**



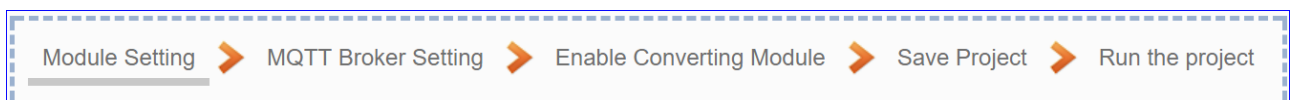
**Note:** The hardware/network connection methods please see the [Chapter 2](#) .

When UA series controller connects the Modbus TCP (via Ethernet, as the picture) and read/write the Modbus I/O via MQTT Broker, user can choose the item [**Modbus TCP / MQTT**] of the “Module Communication Conversion” in the Function Wizard.



**[Step Box]:**

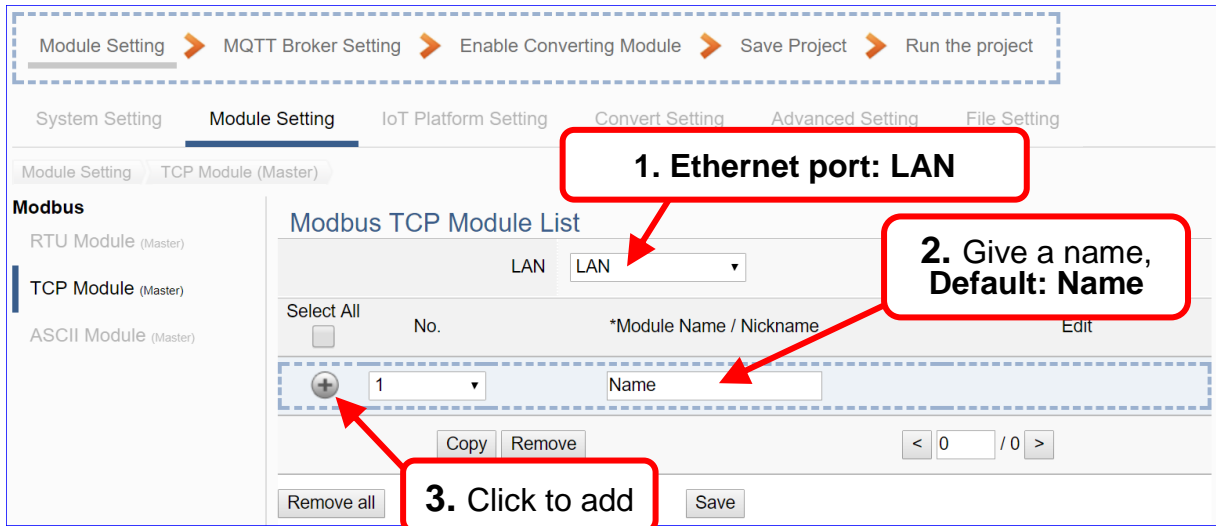
The Step Box of the [**Modbus TCP / MQTT**] has 5 steps as below. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step by step and then can complete the project quickly and rightly.



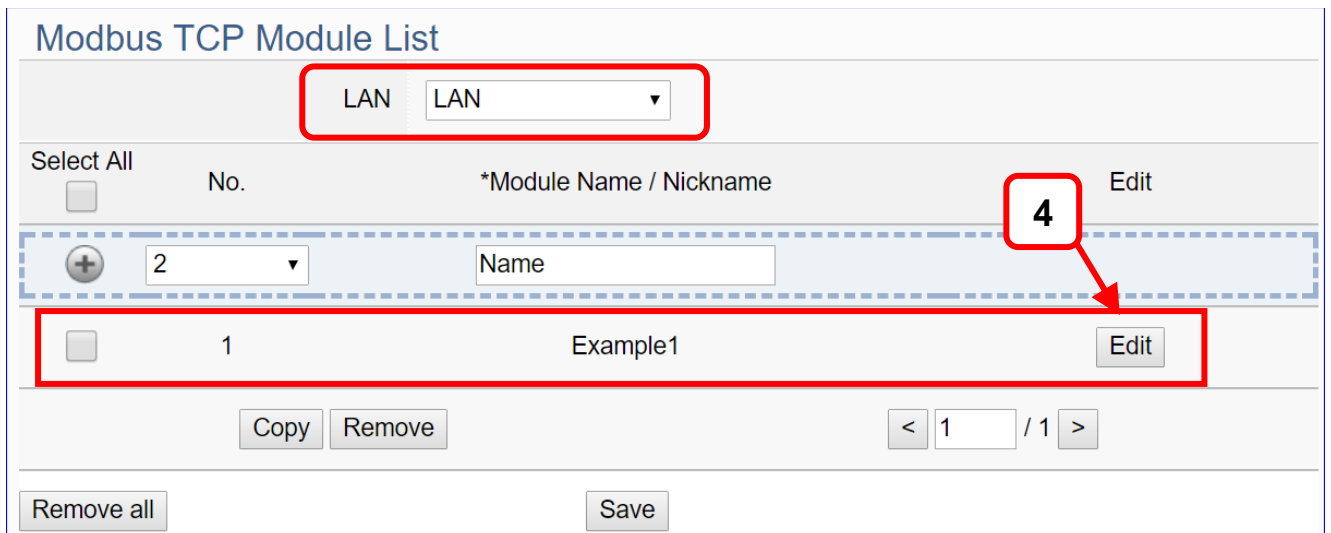
### Step 1. Module Setting

This page is for setting the communication values of the connected modules.

The Ethernet port is LAN for connecting with the TCP module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.



Add a module (No.: 1, Name: Example1) as below, and then click [Edit] button to enter the “Module Content Setting” page.



If set up a wrong module, user can click the box in the left side of the module number and click the [Remove] button to delete the module.

Click [Edit] can enter the [Module Content Setting] page to set up the module and the Modbus address mapping table.

Module Content Setting	
No.	1
Module Name	Example1
IP	0 . 0 . 0 . 0
Port	502
Slave ID	1
Timeout	500
Polling Rate	500
Modbus Mapping Table Setting	
Data Model	01 Coil Status(0x)
Start Address	0
Data Number	1
Create Tables	<input type="button" value="Add"/>

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
IP	The IP address of the connected module. Default: 0.0.0.0
Port	The port number for Modbus TCP. Default: 502
Slave ID	Set the Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Polling Rate	Set a time interval for the command. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models "01" ~ "04" for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI)
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.

The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting	Nickname Setting																																				
Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)																																				
<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>2</td></tr> <tr><td>Type</td><td>Bool</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	2	Type	Bool	<input type="button" value="Edit"/>		<table border="1"> <tr><td>Address</td><td><input type="text" value="0"/></td></tr> <tr><td>Number</td><td><input type="text" value="1"/></td></tr> <tr><td>Type</td><td>Bool</td></tr> <tr><td colspan="2"><input type="button" value="Delete"/></td></tr> <tr><td colspan="2"><input type="button" value="Save"/></td></tr> <tr><td colspan="2"><input type="button" value="Cancel"/></td></tr> </table>	Address	<input type="text" value="0"/>	Number	<input type="text" value="1"/>	Type	Bool	<input type="button" value="Delete"/>		<input type="button" value="Save"/>		<input type="button" value="Cancel"/>		<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>1</td></tr> <tr><td>Type</td><td>Short</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	1	Type	Short	<input type="button" value="Edit"/>		<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>1</td></tr> <tr><td>Type</td><td>Float</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	1	Type	Float	<input type="button" value="Edit"/>	
Address	0																																						
Number	2																																						
Type	Bool																																						
<input type="button" value="Edit"/>																																							
Address	<input type="text" value="0"/>																																						
Number	<input type="text" value="1"/>																																						
Type	Bool																																						
<input type="button" value="Delete"/>																																							
<input type="button" value="Save"/>																																							
<input type="button" value="Cancel"/>																																							
Address	0																																						
Number	1																																						
Type	Short																																						
<input type="button" value="Edit"/>																																							
Address	0																																						
Number	1																																						
Type	Float																																						
<input type="button" value="Edit"/>																																							
Press Save to finish editing.																																							
<input type="button" value="OK"/> <input type="button" value="Cancel"/>																																							

Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

Modbus Mapping Table		Address Setting	Nickname Setting	
<b>01 Coil Status(0x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
1	<input type="text" value="Tag1"/>	Bool	<input type="text"/>	
<b>02 Input Status(1x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
<b>03 Holding Registers(4x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input type="text"/>
<b>04 Input Registers(3x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input type="text"/>
		<input type="button" value="OK"/>	<input type="button" value="Cancel"/>	

Modbus Mapping Table – Nickname Setting	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

## Step 2. MQTT Broker Setting

Click the next step, and enter the **Step 2 [MQTT Broker Setting]** of the UI setting.

This page is for setting the IoT platform and the MQTT Broker connection, e.g. the local or remote broker, port, login information, etc.

We select the “Modbus RTU / MQTT” conversion at the beginning, so this step will auto enter the **[MQTT Connection > Local Broker]** page of IoT Platform Setting. The “Step Box” will prevent the user from selecting the wrong platform. User can choose the local or remote broker for the MQTT connection.

### MQTT Connection > Local Broker Setting

Port	The COM port of the Local MQTT Broker. System default: 1883
Anonymous Login	Check to allow anonymous login. Default: Check.
Save	Click to save the setting of this page.

### MQTT Connection > Remote Broker List

Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
	Click to add a new remote Broker.
Save	Click to save the settings of this page.

After creating a new Remote Broker (as below) :

MQTT Connection > Remote Broker List	
Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Edit / Remove	Click [Edit] can set the Broker. Click the left box and [remove] can delete the Broker.
Save	Click to save the settings of this item.

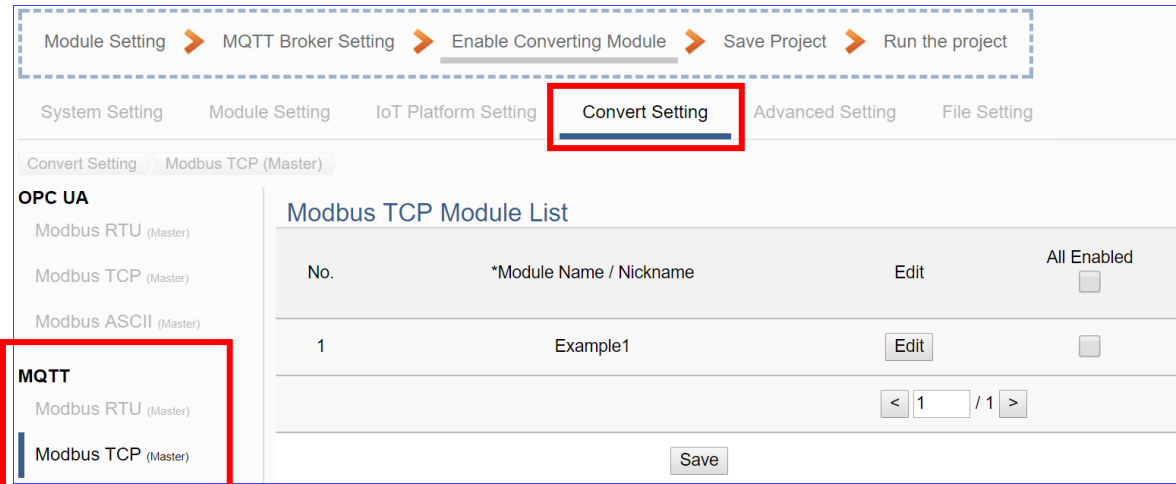
MQTT Connection > Remote Broker > Broker Content Settings	
Broker Name	The name of the remote MQTT Broker. (Editable)
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Keep Alive Time	The keep alive time. Default: 60 (second)
SSL/TLS	Check to enable the supporting of SSL/TLS security communication. Default: uncheck.
Anonymous Login	Check to allow anonymous login. Default: Check.
OK	Click to save the settings and exit.

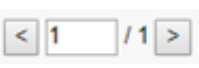


### Step 3. Enable Converting Module

Click the next step, and enter the **Step 3 [Enable Converting Module]** UI setting  
 This step is for enabling the module for the Modbus TCP / MQTT conversion.

We select the “Modbus TCP / MQTT” conversion at the beginning, so this step will auto enter the **[MQTT > Modbus TCP (Master)]** page of Conversion setting. The “Step Box” will prevent the user from selecting the wrong platform.



Convert Setting > MQTT > Modbus TCP (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	Click to enter the “MQTT Client Setting” page to set up the Topic, QoS, Publish, Subscribe ...
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “MQTT Client Setting” page:

MQTT Client Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>
Will Topic	<input type="text"/>
Will	<input type="text"/>
MQTT Connection	<input checked="" type="checkbox"/> Broker (Local) <input type="checkbox"/> Broker1 (Remote)

Convert Setting > MQTT > Modbus TCP (Master) – MQTT Client Setting	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
Scan Rate(ms)	Set an update frequency for the task data. Default: 1000 (Unit: ms)
Dead Bend	Give a dead bend value for updating a float signal. Default: 0
Will Topic	Enter the title of a disconnect notice. Default: Null.
Will	Enter a disconnect notice. Default: Null.
MQTT Connection	Check the Broker want to use Local Broker or Remote Broker.

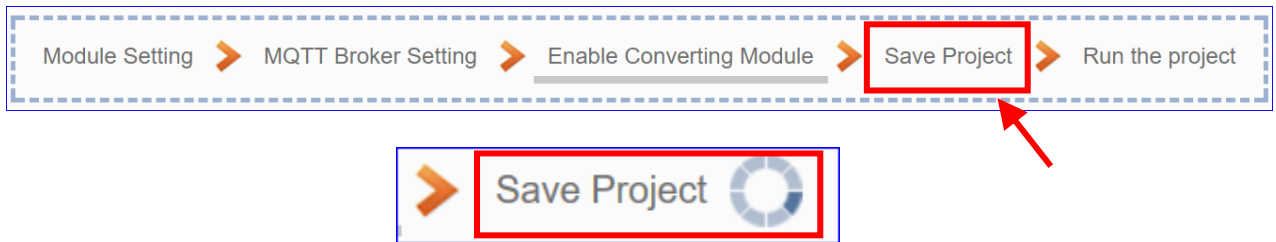
Publish & Subscribe									
Details <input type="button" value="Show"/> <input type="button" value="Hide"/>									
Name	Attribute	Data Type	Subscribe Topic	Subscribe QoS	Publish Topic	Publish QoS	Retain	Enabled	
Tag0	Read	Short	/MTCP_No.1_Example1/Inpu t_Registers/Tag0/Subscribe	2		2	<input type="checkbox"/>	<input type="checkbox"/>	
Tag0	Read / Write	Short	/MTCP_No.1_Example1/Hol ding_Registers/Tag0/Subscri be	2	/MTCP_No.1_Example1/Hol ding_Registers/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>	
Tag0	Read	Bool	/MTCP_No.1_Example1/Inpu t_Status/Tag0/Subscribe	2		2	<input type="checkbox"/>	<input type="checkbox"/>	
Tag0	Read / Write	Bool	/MTCP_No.1_Example1/Coil _Status/Tag0/Subscribe	2	/MTCP_No.1_Example1/Coil _Status/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>	
Tag1	Read / Write	Bool	/MTCP_No.1_Example1/Coil _Status/Tag1/Subscribe	2	/MTCP_No.1_Example1/Coil _Status/Tag1/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="button" value="OK"/> <input type="button" value="Cancel"/>									

Convert Setting > MQTT > Modbus TCP (Master) – Publish & Subscribe	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Name	The variable name of the mapping address. (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Subscribe Topic	The topic of receiving/subscribing data message.
Subscribe Qos	The subscribe Qos (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Publish Topic	The topic of sending/publishing data message.
Publish Qos	The publish Qos (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Retain	Check [Retain] box of the top row can store the broker message for all variables in list. Check the box of each variable can store the broker message just that variable. Default: Uncheck.
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

### Step 4. Save Project

The setting of this example is finished now, and then to save the whole project and run the project. So the last two steps will not show setting pages, but show some displays.

Click the next step **[Save Project]**, the Step Box will show an animation as below picture, that means the project is saving. When the animation vanished, the project is saved completely.

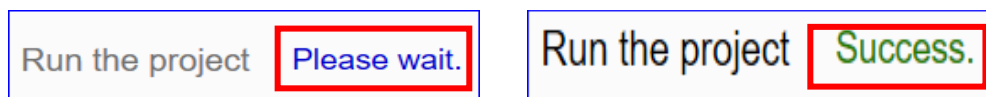


### Step 5. Run the Project

The project, after saving, needs to be executed. Click the next step **[Run the Project]**.



The Step Box will show the words **“Please wait”** (as below), that means the system is deleting the old project in the UA controller, and will upload the new project into the UA series and run the new project. When the words **“Please wait”** disappears, the new words **“Success”** appears (as below), that means the UA controller is running new project successfully.



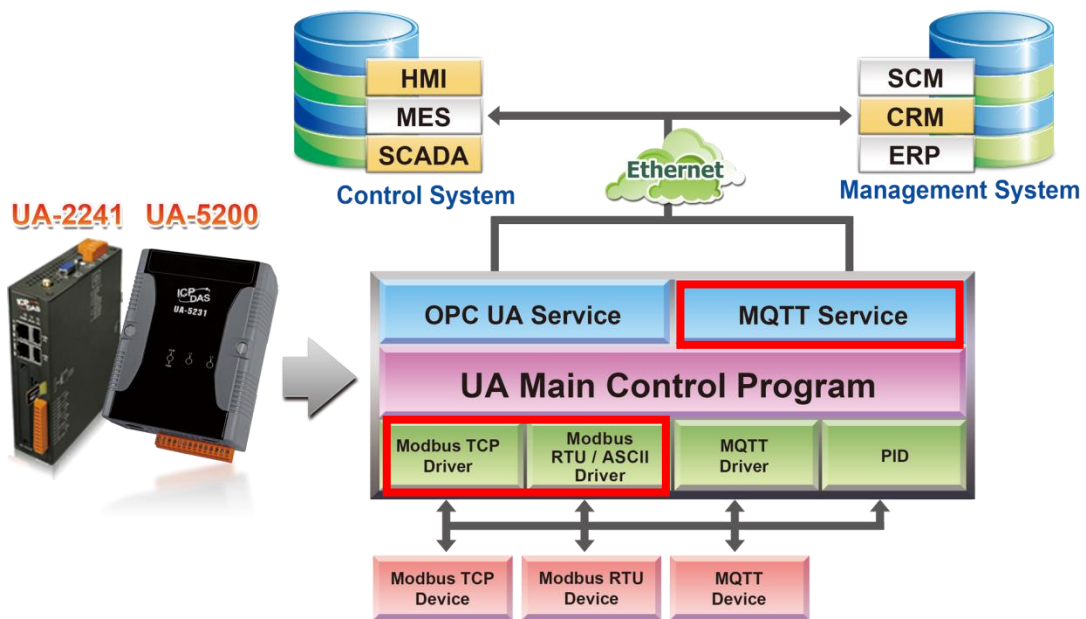
And then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication.

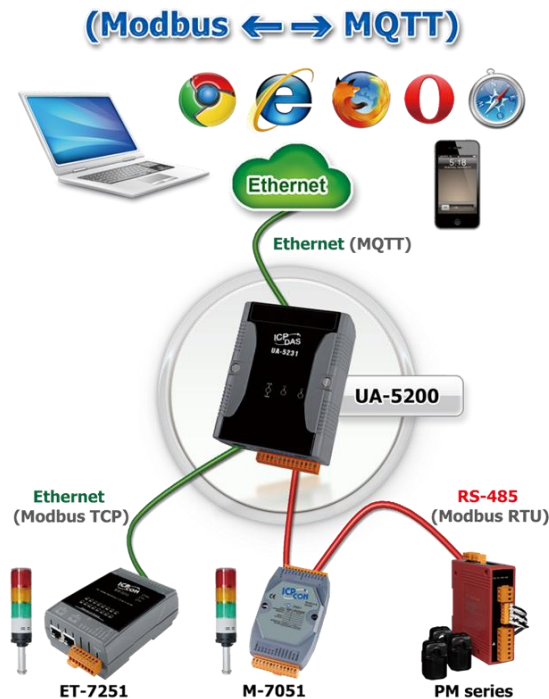
### 4.1.3. Modbus / MQTT JSON Conversion

Modbus / MQTT JSON Conversion include the conversion of MQTT and Modbus RTU / TCP / ASCII three protocols. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and combine several messages that converted in JSON format into a group to read and write the multiple channels of the Modbus RTU devices that connected to the controller.

#### Modbus / MQTT JSON Function Diagram:

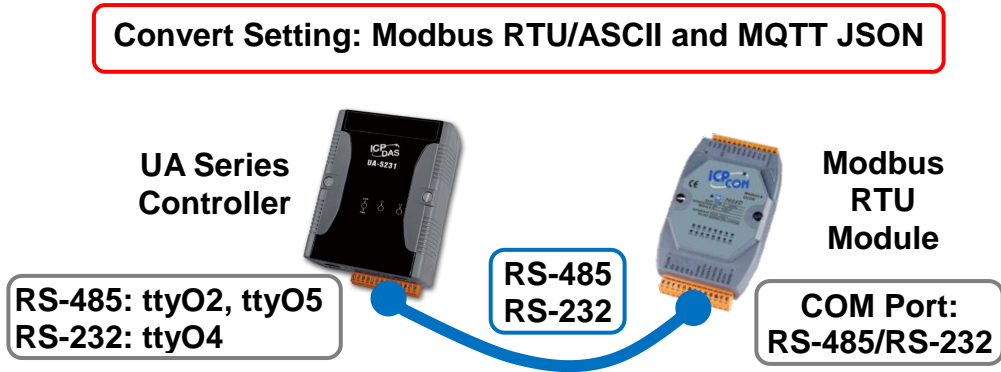


#### Application Solution:



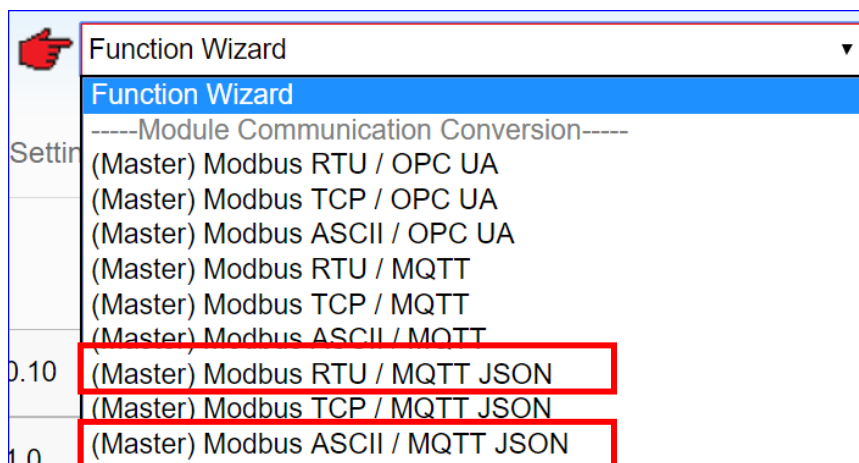
The settings of Modbus RTU/ASCII are the same. Here will introduce them together as a setting sample for Modbus / MQTT JSON conversion.

● **Modbus RTU / ASCII 與 MQTT JSON 轉換傳輸:**



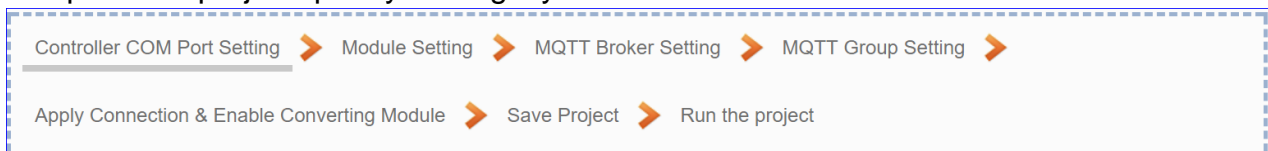
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus RTU or ASCII module (via RS-485 / RS-232, as the picture) and read/write the Modbus I/O via MQTT Broker, user can choose the item [**Modbus RTU / MQTT JSON**] or [**Modbus ASCII / MQTT JSON**] of the “Module Communication Conversion” in the Function Wizard.



**[Step Box]:**

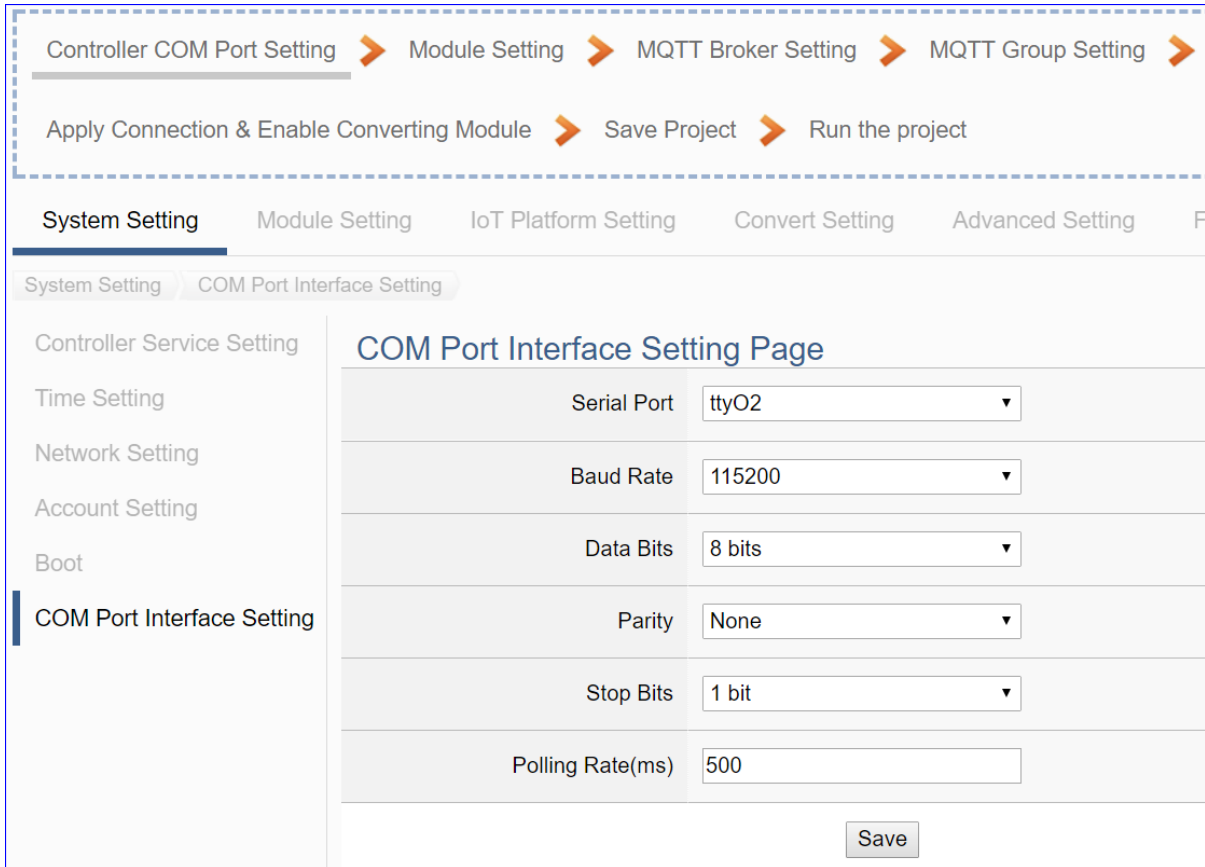
The Step Box of the [**Modbus RTU / MQTT JSON**] and [**Modbus ASCII / MQTT JSON**] has the same 7 steps, here will introduce them together. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step by step and then can complete the project quickly and rightly.



### Step 1. Controller COM Port Setting

This page allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication.

The user can find the the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).

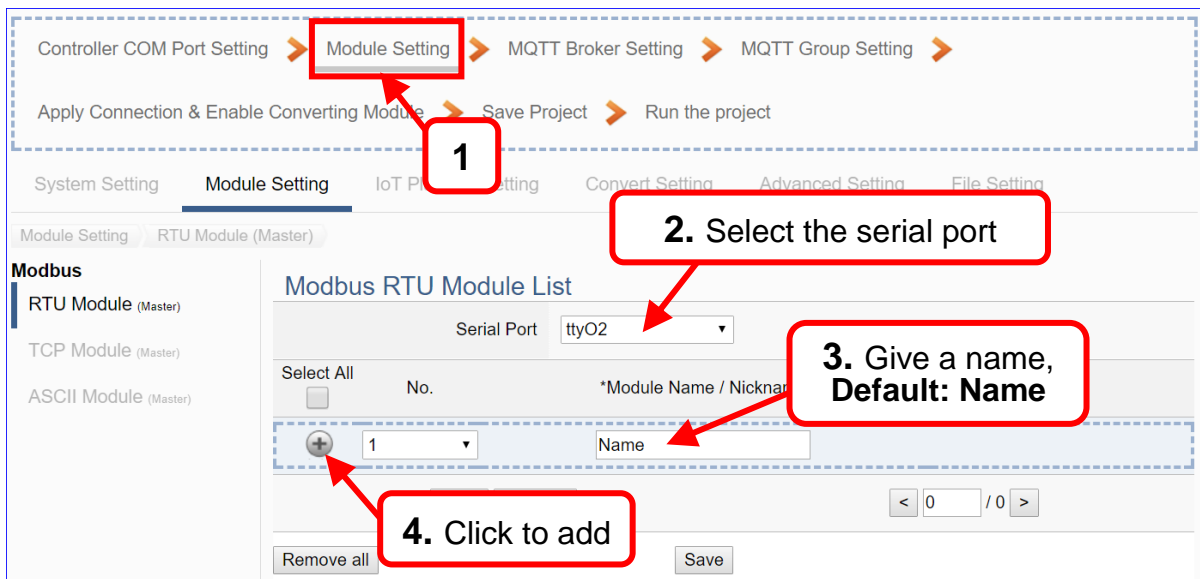


COM Port Interface Setting Page	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485 ; ttyO4: RS-232 ; ttyO5: RS-485
Baud Rate	Choose a baud rate to communicate with the module: 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. The UA controller and the I/O module need have the same baud rate.
Data Bits	The number of bits used to represent one byte of data: 7 bits or 8 bits. Default: 8 Bits.
Parity	Choose one way for the parity checking. Options: None, Even, and Odd. Default: None.
Stop Bits	Choose the number of stop bit: 1 bit or 2 bits. Default: 1.
Polling Rate(ms)	Set a time interval for the command. Default: 500 ms
Save	Click [Save] button could save the settings of this page.

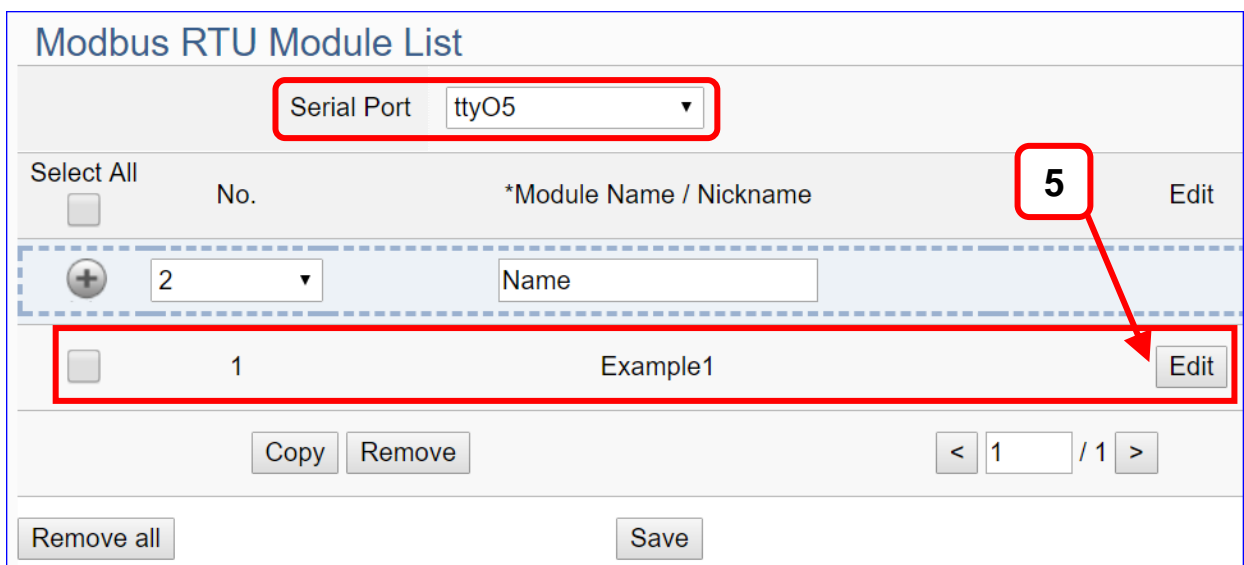
## Step 2. Module Setting

Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First choose the serial port that connected with the module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.



Add a module (No.: 1, Name: Example1) as below, and then click [Edit] button to enter the “Module Content Setting” page.



If set up a wrong module, user can click the box in the left side of the module number and click the [Remove] button to delete the module.



[**Module Content Setting**] page can set up the module and the Modbus address mapping table:

Module Content Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Name1"/>
Slave ID	<input type="text" value="1"/>
Timeout	<input type="text" value="500"/>
Modbus Mapping Table Setting	
Data Model	<input type="text" value="03 Holding Registers(4x)"/>
Start Address	<input type="text" value="0"/>
Data Number	<input type="text" value="1"/>
Type	<input type="text" value="16-bit Short"/>
Create Tables	<input type="button" value="Add"/>

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
Slave ID	Set the module Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models "01" ~ "04" for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI) <div style="border: 1px solid black; padding: 2px; display: inline-block;">                     01 Coil Status(0x)                      02 Input Status(1x)                      03 Holding Registers(4x)                      04 Input Registers(3x)                 </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.

The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting		Nickname Setting			
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)		Input Registers(3x)	
Address	0	Address	<input type="text" value="0"/>	Address	0	Address	0
Number	2	Number	<input type="text" value="1"/>	Number	1	Number	1
Type	Bool	Type	Bool	Type	Short	Type	Float
<input type="button" value="Edit"/>		<input type="button" value="Delete"/> <input type="button" value="Save"/>		<input type="button" value="Edit"/>		<input type="button" value="Edit"/>	
		<input type="button" value="Cancel"/>					
Press Save to finish editing.							
				<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	

Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

Modbus Mapping Table
Address Setting
Nickname Setting

**01 Coil Status(0x)**

Table Display
Show
Hide

Address	Variable name	Data Type	Description
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>
1	<input type="text" value="Tag1"/>	Bool	<input type="text"/>

**02 Input Status(1x)**

Table Display
Show
Hide

Address	Variable name	Data Type	Description
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>

**03 Holding Registers(4x)**

Table Display
Show
Hide

Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input type="text"/>

**04 Input Registers(3x)**

Table Display
Show
Hide

Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input type="text"/>

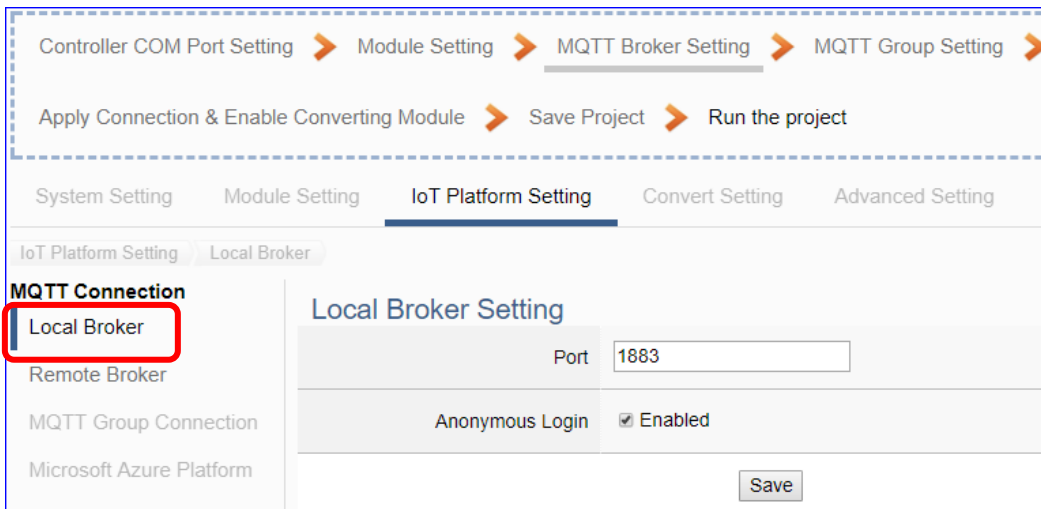
OK
Cancel

<b>Modbus Mapping Table – Nickname Setting</b>	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

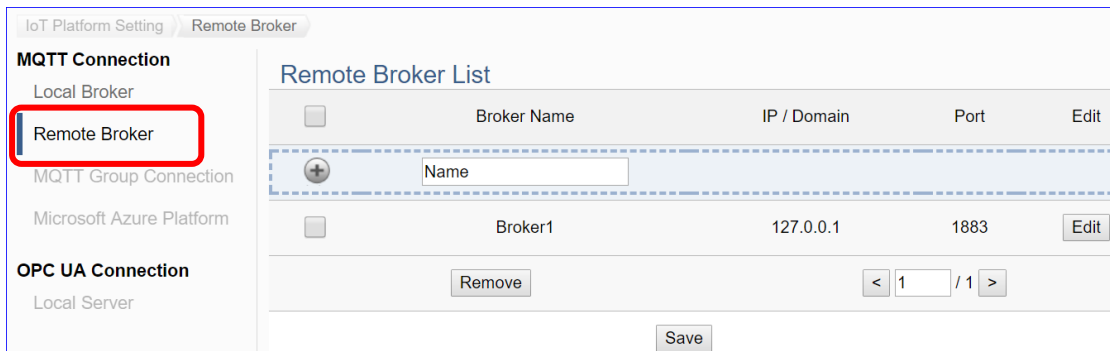
### Step 3. MQTT Broker Setting

Click the next step, and enter the **Step 3 [MQTT Broker Setting]** of the UI setting. This page is for setting the IoT platform and the MQTT Broker connection, e.g. the local or remote broker, port, login information, etc.

We select the “Modbus RTU (or ASCII) / MQTT JSON” conversion at the beginning, so this step will auto enter the **[MQTT Connection > Local Broker]** page of IoT Platform Setting. The “Step Box” will prevent the user from selecting the wrong platform. User can choose the local or remote broker for the MQTT connection.



MQTT Connection > Local Broker Setting	
Port	The COM port of the Local MQTT Broker. System default: 1883
Anonymous Login	Check to allow anonymous login. Default: Check.
Save	Click to save the setting of this page.



MQTT Connection > Remote Broker List	
Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
	Click to add a new remote Broker.
Save	Click to save the settings of this page.

After creating a new Remote Broker (as below) :

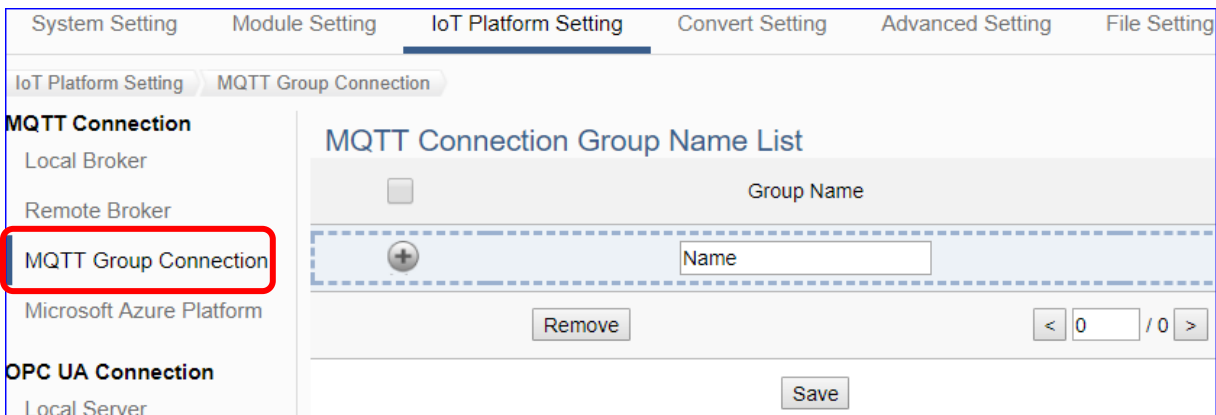
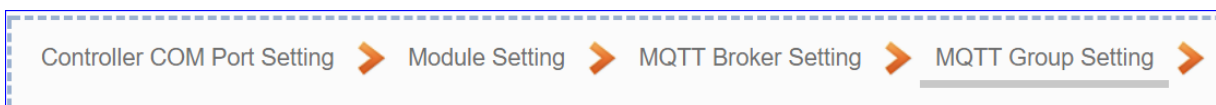
MQTT Connection > Remote Broker List	
Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Edit / Remove	Click [Edit] can set the Broker. Click the left box and [remove] can delete the Broker.
Save	Click to save the settings of this item.

MQTT Connection > Remote Broker > Broker Content Settings	
Broker Name	The name of the remote MQTT Broker. (Editable)
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Keep Alive Time	The keep alive time. Default: 60 (second)
SSL/TLS	Check to enable the supporting of SSL/TLS security communication. Default: uncheck.
Anonymous Login	Check to allow anonymous login. Default: Check.
OK	Click to save the settings and exit.

### Step 4. MQTT Group Setting

Click the next step, and enter the **Step 4 [MQTT Group Setting]** of the UI setting. This page is for setting the MQTT Group connection, Setting with the MQTT JSON function in the Convert Transmission, It can make the I/O module messages in groups and then mapping to the user-defined publish and subscribe topics.

We select the “Modbus RTU (or ASCII) / MQTT JSON” conversion at the beginning, so this step will auto enter the **[MQTT Connection > MQTT Group Connection]** page of IoT Platform Setting. The “Step Box” will prevent the user from selecting the wrong platform.



MQTT Connection > MQTT Group Connection > MQTT Connection Group Name List	
Group Name	MQTT group name, user can define, e.g. Group1. Default: Name.
	Click to add a new MQTT Group.
	The page number of the group list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the setting of this page.

After creating a new group (as below):

**MQTT Connection Group Name List**

<input type="checkbox"/>	Group Name	Edit
<input style="border: 1px dashed blue;" type="button" value="+"/>	Name <input type="text"/>	
<input type="checkbox"/>	Name	Edit

1 / 1

Click [Edit] button to enter the [MQTT Client Setting] page:

**MQTT Client Setting**

No.	<input type="text" value="1"/>
Group Name	<input type="text" value="Name"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>
Will Topic	<input type="text"/>
Will	<input type="text"/>
MQTT Connection	<input checked="" type="checkbox"/> Broker (Local) <input type="checkbox"/> Broker1 (Remote)

IoT Platform Setting > MQTT Group Connection > MQTT Client Setting	
No.	The group number in the MQTT Client list (Not editable here)
Group Name	Give a name, e.g. Group1. Default: Name.
Scan Rate(ms)	Set an update frequency for the data. Default: 1000 (Unit: ms)
Dead Bend	Give a dead bend value for updating a float signal. Default: 0
Will Topic	Enter the title of a disconnect notice. Default: Null.
Will	Enter a disconnect notice. Default: Null.
MQTT Connection	Check the Broker want to use Local Broker or Remote Broker.

Publish & Subscribe	
Publish Topic	<input type="text" value="/Name/Publish"/>
Publish QoS	<input type="text" value="2"/>
Subscribe Topic	<input type="text" value="/Name/Subscribe"/>
Subscribe QoS	<input type="text" value="2"/>
Retain	<input type="text" value="No"/>
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

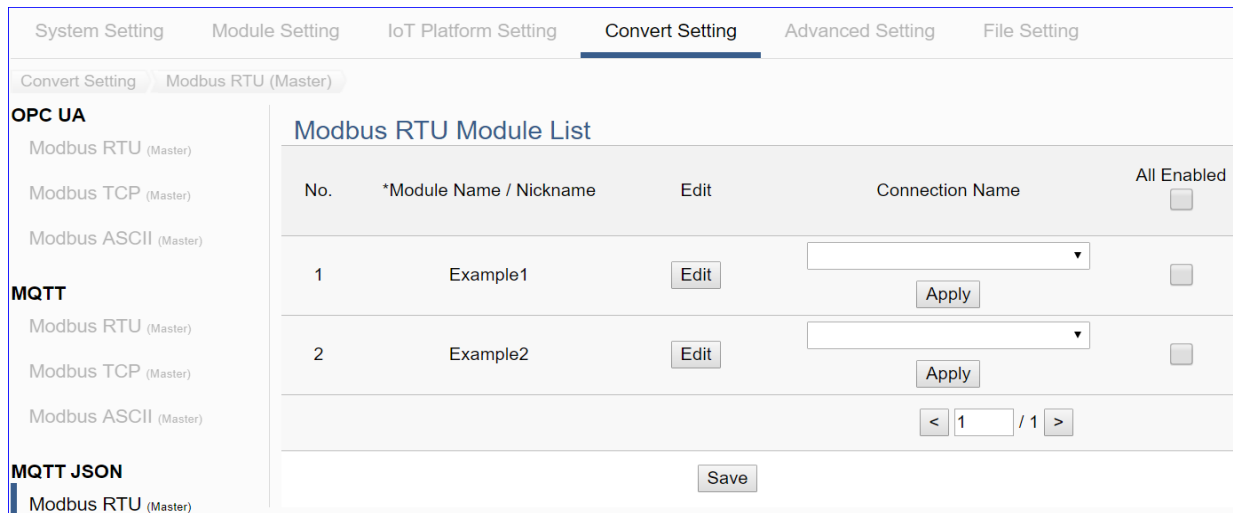
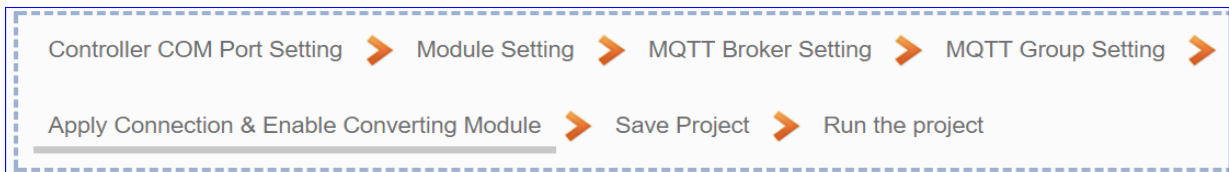
IoT Platform Setting > MQTT Group Connection > MQTT Client Setting – Publish & Subscribe	
Publish Topic	The topic of sending/publishing data message.
Publish Qos	The publish Qos (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Subscribe Topic	The topic of receiving/subscribing data message.
Subscribe Qos	The subscribe Qos (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Retain	Whether to store a broker message. Default: No
OK	Click to save the settings and exit.




## Step 5. Apply Connection & Enable Converting Module

Click the next step, and enter the **Step 5 [Apply Connection & Enable Converting Module]** UI setting. This page is for applying the connection and enabling the converting module.

We select the “Modbus RTU (or ASCII) / MQTT JSON” conversion at the beginning, so this step will auto enter the [**Convert Setting > MQTT JSON - Modbus RTU (or ASCII) (Master)**] page of Convet setting. The “Step Box” will prevent the user from selecting the wrong platform.



Convert Setting > MQTT JSON > Modbus RTU (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name	The module name set in the module list (Not editable here)
Connection Name	Select a group connection name, and then click [Apply].
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	If user wants to enable some I/O channels for conversion, click [Edit] of that module to enter the “Variable Tale” setting.
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “Module Content Setting” page:

### Module Content Setting

No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>

### Variable Table

Details	<input type="button" value="Show"/>	<input type="button" value="Hide"/>
---------	-------------------------------------	-------------------------------------

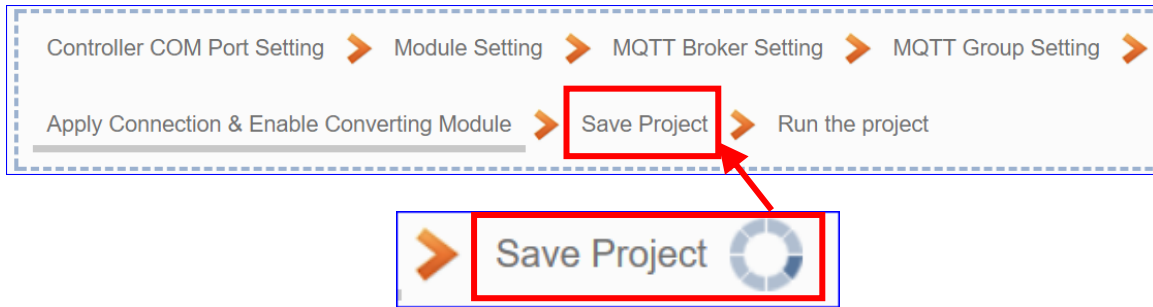
Variable Name	Alias	Attribute	Data Type	Connection Name	Enabled
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read"/>	Float	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read / Write"/>	Short	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag1"/>	<input type="text" value="Tag1"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag1"/>	<input type="text" value="Tag1"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>	<input type="checkbox"/>

<b>Convert Setting &gt; MQTT JSON &gt; Modbus RTU (Master) – Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Convert Setting &gt; MQTT JSON &gt; Modbus RTU (Master) – Variable Table</b>	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Variable Name	The variable name of the mapping address. (Not editable here)
Alias	The the alias name for the variable. (Editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Connection Name	Select the group name that set in the group list page.
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

### Step 6. Save Project

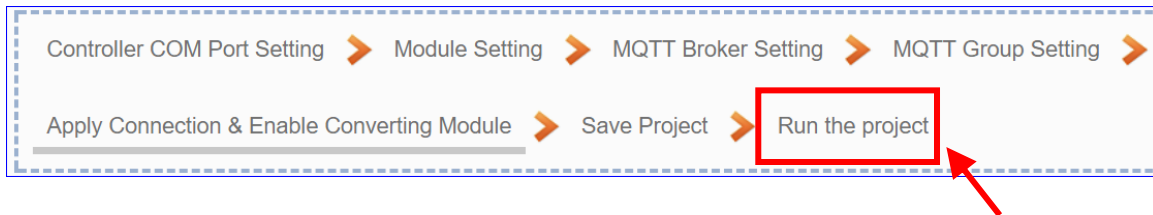
The setting of this example is finished now, and then to save the whole project and run the project. So the last two steps will not show setting pages, but show some displays.

Click the next step [**Save Project**], the Step Box will show an animation as below picture, that means the project is saving. When the animation vanished, the project is saved completely.

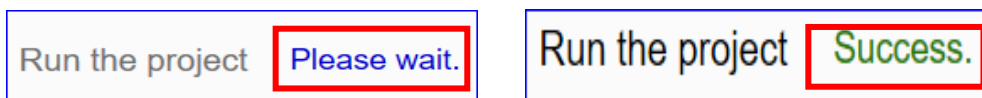


### Step 7. Run the Project

The project, after saving, needs to be executed. Click the next step [**Run the Project**].



The Step Box will show the words “**Please wait**” (as below), that means the system is deleting the old project in the UA controller, and will upload the new project into the UA series and run the new project. When the words “**Please wait**” disappears, the new words “**Success**” appears (as below), that means the UA controller is running new project successfully.



And then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

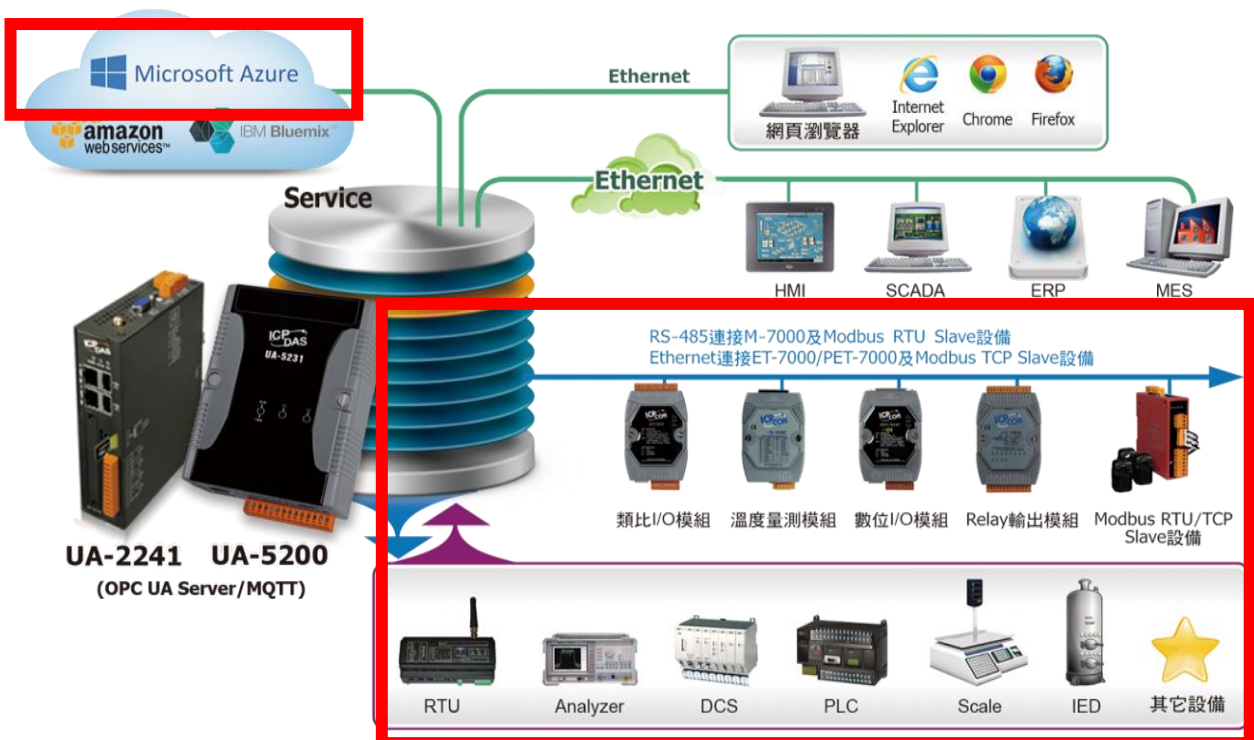
The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication.

## 4.2. Module Connecting to Azure

"Module Connecting to Azure" is a common way to integrate IoT devices into the cloud. Many of the applications use MQTT connection to the cloud for the setting is fast and easy. The UA series also provides the MQTT function for module to connect to the Azure platform and allows users to publish messages to Microsoft Azure and receive messages from Microsoft Azure. This section will introduce the setting steps and the function parameters of the "Module Connecting to Azure". There are 3 items in this category for 3 protocol types. Here will introduce the Modbus TCP / Azure for this category.

-----Module Connecting to Azure-----  
 (Master) Modbus RTU / Azure  
 (Master) Modbus TCP / Azure  
 (Master) Modbus ASCII / Azure

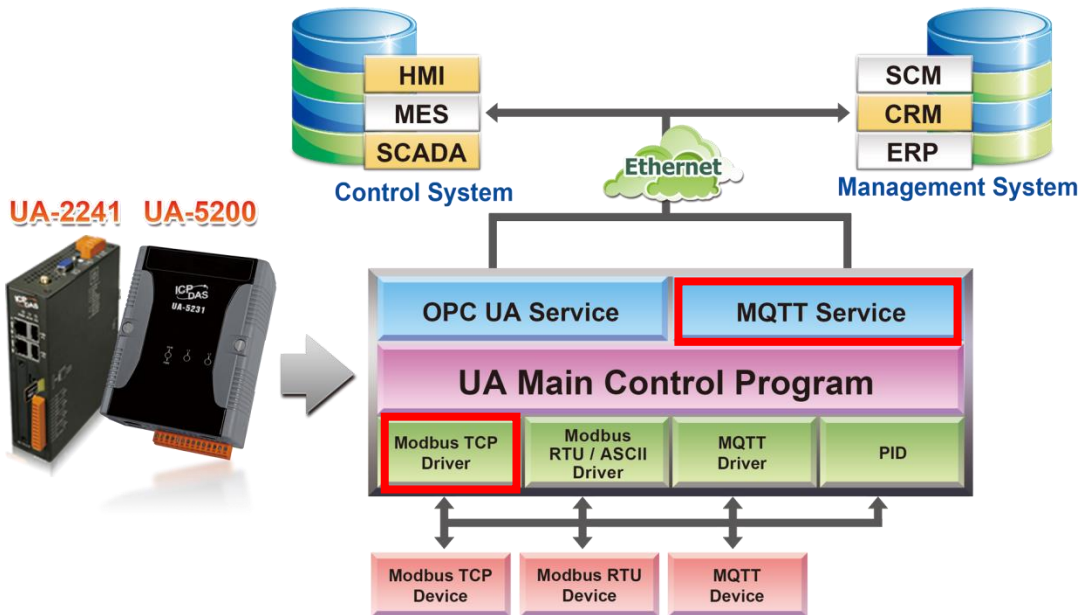
<b>Modbus RTU / Azure</b>	Allow the Modbus RTU connecting to the Microsoft Azure platform and can publish messages to Microsoft Azure and receive messages from Microsoft Azure.
<b>Modbus TCP / Azure</b>	Allow the Modbus RTU connecting to the Microsoft Azure platform and can publish messages to Microsoft Azure and receive messages from Microsoft Azure. <b>(Section 4.2.1)</b>
<b>Modbus ASCII / Azure</b>	Allow the Modbus RTU connecting to the Microsoft Azure platform and can publish messages to Microsoft Azure and receive messages from Microsoft Azure.



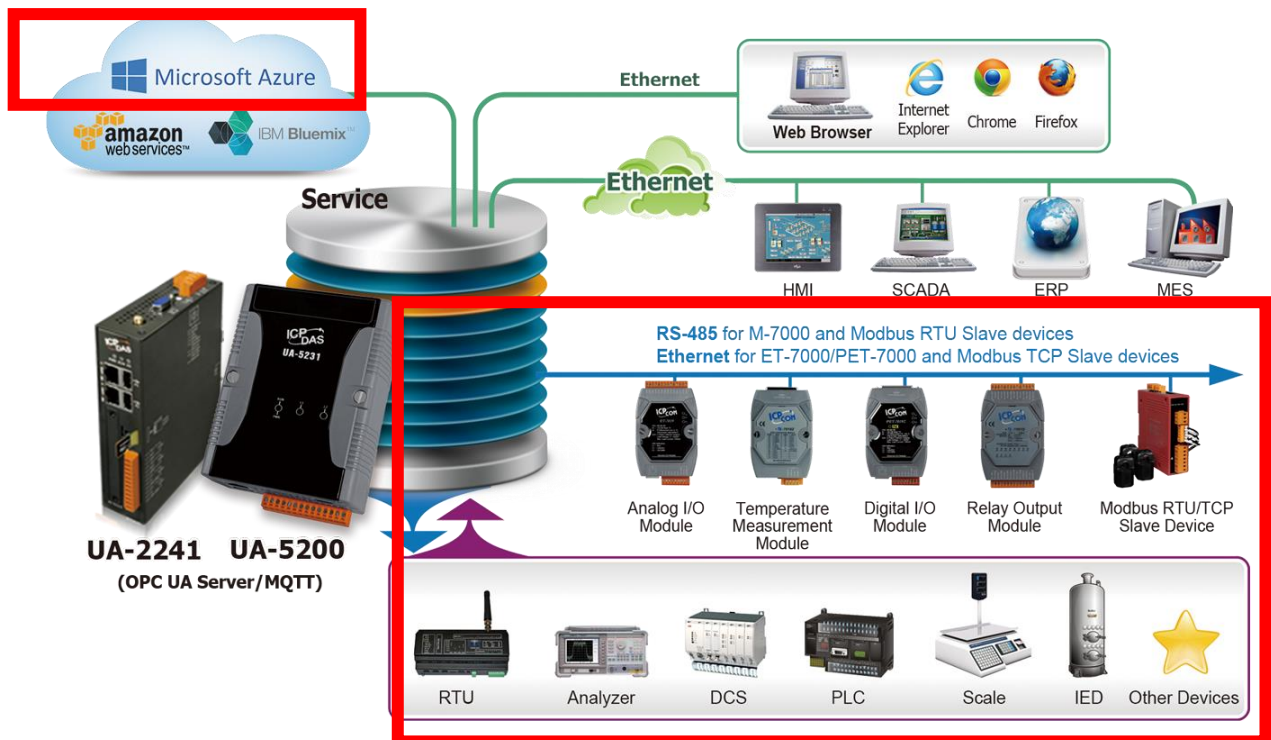
### 4.2.1. Modbus TCP / Azure Connecting

The UA series provides the MQTT function for module to connect to the Microsoft Azure platform and allows users to publish messages to Azure and receive messages from Azure. This section will introduce the setting steps and the function parameters. There are 3 items about Azure function in the “Function Wizard”. Here will introduce the Modbus TCP / Azure.

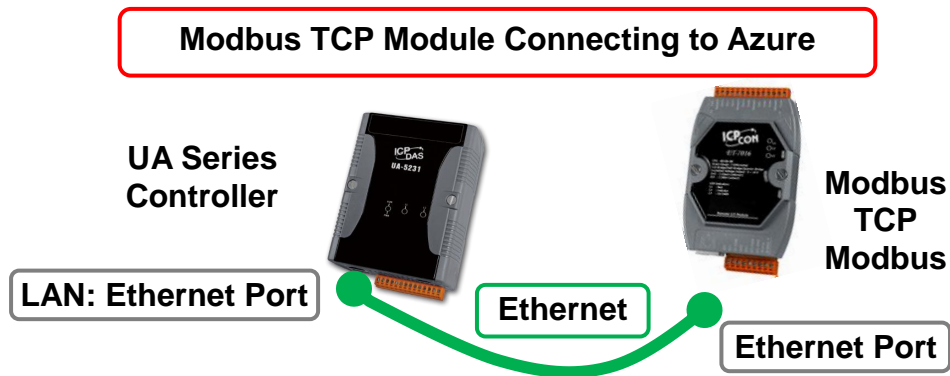
#### Function Diagram for Modbus TCP / Azure:



#### Application Solution:

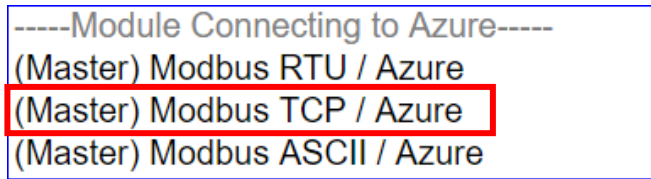


● **Modbus TCP Module Connecting to Azure**



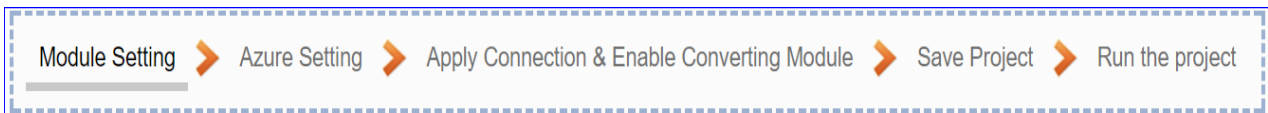
**Note:** The hardware/network connection methods please see the [Chapter 2](#) .

When UA series controller connects the Modbus TCP (via Ethernet, as the picture), read/write the Modbus I/O via MQTT Broker and transfer the data to the Microsoft Azure platform, user can choose the item **[Modbus TCP / Azure]** of the “Module Connecting to Azure” in the Function Wizard.

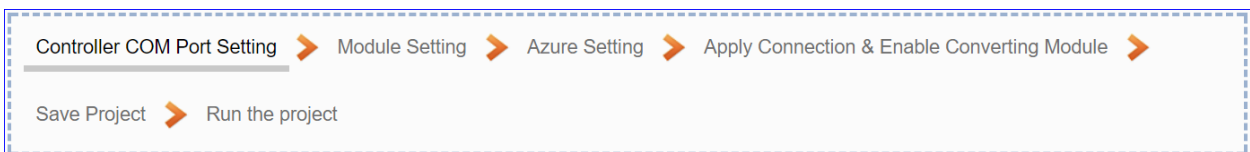


**[Step Box]:**

The Step Box of the **[Modbus TCP / Azure]** has 5 steps as below. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step by step and then can complete the project quickly and rightly.



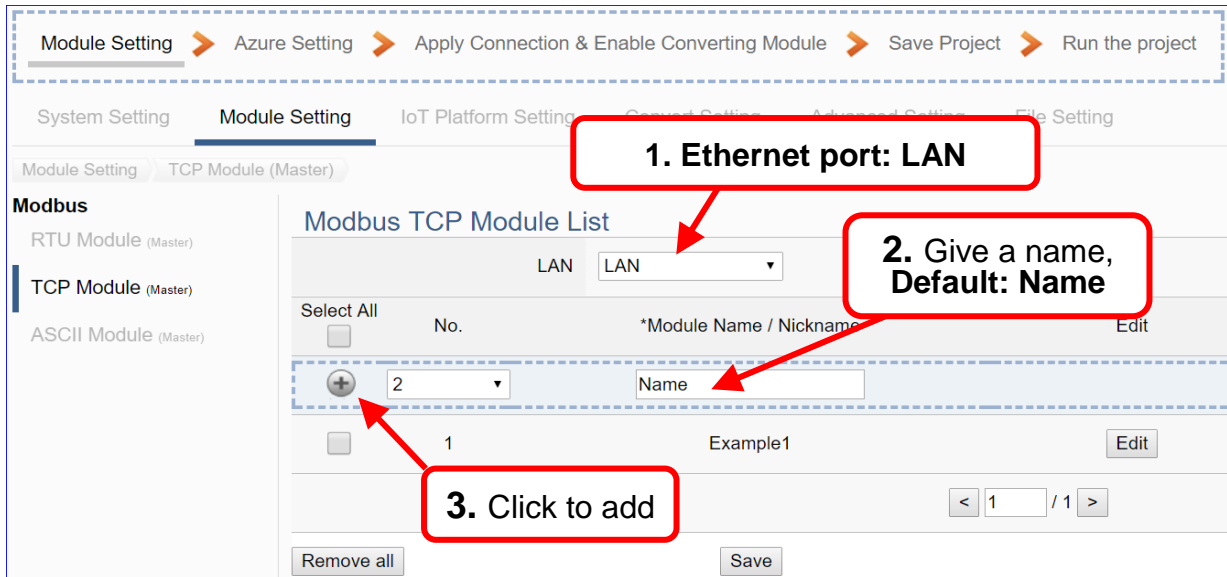
In addition, the Step Box of [Modbus RTU / Azure] or [Modbus ASCII / Azure] has 6 steps. The different step is “Controller COM Port Setting” that can refer to Section 4.1.1 or 4.1.3.



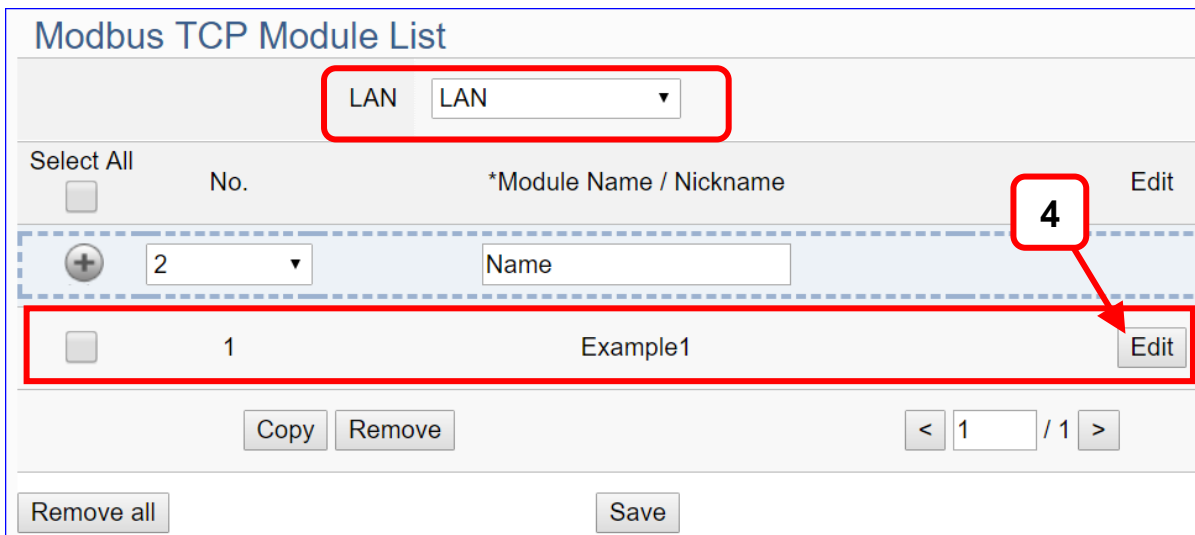
### Step 1. Module Setting

This page is for setting the communication values of the connected modules.

The Ethernet port is LAN for connecting with the TCP module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.



Add a module (No.: 1, Name: Example1) as below, and then click [Edit] button to enter the “Module Content Setting” page.



If set up a wrong module, user can click the box in the left side of the module number and click the [Remove] button to delete the module.

Click [Edit] can enter the [Module Content Setting] page to set up the module and the Modbus address mapping table.

Module Content Setting	
No.	1
Module Name	Example1
IP	0 . 0 . 0 . 0
Port	502
Slave ID	1
Timeout	500
Polling Rate	500
Modbus Mapping Table Setting	
Data Model	01 Coil Status(0x)
Start Address	0
Data Number	1
Create Tables	Add

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
IP	The IP address of the connected module. Default: 0.0.0.0
Port	The port number for Modbus TCP. Default: 502
Slave ID	Set the Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Polling Rate	Set a time interval for the command. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models "01" ~ "04" for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI)
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.



The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting		Nickname Setting	
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)	
Input Registers(3x)					
Address	0	Address	<input type="text" value="0"/>	Address	0
Number	2	Number	<input type="text" value="1"/>	Number	1
Type	Bool	Type	Bool	Type	Short
<input type="button" value="Edit"/>		<input type="button" value="Delete"/> <input type="button" value="Save"/>		<input type="button" value="Edit"/>	
		<input type="button" value="Cancel"/>			
Press Save to finish editing.					
		<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	

Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

Modbus Mapping Table
Address Setting
Nickname Setting

**01 Coil Status(0x)**

Table Display

Address	Variable name	Data Type	Description
0	<input type="text" value="Tag0"/>	Bool	<input style="width: 100%;" type="text"/>
1	<input type="text" value="Tag1"/>	Bool	<input style="width: 100%;" type="text"/>

**02 Input Status(1x)**

Table Display

Address	Variable name	Data Type	Description
0	<input type="text" value="Tag0"/>	Bool	<input style="width: 100%;" type="text"/>

**03 Holding Registers(4x)**

Table Display

Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input style="width: 100%;" type="text"/>

**04 Input Registers(3x)**

Table Display

Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input style="width: 100%;" type="text"/>

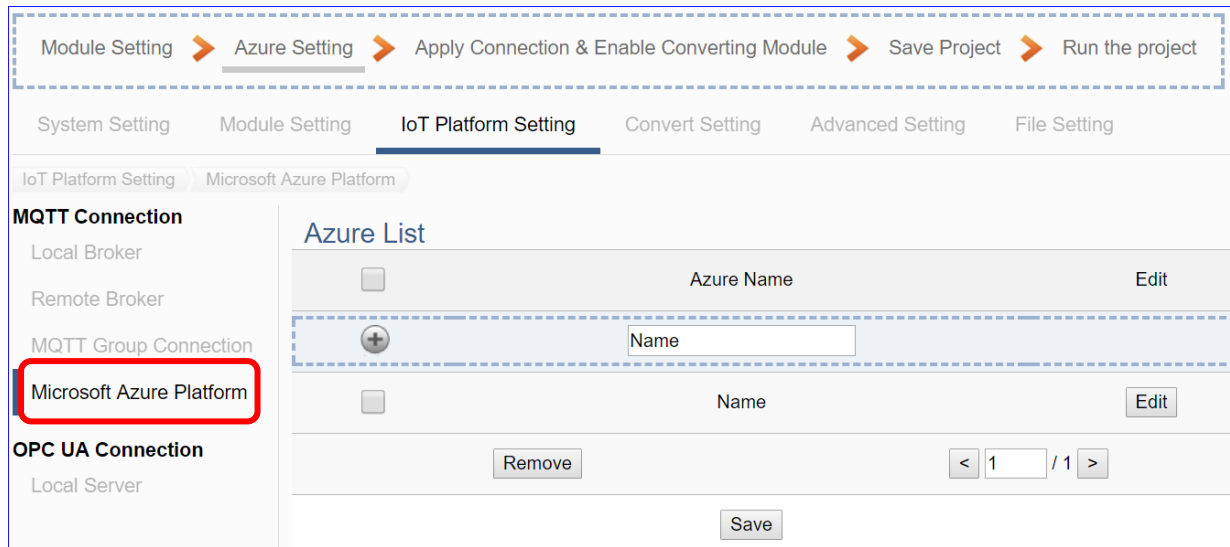
<b>Modbus Mapping Table – Nickname Setting</b>	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

## Step 2. Azure Setting

Click the next step, and enter the **Step 2 [Azure Setting]** of the UI setting.

This page is for setting the Microsoft Azure Platform related information of the MQTT Connection in the IoT platform, e.g. the name, SAS Token, etc.

We select the “Modbus TCP / Asure” connecting item at the beginning, so this step will auto enter the **[MQTT Connection > Microsoft Azure Platform]** page of IoT Platform Setting. The “Step Box” will prevent the user from selecting the wrong platform.



MQTT Connection > Microsoft Azure Platform > Azure List	
Azure Name	Azure name. User can define the name. Default: Name.
	Click to add a new Azure list.
Edit / Remove	Click [Edit] can set the Azure list. Click the left box and [remove] can delete the Azure list.
	The page number of the Azure list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “**Azure Content Settings**” page:

### Azure Content Settings

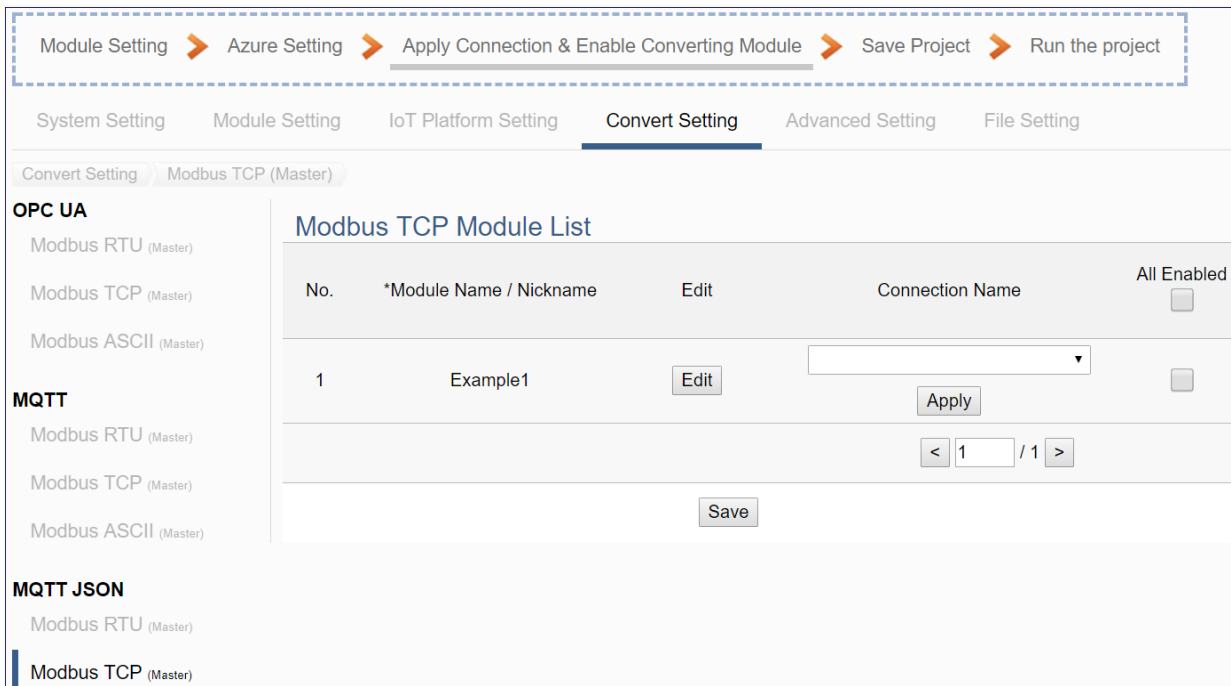
Azure Name	<input type="text" value="Name"/>
SAS Token	<div style="border: 1px solid gray; padding: 2px; min-height: 100px;">                     HostName=;DeviceId=;SharedAccessSignature=                 </div>
Keep Alive Time(second)	<input type="text" value="60"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>
CDS	<input type="checkbox"/> Enabled


<b>MQTT Connection &gt; Microsoft Azure Platform &gt; Azure List &gt; Azure Content Settings</b>									
Azure Name	Azure name. User can define the name. Default: Name.								
SAS Token	Input the SAS Token which you previously registered for the UA controller from Microsoft Azure. For the procedure to generate a SAS Token, please refer to the "Documentation > Azure IoT Hub > IoT Hub MQTT support" section on the Microsoft Azure Web Site for detailed information.								
Keep Alive Time(second)	Set the time in second that pass away without communication between the UA controller and Microsoft Azure. Default: 60 second.								
Scan Rate(ms)	Set an update frequency for the task data. Default: 1000 (Unit: ms)								
Dead Band	Give a dead bend value for updating a float signal. Default: 0								
CDS (Connected Device Studio)	If user wants to publish the messages compliant with the Microsoft CDS platform, user must check the "CDS" to Enabled and fill in the Company ID, Equipment ID and Message ID that applied from the Microsoft CDS platform.								
	<div style="border: 1px solid gray; padding: 5px; margin: 5px 0;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; padding: 2px;">CDS</td> <td style="padding: 2px;"><input checked="" type="checkbox"/> Enabled</td> </tr> <tr> <td style="padding: 2px;">Company ID</td> <td style="padding: 2px;"><input type="text" value="0"/></td> </tr> <tr> <td style="padding: 2px;">Equipment ID</td> <td style="padding: 2px;"><input type="text"/> <span style="color: red; font-size: small;">Please enter english and numbers.</span></td> </tr> <tr> <td style="padding: 2px;">Message ID</td> <td style="padding: 2px;"><input type="text"/></td> </tr> </table> </div>	CDS	<input checked="" type="checkbox"/> Enabled	Company ID	<input type="text" value="0"/>	Equipment ID	<input type="text"/> <span style="color: red; font-size: small;">Please enter english and numbers.</span>	Message ID	<input type="text"/>
CDS	<input checked="" type="checkbox"/> Enabled								
Company ID	<input type="text" value="0"/>								
Equipment ID	<input type="text"/> <span style="color: red; font-size: small;">Please enter english and numbers.</span>								
Message ID	<input type="text"/>								
OK	Click to save and exit this page.								

### Step 3. Apply Connection & Enable Converting Module

Click the next step, and enter the **Step 3 [Apply Connection & Enable Converting Module]** UI setting. This page is for applying the connection and enabling the converting module.

We select the “Modbus TCP / Azure” at the beginning, and UA system connecting to Azure throughs MQTT JSON group method, so this step will auto enter the **[Convert Setting > MQTT JSON - Modbus TCP (Master)]** page of Convet setting. The “Step Box” will prevent the user from selecting the wrong platform.



Convert Setting > MQTT JSON > Modbus TCP (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
Edit	If user wants to enable some I/O channels for conversion, click [Edit] of that module to enter the “Variable Tale” setting.
Connection Name	Select an Azure connectionn name, and then click [Apply].
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “Module Content Setting” page:

### Module Content Setting

No.	<input style="width: 90%;" type="text" value="1"/>
Module Name	<input style="width: 90%;" type="text" value="Example1"/>

### Variable Table

Details	<input type="button" value="Show"/>	<input type="button" value="Hide"/>
---------	-------------------------------------	-------------------------------------

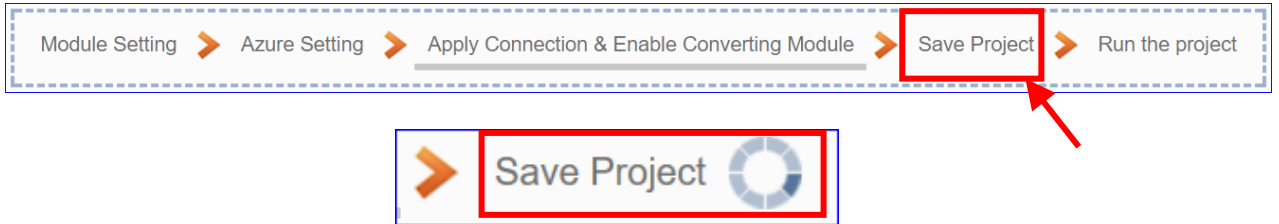
Variable Name	Alias	Attribute	Data Type	Connection Name	Enabled
<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Read"/>	Short	<input style="width: 90%;" type="text"/>	<input type="checkbox"/>
<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Read / Write"/>	Short	<input style="width: 90%;" type="text"/>	<input type="checkbox"/>
<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Read"/>	Bool	<input style="width: 90%;" type="text"/>	<input type="checkbox"/>
<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Tag0"/>	<input style="width: 90%;" type="text" value="Read / Write"/>	Bool	<input style="width: 90%;" type="text"/>	<input type="checkbox"/>
<input style="width: 90%;" type="text" value="Tag1"/>	<input style="width: 90%;" type="text" value="Tag1"/>	<input style="width: 90%;" type="text" value="Read / Write"/>	Bool	<input style="width: 90%;" type="text"/>	<input type="checkbox"/>

<b>Convert Setting &gt; MQTT JSON &gt; Modbus TCP (Master) Module List –Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Convert Setting &gt; MQTT JSON &gt; Modbus TCP (Master) Module List – Variable Table</b>	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Variable Name	The variable name of the mapping address. (Not editable here)
Alias	The the alias name for the variable. (Editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Connection Name	Select the Azure connection name that set in the [Azure Setting] step.
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

### Step 4. Save Project

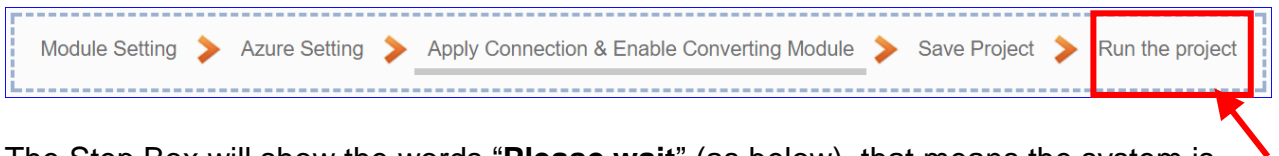
The setting of this example is finished now, and then to save the whole project and run the project. So the last two steps will not show setting pages, but show some displays.

Click the next step [**Save Project**], the Step Box will show an animation as below picture, that means the project is saving. When the animation vanished, the project is saved completely.

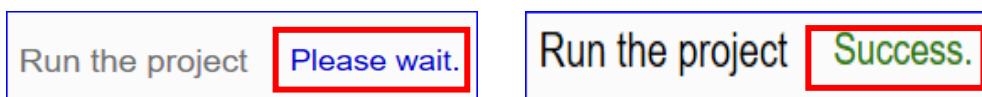


### Step 5. Run the Project

The project, after saving, needs to be executed. Click the next step [**Run the Project**].



The Step Box will show the words “**Please wait**” (as below), that means the system is deleting the old project in the UA controller, and will upload the new project into the UA series and run the new project. When the words “**Please wait**” disappears, the new words “**Success**” appears (as below), that means the UA controller is running new project successfully.



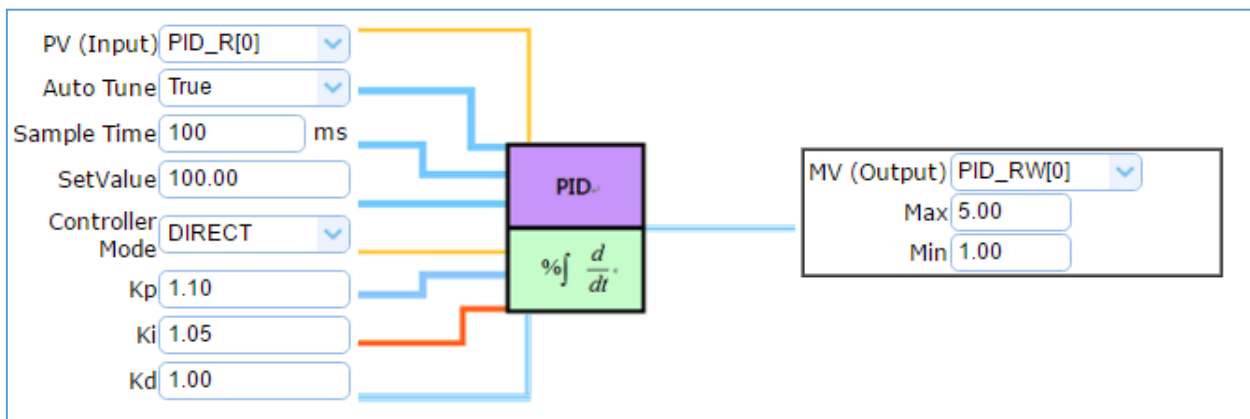
And then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

The new project now completes the setting, uploading and running in the UA controller and can process the new function project.

### 4.3. PID

PID (Proportional-Integral-Derivative) control is the most widely used in industrial control systems. A regulator which is controlled in accordance with Proportional, Integral and Derivative is called PID control for short, also called PID regulator. When the user cannot fully grasp or measure parameters of the control system, the PID regulator is the best solution.

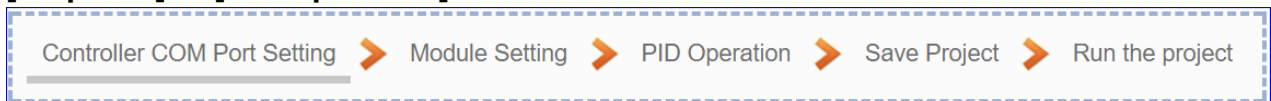
The PID controller is a common feedback loop component in industrial control applications. The controller compares the collected data with a reference value and then uses this difference to calculate a new input value whose purpose is to allow the system data to reach or remain at the reference value.



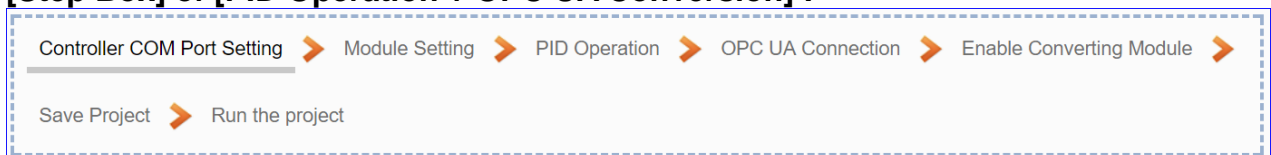
This chapter introduces the setting steps and the function parameters of the PID. There are 2 items about “PID” function in the “Function Wizard”. The 2<sup>nd</sup> item [PID Operation + OPC UA Communication Conversion] is combining the 1<sup>st</sup> item [PID Operation] and the [Section 4.1.1 Modbus / OPC UA Conversion](#) .



**[Step Box] of [PID Operation] :**



**[Step Box] of [PID Operation + OPC UA Conversion] :**

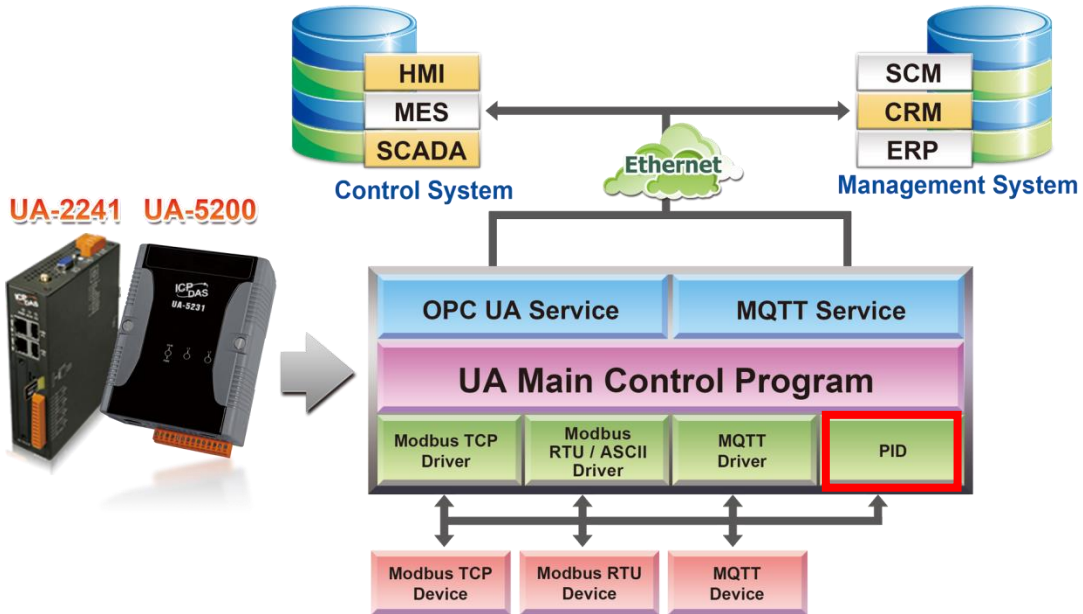




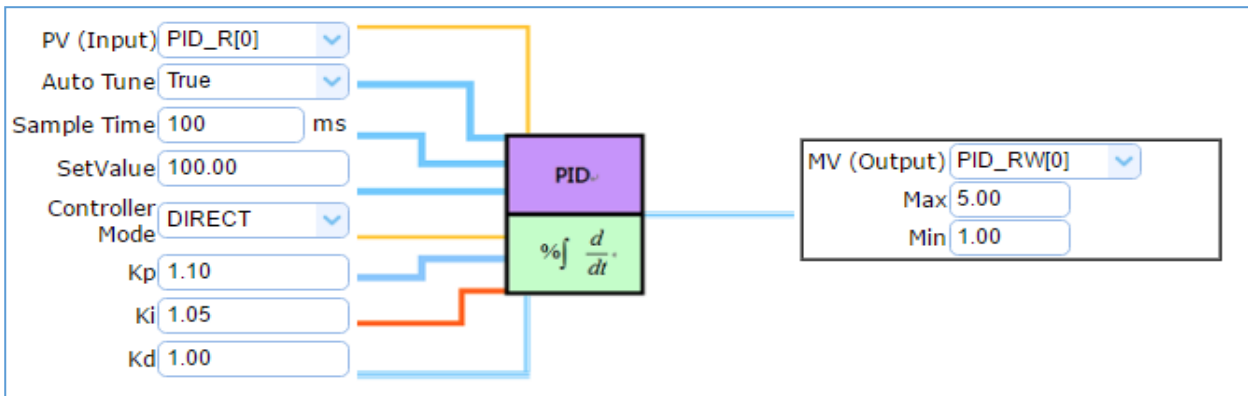
### 4.3.1. PID Operation

In the PID Operation function, UA controller collects the module's data to operate via the feedback loop component of PID control. The controller compares the collected data with a reference value and then uses this difference to calculate a new input value whose purpose is to allow the system data to reach or remain at the reference value. This section will introduce the setting steps and the function parameters of the [PID Operation].

#### Function Diagram for PID Operation:

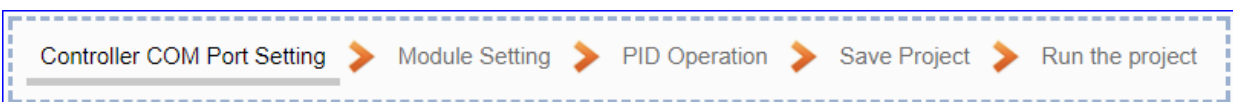


#### Application Solution Example:



#### [Step Box]:

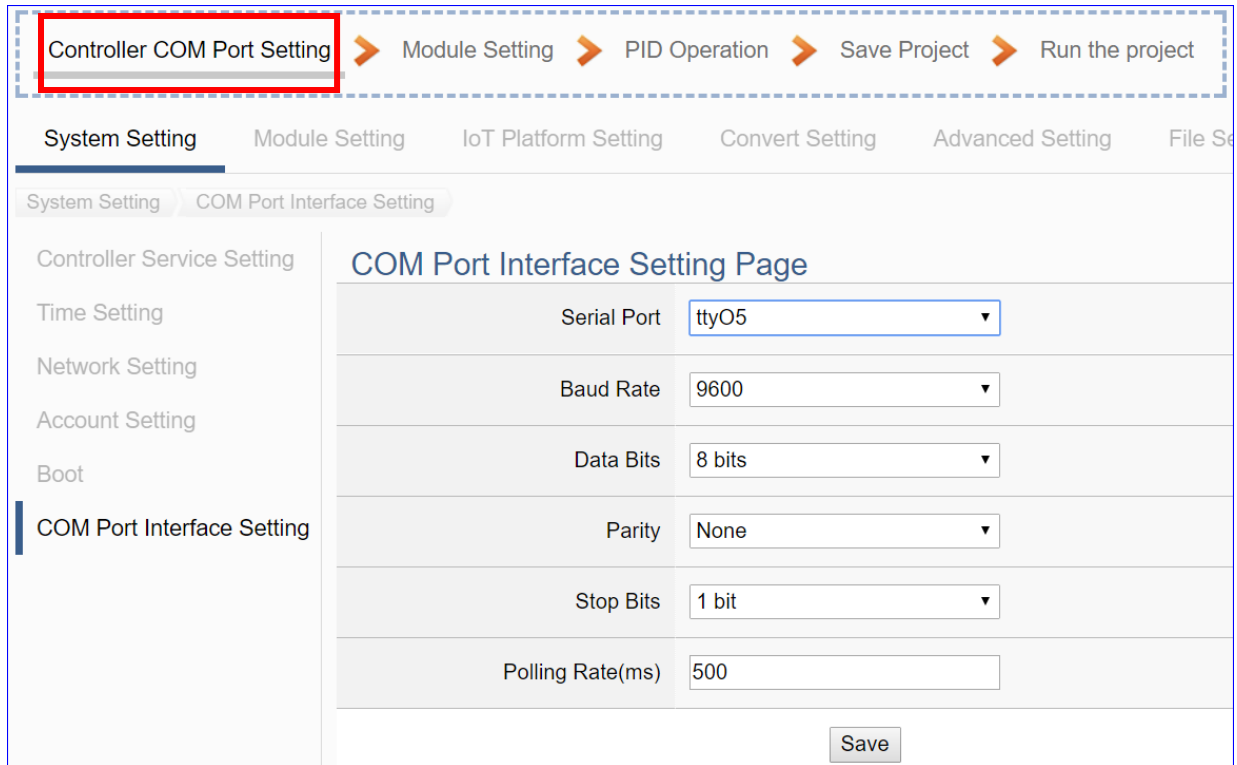
The Step Box of the [PID Operation] has 5 steps as below. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step by step and then can complete the project.



### Step 1. Controller COM Port Setting

This page allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication.

The user can find the the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).

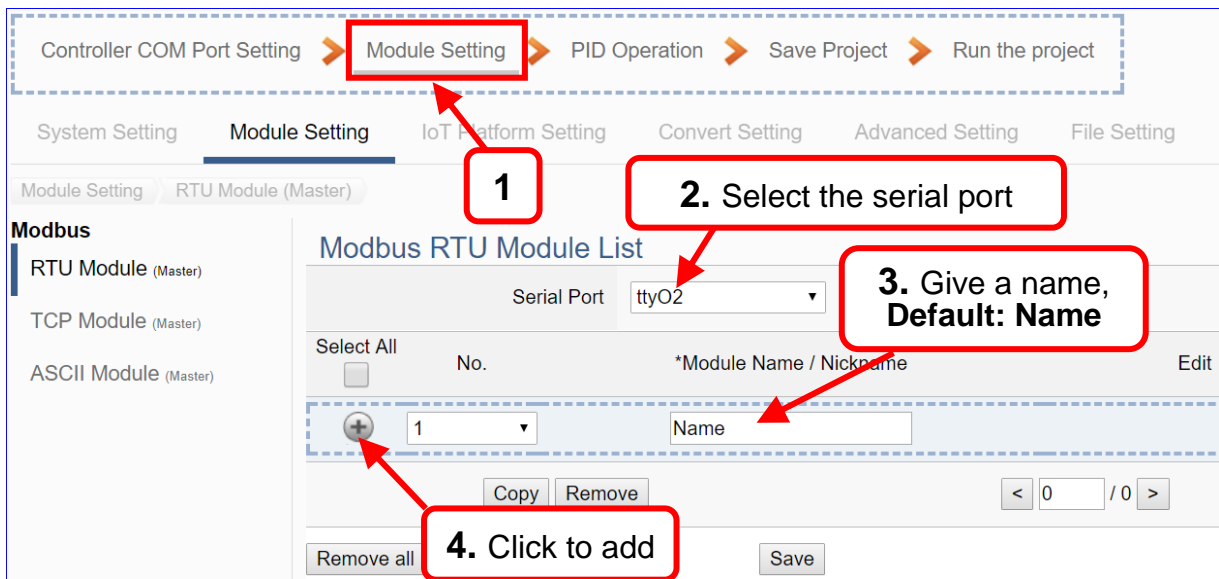


COM Port Interface Setting Page	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485 ; ttyO4: RS-232 ; ttyO5: RS-485
Baud Rate	Choose a baud rate to communicate with the module: 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. The UA controller and the I/O module need have the same baud rate.
Data Bits	The number of bits used to represent one byte of data: 7 bits or 8 bits. Default: 8 Bits.
Parity	Choose one way for the parity checking. Options: None, Even, and Odd. Default: None.
Stop Bits	Choose the number of stop bit: 1 bit or 2 bits. Default: 1.
Polling Rate(ms)	Set a time interval for the command. Default: 500 ms
Save	Click [Save] button could save the settings of this page.

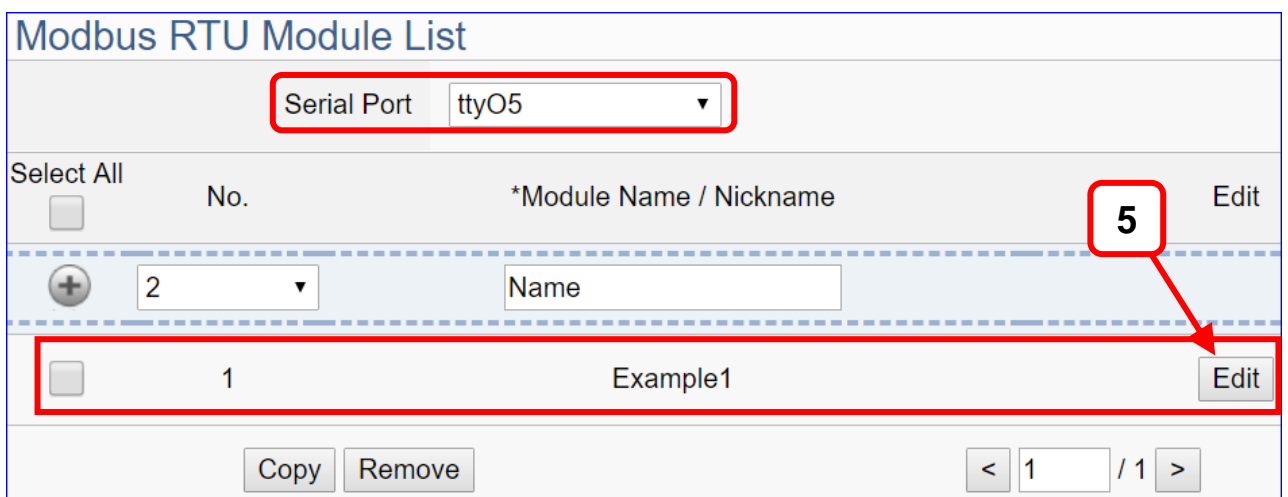
## Step 2. Module Setting

Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First choose the serial port that connected with the module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.



Add a module (No.: 1, Name: Example1) as below, and then click [Edit] button to enter the “Module Content Setting” page.



If set up a wrong module, user can click the box in the left side of the module number and click the [Remove] button to delete the module.

[**Module Content Setting**] page can set up the module and the Modbus address mapping table:

Module Content Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>
Slave ID	<input type="text" value="1"/>
Timeout	<input type="text" value="500"/>
Modbus Mapping Table Setting	
Data Model	<input type="text" value="01 Coil Status(0x)"/>
Start Address	<input type="text" value="0"/>
Data Number	<input type="text" value="1"/>
Create Tables	<input type="button" value="Add"/>

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
Slave ID	Set the module Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models "01" ~ "04" for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI) <div style="border: 1px solid black; padding: 2px; margin-top: 5px;"> <span style="background-color: #007bff; color: white; padding: 2px;">01 Coil Status(0x)</span>  <span style="padding: 2px;">02 Input Status(1x)</span>  <span style="padding: 2px;">03 Holding Registers(4x)</span>  <span style="padding: 2px;">04 Input Registers(3x)</span> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.

The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting	Nickname Setting																																				
Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)																																				
<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>2</td></tr> <tr><td>Type</td><td>Bool</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	2	Type	Bool	<input type="button" value="Edit"/>		<table border="1"> <tr><td>Address</td><td><input type="text" value="0"/></td></tr> <tr><td>Number</td><td><input type="text" value="1"/></td></tr> <tr><td>Type</td><td>Bool</td></tr> <tr><td colspan="2"><input type="button" value="Delete"/></td></tr> <tr><td colspan="2"><input type="button" value="Save"/></td></tr> <tr><td colspan="2"><input type="button" value="Cancel"/></td></tr> </table>	Address	<input type="text" value="0"/>	Number	<input type="text" value="1"/>	Type	Bool	<input type="button" value="Delete"/>		<input type="button" value="Save"/>		<input type="button" value="Cancel"/>		<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>1</td></tr> <tr><td>Type</td><td>Short</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	1	Type	Short	<input type="button" value="Edit"/>		<table border="1"> <tr><td>Address</td><td>0</td></tr> <tr><td>Number</td><td>1</td></tr> <tr><td>Type</td><td>Float</td></tr> <tr><td colspan="2"><input type="button" value="Edit"/></td></tr> </table>	Address	0	Number	1	Type	Float	<input type="button" value="Edit"/>	
Address	0																																						
Number	2																																						
Type	Bool																																						
<input type="button" value="Edit"/>																																							
Address	<input type="text" value="0"/>																																						
Number	<input type="text" value="1"/>																																						
Type	Bool																																						
<input type="button" value="Delete"/>																																							
<input type="button" value="Save"/>																																							
<input type="button" value="Cancel"/>																																							
Address	0																																						
Number	1																																						
Type	Short																																						
<input type="button" value="Edit"/>																																							
Address	0																																						
Number	1																																						
Type	Float																																						
<input type="button" value="Edit"/>																																							
Press Save to finish editing.																																							
<input type="button" value="OK"/> <input type="button" value="Cancel"/>																																							

Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

Modbus Mapping Table
Address Setting
Nickname Setting

**01 Coil Status(0x)**

Table Display

Address	Variable name	Data Type	Description
0	<input style="width: 80%;" type="text" value="Tag0"/>	Bool	<input style="width: 95%;" type="text"/>
1	<input style="width: 80%;" type="text" value="Tag1"/>	Bool	<input style="width: 95%;" type="text"/>

**02 Input Status(1x)**

Table Display

Address	Variable name	Data Type	Description
0	<input style="width: 80%;" type="text" value="Tag0"/>	Bool	<input style="width: 95%;" type="text"/>

**03 Holding Registers(4x)**

Table Display

Address	Variable name	Data Type	Swap	Description
0	<input style="width: 80%;" type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input style="width: 95%;" type="text"/>

**04 Input Registers(3x)**

Table Display

Address	Variable name	Data Type	Swap	Description
0	<input style="width: 80%;" type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input style="width: 95%;" type="text"/>

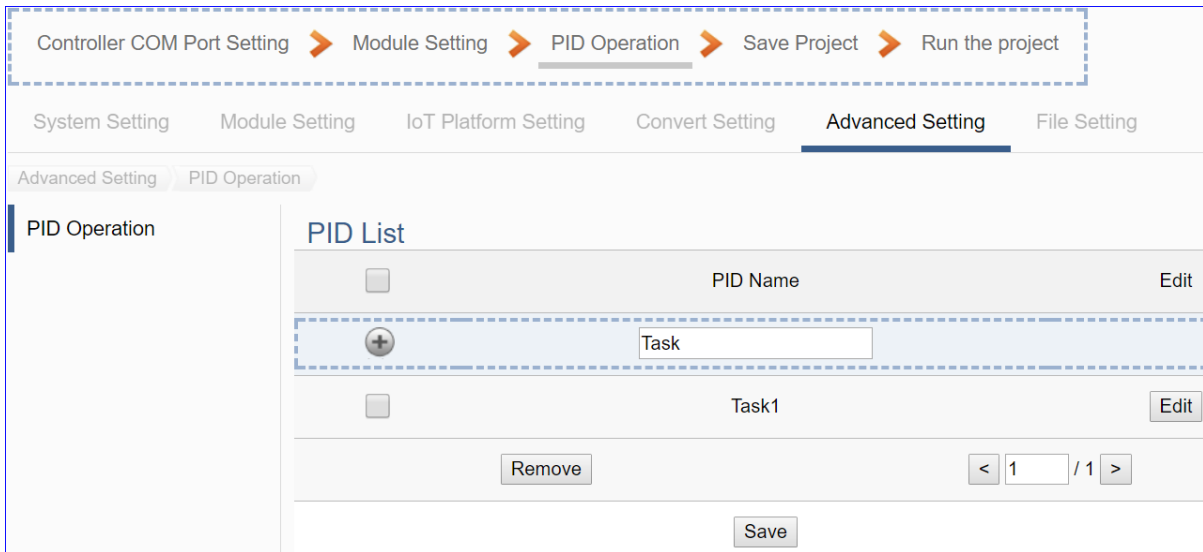
<b>Modbus Mapping Table – Nickname Setting</b>	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

### Step 3. PID Operation

Click the next step, and enter the **Step 3 [PID Operation]** of the UI setting.

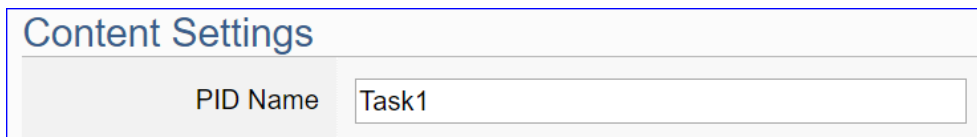
This page is for setting the Task and related parameters of the PID Operation, e.g. I/O module, I/O channels, variables, setpoint, control mode ....

We select the “**PID Operation**” at the beginning, so this step will auto enter the setting page [**Advanced Setting > PID Operation**]. The “Step Box” will prevent the user from selecting the wrong platform.



Advanced Setting > PID Operation > PID List	
PID Name	PID name, user can define, e.g. Task1. Default: Task.
	Click to add a new PID Task.
Edit / Remove	Click [Edit] can set the PID content. Click the left box and [remove] can delete the PID list.
	The page number of the PID list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the setting of this page.

Click [Edit] button to enter the [Content Settings] page:



Advanced Setting > PID Operation > Content Settings	
PID Name	PID name, user can define, e.g. Task1. Default: Task.

Input Item	
Module selection	Type : <input type="text"/> <span style="color: red;">Please select the module type.</span>
	No. : <input type="text"/> <span style="color: red;">Please select the number. When no option is available, add a module.</span>
	Name : <input type="text"/>
Variable selection	Attribute <input type="text"/> <span style="color: red;">Please select item.</span>
	Type : <input type="text"/> <span style="color: red;">Please select item.</span>
	Name : <input type="text"/> <span style="color: red;">Please select name. When there is no option, add the variables in the module.</span>
Auto Tune	<input checked="" type="checkbox"/> Enabled
Sample Time(ms)	<input type="text" value="500"/>
Setpoint	<input type="text" value="0"/>
Controller Mode	<input type="text" value="DIRECT"/>
Kp	<input type="text" value="1"/>
Ki	<input type="text" value="1"/>
Kd	<input type="text" value="1"/>

Advanced Setting > PID Operation > Input Item	
Module selection	Choose a predefined module for input data of the PID. Select the type, number and name of the input module. If no option is available, add a new module.
Variable selection	Choose a predefined float variable as the input parameter for PID operation. Select the attribute, type and name of the float variable.
Auto Tune	Enable: Auto-tuning PID parameters for your system. Default: check. Unenable: Tuning PID parameters manually, e.g. Kp, Ki, Kd.
Sample Time (ms)	Set the sampling time. (Unit: ms) Default: 500 ms.
Setpoint	The target value for PID control. Default: 0.
Controller Mode	DIRECT: Set it as positive output value. Default: DIRECT. REVERSE: Set it as reverse output value.
Kp	Set the Proportional gain. Default: 1.
Ki	Set the Integral gain. Default: 1.
Kd	Set the Derivative gain. Default: 1.

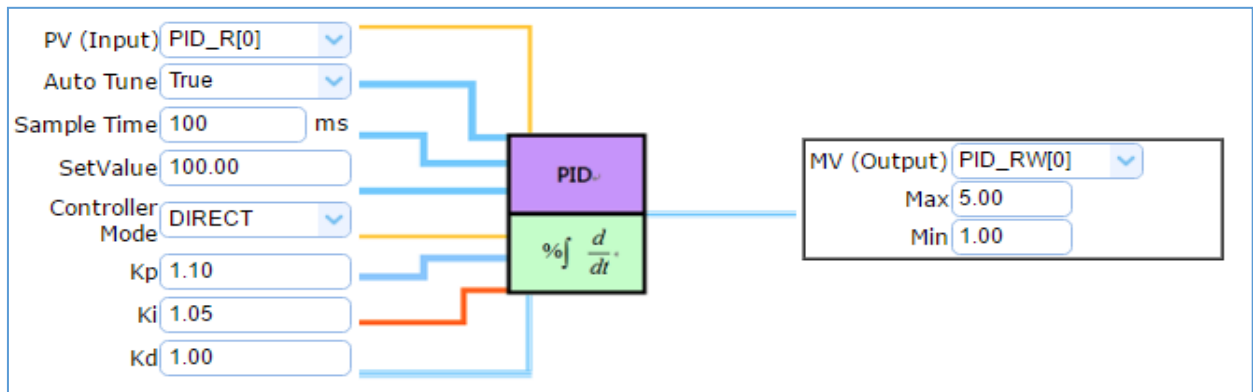


**Output Item**

Module selection	Type :	<input type="text"/>	Please select the module type.
	No. :	<input type="text"/>	Please select the number. When no option is available, add a module.
	Name :	<input type="text"/>	
Variable selection	Attribute	<input type="text"/>	Please select item.
	Type :	<input type="text"/>	Please select item.
	Name :	<input type="text"/>	Please select name. When there is no option, add the variables in the module.
Max	<input type="text"/>	0	
Min	<input type="text"/>	0	

Advanced Setting > PID Operation > Output Item	
Module selection	Choose a predefined module for output data of the PID. Select the type, number and name of the input module. If no option is available, add a new module.
Variable selection	Choose a predefined float variable as the output parameter for PID operation. Select the attribute, type and name of the float variable.
Max	Set the upper-limit value for the variable. Default: 0.
Min	Set the lower-limit value for the variable. Default: 0.
OK	Click to save the settings of the page and back to the PID list page.

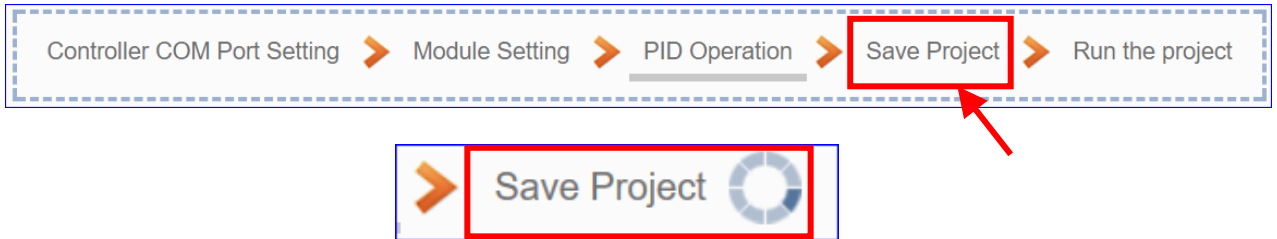
Example:



### Step 4. Save Project

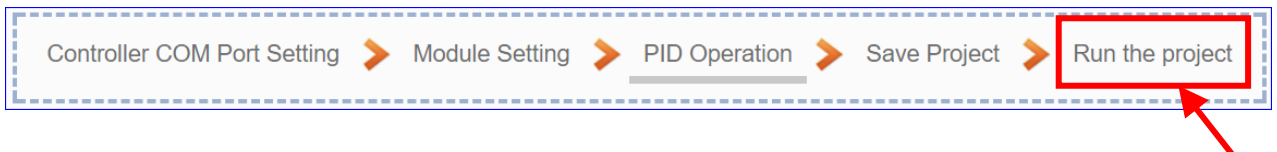
The setting of this example is finished now, and then to save the whole project and run the project. So the last two steps will not show setting pages, but show some displays.

Click the next step [**Save Project**], the Step Box will show an animation as below picture, that means the project is saving. When the animation vanished, the project is saved completely.



### Step 5. Run the Project

The project, after saving, needs to be executed. Click the next step [**Run the Project**].



The Step Box will show the words "**Please wait**" (as below), that means the system is deleting the old project in the UA controller, and will upload the new project into the UA series and run the new project. When the words "**Please wait**" disappears, the new words "**Success**" appears (as below), that means the UA controller is running new project successfully.



And then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

The new project now completes the setting, uploading and running in the UA controller and can process the new function project.

## 5. System Setting

**System Setting** is the first item of the Main Menu and the first screen view when logging the UA Web UI. The System Setting provides the functions for system management of the UA series controller and displays the version information of the system (Higher-left picture).

[System Setting] includes six sub-menu functions (Lower-left picture) and the function descriptions are listed on the page of the Main Menu, such as the controller service, time, network, account, boot and COM port interface settings. This chapter will introduce these function items and setting parameters.

System Setting	
Middleware Version	Version 1.0.0.10
Main Program	Version 1.0.1.0
Web Interface	Version : 2.0.0 Date : 2017/10/31

System Setting	
Controller Service Setting	Controller Service Setting provides the function to display and set the running status of the controller service about the project, MQTT broker and DDNS.
Time Setting	Time Setting provides the function to display and set the date, time and time zone of the controller. (Include manually, synchronization, etc.)
Network Setting	Network Setting provides the function to display and set the network settings. (Include IP, host controller, DDNS, etc.)
Account Setting	Account Setting provides the function to set the username and password of the web UI.
Boot	Boot function provides the function to reboot the controller, and enable the function to run the project, MQTT broker or DDNS at startup.
COM Port Interface Setting	COM Port Interface Setting allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication.

The setting procedure for the UA series controllers is to set up from the left to the right of the main menu functions. User can find the procedure information in the following chapters.

[3.3 Function Setting Procedure](#)

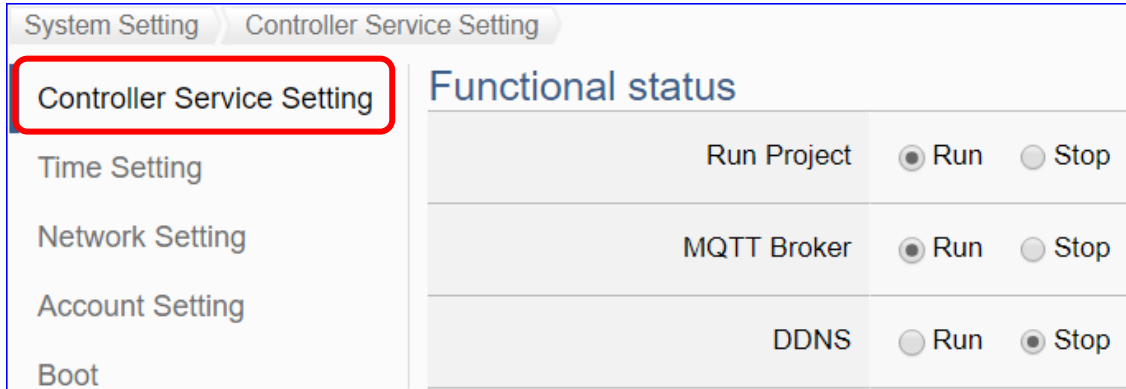
[2. Quick Start](#)

[4. Function Wizard](#)

About the Web UI login information and the UI environment, please refer to [3. Web UI Login and Environment Overview](#).

## 5.1. Controller Service Setting

Controller Service Setting provides the function to display and set the running status of the controller service about the project, MQTT Broker and DDNS.

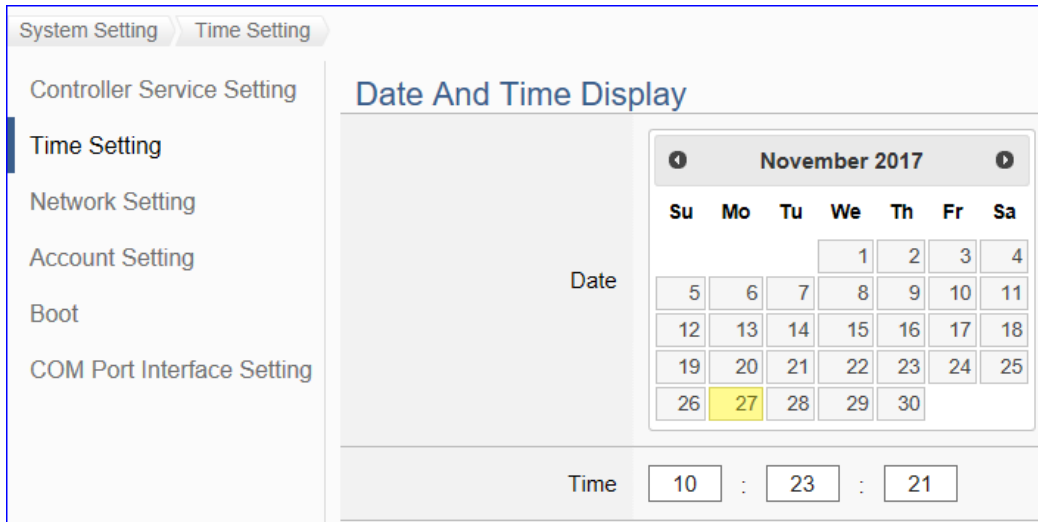


System Setting > Controller Service Setting > Functional status	
Run Project	Display the current status of project running in the UA series controller and provide “Run” and “Stop” button to switch the status. Default: Run.
MQTT Broker	Display the current status of MQTT Broker of the UA series controller and provide “Run” and “Stop” button to switch the status. Default: Run.
DDNS	Display the current status of DDNS Client of the UA series controller and provide “Run” and “Stop” button to switch the status. Default: Stop.

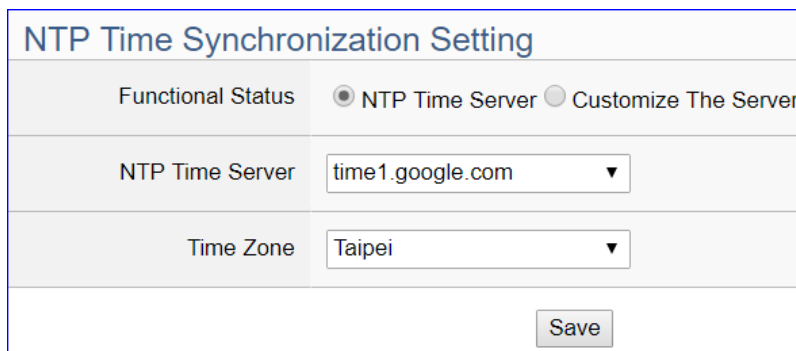
## 5.2. Time Setting

Time Setting provides the function to display and set the date, time and time zone of the controller, including manually, synchronization, etc.

Time Setting provides 3 functions: Data and Time Display, NTP Time Synchronization Setting and Set the Time Manually.



System Setting > Time Setting > Date And Time Display	
Date	Display the date of the UA series controller. The yellow block means current day. User can switch to show the date in other month.
Time	Display the current time of the UA series controller, including hour, minute and second.



System Setting > Time Setting > NTP Time Synchronization Setting	
Functional Status / NTP Time Server	Set up one NTP Time Server from the google (4), indows and nist (4) servers for synchronization. Click “Customize The Server” and enter the IP address or the domain name can set up user own time server.
Time Zone	Set up the time zone.
Save	Click to save the settings of this item.

### Set The Time Manually

Time Setting	<input type="text" value="2017"/> / <input type="text" value="11"/> / <input type="text" value="27"/> <input type="text" value="11"/> : <input type="text" value="30"/> : <input type="text" value="16"/>
Read The Local Computer Time	<input type="button" value="Read"/>
Time Zone	<input type="text" value="Taipei"/> ▼

<b>System Setting &gt; Time Setting &gt; Set The Time Manually</b>	
Time Setting	Set the system time of the UA controller by manually. Directly enter the new year/month/date and hour:minute:second.
Read The Local Computer Time	Click [Read] can copy the current time of the using computer to the "Time Setting" of this item.
Time Zone	Set up the time zone.
Save	Click to save the settings of this item and update the data of "Time Setting" to the "Date And Time Display" on the top of this page.

### 5.3. Network Setting

Network Setting provides the function to display and set the network settings, including IP address, host controller, DDNS, etc.

**Network Setting(LAN1)**

Connection Mode  Specify an IP address  
 Obtain an IP address automatically(DHCP)

IP  .  .  .

Mask  .  .  .

Gateway  .  .  .

---

**Hostname Setting**

Hostname

System Setting > Network Setting > Network Setting (LAN1)	
Connection Mode	<p><b>Specify an IP address:</b> It's the fixed IP mode. Users input the values in the fields of IP, Mask and Gateway according to customer's network. Detail information for the factory default value of UA controller network refers to the <a href="#">Section 2.2.1</a>.</p> <p><b>Obtain an IP address automatically (DHCP):</b> It's the Dynamic Host Configuration Protocol mode. The system assigns the IP, Mask and Gateway automatically.</p>
IP	The IP address of this UA-5200. Factory Default: 192.168.255.1
Mask	The mask address of this UA-5200. Factory Default: 255.255.0.0
Gateway	The gateway address of this UA-5200. Factory Default: 192.168.1.1
Save	Click to save the settings of this item.
System Setting > Network Setting > Hostname Setting	
Hostname	The host name of this UA-5200. Default: system value. User can give a new name, but can not be null.
Save	Click to save the settings of this item.

### Dynamic DNS Setting

Service Provider	<input style="width: 90%;" type="text" value="NO-IP"/>
*Username	<input style="width: 90%;" type="text" value="undefined"/>
*Password	<input style="width: 90%;" type="password" value="....."/>
*Domain Name	<input style="width: 90%;" type="text" value="undefined"/>

<b>System Setting &gt; Network Setting &gt; Dynamic DNS Setting</b>	
Service Provider	Select the company of the DDNS service. Default: NO-IP. Supports: NO-IP, ChangeIP.com, DynDNS, FreeDNS.
*Username	Set up the login user name. The star * means the field can not be null. Default: undefined.
*Password	Set up the login password. The star * means the field can not be null.
*Domain Name	Define the parked domain name of the DDNS. The star * means the field can not be null. Default: undefined.
Save	Click to save the settings of this item.

\* The star “ \* ” means the field can not be null.



## 5.4. Account Setting

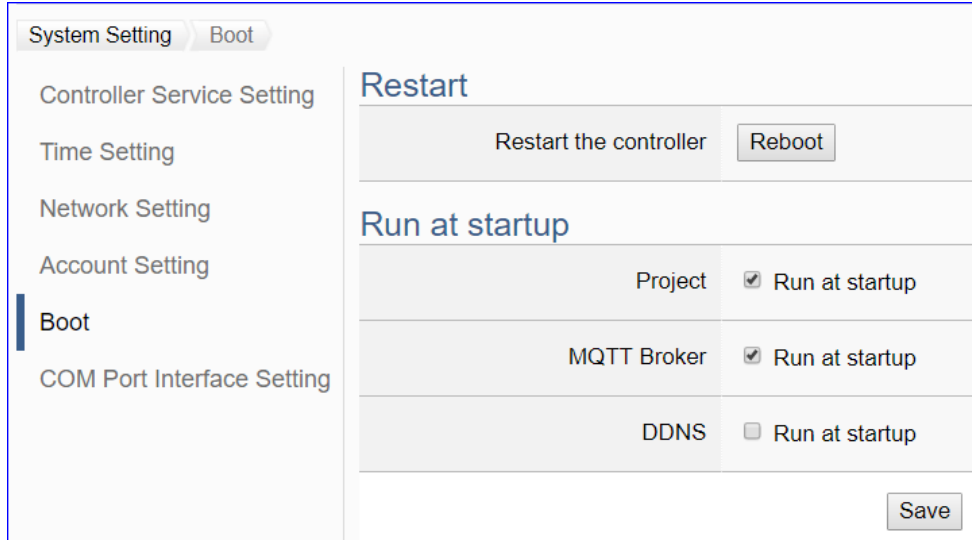
Account Setting provides the function to set the login username and password of the UA-5200's web UI.

The factory default username and password of the UA Web UI: root / root. The detail information for the factory default network values of UA controller please refers to the [Section 2.2.1.](#)

System Setting > Account Setting > Account Settings Page	
Username	The login username for the UA-5200's Web UI. Factory default: root
Password	The login password for the UA-5200's Web UI. Factory default: root
Retype Password	Retype the password for conform the operation when setting the new account information.
Save	Click to save the settings of this page.

## 5.5. Boot

Boot function provides the function to reboot the UA series controller, and enable the function to run the project, MQTT broker or DDNS at startup.



System Setting > Boot > Restart	
Restart the controller	Click “Reboot” can restart the UA controller at once.
System Setting > Boot > Run at startup	
Project	Check the “Run at startup” box can set the project to run at the UA controller startup. Default: check.
MQTT Broker	Check the “Run at startup” box can set the MQTT Broker to run at the UA controller startup. Default: check.
DDNS	Check the “Run at startup” box can set the DDNS to run at the UA controller startup. Default: uncheck.
Save	Click to save the settings of this page.

## 5.6. COM Port Interface Setting

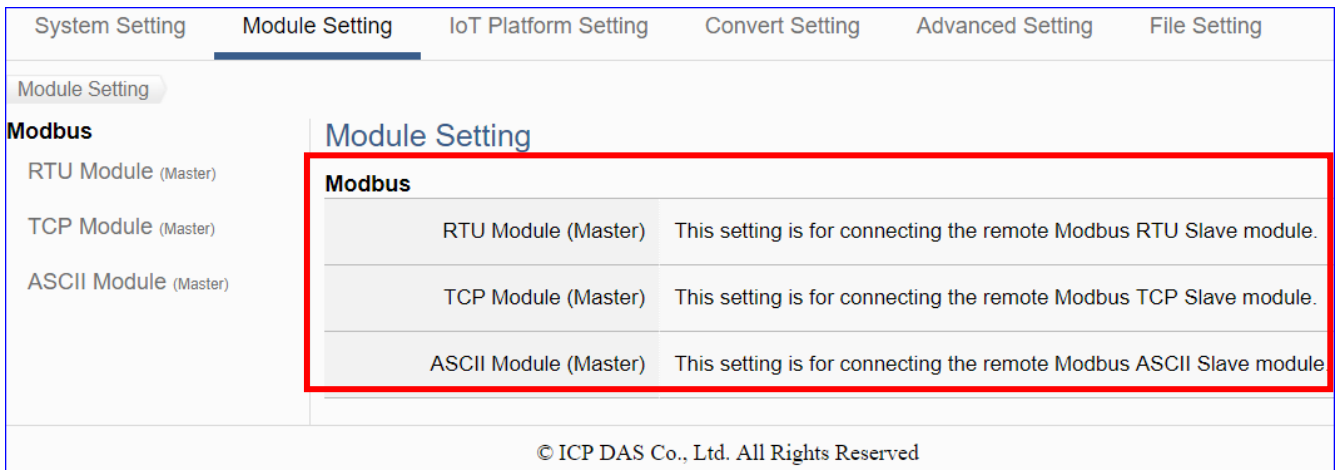
COM Port Interface Setting allows display and set the COM port interface of the UA series controller for the RS-232/RS-485 serial communication.

System Setting > COM Port Interface Setting > COM Port Interface Setting Page	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485 ; ttyO4: RS-232 ; ttyO5: RS-485. Default: ttyO2.
Baud Rate	Choose a baud rate to communicate with the module: 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. The UA controller and the I/O module need have the same baud rate. Default: 115200.
Data Bits	The number of bits used to represent one byte of data: 7 bits or 8 bits. Default: 8 Bits.
Parity	Choose one way for the parity checking. Options: None, Even, and Odd. Default: None.
Stop Bits	Choose the number of stop bit: 1 bit or 2 bits. Default: 1.
Polling Rate(ms)	Set a time interval for the command. Default: 500 ms
Save	Click to save the settings of this page.

## 6. Module Setting

**Module Setting** is the second item of the Main Menu. The Module Setting provides the functions for UA series controller to connect the remote Modbus Slave module including the Modbus RTU/TCP/ASCII module.

[Module Setting] includes three sub-menu functions (see the picture below) and the function descriptions are listed on the page of the Main Menu, such as the Modbus RTU Module (Master), TCP Module (Master) and ASCII Module (Master). The Module Setting will support more modules in the future. This chapter will introduce the current function items and setting parameters.



The setting procedure for the UA series controllers is to set up from the left to the right of the main menu functions. User can find the procedure information in the following chapters.

[3.3 Function Setting Procedure](#)

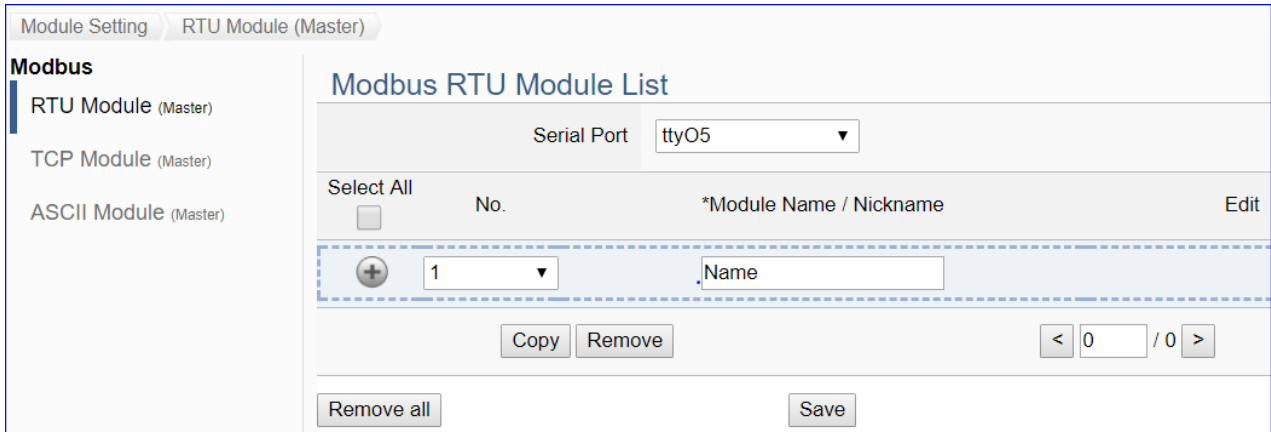
[2. Quick Start](#)

[4. Function Wizard](#)

About the Web UI login information and the UI environment, please refer to [3. Web UI Login and Environment Overview](#).

## 6.1. Modbus RTU (Master)

This setting is for UA Controller connecting the remote Modbus RTU Slave module.



This page is for setting the communication values with the connected modules. First choose the serial port that connected with the module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.

### Setting Steps:

1. Select the module connecting Serial port
2. Give the module name or nickname, e.g. Example2. Default: Name
3. Click the button [ + ] to add a new module
4. Click the button [Edit] to enter the Module Content Setting page
5. Set up the Modbus Mapping Table for the UA controller and module I/O channels



The function items and setting parameters of the [Modbus RTU Module List]:

### Modbus RTU Module List

Serial Port ttyO5 ▼

Select All	No.	*Module Name / Nickname	Edit
<input type="checkbox"/>	3	Name	
<input type="checkbox"/>	1	Example1	<input type="button" value="Edit"/>

< 1 / 1 >

<b>Module Setting &gt; Modbus - RTU Module (Master) &gt; Modbus RTU Module List</b>	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485; ttyO4: RS-232; ttyO5: RS-485. Default: ttyO2.
<input type="button" value="Add"/>	Click to add a list of module.
<input type="checkbox"/>	Check the box in the left of the module is to select that module list, can delete or copy the module. Check the box "Select All" will select all modules in the list.
No.	The module number in the module list (System arrange, not editable)
*Module Name / Nickname	Module name or nick name. User can give a new name. (The star * means this field can not be null.)
Edit	Click to set the module in the Module Content Setting page.
Copy	Select the module wants to copy by check the box and click [Copy] can copy module by assigning port and Number. Yes: copy the module and exit. No: exit without copy.
	<div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: auto;"> <p style="text-align: center; margin: 0;"><b>Copy module</b></p> <p style="margin: 5px 0;">Copy to : <span style="border: 1px solid gray; padding: 2px;">ttyO5</span> ▼</p> <p style="margin: 5px 0;">Quantity : <input style="width: 50px;" type="text"/></p> <p style="text-align: center; margin: 5px 0;"> <input type="button" value="Yes"/> <input type="button" value="No"/> </p> </div>
Remove	Click to delete the checked module(s)
Remove all	Click to delete all modules linked with the selected port. Remove: delete the modules and exit. No: exit without delete module.
	<div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: auto;"> <p style="text-align: center; margin: 0;"><b>Remove module</b></p> <p style="margin: 5px 0;">Select : <span style="border: 1px solid gray; padding: 2px;">ttyO2</span> ▼</p> <p style="text-align: center; margin: 5px 0;"> <input type="button" value="Remove"/> <input type="button" value="No"/> </p> </div>
<input type="button" value="Page 1 of 1"/>	The page number / total pages of the module list. Click < or > to go to the previous or the next page.
Save	Click to save the settings of this page.

Click [Edit] button to enter the “Module Content Setting” page.

(Master) Module Content Setting

### Module Content Setting

No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Name"/>
Slave ID	<input type="text" value="1"/>
Timeout	<input type="text" value="500"/>

### Modbus Mapping Table Setting

Data Model	<input type="text" value="01 Coil Status(0x) ▼"/>
Start Address	<input type="text" value="0"/>
Data Number	<input type="text" value="1"/>
Create Tables	<input type="button" value="Add"/>

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
Slave ID	Set the module Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models “01” ~ “04” for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI) <div style="float: right; border: 1px solid black; padding: 2px; margin-top: 5px;"> <div style="background-color: #007bff; color: white; padding: 2px;">01 Coil Status(0x)</div> <div style="padding: 2px;">02 Input Status(1x)</div> <div style="padding: 2px;">03 Holding Registers(4x)</div> <div style="padding: 2px;">04 Input Registers(3x)</div> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.

The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting		Nickname Setting	
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)	
Input Registers(3x)					
Address	0	Address	<input type="text" value="0"/>	Address	0
Number	2	Number	<input type="text" value="1"/>	Number	1
Type	Bool	Type	Bool	Type	Short
<input type="button" value="Edit"/>		<input type="button" value="Delete"/>	<input type="button" value="Save"/>	<input type="button" value="Edit"/>	
		<input type="button" value="Cancel"/>			
Press Save to finish editing.					
		<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	

Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

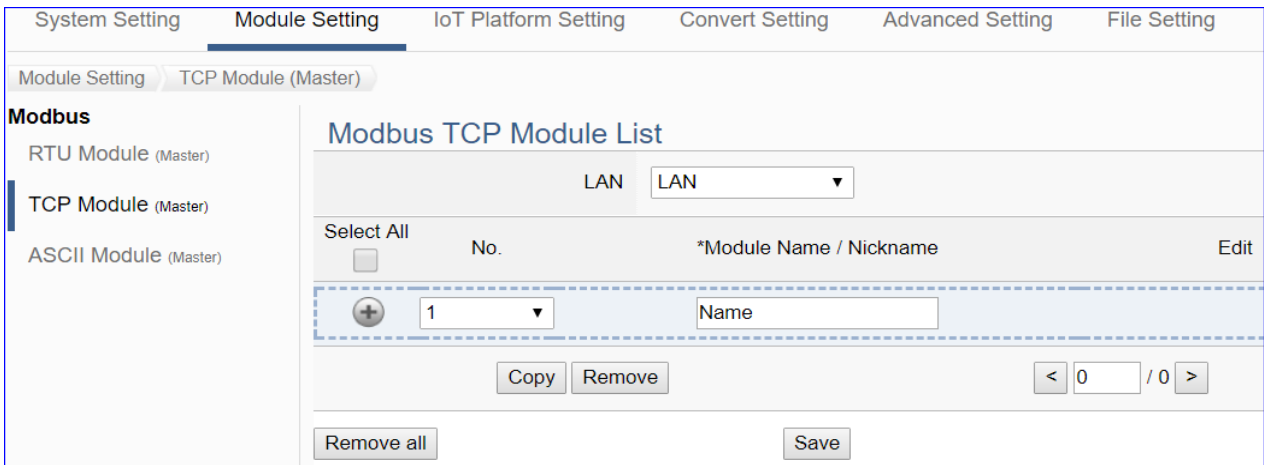


Modbus Mapping Table		Address Setting	Nickname Setting	
<b>01 Coil Status(0x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
1	<input type="text" value="Tag1"/>	Bool	<input type="text"/>	
<b>02 Input Status(1x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
<b>03 Holding Registers(4x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input type="text"/>
<b>04 Input Registers(3x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input type="text"/>
		<input type="button" value="OK"/>	<input type="button" value="Cancel"/>	

Modbus Mapping Table – Nickname Setting	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

## 6.2. Modbus TCP (Master)

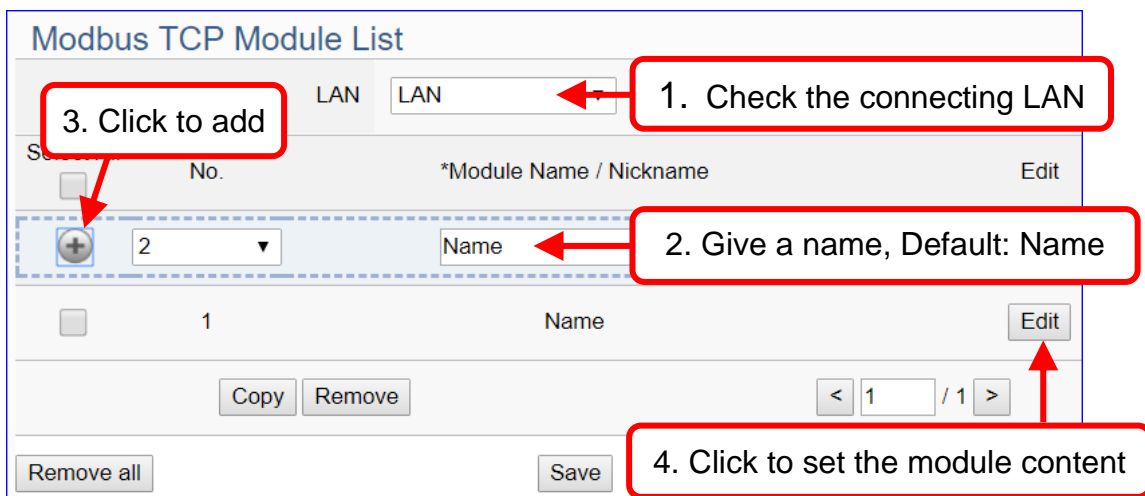
This setting is for UA Controller connecting the remote Modbus TCP Slave module.



This page is for setting the communication values with the connected modules. First choose the Ethernet LAN port that connected with the module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.

### Setting Steps:

1. Select the module connecting Ethernet LAN port
2. Give the module name or nickname, e.g. Example2. Default: Name
3. Click the button [ + ] to add a new module
4. Click the button [Edit] to enter the Module Content Setting page
5. Set up the Modbus Mapping Table for the UA controller and module I/O channels



The function items and setting parameters of the [Modbus TCP Module List]:

Module Setting > Modbus - RTU Module (Master) > Modbus RTU Module List		
LAN	Choose the LAN port of UA controller that links with the TCP module. UA-52xx has one LAN port; the coming UA-2xxx has 2 LAN ports.	
	Click to add a list of module.	
<input type="checkbox"/>	Check the box in the left of the module is to select that module list, can delete or copy the module. Check the box "Select All" will select all modules in the list.	
No.	The module number in the module list (System arrange, not editable)	
*Module Name / Nickname	Module name or nick name. User can give a new name. (The star * means this field can not be null.)	
Edit	Click to set the module in the Module Content Setting page.	
Copy	Select the module wants to copy by check the box and click [Copy] can copy module by assigning port and quantity. Yes: copy the module and exit. No: exit without copy.	
Remove	Click to delete the checked module(s)	
Remove all	Click to delete all modules linked with the selected port. Remove: delete the modules and exit. No: exit without delete module.	
	The page number / total pages of the module list. Click < or > to go to the previous or the next page.	
Save	Click to save the settings of this page.	

Click [Edit] can enter the [Module Content Setting] page to set up the module and the Modbus address mapping table.

Module Content Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Name"/>
IP	<input type="text" value="0"/> <input type="text" value="0"/> <input type="text" value="0"/> <input type="text" value="0"/>
Port	<input type="text" value="502"/>
Slave ID	<input type="text" value="1"/>
Timeout	<input type="text" value="500"/>
Polling Rate	<input type="text" value="500"/>
Modbus Mapping Table Setting	
Data Model	<input type="text" value="01 Coil Status(0x)"/>
Start Address	<input type="text" value="0"/>
Data Number	<input type="text" value="1"/>
Create Tables	<input type="button" value="Add"/>

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
IP	The IP address of the connected module. Default: 0.0.0.0
Port	The port number for Modbus TCP. Default: 502
Slave ID	Set the Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Polling Rate	Set a time interval for the command. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models "01" ~ "04" for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI)
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.

The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting		Nickname Setting	
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)	
Input Registers(3x)					
Address	0	Address	<input type="text" value="0"/>	Address	0
Number	2	Number	<input type="text" value="1"/>	Number	1
Type	Bool	Type	Bool	Type	Short
<input type="button" value="Edit"/>		<input type="button" value="Delete"/> <input type="button" value="Save"/>		<input type="button" value="Edit"/>	
		<input type="button" value="Cancel"/>		<input type="button" value="Edit"/>	
Press Save to finish editing.					
		<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	

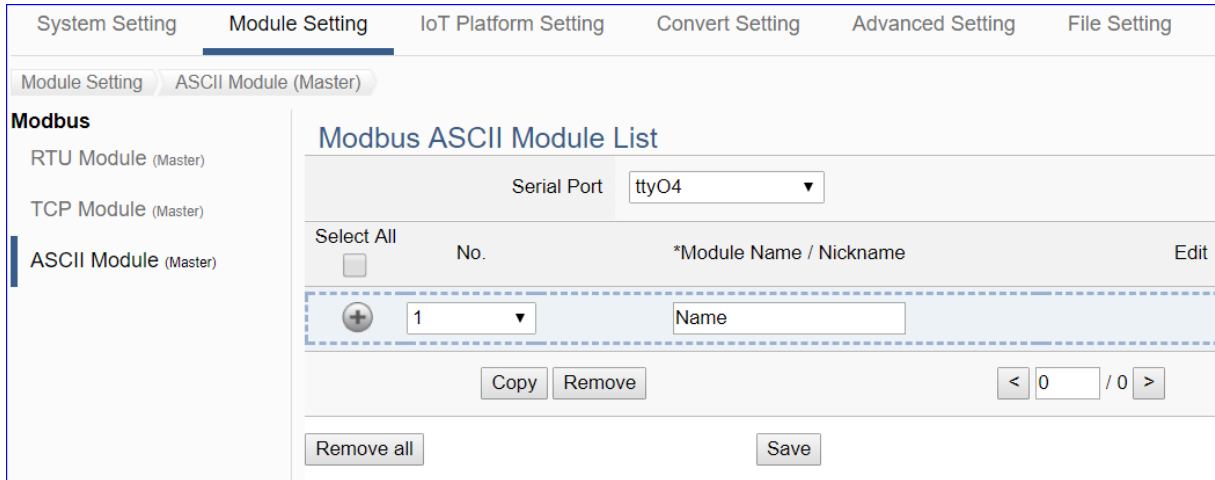
Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

Modbus Mapping Table		Address Setting	Nickname Setting	
<b>01 Coil Status(0x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
1	<input type="text" value="Tag1"/>	Bool	<input type="text"/>	
<b>02 Input Status(1x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
<b>03 Holding Registers(4x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input type="text"/>
<b>04 Input Registers(3x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input type="text"/>
		<input type="button" value="OK"/>	<input type="button" value="Cancel"/>	

Modbus Mapping Table – Nickname Setting	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

### 6.3. Modbus ASCII (Master)

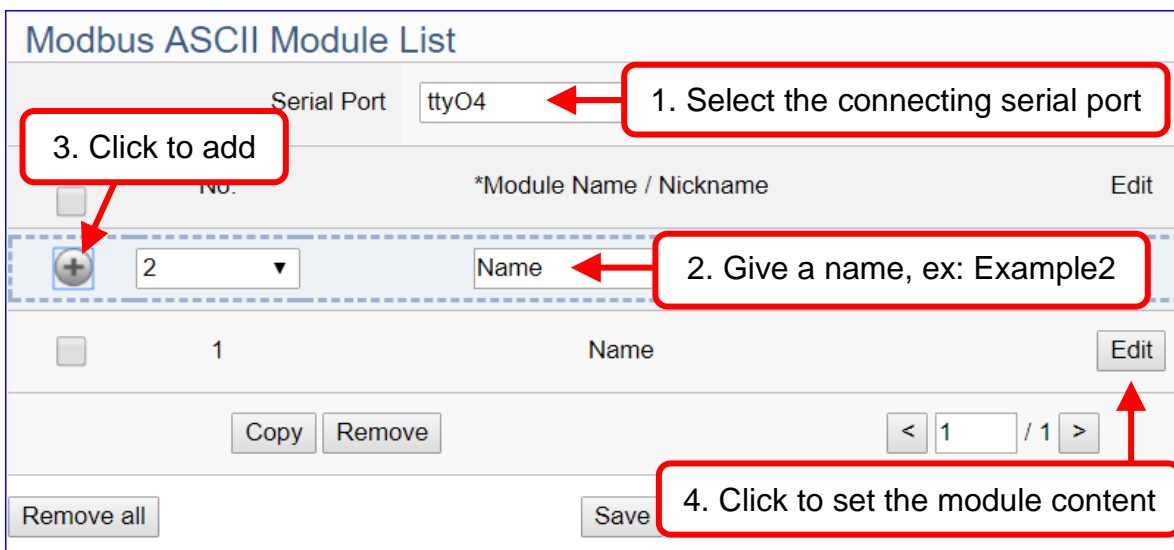
This setting is for UA Controller connecting the remote Modbus ASCII Slave module.



This page is for setting the communication values with the connected modules. First choose the serial port that connected with the module, and each module can give a name (Default name: Name). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.

**Setting Steps:**

1. Select the module connecting Serial port
2. Give the module name or nickname, e.g. Example2. Default: Name
3. Click the button [ + ] to add a new module
4. Click the button [Edit] to enter the Module Content Setting page
5. Set up the Modbus Mapping Table for the UA controller and module I/O channels




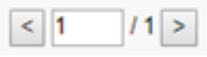
The function items and setting parameters of the [Modbus ASCII Module List]:

### Modbus ASCII Module List

Serial Port

Select All	No.	*Module Name / Nickname	Edit
<input type="checkbox"/>	2	Name	
<input type="checkbox"/>	1	Name	Edit

< 1 / 1 >

Module Setting > Modbus - ASCII Module (Master) > Modbus ASCII Module List	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485; ttyO4: RS-232; ttyO5: RS-485. Default: ttyO2.
	Click to add a list of module.
<input type="checkbox"/>	Check the box in the left of the module is to select that module list, can delete or copy the module. Check the box "Select All" will select all modules in the list.
No.	The module number in the module list (System arrange, not editable)
*Module Name / Nickname	Module name or nick name. User can give a new name. (The star * means this field can not be null.)
Edit	Click to set the module in the Module Content Setting page.
Copy	Select the module wants to copy by check the box and click [Copy] can copy module by assigning port and Quantity. Yes: copy the module and exit. No: exit without copy.
	<div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center; margin: 0;"><b>Copy module</b></p> <p style="margin: 5px 0;">Copy to : <input style="width: 80px;" type="text" value="ttyO5"/></p> <p style="margin: 5px 0;">Quantity : <input style="width: 80px;" type="text"/></p> <p style="text-align: center; margin: 5px 0;"> <input type="button" value="Yes"/> <input type="button" value="No"/> </p> </div>
Remove	Click to delete the checked module(s)
Remove all	Click to delete all modules linked with the selected port. Remove: delete the modules and exit. No: exit without delete module.
	<div style="border: 1px solid gray; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="text-align: center; margin: 0;"><b>Remove module</b></p> <p style="margin: 5px 0;">Select : <input style="width: 80px;" type="text" value="ttyO2"/></p> <p style="text-align: center; margin: 5px 0;"> <input type="button" value="Remove"/> <input type="button" value="No"/> </p> </div>
	The page number / total pages of the module list. Click < or > to go to the previous or the next page.
Save	Click to save the settings of this page.



Click [Edit] button to enter the “Module Content Setting” page.

(Master)
Module Content Setting

### Module Content Setting

No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Name"/>
Slave ID	<input type="text" value="1"/>
Timeout	<input type="text" value="500"/>

### Modbus Mapping Table Setting

Data Model	<input type="text" value="01 Coil Status(0x)"/>
Start Address	<input type="text" value="0"/>
Data Number	<input type="text" value="1"/>
Create Tables	<input type="button" value="Add"/>

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
Slave ID	Set the module Slave ID of the UA-5200. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Modbus Mapping Table Setting	
Data Model	System provides 4 Modbus data models “01” ~ “04” for mapping to address of DO, DI, AO and AI. (ex. 01: DO channels, 02: DI, 03: AO, 04: AI) <div style="float: right; border: 1px solid #ccc; padding: 2px; margin-top: 5px;"> <span style="background-color: #007bff; color: white; padding: 2px;">01 Coil Status(0x)</span>  <span style="padding: 2px;">02 Input Status(1x)</span>  <span style="padding: 2px;">03 Holding Registers(4x)</span>  <span style="padding: 2px;">04 Input Registers(3x)</span> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to set follow the UA series to start from 0.
Data Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. Default: 1.
Type	This item only when the data model is 03 or 04. Choose the suitable data type: 16-bit Short, 16-bit Unsigned Short, 32-bit Long, 32-bit Unsigned Long, 32-bit Float, 64-bit Double.
Create Tables	Click [Add] button, it will add a table in the Modbus mapping table.

The finished Modbus Mapping Table as below is in order of DO, DI, AO and AI.

Modbus Mapping Table		Address Setting		Nickname Setting	
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)	
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)	
Address	0	Address	<input type="text" value="0"/>	Address	0
Number	2	Number	<input type="text" value="1"/>	Number	1
Type	Bool	Type	Bool	Type	Short
<input type="button" value="Edit"/>		<input type="button" value="Delete"/>	<input type="button" value="Save"/>	<input type="button" value="Edit"/>	
		<input type="button" value="Cancel"/>			
Press Save to finish editing.					
		<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	

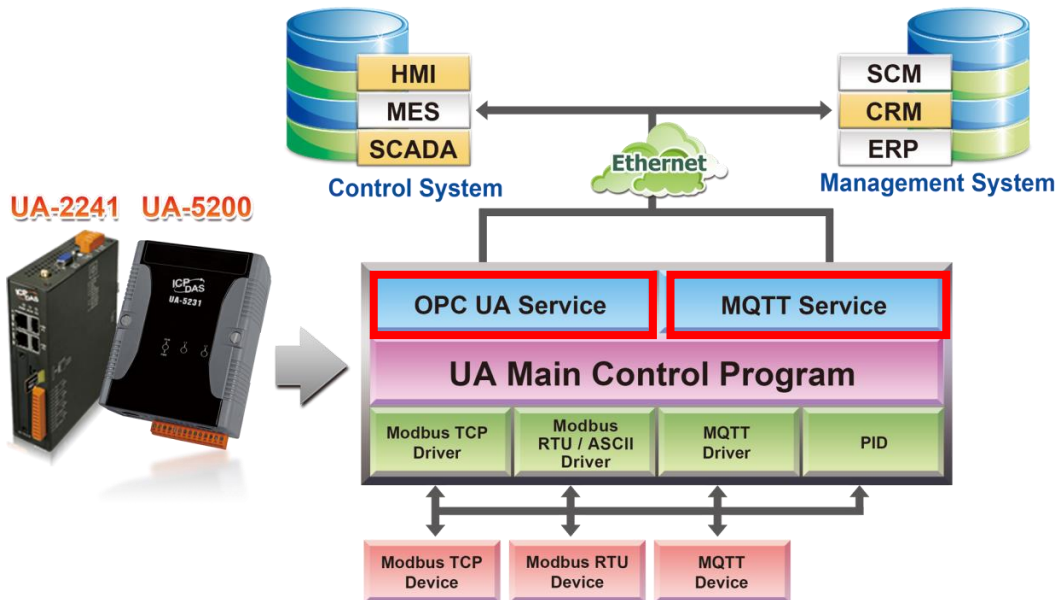
Modbus Mapping Table – Address Setting	
Address Setting	The “Address Setting” page of the Modbus Mapping Table
Nickname Setting	Click can switch to the The “Nickname Setting” page of the Modbus Mapping Table. (Next page)
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Address	The start address of the Modbus command. Default: 0. <b>Note:</b> the address of UA controller is start from 0, even if some modules are start from 1, here it needs to follow the UA series to start from 0.
Number	The number of the Modbus address. Need to give enough number for the DO, DI, AO, AI channels of the module. At least 1.
Type	DO/DI type: Bool (Boolean) AO/AI type: depend on setting of [Modbus Mapping Table Setting]
Edit	Click to change the address and Number.
Delete	Click to delete this address table.
Save	Click to save and exit this table editing.
Cancel	Click to exit without saving and back to the module list page.
OK	Click to save this page settings and back to the module list page.

Modbus Mapping Table		Address Setting	Nickname Setting	
<b>01 Coil Status(0x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
1	<input type="text" value="Tag1"/>	Bool	<input type="text"/>	
<b>02 Input Status(1x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Description	
0	<input type="text" value="Tag0"/>	Bool	<input type="text"/>	
<b>03 Holding Registers(4x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Short	<input type="checkbox"/>	<input type="text"/>
<b>04 Input Registers(3x)</b>				
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>	
Address	Variable name	Data Type	Swap	Description
0	<input type="text" value="Tag0"/>	Float	<input type="checkbox"/>	<input type="text"/>
		<input type="button" value="OK"/>	<input type="button" value="Cancel"/>	

Modbus Mapping Table – Nickname Setting	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Variable name	The variable name of the mapping address. Default: Tag0 and auto arrange the number. User can define the name.
Data Type	Display data type of the variable. (Not editable)
Swap	Check to swap the byte order (Lo-Hi/Hi-Lo) for 4-byte or 8-byte.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

## 7. IoT Platform Setting

**IoT Platform Setting** is the third item of the Main Menu. It manages the interaction of the UA series connecting with the host computer in the Internet of Things. It provides OPC UA and MQTT protocols connection services via the Ethernet interface for data transmission.



[IoT Platform Setting] includes five sub-menu functions in MQTT and OPC UA two connections and the function descriptions are listed on the page of the Main Menu, such as the Local Broker, Remote Broker, MQTT Group Connection and Microsoft Azure Platform in the MQTT Connection category, and the Local Server in the OPC UA Connection category. This chapter will introduce these function items and setting parameters.

System Setting	Module Setting	IoT Platform Setting	Convert Setting	Advanced Setting	File Setting
IoT Platform Setting					
<b>MQTT Connection</b>					
Local Broker					
Remote Broker					
MQTT Group Connection					
Microsoft Azure Platform					
<b>OPC UA Connection</b>					
Local Server					

IoT Platform Setting	
<b>MQTT Connection</b>	
Local Broker	This setting provides to build a user MQTT Broker via the built-in MQTT Broker service of the controller.
Remote Broker	This function can set up the MQTT connection with the remote Broker. User can publish and subscribe messages to the remote Broker through this connection.
MQTT Group Connection	This function can set up the MQTT connection with local and remote brokers. Setting with the MQTT JSON function in the Convert Transmission, It can make the I/O module messages in groups and then mapping to the user-defined publish and subscribe topics.
Microsoft Azure Platform	The system features the connection ability to the Microsoft Azure platform. It allows users to publish messages to Microsoft Azure and receive messages from Microsoft Azure.
<b>OPC UA Connection</b>	
Local Server	This function provides the settings for the OPC UA Server.

## 7.1. MQTT Local Broker

UA series controller built-in MQTT Broker that compliance with MQTT v3.1.1 protocol and supporting MQTT message distribution management. When using MQTT communication, there is no need to build a new Broker system.

The screenshot shows a web interface for 'IoT Platform Setting'. The 'Local Broker' sub-tab is active. On the left, under 'MQTT Connection', 'Local Broker' is selected. The main area, titled 'Local Broker Setting', contains two rows of settings: 'Port' with a text input field containing '1883', and 'Anonymous Login' with a checked checkbox and the label 'Enabled'. A 'Save' button is located at the bottom right of the settings area.

MQTT Connection > Local Broker Setting	
Port	MQTT Local Broker's COM port. System default: 1883
Anonymous Login	Check to allow anonymous login. Default: Check Enabled.
Save	Click to save the settings of this page.

## 7.2. MQTT Remote Broker

UA series controller built-in MQTT Broker(See [Section 7.1](#)), but when users want to use the external MQTT Broker, UA system also provides the settings to connect and publish/subscribe messages with the MQTT Remote Broker.

This page can set up the MQTT connection with the remote Broker. User can publish and subscribe messages to the remote Broker through this connection.

Broker Name	IP / Domain	Port	Edit
<input type="text" value="Name"/>			

Remove < 0 / 0 >

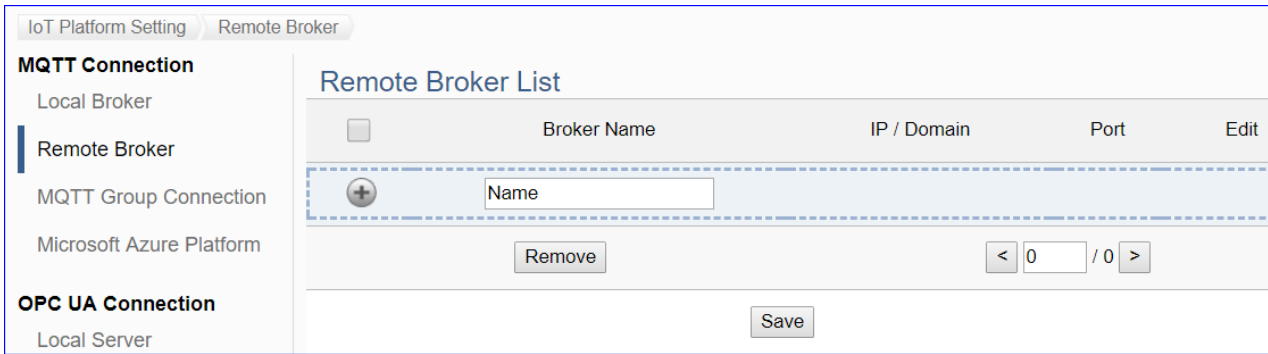
Save

### Setting Sequence for the MQTT Connection:

1. Add and set up a connection Broker name in the Remote Broker List.
2. Set up the contents of the Topic messages published/subscribed by other external MQTT devices for mapping to the Variables Table of the UA-5200 controller.
3. Convert the data contents of the MQTT device to communicate with other protocols.

For the certificate about the communication security, please refer to [Chapter 12](#).

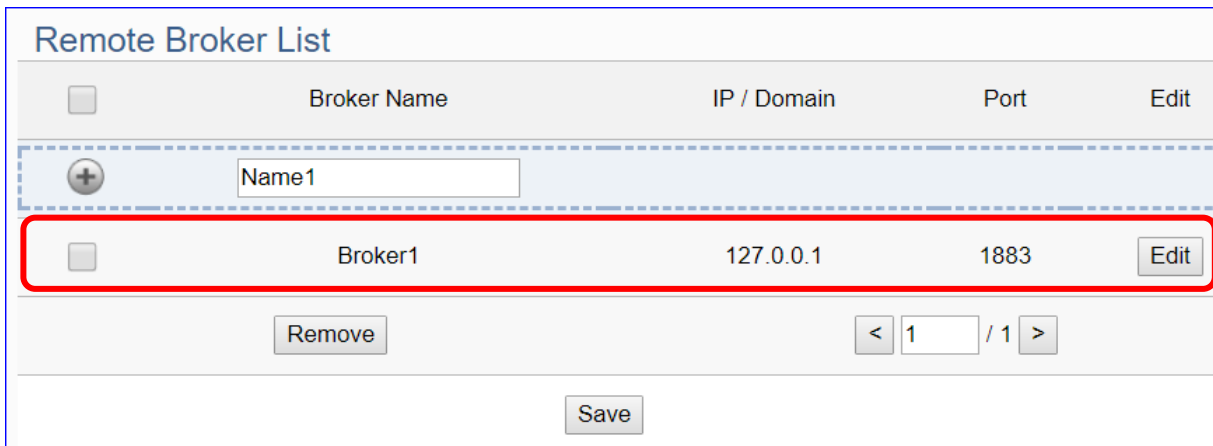
This section will introduce the function items and setting parameters.



**MQTT Connection > Remote Broker > Remote Broker List**

Broker Name	MQTT Remote Broker name. User can give a new name, e.g. Broker1. Default: Name.
	Click to add a list of remote Broker.
Save	Click to save the settings of this page.

After adding a list of the Remote Broker:



**MQTT Connection > Remote Broker > Remote Broker List**

Broker Name	The MQTT remote Broker name.
IP / Domain	The IP address or domain name of the remote Broker.
Port	The communication port of the remote Broker.
<input type="checkbox"/>	Check the box in the left of the Broker is to select that Broker, can delete or copy the Broker. Check the box on the top of the list will select all Brokers in the list.
Edit	Click to set up the remote Broker in the Broker Content Setting page.
Remove	Click to delete the checked Broker(s)
	The page number / total pages of the Broker list. Click < or > to go to the previous or the next page.
Save	Click to save the settings of this page.

Click [Edit] to set up the group in the Broker Content Setting page.

### Broker Content Settings

Broker Name	Broker1
IP / Domain	127.0.0.1
Port	1883
Keep Alive Time(second)	60
SSL/TLS	<input type="checkbox"/> Enabled
Anonymous Login	<input checked="" type="checkbox"/> Enabled

<b>MQTT Connection &gt; Remote Broker List &gt; Broker Content Settings</b>	
Broker Name	The name of the remote MQTT Broker. User can define a new name.
IP / Domain	Set the IP address or domain name of the Remote MQTT Broker. Default: 127.0.0.1
Port	The remote Broker port. Default: 1883.
Keep Alive Time (second)	The Keep alive time. Default: 60 second.
SSL/TLS	Check to enable the supporting of SSL/TLS security communication. Default: Uncheck.
Anonymous Login	Check to allow anonymous login. Default: Check Enabled.
OK	Click to save the setting and exit this page. Click [Cancel] to exit this page without saving.



## 7.3. MQTT Group Connection

This function can set up the MQTT connection with local and remote brokers. Setting with the MQTT JSON function in the Convert Setting, It can make the I/O module messages in groups and then mapping to the user-defined publish and subscribe topics.

If the MQTT Group connection needs to use an external MQTT remote Broker, you need to set the remote Broker connection first and then set the connection group list. This page is for the setting of new, remove and set up the connection group list and their function parameters.

### Setting Sequence for the MQTT Group Connection:

1. Set up a connection MQTT Broker of Local or Remote Broker.
2. Add and set up a MQTT connection group name in the List.
3. Set up the contents of the Topic messages published/subscribed by other external MQTT devices that supporting JSON format for mapping to the Variables Table of the UA-5200 controller.
4. Convert the data contents of the MQTT device into JSON format of groups to communicate with other protocols.

For the certificate about the communication security, please refer to [Chapter 12](#).

This section will introduce the function items and setting parameters.

IoT Platform Setting > MQTT Connection > MQTT Connection Group Name List	
Group Name	MQTT connection group name. User can give a new name, e.g. Group1. Default: Name.
	Click to add a list of MQTT connection group.
Save	Click to save the settings of this page.

After adding a list of the MQTT connection group:

IoT Platform Setting > MQTT Connection > MQTT Connection Group Name List	
Group Name	The MQTT connection group name.
<input type="checkbox"/>	Check the box in the left of the Group name is to select that group, can delete or copy the group. Check the box on the top of the list will select all groups in the list.
Edit	Click to set up the group in the MQTT Client Setting page.
Remove	Click to delete the checked group(s)
	The page number / total pages of the group list. Click < or > to go to the previous or the next page.
Save	Click to save the settings of this page.

Click [Edit] to set up the group in the MQTT Client Setting page.

MQTT Client Setting	
No.	1
Group Name	Name
Scan Rate(ms)	1000
Dead Band	0
Will Topic	
Will	
MQTT Connection	<input checked="" type="checkbox"/> Broker (Local) <input type="checkbox"/> Broker1 (Remote)

IoT Platform Setting > MQTT Connection > MQTT Client Setting	
No.	The MQTT Client Number. (Uneditable)
Group Name	The name of the Group. User can define a new name.
Scan Rate(ms)	Set an update frequency for the data. Unit: ms. Default: 1000 ms.
Dead Band	Give a dead bend value for updating a float signal. Default: 0
Will Topic	The title of a disconnect notice. Default: Null.
Will	The disconnect notice. Default: Null.
MQTT Connection	Check the Broker for this MQTT connection, Local Broker or Remote Broker. Remote Broker option will appear only when set in advance.

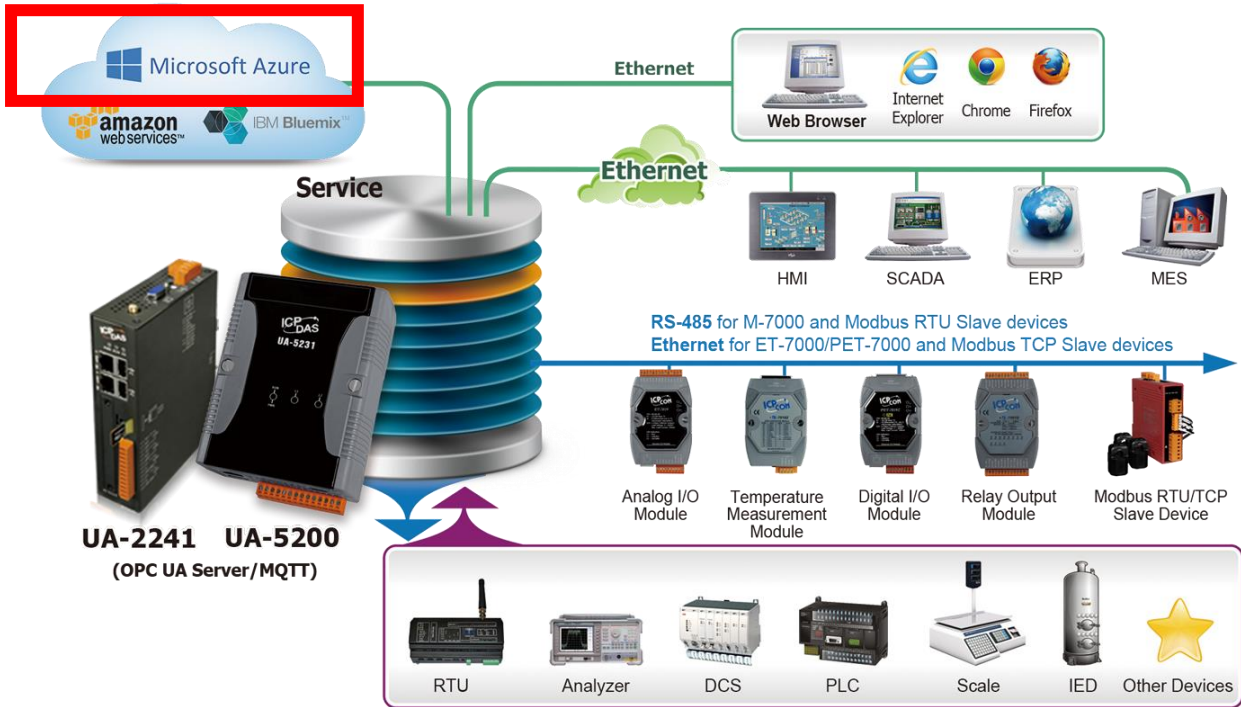
### Publish & Subscribe

Publish Topic	<input type="text" value="/Name/Publish"/>
Publish QoS	<input style="border-bottom: 1px solid black; border-right: 1px solid black; border-left: 1px solid black; border-top: 1px solid black; width: 50px; text-align: right; font-size: small; color: gray; vertical-align: bottom;" type="text" value="2"/> ▼
Subscribe Topic	<input type="text" value="/Name/Subscribe"/>
Subscribe QoS	<input style="border-bottom: 1px solid black; border-right: 1px solid black; border-left: 1px solid black; border-top: 1px solid black; width: 50px; text-align: right; font-size: small; color: gray; vertical-align: bottom;" type="text" value="2"/> ▼
Retain	<input style="border-bottom: 1px solid black; border-right: 1px solid black; border-left: 1px solid black; border-top: 1px solid black; width: 50px; text-align: right; font-size: small; color: gray; vertical-align: bottom;" type="text" value="No"/> ▼

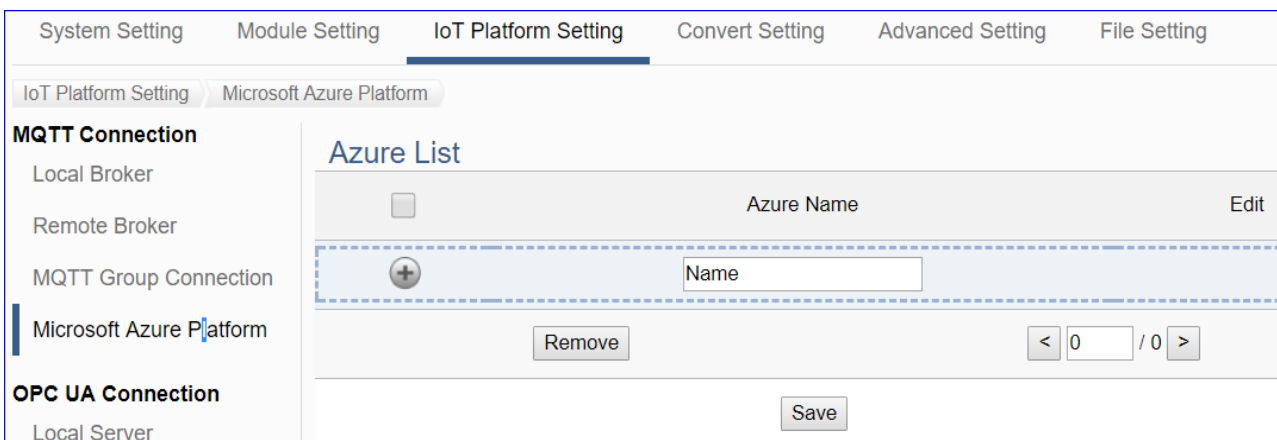
<b>IoT Platform Setting &gt; MQTT Connection &gt; MQTT Client Setting – Publish &amp; Subscribe</b>	
Publish Topic	The topic of sending/publishing data message.
Publish QoS	The publish Qos (Quality of Service) levels. Default: 2. 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Subscribe Topic	The topic of receiving/subscribing data message.
Subscribe QoS	The subscribe Qos (Quality of Service) levels. Default: 2. 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Retain	Whether the Broker to store the message. Default: No.
OK	Click to save the setting and exit this page. Click [Cancel] to exit this page without saving.

## 7.4. MQTT Connection - Microsoft Azure Platform

Microsoft Azure Platform is a common platform to integrate IoT devices into the cloud. Many of the applications use MQTT connection to the cloud for the setting is fast and easy. The UA series also provides the MQTT function for module to connect to the Azure platform and allows users to publish messages to Microsoft Azure and receive messages from Microsoft Azure.



This page will introduce the settings for UA series controller using MQTT service to connect to the Microsoft Azure Platform. It includes new, remove and set up the Azure list and the function parameters



The screenshot shows a web interface titled "Azure List". At the top, there is a header bar with a checkbox, the text "Azure Name", and an "Edit" button. Below this is a dashed blue border containing a "+" icon and a text input field with the value "Name". Underneath the dashed border is a "Remove" button and a pagination control showing "< 0 / 0 >". At the bottom of the interface is a "Save" button.

IoT Platform Setting > MQTT Connection > Microsoft Azure Platform > Azure List	
Azure Name	Azure name. User can give a new name1. Default: Name.
	Click to add a list of Azure.

After adding a list of the Asure:

The screenshot shows the "Azure List" interface after one item has been added. The header bar remains the same. The dashed blue border now contains a "+" icon and a text input field with the value "Name1". Below the dashed border, there is a new row with a checkbox, the text "Name", and an "Edit" button. The "Remove" button and pagination control now show "< 1 / 1 >". The "Save" button is still at the bottom.

IoT Platform Setting > MQTT Connection > Microsoft Azure Platform > Azure List	
Azure Name	Azure name. User can define the name. Default: Name.
	Click to add a new Azure list.
<input type="checkbox"/>	Check the box in the left of a Azure name is to select that Azure, can delete or copy the Azure. Check the box on the top of the list will select all Azures in the list.
Edit	Click to set up the Azure in the Azure Content Setting page.
Remove	Click to delete the checked Azure(s).
	The page number / total pages of the Azure list. Click < or > to go to the previous or the next page.
Save	Click to save the settings of this page.

Click [Edit] to set up the Azure in the Azure Content Setting page.

### Azure Content Settings

Azure Name	<input type="text" value="Name"/>
SAS Token	<input "="" style="width: 100%; height: 100%;" type="text" value="HostName=;DeviceId=;SharedAccessSignature="/>
Keep Alive Time(second)	<input type="text" value="60"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>
CDS	<input checked="" type="checkbox"/> Enabled

IoT Platform Setting > MQTT Connection > Microsoft Azure Platform > Azure Content Settings									
Azure Name	Azure name. User can define the name. Default: Name.								
SAS Token	Input the SAS Token which you previously registered for the UA controller from Microsoft Azure. For the procedure to generate a SAS Token, please refer to the "Documentation > Azure IoT Hub > IoT Hub MQTT support" section on the Microsoft Azure Web Site for detailed information.								
Keep Alive Time(second)	Set the time in second that pass away without communication between the UA controller and Microsoft Azure. Default: 60 second.								
Scan Rate(ms)	Set an update frequency for the task data. Default: 1000 (Unit: ms)								
Dead Band	Give a dead bend value for updating a float signal. Default: 0								
CDS (Connected Device Studio)	<p>If user wants to publish the messages compliant with the Microsoft CDS platform, user must check the "CDS" and fill in the Company ID, Equipment ID and Message ID that applied from the Microsoft CDS platform. Default: Uncheck.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">CDS</td> <td><input checked="" type="checkbox"/> Enabled</td> </tr> <tr> <td>Company ID</td> <td><input type="text" value="0"/></td> </tr> <tr> <td>Equipment ID</td> <td><input type="text"/> <span style="color: red; font-size: small;">Please enter english and numbers.</span></td> </tr> <tr> <td>Message ID</td> <td><input type="text"/></td> </tr> </table> </div>	CDS	<input checked="" type="checkbox"/> Enabled	Company ID	<input type="text" value="0"/>	Equipment ID	<input type="text"/> <span style="color: red; font-size: small;">Please enter english and numbers.</span>	Message ID	<input type="text"/>
CDS	<input checked="" type="checkbox"/> Enabled								
Company ID	<input type="text" value="0"/>								
Equipment ID	<input type="text"/> <span style="color: red; font-size: small;">Please enter english and numbers.</span>								
Message ID	<input type="text"/>								
OK	Click to save and exit this page.								

## 7.5. OPC UA Connection - Local Server

UA series controller built-in OPC UA Server service can integrate the I/O products and the third-party devices, import their data to the back-end SCADA management system or the big-data analysis/decision system, to satisfy the reliability, interoperability and security needs of the Industrial 4.0 automation system.

This page provides the settings for the UA series built-in OPC UA Server.

OPC UA Connection > Local Server – Server	
Server Name	Display the active OPC UA Server name. Not editable. System value: ICPDAS_OPC_UA_Server
Port	The communication port number of the OPC UA Server. System Default: 48010.
Save	Click to save the settings of this item.
OPC UA Connection > Local Server – User Identity Tokens	
Anonymous Login	Check to enable the anonymous login of clients. Default: check.
User Password Login	Check to enable the user password login of clients. Default: uncheck.
Certificate Login	Check to enable the certificate login of clients. Default: uncheck.
Save	Click to save the settings of this item.



## 8. Convert Setting

**Convert Setting** is the fourth item of the Main Menu for the communication conversion.

The Convert Setting has 9 sub-menu items in 3 protocol types including OPC UA, MQTT and MQTT JSON. And each protocol type has 3 convert settings items for conversion with the Modbus RTU/TCP/ASCII (Master) protocols and the function descriptions are listed on the page of the Main Menu. This chapter will introduce these function items and setting parameters.

System Setting	Module Setting	IoT Platform Setting	Convert Setting	Advanced Setting	File Setting
Convert Setting					
<b>OPC UA</b>					
Modbus RTU (Master)					
Modbus TCP (Master)					
Modbus ASCII (Master)					
<b>MQTT</b>					
Modbus RTU (Master)					
Modbus TCP (Master)					
Modbus ASCII (Master)					
<b>MQTT JSON</b>					
Modbus RTU (Master)					
Modbus TCP (Master)					
Modbus ASCII (Master)					

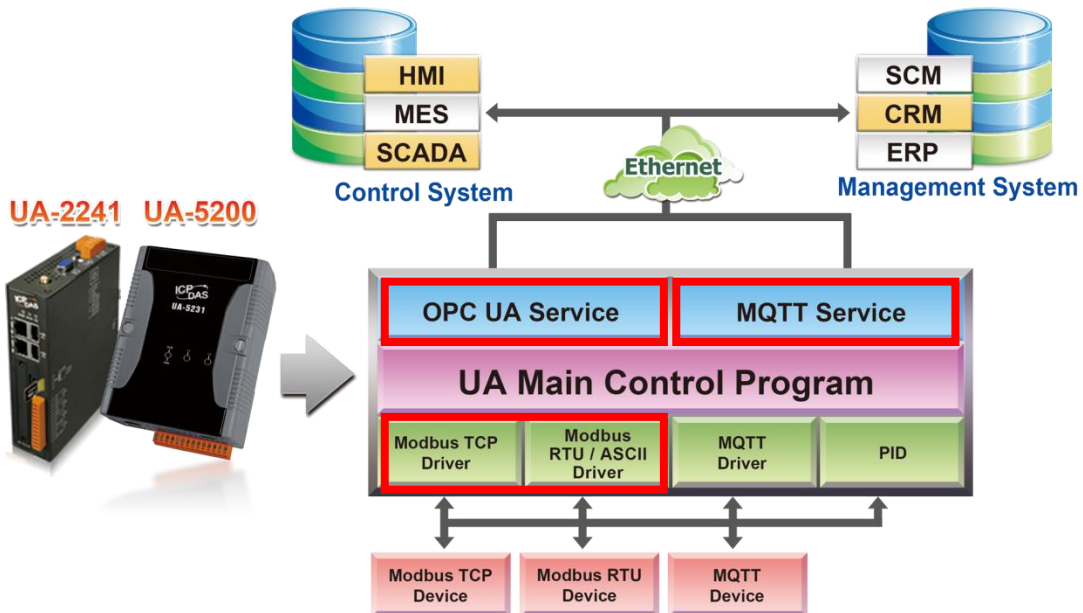
  

Convert Setting	
<b>OPC UA</b>	
Modbus RTU (Master)	Provides OPC UA and Modbus RTU (Master) communication protocol conversion. With this function, the OPC UA Server can read and write the Modbus RTU device that connected to the controller.
Modbus TCP (Master)	Provides OPC UA and Modbus TCP (Master) communication protocol conversion. With this function, the OPC UA Server can read and write the Modbus TCP device that connected to the controller.
Modbus ASCII (Master)	Provides OPC UA and Modbus ASCII (Master) communication protocol conversion. With this function, the OPC UA Server can read and write the Modbus ASCII device that connected to the controller.
<b>MQTT</b>	
Modbus RTU (Master)	Provides MQTT and Modbus RTU (Master) communication protocol conversion. With this function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus RTU device that connected to the controller.
Modbus TCP (Master)	Provides MQTT and Modbus TCP (Master) communication protocol conversion. With this function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus TCP device that connected to the controller.
Modbus ASCII (Master)	Provides MQTT and Modbus ASCII (Master) communication protocol conversion. With this function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus ASCII device that connected to the controller.
<b>MQTT JSON</b>	
Modbus RTU (Master)	Provides MQTT and Modbus RTU (Master) communication protocol conversion. With this function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and combine several messages that converted in JSON format into a group to read and write the multiple channels of the Modbus RTU devices that connected to the controller.
Modbus TCP (Master)	Provides MQTT and Modbus TCP (Master) communication protocol conversion. With this function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and combine several messages that converted in JSON format into a group to read and write the multiple channels of the Modbus TCP devices that connected to the controller.
Modbus ASCII (Master)	Provides MQTT and Modbus ASCII (Master) communication protocol conversion. With this function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and combine several messages that converted in JSON format into a group to read and write the multiple channels of the Modbus ASCII device that connected to the controller.

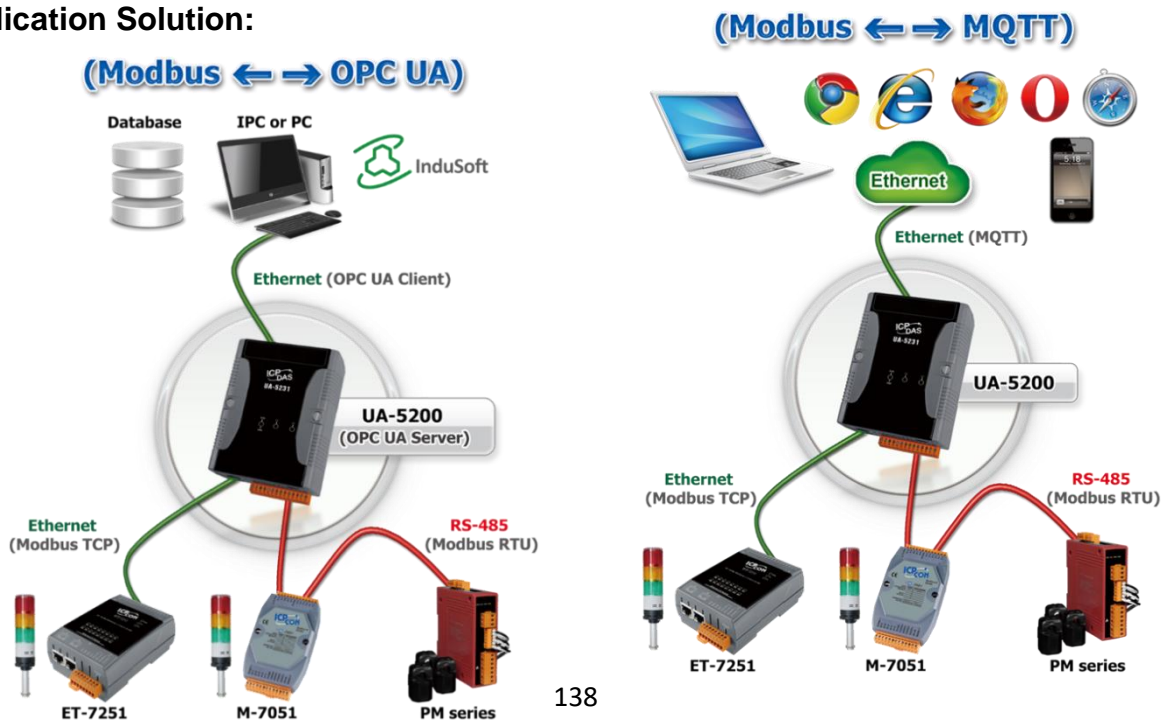
The settings of Modbus RTU/ASCII are the same. Here will introduce them together.

<b>OPC UA</b>	Use <b>OPC UA Service</b> to convert with <b>Modbus RTU/ASCII</b> protocol. <b>(8.1)</b> Use <b>OPC UA Service</b> to convert with <b>Modbus TCP</b> protocol. <b>(8.2)</b>
<b>MQTT</b>	Use <b>MQTT Service</b> to convert with <b>Modbus RTU/ASCII</b> protocol. <b>(8.3)</b> Use <b>MQTT Service</b> to convert with <b>Modbus TCP</b> protocol. <b>(8.4)</b>
<b>MQTT JSON</b>	Use <b>MQTT Service</b> in group of JSON format to convert with <b>Modbus RTU/ASCII</b> protocol. <b>(8.5)</b> Use <b>MQTT Service</b> in group of JSON format to convert with <b>Modbus TCP</b> protocol. <b>(8.6)</b>

**UA Series Function Diagram:**



**Application Solution:**

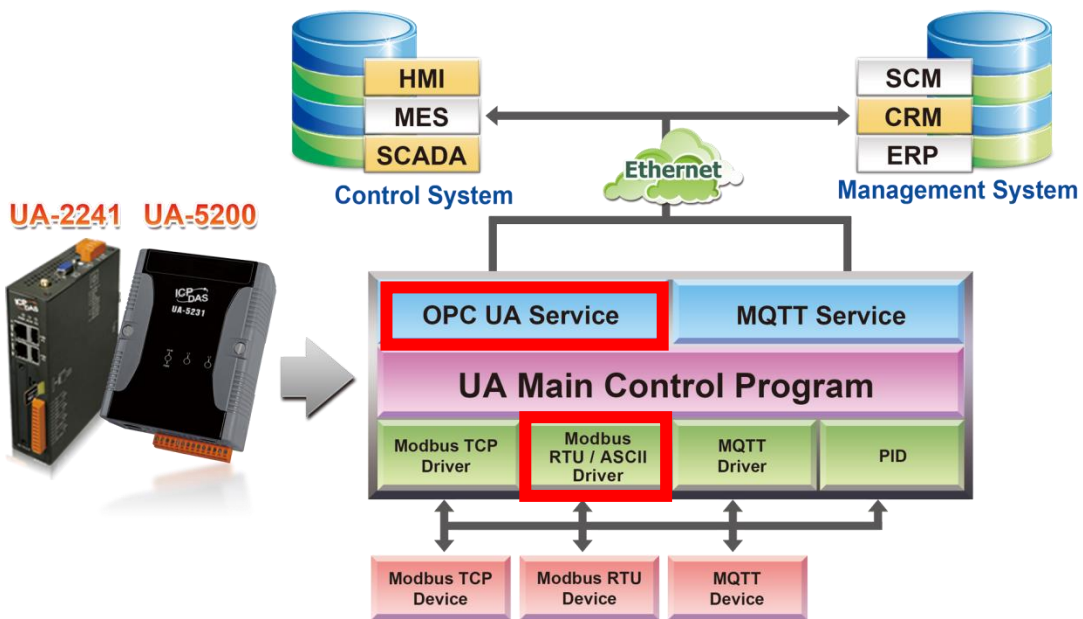


## 8.1. OPC UA and Modbus RTU/ASCII Conversion

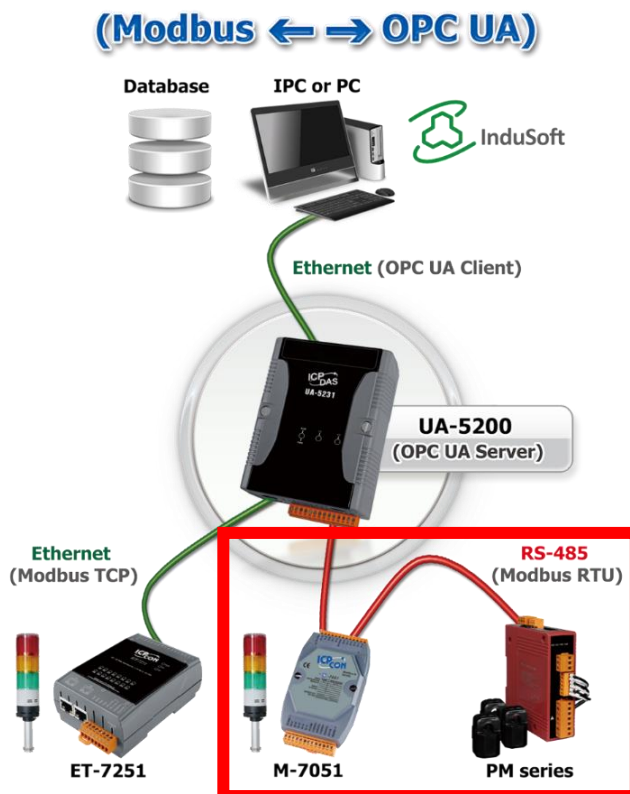
This page provides OPC UA and Modbus RTU/ASCII (Master) communication protocol conversion. With this function, the OPC UA Server can read and write the Modbus RTU / ASCII device that connected to the controller.

The settings of Modbus RTU/ASCII are the same. Here will introduce them together.

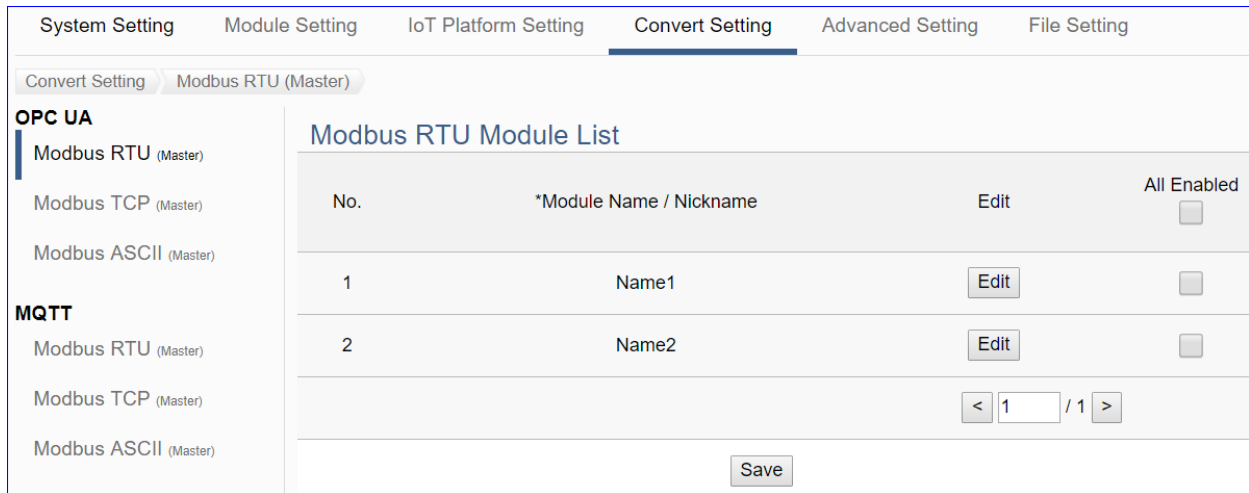
### Function Diagram:




### Application Solution:



When entering the menu [Convert Setting] and the sub-menu [OPC UA] > Modbus RTU (Master) or Modbus ASCII (Master), the Modbus RTU/ASCII modules preset in the [Module Setting] will show up in the Module List. (Refer to Chapter 6 for the Module Setting.)



Convert Setting > OPC UA > Modbus RTU (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	If user wants to enable some I/O channels for conversion, click [Edit] of that module to enter the “Variable Tale” setting. It is normal to set all channels as enabled, and the conversion will not affect the unconnected channels.
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “Module Content Setting” page:

The “Module Content Setting” page after clicking the [Edit] button:

**Module Content Setting**

No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>

**Variable Table**

Name	Attribute	Data Type	Enabled
Tag0	<input type="text" value="Read"/>	Float	<input type="checkbox"/>
Tag0	<input type="text" value="Read / Write"/>	Short	<input checked="" type="checkbox"/>
Tag0	<input type="text" value="Read"/>	Bool	<input checked="" type="checkbox"/>
Tag1	<input type="text" value="Read"/>	Bool	<input type="checkbox"/>
Tag0	<input type="text" value="Read / Write"/>	Bool	<input checked="" type="checkbox"/>
Tag1	<input type="text" value="Read / Write"/>	Bool	<input type="checkbox"/>

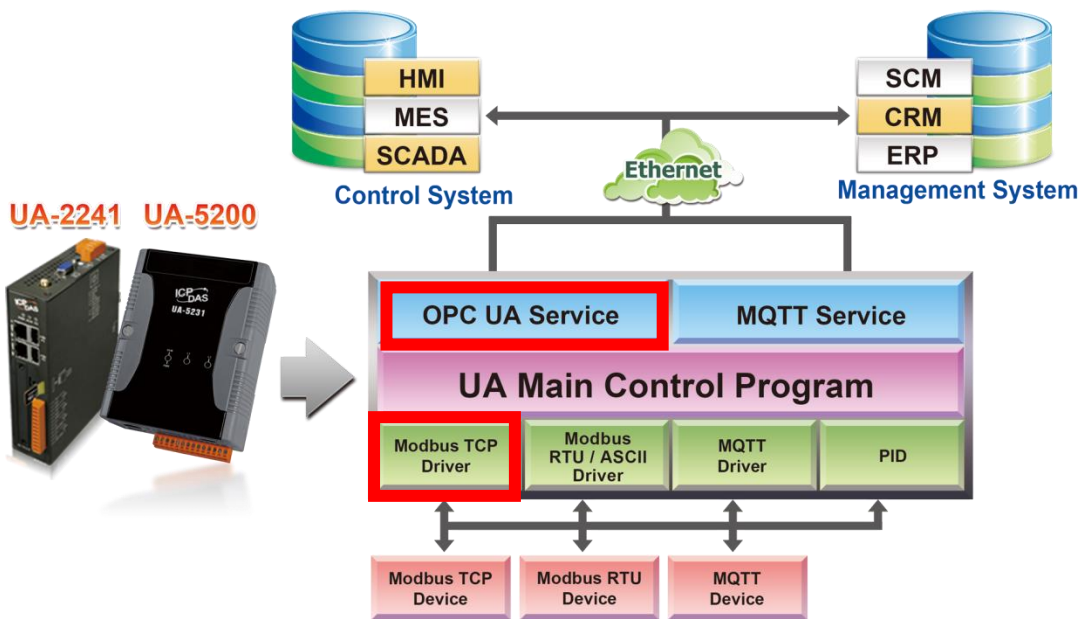
<b>Convert Setting &gt; OPC UA &gt; Modbus RTU (Master) – Module Content</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Convert Setting &gt; OPC UA &gt; Modbus RTU (Master) – Variable Table</b>	
Name	Display the variable name that set in the Modbus Address Mapping Table page (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

When complete the setting, click [OK] to save this page settings and back to the module list page. And remember to click [Save] to save the Convert Setting.

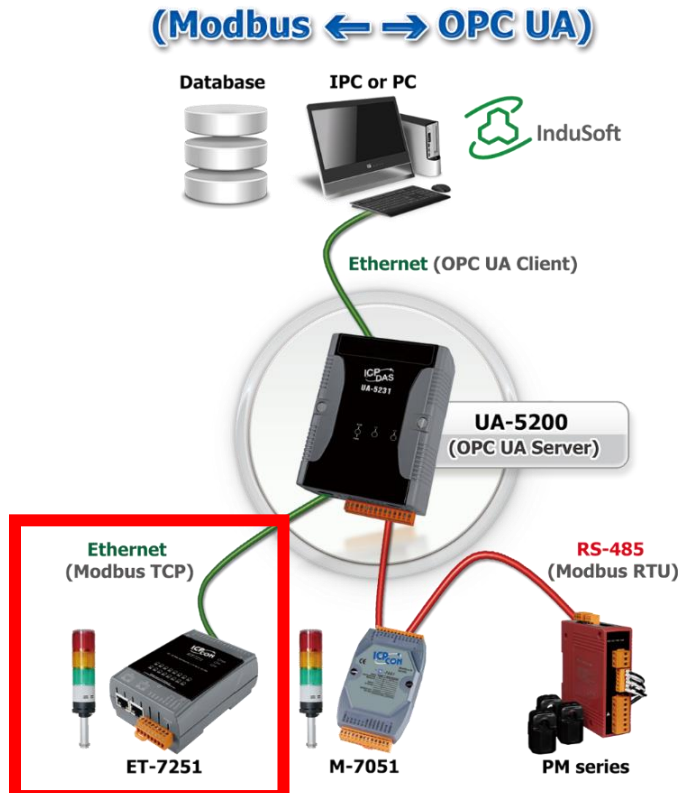
## 8.2. OPC UA and Modbus TCP Conversion

This page provides OPC UA and Modbus TCP (Master) communication protocol conversion. With this function, the OPC UA Server can read and write the Modbus TCP device that connected to the controller.

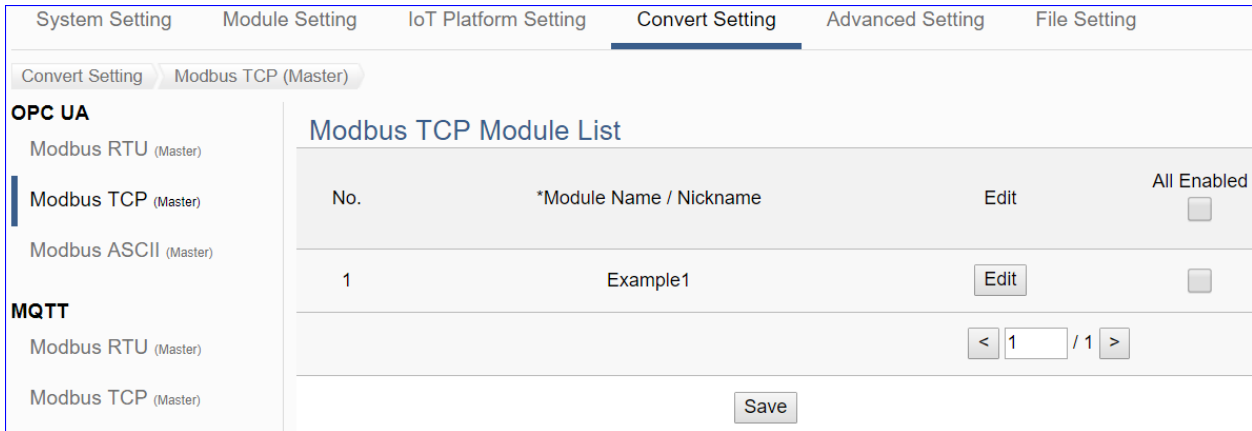
### Function Diagram:




### Application Solution:



When entering the menu [Convert Setting] and the sub-menu [OPC UA] > Modbus RTU (Master) or Modbus ASCII (Master), the Modbus RTU/ASCII modules preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 6](#) for the Module Setting.)



Convert Setting > MQTT > Modbus TCP (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	Click to enter the “MQTT Client Setting” page to set up the Topic, QoX, Publish, Subscribe ...
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “Module Content Setting” page:

The “Module Content Setting” page after clicking the [Edit] button:

Module Content Setting			
No.	<input type="text" value="1"/>		
Module Name	<input type="text" value="Example1"/>		
Variable Table			
Name	Attribute	Data Type	Enabled <input checked="" type="checkbox"/>
Tag0	<input type="text" value="Read"/>	Short	<input checked="" type="checkbox"/>
Tag0	<input type="text" value="Read / Write"/>	Short	<input checked="" type="checkbox"/>
Tag0	<input type="text" value="Read"/>	Bool	<input checked="" type="checkbox"/>
Tag0	<input type="text" value="Read / Write"/>	Bool	<input checked="" type="checkbox"/>
Tag1	<input type="text" value="Read / Write"/>	Bool	<input checked="" type="checkbox"/>
<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	

Convert Setting > OPC UA > Modbus TCP (Master) – Module Content	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
Convert Setting > OPC UA > Modbus TCP (Master) – Variable Table	
Name	Display the variable name that set in the Modbus Address Mapping Table page (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

When complete the setting, click [OK] to save this page settings and back to the module list page. And remember to click [Save] to save the Convert Setting.

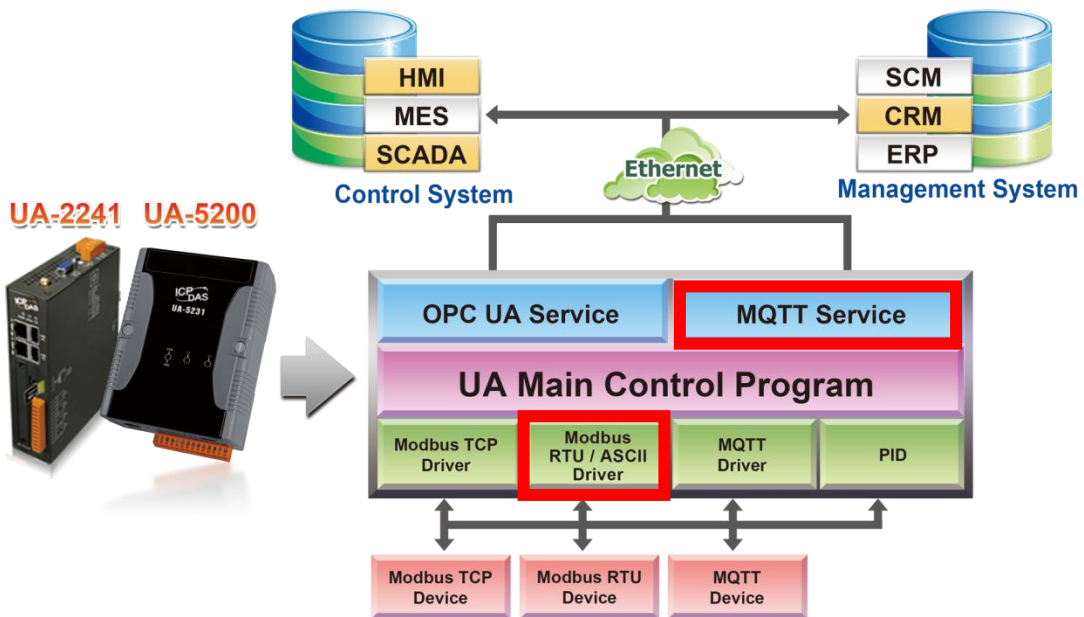


### 8.3. MQTT and Modbus RTU/ASCII Conversion

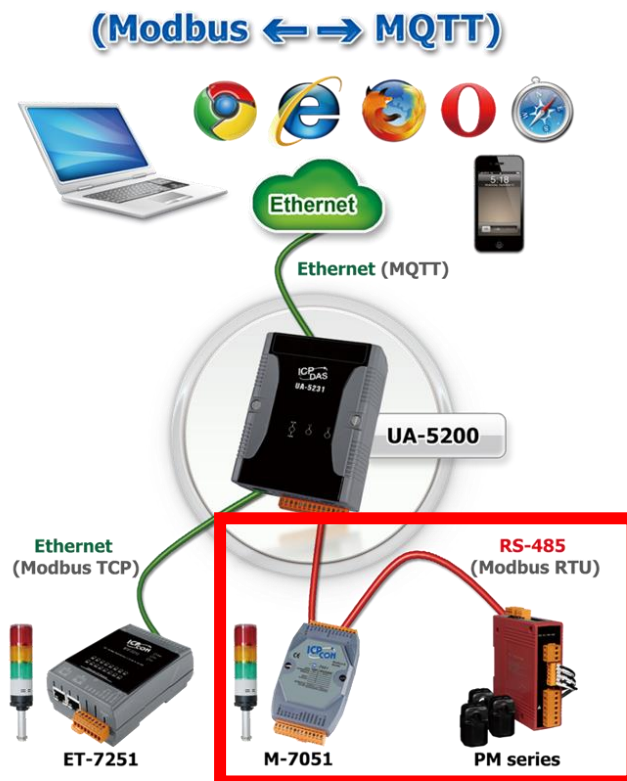
This page provides MQTT and Modbus RTU/ASCII (Master) communication protocol conversion. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus device that connected to the controller.

The settings of Modbus RTU/ASCII are the same. Here will introduce them together. For the certificate about the communication security, please refer to [Chapter 12](#).

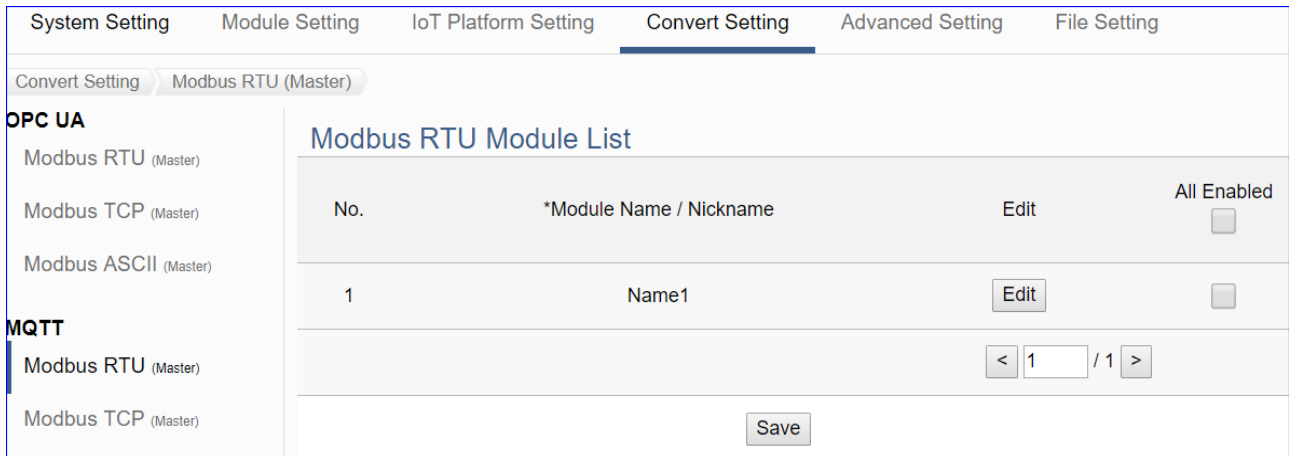
**Function Diagram:**



**Application Solution:**



When entering the menu [Convert Setting] and the sub-menu [MQTT] > Modbus RTU (Master) or Modbus ASCII (Master), the Modbus RTU/ASCII modules preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 6](#) for the Module Setting.)



Convert Setting > MQTT > Modbus RTU (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	Click to enter the “MQTT Client Setting” page to set up the Topic, QoS, Publish, Subscribe ...
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “MQTT Client Setting” page.

The “MQTT Client Setting” page after clicking the [Edit] button:

MQTT Client Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>
Will Topic	<input type="text"/>
Will	<input type="text"/>
MQTT Connection	<input checked="" type="checkbox"/> Broker (Local) <input type="checkbox"/> Broker1 (Remote)

Convert Setting > MQTT > Modbus RTU (Master) – MQTT Client Setting	
No.	The module number in the module list (Uneditable)
Module Name	The module name set in the module list (Not editable here)
Scan Rate(ms)	Set an update frequency for the task data. Default: 1000 (Unit: ms)
Dead Bend	Give a dead bend value for updating a float signal. Default: 0
Will Topic	Enter the title of a disconnect notice. Default: Null.
Will	Enter a disconnect notice. Default: Null.
MQTT Connection	Check the Broker for this MQTT connection, Local Broker or Remote Broker. Remote Broker option will appear only when set in advance.

**Publish & Subscribe**

Details

Name	Attribute	Data Type	Subscribe Topic	Subscribe QoS	Publish Topic	Publish QoS	Retain	Enabled
<input type="text" value="Tag0"/>	Read ▾	Float	/MRTU_No.1_Name1/Input_Registers/Tag0/Subscribe	2 ▾		2 ▾	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	Read / Write ▾	Short	/MRTU_No.1_Name1/Holding_Registers/Tag0/Subscribe	2 ▾	/MRTU_No.1_Name1/Holding_Registers/Tag0/Publish	2 ▾	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	Read ▾	Bool	/MRTU_No.1_Name1/Input_Status/Tag0/Subscribe	2 ▾		2 ▾	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	Read / Write ▾	Bool	/MRTU_No.1_Name1/Coil_Status/Tag0/Subscribe	2 ▾	/MRTU_No.1_Name1/Coil_Status/Tag0/Publish	2 ▾	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="Tag1"/>	Read / Write ▾	Bool	/MRTU_No.1_Name1/Coil_Status/Tag1/Subscribe	2 ▾	/MRTU_No.1_Name1/Coil_Status/Tag1/Publish	2 ▾	<input type="checkbox"/>	<input type="checkbox"/>

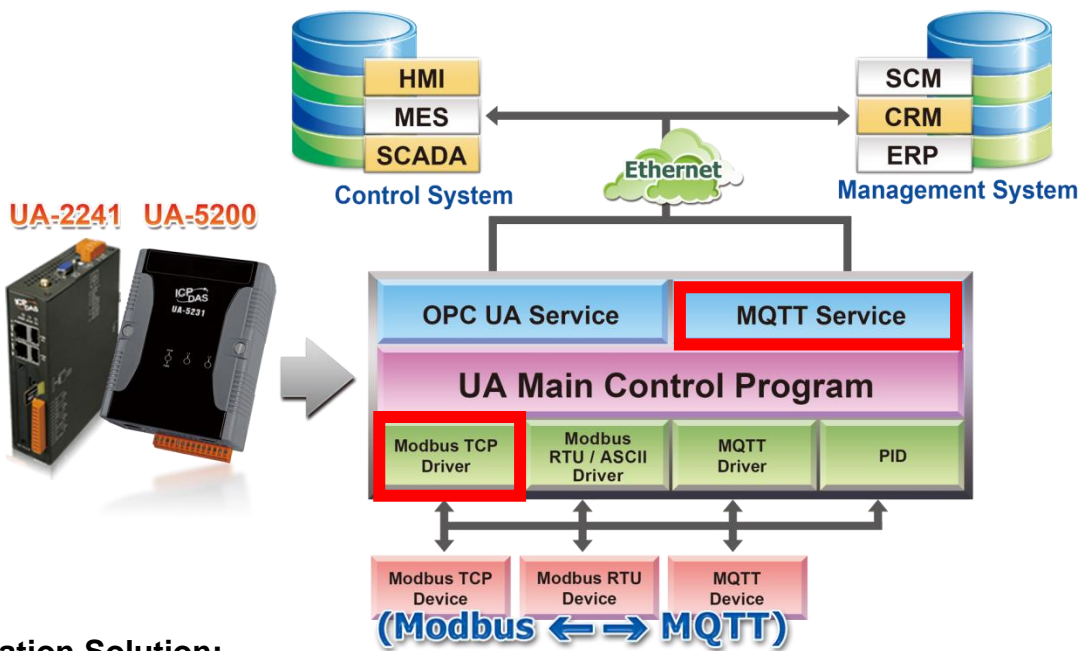
<b>Convert Setting &gt; MQTT &gt; Modbus RTU (Master) – Publish &amp; Subscribe</b>	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Name	The variable name of the mapping address. (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Subscribe Topic	The topic of receiving/subscribing data message.
Subscribe QoS	The subscribe QoS (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Publish Topic	The topic of sending/publishing data message.
Publish QoS	The publish QoS (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Retain	Check [Retain] box of the top row can store the broker message for all variables in list. Check the box of each variable can store the broker message just that variable. Default: Uncheck.
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

## 8.4. MQTT and Modbus TCP Conversion

This page provides MQTT and Modbus TCP (Master) communication protocol conversion. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus device that connected to the controller.

For the certificate about the communication security, please refer to [Chapter 12](#).

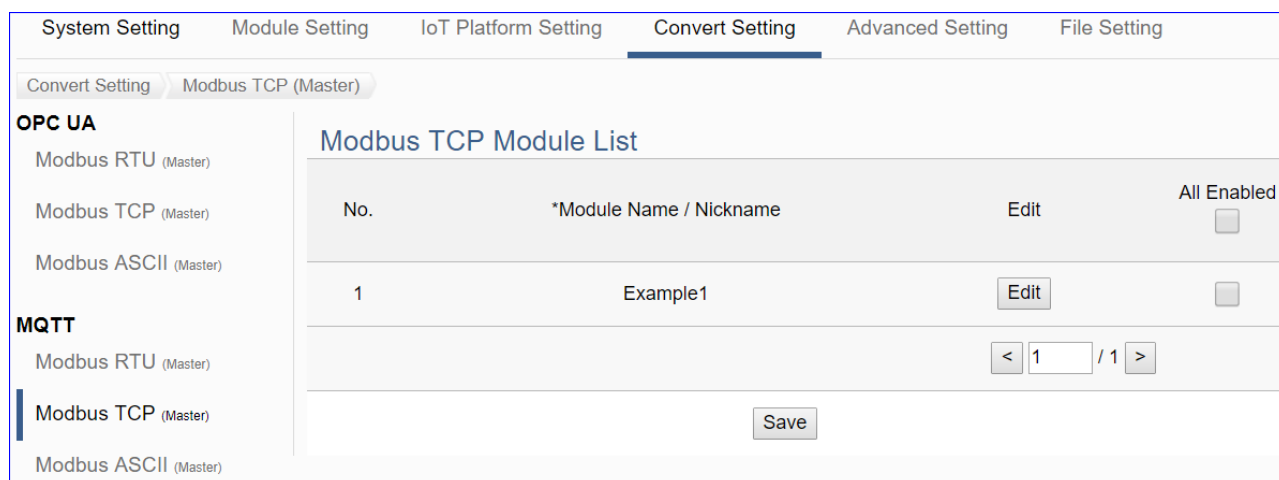
### Function Diagram:




### Application Solution:



When entering the menu [Convert Setting] and the sub-menu [MQTT] > Modbus TCP (Master), the Modbus TCP modules preset in the [Module Setting] will show up in the Module List. (Refer to Chapter 6 for the Module Setting.)



Convert Setting > MQTT > Modbus RTU (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	Click to enter the “MQTT Client Setting” page to set up the Topic, QoS, Publish, Subscribe ...
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “MQTT Client Setting” page.

The “MQTT Client Setting” page after clicking the [Edit] button:

MQTT Client Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>
Will Topic	<input type="text"/>
Will	<input type="text"/>
MQTT Connection	<input checked="" type="checkbox"/> Broker (Local) <input type="checkbox"/> Broker1 (Remote)

Convert Setting > MQTT > Modbus TCP (Master) – MQTT Client Setting	
No.	The module number in the module list (Uneditable)
Module Name	The module name set in the module list (Not editable here)
Scan Rate(ms)	Set an update frequency for the task data. Default: 1000 (Unit: ms)
Dead Bend	Give a dead bend value for updating a float signal. Default: 0
Will Topic	Enter the title of a disconnect notice. Default: Null.
Will	Enter a disconnect notice. Default: Null.
MQTT Connection	Check the Broker for this MQTT connection, Local Broker or Remote Broker. Remote Broker option will appear only when set in advance.

**Publish & Subscribe**

Details

Name	Attribute	Data Type	Subscribe Topic	Subscribe QoS	Publish Topic	Publish QoS	Retain	Enabled
Tag0	Read	Float	/MRTU_No.1_Name1/Input_Registers/Tag0/Subscribe	2		2	<input type="checkbox"/>	<input type="checkbox"/>
Tag0	Read / Write	Short	/MRTU_No.1_Name1/Holding_Registers/Tag0/Subscribe	2	/MRTU_No.1_Name1/Holding_Registers/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>
Tag0	Read	Bool	/MRTU_No.1_Name1/Input_Status/Tag0/Subscribe	2		2	<input type="checkbox"/>	<input type="checkbox"/>
Tag0	Read / Write	Bool	/MRTU_No.1_Name1/Coil_Status/Tag0/Subscribe	2	/MRTU_No.1_Name1/Coil_Status/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>
Tag1	Read / Write	Bool	/MRTU_No.1_Name1/Coil_Status/Tag1/Subscribe	2	/MRTU_No.1_Name1/Coil_Status/Tag1/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>

<b>Convert Setting &gt; MQTT &gt; Modbus TCP (Master) – Publish &amp; Subscribe</b>	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Name	The variable name of the mapping address. (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Subscribe Topic	The topic of receiving/subscribing data message.
Subscribe QoS	The subscribe Qos (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Publish Topic	The topic of sending/publishing data message.
Publish QoS	The publish Qos (Quality of Service) levels. Default: 2 0: Delivering a message at most once. 1: Delivering a message at least once. 2: Delivering a message at exactly once.
Retain	Check [Retain] box of the top row can store the broker message for all variables in list. Check the box of each variable can store the broker message just that variable. Default: Uncheck.
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

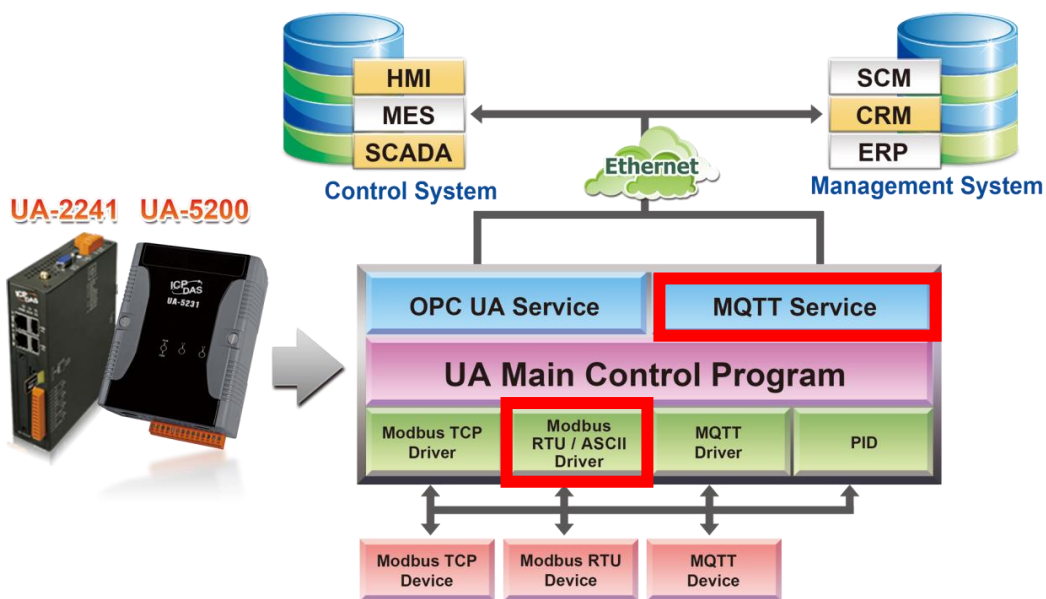


## 8.5. MQTT JSON and Modbus RTU/ASCII Conversion

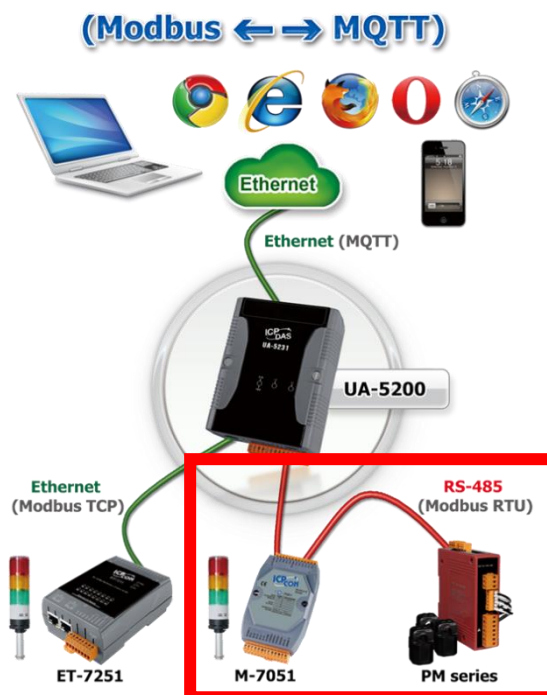
This page provides MQTT JSON and Modbus RTU/ASCII (Master) communication protocol conversion. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and combine several messages that converted in JSON format into a group to read and write the multiple channels of the Modbus RTU/ASCII devices that connected to the controller.

The settings of Modbus RTU/ASCII are the same. Here will introduce them together. For the certificate about the communication security, please refer to [Chapter 12](#).

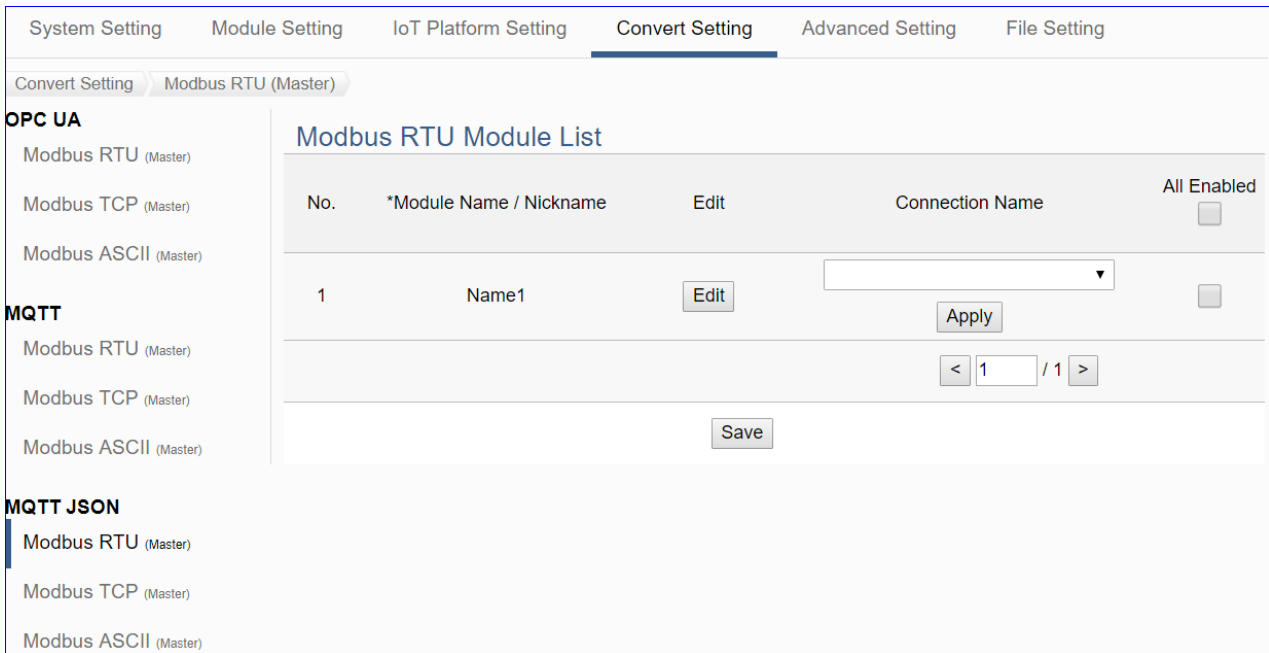
### Function Diagram:




### Application Solution:



When entering the menu [Convert Setting] and the sub-menu [MQTT JSON] > Modbus RTU or Modbus ASCII (Master), the Modbus RTU/ASCII modules preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 6](#) for the Module Setting.)



Convert Setting > MQTT JSON > Modbus RTU (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
Connection Name	Select a group connection name, and then click [Apply].
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
Edit	If user wants to enable some I/O channels for conversion, click [Edit] of that module to enter the “Module Content Setting” and “Variable Tale” page.
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “Module Content Setting” and “Variable Tale” page:

### Module Content Setting

No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>

### Variable Table

Details

Variable Name	Alias	Attribute	Data Type	Connection Name	Enabled
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read"/>	Float	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read / Write"/>	Short	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag1"/>	<input type="text" value="Tag1"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag1"/>	<input type="text" value="Tag1"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>	<input type="checkbox"/>

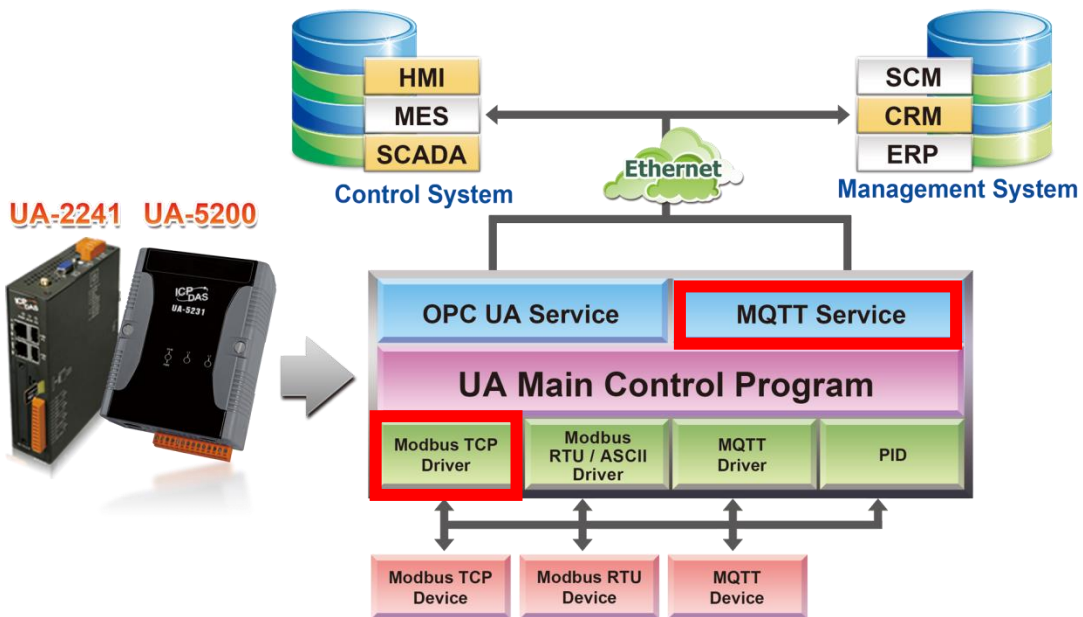
<b>Convert Setting &gt; MQTT JSON &gt; Modbus RTU (Master) – Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Convert Setting &gt; MQTT JSON &gt; Modbus RTU (Master) – Variable Table</b>	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Variable Name	The variable name of the mapping address. (Not editable here)
Alias	The the alias name for the variable. (Editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Connection Name	Select the group name that set in the group list page.
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

## 8.6. MQTT JSON and Modbus TCP Conversion

This page provides MQTT JSON and Modbus TCP (Master) communication protocol conversion. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and combine several messages that converted in JSON format into a group to read and write the multiple channels of the Modbus TCP devices that connected to the controller.

For the certificate about the communication security, please refer to [Chapter 12](#).

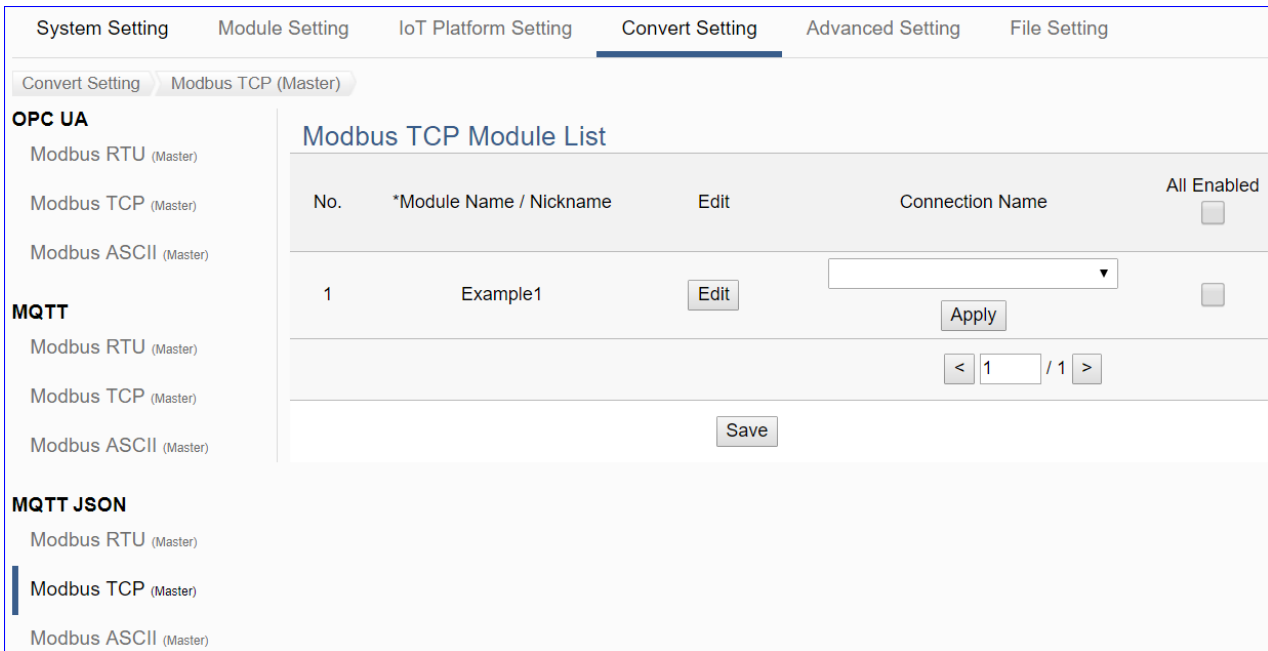
### Function Diagram:

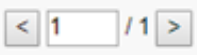


### Application Solution:



When entering the menu [Convert Setting] and the sub-menu [MQTT JSON] > Modbus TCP (Master), the Modbus TCP modules preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 6](#) for the Module Setting.)



Convert Setting > MQTT JSON > Modbus TCP (Master) Module List	
No.	The module number in the module list (Not editable here)
*Module Name / Nickname	The module name set in the module list (Not editable here)
Connection Name	Select a group connection name, and then click [Apply].
Edit	If user wants to enable some I/O channels for conversion, click [Edit] of that module to enter the “Module Content Setting” and “Variable Tale” page.
All Enabled	Check [All Enabled] box to enable all modules in list for conversion. Default: Uncheck. Check the box of each module can enable just that module for conversion.
	The page number of the module list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the settings of this page.

Click [Edit] button could enter the “Module Content Setting” and “Variable Tale” page:

### Module Content Setting

No.	<input type="text" value="1"/>
Module Name	<input type="text" value="Example1"/>

### Variable Table

Details

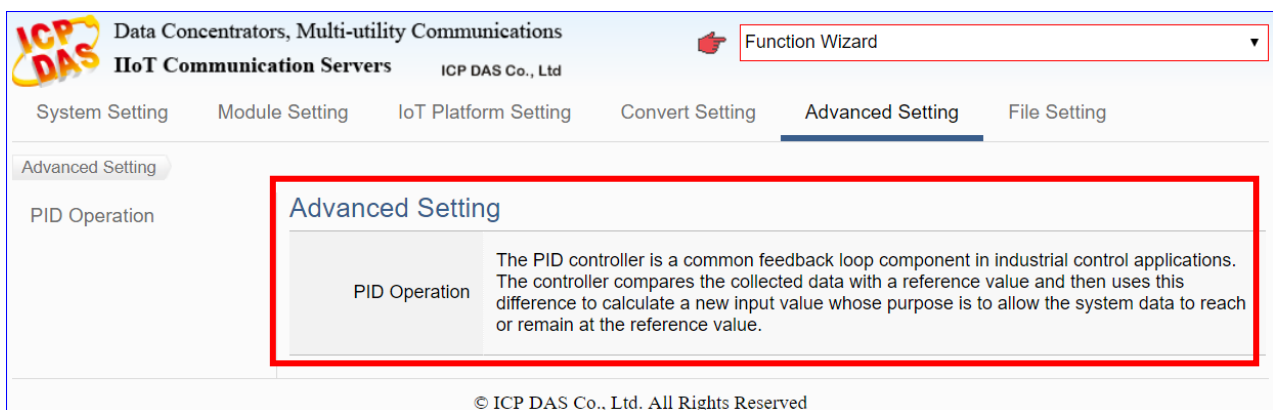
Variable Name	Alias	Attribute	Data Type	Connection Name	Enabled
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	Read ▾	Float	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	Read / Write ▾	Short	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	Read ▾	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag1"/>	<input type="text" value="Tag1"/>	Read ▾	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag0"/>	<input type="text" value="Tag0"/>	Read / Write ▾	Bool	<input type="text"/>	<input type="checkbox"/>
<input type="text" value="Tag1"/>	<input type="text" value="Tag1"/>	Read / Write ▾	Bool	<input type="text"/>	<input type="checkbox"/>

<b>Convert Setting &gt; MQTT JSON &gt; Modbus TCP (Master) – Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Convert Setting &gt; MQTT JSON &gt; Modbus TCP (Master) – Variable Table</b>	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Variable Name	The variable name of the mapping address. (Not editable here)
Alias	The the alias name for the variable. (Editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable that set in the Modbus Address Mapping Table page. (Not editable) Include: Bool, Short, Float...
Connection Name	Select the group name that set in the group list page.
Enabled	Check [Enabled] box of the top row can enable all variables in list. Check the box of each variable can enable just that variable for conversion. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

## 9. Advanced Setting

**Advanced Setting** is the fifth (5<sup>th</sup>) item of the Main Menu, mainly to provide advanced monitoring and control related settings.

Advanced Setting provides virtual device function to allow users to simulate various devices with the real I/O. There is one advanced setting function “PID Operation” and the description is listed on the page of the Main Menu. It will support more functions in the future. This chapter will introduce the function items and setting parameters.



The setting procedure for the UA series controllers is to set up from the left to the right of the main menu functions. User can find the procedure information in the following chapters.

[3.3 Function Setting Procedure](#)

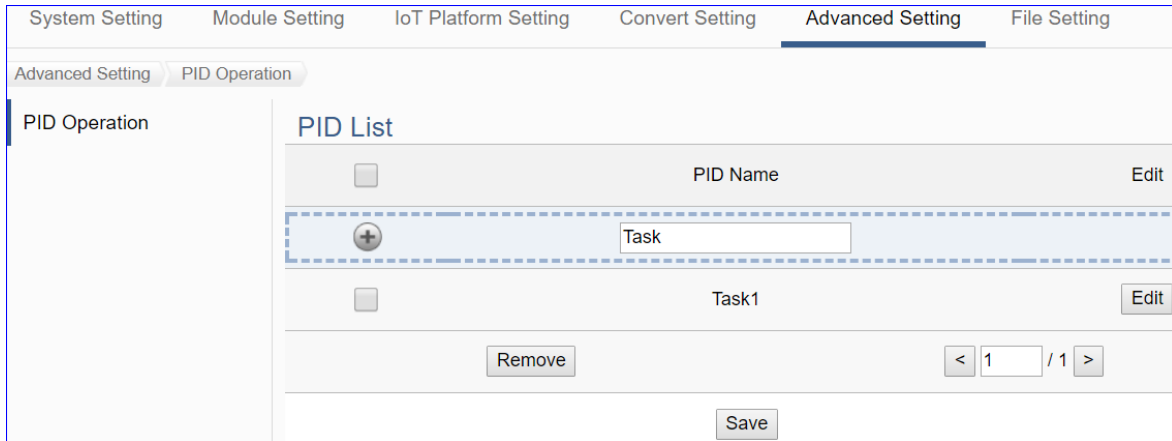
[2. Quick Start](#)

[4. Function Wizard](#)

About the Web UI login information and the UI environment, please refer to [3. Web UI Login and Enviroment Overview](#).

## 9.1. PID Operation

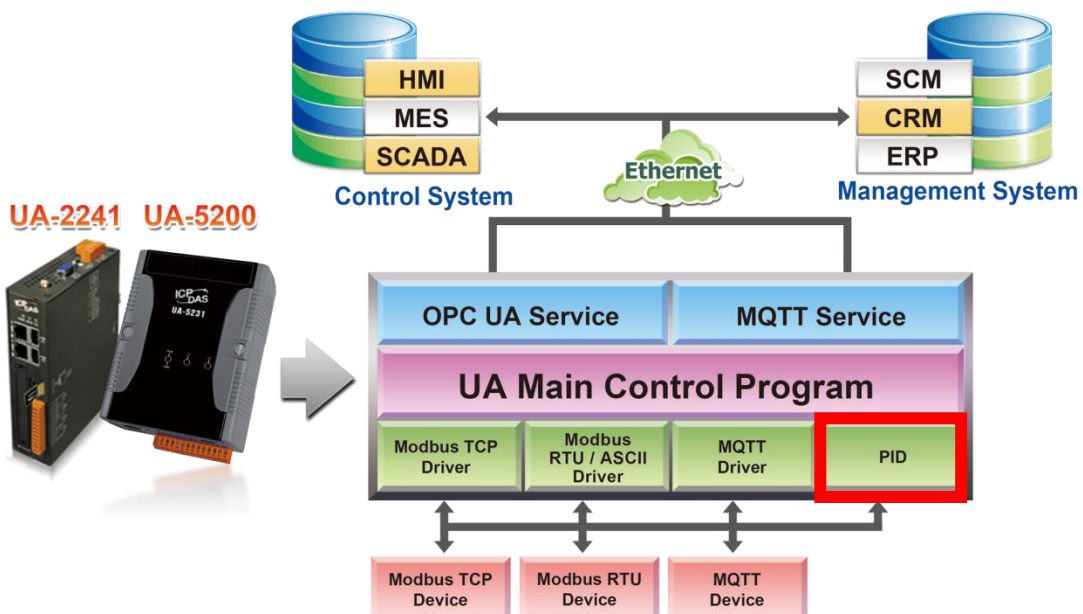
This page is about the virtual device function to allow users to simulate various devices with the real I/O by using the tuning function of PID operation.



PID (Proportional-Integral-Derivative) control is the most widely used in industrial control systems. A regulator which is controlled in accordance with Proportional, Integral and Derivative is called PID control for short, also called PID regulator. When the user cannot fully grasp or measure parameters of the control system, the PID regulator is the best solution.

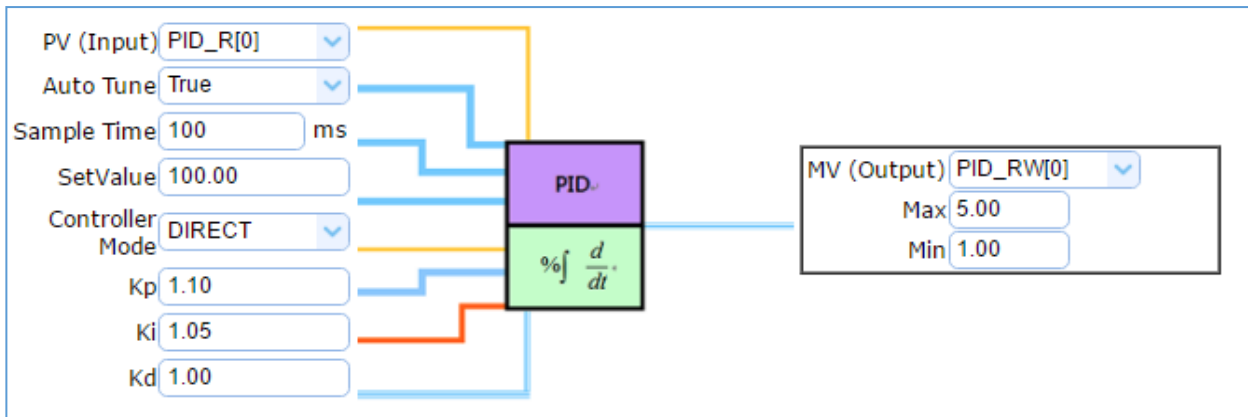
The PID controller is a common feedback loop component in industrial control applications. The controller compares the collected data with a reference value and then uses this difference to calculate a new input value whose purpose is to allow the system data to reach or remain at the reference value.

### Function Diagram:





**PID Operation Solution Example:**



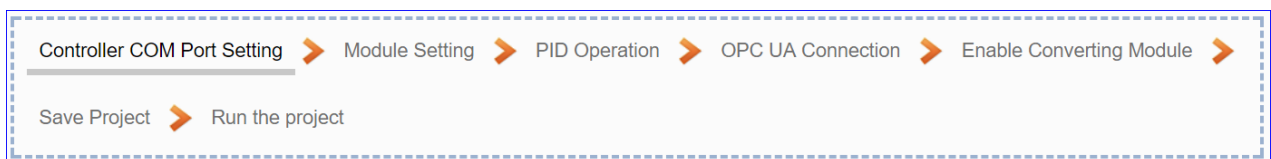
In the PID Operation function, UA controller collects the module’s data to operate via the feedback loop component of PID control. The controller compares the collected data with a reference value and then uses this difference to calculate a new input value whose purpose is to allow the system data to reach or remain at the reference value.

The setting steps of the PID Operation are as below. The descriptions for the steps setting please refer to Section 4.3 “PID” items in the Function Wizard.

**[Step Box] of [PID Operation] :**



**[Step Box] of [PID Operation + OPC UA Conversion] :**



This section will introduce the function items and setting parameters of the PID Operation.

**PID List**

<input type="checkbox"/>	PID Name	Edit
<input style="border: none; background: none; width: 20px; height: 20px; border-radius: 50%;" type="button" value="+"/>	Task <input style="width: 100px;" type="text"/>	
<input type="checkbox"/>	Task1	<input type="button" value="Edit"/>
<input type="button" value="Remove"/>		<input type="button" value="&lt;"/> <input style="width: 20px;" type="text" value="1"/> <input type="button" value="/"/> <input style="width: 20px;" type="text" value="1"/> <input type="button" value="&gt;"/>
<input type="button" value="Save"/>		

<b>Advanced Setting &gt; PID Operation &gt; PID List</b>	
PID Name	PID name, user can define, e.g. Task1. Default: Task.
<input style="border: none; background: none; width: 20px; height: 20px; border-radius: 50%;" type="button" value="+"/>	Click to add a new PID Task.
Edit / Remove	Click [Edit] can set the PID content. Click the left box and [remove] can delete the PID list.
<input type="button" value="&lt;"/> <input style="width: 20px;" type="text" value="1"/> <input type="button" value="/"/> <input style="width: 20px;" type="text" value="1"/> <input type="button" value="&gt;"/>	The page number of the PID list: Current page / Total pages. Click < or > to go to the previous or next page.
Save	Click to save the setting of this page.

Click [Edit] button to enter the [Content Settings] page:

**Content Settings**

PID Name	Task1
----------	-------

<b>Advanced Setting &gt; PID Operation &gt; Content Settings</b>	
PID Name	PID name, user can define, e.g. Task1. Default: Task.

Input Item	
Module selection	Type : <input type="text"/> <small>Please select the module type.</small>
	No. : <input type="text"/> <small>Please select the number. When no option is available, add a module.</small>
	Name : <input type="text"/>
Variable selection	Attribute <input type="text"/> <small>Please select item.</small>
	Type : <input type="text"/> <small>Please select item.</small>
	Name : <input type="text"/> <small>Please select name. When there is no option, add the variables in the module.</small>
Auto Tune	<input checked="" type="checkbox"/> Enabled
Sample Time(ms)	<input type="text" value="500"/>
Setpoint	<input type="text" value="0"/>
Controller Mode	<input type="text" value="DIRECT"/>
Kp	<input type="text" value="1"/>
Ki	<input type="text" value="1"/>
Kd	<input type="text" value="1"/>

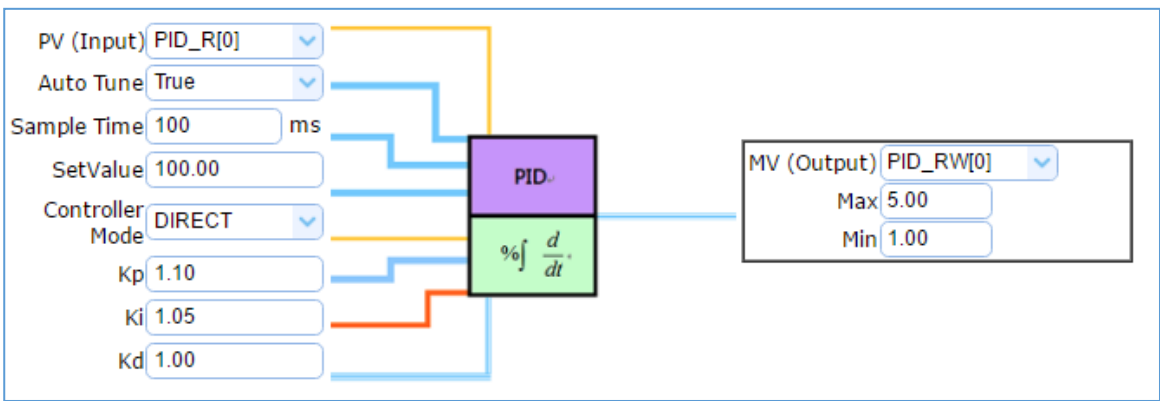
Advanced Setting > PID Operation > Input Item	
Module selection	Choose a predefined module for input data of the PID. Select the type, number and name of the input module. If no option is available, add a new module.
Variable selection	Choose a predefined float variable as the input parameter for PID operation. Select the attribute, type and name of the float variable.
Auto Tune	Enable: Auto-tuning PID parameters for your system. Default: check. Unenable: Tuning PID parameters manually, e.g. Kp, Ki, Kd.
Sample Time (ms)	Set the sampling time. (Unit: ms) Default: 500 ms.
Setpoint	The target value for PID control. Default: 0.
Controller Mode	DIRECT: Set it as positive output value. Default: DIRECT. REVERSE: Set it as reverse output value.
Kp	Set the Proportional gain. Default: 1.
Ki	Set the Integral gain. Default: 1.
Kd	Set the Derivative gain. Default: 1.

**Output Item**

Module selection	Type :	<input type="text"/>	Please select the module type.
	No. :	<input type="text"/>	Please select the number. When no option is available, add a module.
	Name :	<input type="text"/>	
Variable selection	Attribute	<input type="text"/>	Please select item.
	Type :	<input type="text"/>	Please select item.
	Name :	<input type="text"/>	Please select name. When there is no option, add the variables in the module.
Max	<input type="text" value="0"/>		
Min	<input type="text" value="0"/>		

Advanced Setting > PID Operation > Output Item	
Module selection	Choose a predefined module for output data of the PID. Select the type, number and name of the input module. If no option is available, add a new module.
Variable selection	Choose a predefined float variable as the output parameter for PID operation. Select the attribute, type and name of the float variable.
Max	Set the upper-limit value for the variable. Default: 0.
Min	Set the lower-limit value for the variable. Default: 0.
OK	Click to save the settings of the page and back to the PID list page.

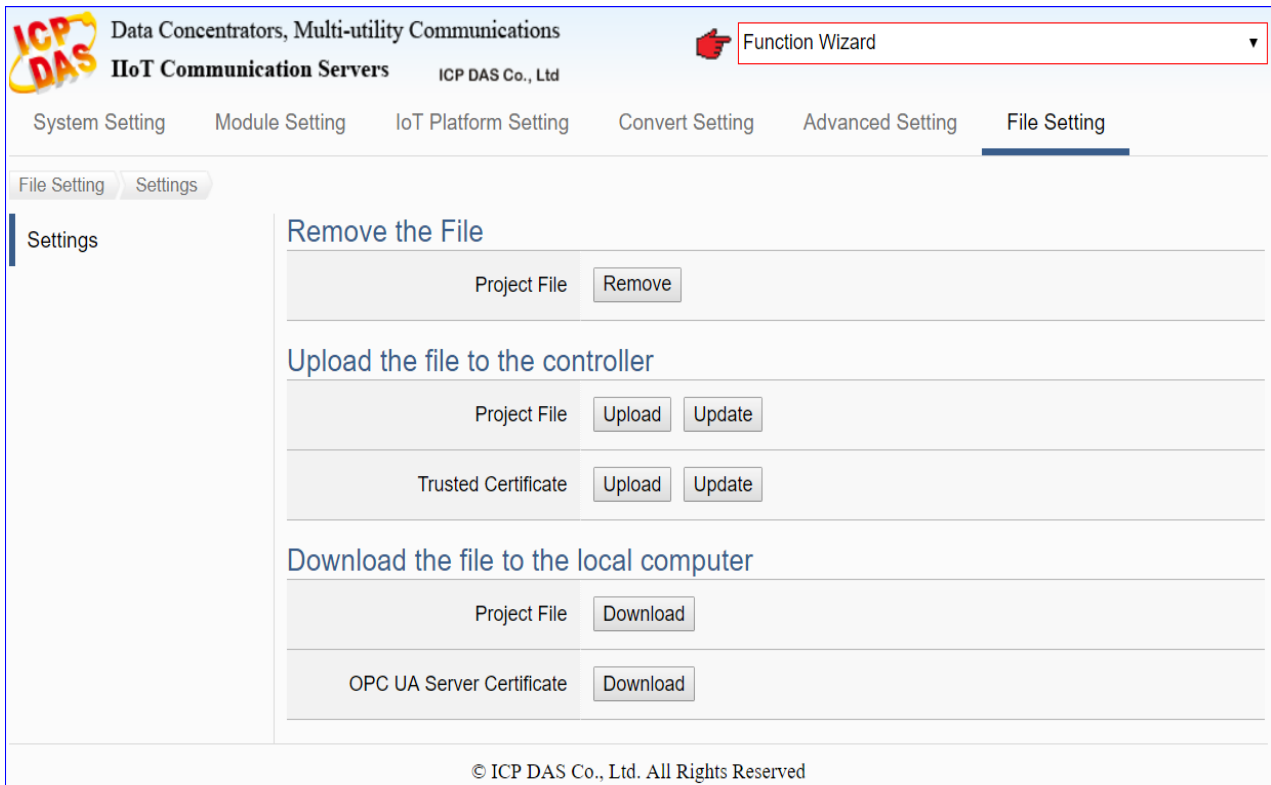
**PID Operation Solution Example:**



# 10. File Setting

**File Setting** is the last (6<sup>th</sup>) item of the Main Menu, mainly to provide the settings about the files, such as remove, update, upload and download the files of the project and certificate.

File Setting provides one sub-menu function “Settings” and has 3 setting items. This chapter will introduce the function items and setting parameters.



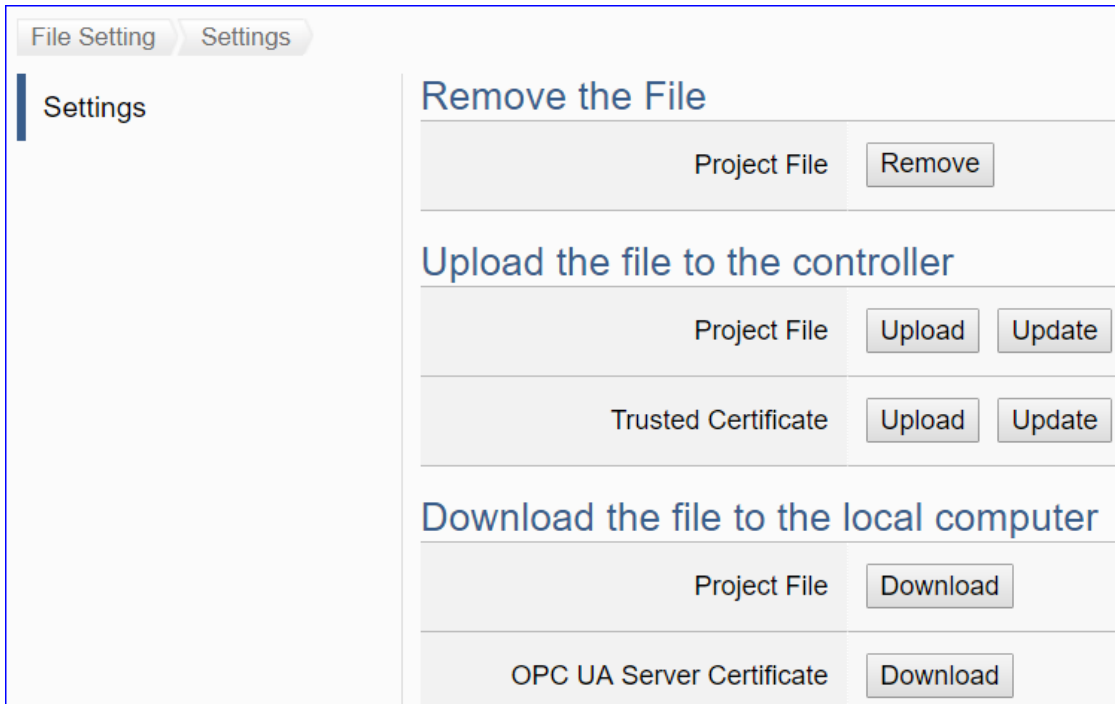
The setting procedure for the UA series controllers is to set up from the left to the right of the main menu functions. User can find the procedure information in the following chapters.

- [3.3 Function Setting Procedure](#)
- [2. Quick Start](#)
- [4. Function Wizard](#)

About the Web UI login information and the UI environment, please refer to [3. Web UI Login and Enviroment Overview](#).

## 10.1. Settings

This page provides 3 setting items: Remove the file, Upload the file to the controller, and Download the file to the local computer.



File Setting > Settings > Remove the File	
Project File	Click [Remove] to delete all project settings current in the UA series controller.
File Setting > Settings > Upload the file to the controller	
Project File	<p><b>Upload:</b> Upload the project with all Web UI settings to the UA series controller. (Exection name of the project file: ".tar")</p> <p><b>Update:</b> Update and run the project file that uploaded into the controller.</p>
Trusted Certificate	<p><b>Upload:</b> Upload the Trusted Certificate file to the UA series controller.</p> <p><b>Update:</b> Update and run the Trusted Certificate file that uploaded into the controller.</p>
File Setting > Settings > Download the file to the local computer	
Project File	<b>Download:</b> Download the project with all Web UI settings to the current computer. (Exection name of the project file: ".tar")
OPC UA Server Certificate	<b>Download:</b> Download the OPC UA Server Certificate file to the current using computer.

# 11. Factory Setting Recovering and Middleware Updating

This chapter will introduce the settings by hardware Rotary Switch, including “Factory Setting Recovering” and “Middleware Updating” that supported since Version 1.0.0.3.

## 11.1. Recovering to Factory Setting (Rotary Switch: 8)

Turn the Rotary Switch of UA-5200 series to “8” can recover to the factory setting.

The steps:

1. Power off the UA-5200 hardware, and turn the Rotary Switch to “8”.



2. Reboot the UA-5200 and wait a long buzzer sound that means of doing the recovering.
3. Wait about five minutes until **two** long buzzer sounds, and then turn the Rotary Switch to “0”.
4. Reboot the UA-5200 again, and the system will recover to the factory settings.

Factory Default Settings of UA-5200			
Network	IP	192.168.255.1	Assign UA-5200 a new IP setting according to your case.
	Netmask	255.255.0.0	
	Gateway	192.168.1.1	
OS Account	Username	root	After login, change your password ASAP. (Refer to <a href="#">Section 5.4</a> )
	Password	icpdas	
Web UI Account	Username	root	
	Password	root	

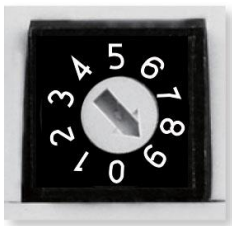
## 11.2. Updating Middleware via USB (Rotary Switch: 9)

Turn the Rotary Switch of UA-5200 series to “9” can update the Middleware version via USB.

**Note:** After the system version is updated, only the last network environment settings (IP, Netmask and Gateway) of the UA series controller will be retained and the rest will be factory recovered.

The steps:

1. Power off the UA-5200 hardware, and turn the Rotary Switch to “9”.



2. Download the Middleware package file of the UA-5200 hardware corresponding model. The download website: <http://icpdas.com.tw/pub/cd/UA-5000/middleware>
3. Save the Middleware package file into an empty FAT32 format USB drive and put to the UA-5200 USB port.
4. Reboot the UA-5200 and wait a long buzzer sound that means of doing the version updating.
5. Wait about five minutes until **two** long buzzer sounds, and then turn the Rotary Switch to “0”.

**Note:**

\* If the buzzer makes 2 short beeps, it means the USB is not connected properly.

Please check and connect the USB again.

\* If the buzzer makes 4 short beeps, it indicates that the installation process is error, please reinstall it. Check if the network cable is properly connected and the installation file is damaged.

6. Reboot the UA-5200 again, and the system will update to the version of the package file.

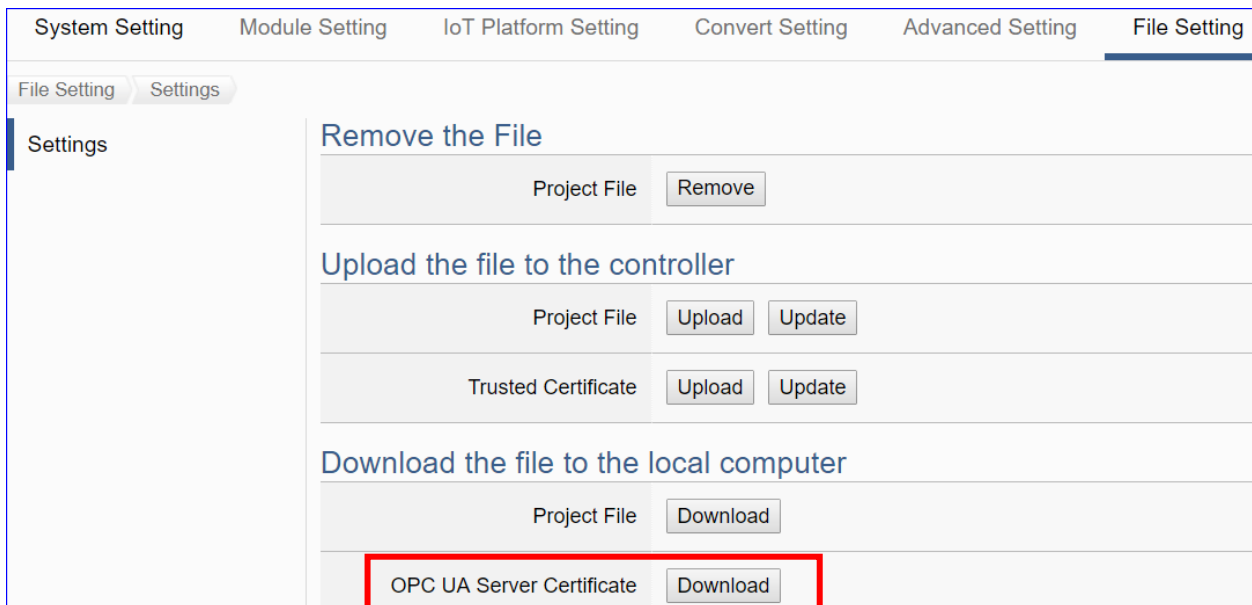


## 12. Security Certificate: Download/Upload/Update

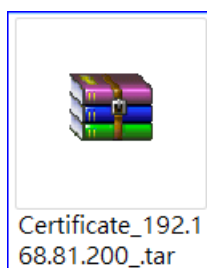
This chapter introduces the security certificate for the UA series controller, and the process to download, upload or update the certificate.

The communication security of the UA series controller, in addition to providing username / password protection mechanism, the SSL/TLS (Secure Socket Layer / Transport Layer Security Transport Layer Security) secure communication mechanism, and also provides the OPC UA trust certificate to protect data transmission security. The OPC UA is secure by default, encryption enabled, and uses advanced certificate handling which includes authentication, authorization, confidentiality and Integrity.

### 12.1. Download the Certificate of UA Controller

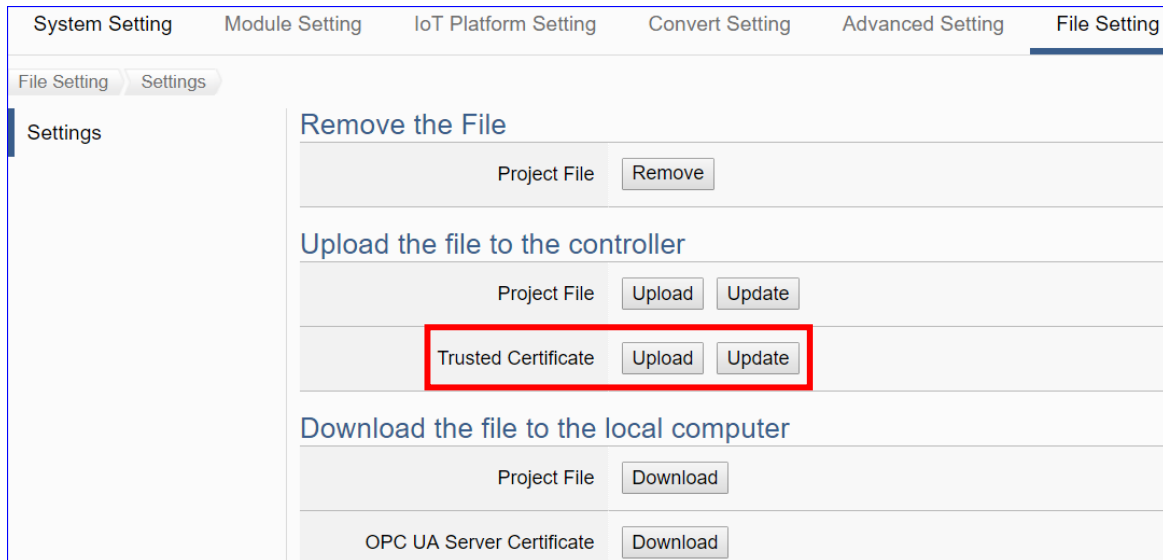


1. Click the main menu [File Setting > Settings > Download the file to the local computer – OPC UA Server Certificate] and then click on the button [Download].
2. Save the OPC UA Server certificate file to your designated folder. The downloaded certificate file (\*.tar) of the UA series controller looks similar to the figure below.



## 12.2. Upload/Update the Certificate to UA Controller

The user can store trusted certificates of the OPC UA client or the MQTT Broker from other device into the UA series controller project for setting up security communications.



1. Get the trusted certificates from OPC UA Client or MQTT Client and save in the PC.
2. Click the main menu [File Setting > Settings > Upload the file to the controller – Trusted Certificate] and click on the button [Upload]. Then select the certificate designated folder to open the file. The certificate will be uploaded to the UA controller.
3. Click the button [Update], then UA system can exchange the certificate authentication.

### Notes for OPC UA Client Certificate:

- The supported name is “\*.der”. The sub-file name must be “.der”, while the file name can follow the user need.
- The supported code format is “DER”.

### Notes for MQTT Client Certificate:

- The supported name is “ca.crt”. The file name must be “ca”, and the sub-file name must be “.crt”.
- The supported code format is “PEM”.

# Appendix A. MQTT JSON Format of the UA Series

## MQTT JSON Example & Format Descriptions:

```
{
  "Variable" : [ {
    "Name" : "Bool_R[0]",
    "Attribute" : "R",
    "Datatype" : "Bool",
    "Value" : 0,
    "Quality" : "Uncertain"
  }, {
    "Name" : "Short_R[0]",
    "Attribute" : "R",
    "Datatype" : "Int16",
    "Value" : 0,
    "Quality" : "Uncertain"
  }, {
    "Name" : "Short_R[1]",
    "Attribute" : "R",
    "Datatype" : "Int16",
    "Value" : 0,
    "Quality" : "Uncertain"
  }, {
    "Name" : "Short_R[2]",
    "Attribute" : "R",
    "Datatype" : "Int16",
    "Value" : 0,
    "Quality" : "Uncertain"
  }, {
    "Name" : "Short_RW[2]",
    "Attribute" : "RW",
    "Datatype" : "Int16",
    "Value" : 0,
    "Quality" : "Uncertain"
  }
  ]
}
```

Name	Descriptions
Variable	The array name of JSON. Its structure includes several member data as below.
Name	The member name of the array element
Attribute	The member attribute of the array element: "R" : can read "W" : can write "RW" : can read and write
Datatype	The member's data type of the array element: "Bool" "Int8" "UInt8" "UInt16" "Int16" "UInt32" "Int32" "UInt64" "Int64" "Float" "Double" "String"
Value	The member's current value of the array element
Quality	The member's current status of the array element: "Uncertain" "Good" "Bad"

## Appendix B. Technical Reference Websites

- OPC UA

<https://opcfoundation.org/>

- MQTT

<http://mqtt.org/>

- Modbus

<http://modbus.org/>