



# UA-2200 / UA-5200 User Manual

## V6.5, 2021/09

### IIoT Communication Server



**UA-2241M**



**UA-2241MX-4GE**  
**UA-2241MX-4GC**



**UA-5231**



**UA-5231M**



**UA-5231M-4GE**  
**UA-5231M-4GC**  
**UA-5231M-3GWA**

Technique: Sun Chen & Tim Chen; Translation & Edition: Eva Li;  
R&D Dept., ICP DAS CO., LTD.  
Latest edited by: Eva Li; V6.5, 09/2021

## 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 [service@icpdas.com](mailto:service@icpdas.com) .

For more product information, please refer to <https://www.icpdas.com> .

# Table of Contents

<b>Table of Contents</b> .....	<b>3</b>
<b>Document Version Modify List:</b> .....	<b>10</b>
<b>1. UA-2200/UA-5200: IIoT Communication Server</b> .....	<b>13</b>
1.1 Introduction.....	13
1.2 Features .....	14
1.3 Specifications.....	16
Hardware Specifications: UA-2200 Series.....	16
Hardware Specifications: UA-5200 Series.....	18
Software Specifications: UA-2200 and UA-5200 Series.....	20
1.4 Appearance.....	22
1.5 Pin Assignment .....	26
UA-5200 Series.....	26
Versions shipped before 2021/09 .....	26
Versions shipped after 2021/09 .....	26
Versions shipped between 2021/05 ~ 2021/09 (Notice).....	26
UA-2200 Series.....	27
1.6 Dimensions .....	28
<b>2. Quick Start 1: Hardware/Network Connection</b> .....	<b>30</b>
2.1 Hardware Connection .....	30
2.1.1 Preparations for Devices.....	30
2.1.2 Hardware Wiring .....	30
2.2 Network Connection.....	31
2.2.1 Connection by Factory Default Settings (For New UA).....	32
2.2.2 Connection by Utility Searching .....	35
<b>3. Quick Start 2: Web UI / Setting Steps</b> .....	<b>39</b>
3.1 Web UI Environment Overview .....	40
3.2 Setting Steps for Project / List.....	41
<b>4. Function Wizard: Project Quick Setup &amp; Examples</b> .....	<b>44</b>
4.1 Module Communication Conversion.....	45
4.1.1 Function Wizard: Modbus / OPC UA (RTU Example, Use M-7055D) .....	46

Step 1. Controller COM Port Setting .....	48
Step 2. Module Setting .....	49
Step 3. OPC UA Certificate .....	57
Step 4. OPC UA Connection .....	58
Step 5. Enable Converting Module .....	59
Step 6. Save Project.....	61
Step 7. Run the Project .....	61
4.1.2 Function Wizard: MQTT / OPC UA (Use MQ-7255M).....	62
Step 1. MQTT Certificate .....	64
Step 2. MQTT Broker Setting .....	65
Step 3. Module Setting .....	68
Step 4. OPC UA Certificate .....	72
Step 5. OPC UA Connection .....	73
Step 6. Enable Converting Module .....	74
Step 7. Save Project.....	75
Step 8. Run the Project .....	75
4.1.3 Function Wizard: EIP / OPC UA (Use EIP-2060).....	76
Step 1. Module Setting .....	78
Step 2. OPC UA Certificate .....	81
Step 3. OPC UA Connection .....	82
Step 4. Enable Converting Module .....	83
Step 5. Save Project.....	84
Step 6. Run the Project .....	84
4.1.4 Function Wizard: Modbus / MQTT (TCP Example, Use DL-302).....	85
Step 1. Module Setting .....	87
Step 2. MQTT Certificate.....	93
Step 3. MQTT Broker Setting .....	94
Step 4. Enable Converting Module .....	97
Step 5. Save Project.....	100
Step 6. Run the Project .....	100
4.1.5 Function Wizard: EIP / MQTT (Use EIP-2060) .....	101
Step 1. Module Setting .....	103
Step 2. MQTT Certificate.....	106
Step 3. MQTT Broker Setting .....	107
Step 4. Enable Converting Module .....	110
Step 5. Save Project.....	111
Step 6. Run the Project .....	111
4.1.6 Function Wizard: Modbus / MQTT JSON (RTU Example, Use M-7055D) .....	112

Step 1. Controller COM Port Setting.....	114
Step 2. Module Setting.....	115
Step 3. MQTT Certificate.....	121
Step 4. MQTT Broker Setting.....	122
Step 5. MQTT Group Setting.....	125
Step 6. Apply Connection & Enable Converting Module.....	128
Step 7. Save Project.....	129
Step 8. Run the Project.....	129
4.2 Module Connecting to Azure.....	130
4.2.1 Function Wizard: Modbus TCP / Azure (Use DL-302).....	131
Step 1. Module Setting.....	133
Step 2. MQTT Certificate.....	139
Step 3. Azure Setting.....	140
Step 4. Apply Connection & Enable Converting Module.....	142
Step 5. Save Project.....	143
Step 6. Run the Project.....	143
4.3 Data Log.....	144
4.3.1 Function Wizard: Modbus / Local Data Logger (RTU Example, tM-AD4P2C2 + DL-302).....	145
Step 1. Controller COM Port Setting.....	147
Step 2. Module Setting.....	150
Step 3. Local Data Logger.....	158
Step 4. Module Data logger.....	159
Step 5. Save Project.....	160
Step 6. Run the Project.....	160
Get the Log File of the Local Data Logger:.....	161
Example of the CVS file and fields for the local data log file:.....	162
4.3.2 Function Wizard: Modbus / Local Data Logger (TCP Example, DL-302).....	163
Step 1. Module Setting.....	165
Step 2. Local Data Logger.....	171
Step 3. Module Data logger.....	172
Step 4. Save Project.....	173
Step 5. Run the Project.....	173
Get the Log File of the Local Data Logger:.....	174
Example of the CVS file and fields for the local data log file:.....	175
4.3.3 Function Wizard: Modbus / MS SQL (RTU Example, tM-AD4P2C2 + DL-302). .....	176
Step 1. Controller COM Port Setting.....	178
Step 2. Module Setting.....	181

Step 3. MS SQLL Connection Setting.....	189
Step 4. Apply Connection & Enable Recording Module .....	191
Step 5. Save Project.....	192
Step 6. Run the Project .....	192
MS SQL Remote Database Example Descriptions: .....	193
4.3.4 Function Wizard: Modbus / MS SQL (TCP Example, Use DL-302) .....	195
Step 1. Module Setting .....	197
Step 2. MS SQL Connection Setting.....	203
Step 3. Apply Connection & Enable Recording Module .....	205
Step 4. Save Project.....	206
Step 5. Run the Project .....	206
MS SQL Remote Database Example Descriptions: .....	207
4.3.5 Function Wizard: Modbus / MySQL(MariaDB) (RTU Example, use M-7026)....	209
Step 1. Controller COM Port Setting.....	211
Step 2. Module Setting .....	212
Step 3. MySQL/MariaDB Connection Setting .....	218
Step 4. Apply Connection & Enable Recording Module .....	220
Step 5. Save Project.....	221
Step 6. Run the Project .....	221
MySQL/MariaDB Remote Database Example Descriptions:.....	222
4.3.6 Function Wizard: Modbus / MySQL(MariaDB) (TCP Multi-Module Example) ...	226
Step 1. Module Setting .....	228
Step 2. MySQL/MariaDB Connection Setting .....	237
Step 3. Apply Connection & Enable Recording Module .....	239
Step 4. Save Project.....	240
Step 5. Run the Project .....	240
MySQL/MariaDB Remote Database Example Descriptions:.....	241
4.4 PID.....	245
4.4.1 Function Wizard: PID Operation (Use AIO Module M-7026).....	246
Step 1. Controller COM Port Setting.....	247
Step 2. Module Setting .....	248
Step 3. PID Operation .....	253
Parameters Descriptions for Input Item of PID Operation .....	254
Setting Example for Input Item of PID Operation .....	255
Parameters Descriptions for Output Item of PID Operation .....	256
Setting Example for Output Item of PID Operation .....	257
Step 4. Save Project.....	258
Step 5. Run the Project .....	258

4.5 APP Message Notify .....	259
4.5.1 Function Wizard: IFTTT Condition Trigger (Line, Twitter) (Use M-7055D) .....	260
Step 1. Controller COM Port Setting.....	261
Step 2. Module Setting .....	262
Step 3. IFTTT Condition Trigger .....	268
Step 4. Save Project.....	273
Step 5. Run the Project .....	273
Step 6. I/O Status .....	273
<b>5 Main Menu: Parameter Settings .....</b>	<b>275</b>
5.1 Main Menu: System Setting .....	275
5.1.1 Controller Service Setting .....	276
5.1.2 Time Setting.....	277
5.1.3 Network Setting.....	279
** Network Setting (Mobile Network):.....	282
5.1.4 Account Setting .....	284
5.1.5 Boot .....	285
5.1.6 COM Port Interface Setting .....	286
5.2 Main Menu: Module Setting.....	287
5.2.1 Modbus RTU (Master).....	288
** Scaling.....	293
** Bitwise .....	294
5.2.2 Modbus TCP (Master).....	295
** Scaling.....	300
** Bitwise .....	301
5.2.3 Modbus ASCII (Master).....	302
5.2.4 MQTT Module .....	307
5.2.5 EtherNet/IP ICP DAS Module.....	311
5.2.6 Internal Module .....	315
5.3 Main Menu: IoT Platform Setting.....	319
5.3.1 MQTT Local Broker.....	320
5.3.2 MQTT Remote Broker .....	321
5.3.3 MQTT Group Connection.....	325
5.3.4 MQTT Connection - Microsoft Azure Platform .....	329
5.3.5 OPC UA Connection - Local Server .....	332

5.4 Main Menu: Convert Setting.....	333
5.4.1 OPC UA and Modbus RTU/ASCII Conversion.....	335
5.4.2 OPC UA and Modbus TCP Conversion .....	338
5.4.3 OPC UA and MQTT Conversion .....	341
5.4.4 OPC UA and EtherNet/IP Conversion .....	344
5.4.5 OPC UA 與 Internal 轉換設定.....	347
5.4.6 MQTT and Modbus RTU/ASCII Conversion.....	349
5.4.7 MQTT and Modbus TCP Conversion .....	353
5.4.8 MQTT and EtherNet/IP Conversion.....	357
5.4.9 MQTT JSON and Modbus RTU/ASCII Conversion.....	361
5.4.10 MQTT JSON and Modbus TCP Conversion .....	364
5.5 Main Menu: Advanced Setting.....	367
5.5.1 PID Operation .....	368
5.5.2 IFTTT Condition Trigger .....	373
5.5.3 Data Logger: Local Data Logger .....	380
CVS local data log file: fields and example .....	382
5.5.4 Data Logger: MS SQL.....	383
MS SQL Remote Database Example Descriptions: .....	386
5.5.5 Data Logger: MySQL / MariaDB.....	388
MySQL/MariaDB Remote Database Example Descriptions:.....	391
5.6 Main Menu: Logger Setting .....	395
5.6.1 Local Data Logger: RTU / TCP Module (Master).....	396
5.6.2 MS SQL: RTU / TCP Module (Master) .....	399
5.6.3 MySQL / MariaDB: RTU / TCP Module (Master) .....	402
5.7 Main Menu: I/O Status .....	405
5.8 Main Menu: File Setting .....	406
5.8.1 Project File.....	407
5.8.2 OPC UA Certificate .....	408
5.8.3 MQTT Certificate.....	410
5.8.4 Log File Download .....	412
5.8.5 Firmware Update .....	413
<b>6. Factory Setting Recovering and Firmware Updating.....</b>	<b>414</b>
6.1 Recovering to Factory Setting (Rotary Switch: 8).....	414
6.2 Updating Firmware A – via Web UI of UA.....	415



6.3 Updating Firmware B - via USB (Rotary Switch: 9) ..... 416

6.4 Updating Firmware C - via MicroSD Card ..... 417

**7 Security Certificate: Download / Upload.....422**

    OPC UA Server Certificate management..... 422

    MQTT Client Certificate management..... 422

    7.1 Download the Certificate from UA Controller ..... 424

    7.2 Upload the Certificate to UA Controller..... 425

        7.2.1 OPC UA Certificate ..... 425

        7.2.2 MQTT Certificate..... 426

**Appendix A. MQTT JSON Format of the UA Series.....427**

**Appendix B. Protocol Technical Reference.....428**

**Appendix C. LED Indicators.....429**

**Appendix D. Mounting the XV-board for UA-2241M.....432**

## Document Version Modify List:

Version	Description
V6.5	<p>Date: 2021/03</p> <ol style="list-style-type: none"> <li>1. Hardware: UA-5231 Series <b>Flash</b> 512MB expand to eMMC <b>8GB</b> (CH1) Hardware: UA-5231 Series <b>tty02</b> become 2500VDC Isolated. All COM Ports of UA series has Isolated. (CH1)</li> <li>2. Hardware: UA-2241M-4Gx change model name to <b>UA-2241MX-4Gx</b> (CH1)</li> <li>3. <b>Module Setting &gt; Modbus RTU: Add "Write Retry"</b> function setting (CH4.1, CH5.2.1)</li> <li>4. <b>Module Setting: Add "Internal Module"</b> function, to create virtual internal variables for reading and writing or as an intermediary to provide data exchange of communication protocols. (CH5.2.6 &amp; CH5.4.5, CH4.1) * User can add internal variables: max. 8 virtual modules, 100 tags (internal variables per module). * Protocol Communication Conversion: provide Internal to OPC UA (Server)</li> <li>5. <b>Convert Setting &gt; OPC UA: Add "Internal"</b> function, provide <b>OPC UA Client to Client</b> communication conversion. (Add CH5.4.5)</li> <li>6. <b>Advanced Setting &gt; Data Logger &gt; MS SQL: Add "Log Mode"</b> (Cycle, Data Change) &amp; <b>"Date Time Format"</b> items (CH4.3, CH5.5.4)</li> <li>7. <b>Advanced Setting &gt; Data Logger &gt; MySQL/MariaDB: Add "Log Mode"</b> (Cycle, Data Change) &amp; <b>"Date Time Format"</b> items (CH4.3, CH5.5.5)</li> <li>8. <b>Logger Setting &gt; MS SQL / MySQL / MariaDB: Add MQTT Module</b> support (CH4.3, CH5.6.2 ~ CH5.6.3)</li> <li>9. <b>File Setting: Add Firmware Update</b>, to update the firmware from UI website. (Add CH5.8.5)</li> <li>10. Above 2 hardware items (1~2) update to CH1</li> <li>11. Above 7 software items (3~9) update to CH5. Previous Function demos lists in the CH4 , but the new function demos of this time will be listed to the UA series FAQ website: <a href="https://www.icpdas.com/en/faq/index.php?kind=326">https://www.icpdas.com/en/faq/index.php?kind=326</a> * <b>Module Communication Conversion &gt; Internal / OPC UA</b> * <b>Data Log &gt; MQTT / MS SQL</b> * <b>Data Log &gt; MQTT / MySQL(MariaDB)</b></li> <li>12. <b>Function Wizard &gt; Data Log</b>, add the multiple modules examples (CH4.3)</li> <li>13. Add Monitor and restart function: System Setting &gt; Controller Service Setting &gt; Run Project (CH5.1)</li> <li>14. To prevent OS permission passwords from being misused, remove following:             <ol style="list-style-type: none"> <li>a. Remove the Factory Default OS Account/PWD item of UA Series (CH2.2.1, 5.1.4, 6.1)</li> <li>b. Remove Appendix D. Changing OS Password (Appendix D)</li> </ol> </li> </ol>

<p>V6.4</p>	<p>Date: 2020/07</p> <ol style="list-style-type: none"> <li>1. Add 4G new Models: UA-2241M-4GE / UA-2241M-4GC</li> <li>2. Change UA-5000 Utility to UA Series Utility (for all UA series)</li> <li>3. Change the webpage links to the new web site.</li> <li>4. IFTTT is no longer support Facebook, delete the Facebook texts.</li> </ol>
<p>V6.3</p>	<p>Date: 2020/03</p> <ol style="list-style-type: none"> <li>1. The manual version number is updated to V6.3 and changed to follow the web interface version number, because the user manual has been greatly revised, including: adding a number of new features, software Firmware to Firmware, web interface update and new logo icon.</li> <li>2. The chapter structure is greatly modified: the chapters related to the menu setting CH5-CH12 are combined to CH5. New CH5: Main Menu: Parameter Settings. New CH4: Function Wizard: Project Quick Setup. Other CH: update data.</li> <li>3. New function: support new remote database <b>MySQL</b> and <b>MariaDB</b>. The original remote database changed the name to <b>MS SQL</b>. (CH4.3, CH5.5)</li> <li>4. New function: add new function <b>Scaling</b> and <b>Bitwise</b>. (CH4, CH5.2)</li> <li>5. New function: 4G Network Setting add <b>unlock the SIM card Pin#</b> (CH5.1.3)</li> <li>6. New function: add <b>Log File Download</b> (CH5.8)</li> <li>7. New/Modify function: <b>Certificate</b> management (CH4 new step, CH5.8 UI)</li> <li>8. Firmware update: combine 2 methods to one chapter. The microSD method move to CH6.3 together with the CH6.2 USB method. (CH6)</li> <li>9. Appendix D: update the LED indicators descriptions.</li> <li>10. Add Appendix F: Add Mounting the XV-board for UA-2241M</li> </ol>
<p>V4.7</p>	<p>Date: 2019/08</p> <ol style="list-style-type: none"> <li>1. CH1.4 UA-2241M supports one I/O expansion bus for optional XV-board. ex: optional XV511i to expand 4 RS-485 ports</li> <li>2. CH1.5 Add the I/O expansion into the appearance figure of UA-2241M.</li> <li>3. Add appendix G: Mounting the XV-board for UA-2241M</li> </ol>
<p>V4.6</p>	<p>Date: 2019/07</p> <ol style="list-style-type: none"> <li>1. UA-2241M supports ttyO2(RS-485) 2500 VDC isolated</li> <li>2. Cloud Support: Add MS Azure and Amazon Web Service, Delete IoTstar</li> <li>3. CH1 Introductions, Features, Functions: follow the new DS</li> <li>4. Add LAN2 settings (for UA-2200) and Note of System Setting &gt; Network</li> </ol>

	5. Appendix: change 3G/4G illustrators of LED Indicators
V4.5	<p>Data 2019/01</p> <ol style="list-style-type: none"> <li>1. Add new model: UA-2241M (Two Ethernet ports)</li> <li>2. Add new function: support EtherNet/IP protocol (Max. 50 EIP-2000)</li> <li>3. Add new function: Data Logger (Local Log, Remote Database)</li> <li>4. Add new function: support IoTstar cloud software (coming soon)</li> <li>5. Web UI supports auto-setting ICP DAS Modbus RTU/TCP and EIP modules</li> <li>6. Delete CDS setting in Azure connection</li> <li>7. Add Appendix F: Changing OS Password</li> </ol>
V4.3	<p>Date: 2018/07</p> <ol style="list-style-type: none"> <li>1. Add 4G new products: UA-5231M-4GE / UA-5231M-4GC (CH1, CH5)</li> <li>2. Modify introduction, features, function diagram and function features (CH1)</li> <li>3. Update specifications, appearance and dimensions (CH1)</li> <li>4. Add LED Indicators descriptions for UA Series (Appendix E.)</li> <li>5. Update Document Version List (Postscript)</li> </ol>
V4.2	<p>Date: 2018/04</p> <ol style="list-style-type: none"> <li>1. Add new 3G function: UA-5231M-3GWA (CH1~3)</li> <li>2. Add new function in the Function Wizard: (CH4) <ol style="list-style-type: none"> <li>a. Module Communication Conversion: Add MQTT / OPC UA</li> <li>b. Add "APP Message Notify" new type, and add new function "IFTTT Condition Trigger (Line, Facebook, Twitter)"</li> </ol> </li> <li>3. Add function: System Setting &gt; Network Setting &gt; Mobile Network (3G) (CH5)</li> <li>4. Add new function: Module Setting &gt; MQTT Module (CH6)</li> <li>5. Add new function: Convert Setting &gt; OPC UA &gt; MQTT (CH8)</li> <li>6. Add new function: Advanced Setting &gt; IFTTT Condition Trigger (CH9)</li> <li>7. Add new function menu: I/O Status (CH10)</li> <li>8. Modify: Recovering to Factory Setting (Rotary Switch: 8) (CH12)</li> <li>9. Add IFTTT Website Setting (Appendix C)</li> <li>10. Add Updating Middleware via MicroSD Card Manually (CH12.2 &amp; App. D)</li> <li>11. Add: Postscript: Document Version List</li> </ol>

# 1. UA-2200/UA-5200: IIoT Communication Server

This chapter introduces UA series and its functions, software/hardware specifications.

## 1.1 Introduction

**UA-2200/UA-5200 Series** is a series of **IIoT (Industrial IoT) Communication Server** for integrating the system and devices of IT and OT. UA features the IIoT Gateway function that allows users to access the remote I/O modules and controllers via Modbus TCP/RTU/ASCII, MQTT, and EtherNet/IP communication protocols. IIoT gateway function can also convert these I/O data to OPC UA or MQTT protocols for the needs of connecting with the MES, ERP, SCADA and Cloud services. Besides, UA features the Data Logger function that allows users to write the I/O data directly into the remote database, and save to the local file as the historical records. UA supports Cloud platform that can connect to Amazon AWS, Microsoft Azure or other Cloud platforms to send over the I/O data; and support Cloud logic service platform "IFTTT" which can connect many web APPs that allows users to receive first-hand notification messages through the most commonly used mobile APPs when an event triggered. UA Series enhances the networking and interoperability between IT and OT. Through UA series, users can easily deploy for Industrial IoT.

- **UA Series:**

<b>UA-2200 Series</b>		<b>UA-2241M</b>	<b>UA-2241MX-4GE UA-2241MX-4GC</b>
			
<b>UA-5200 Series</b>	<b>UA-5231</b>	<b>UA-5231M</b>	<b>UA-5231M-4GE UA-5231M-4GC UA-5231M-3GWA</b>
			

## 1.2 Features

### ■ Built-in OPC UA Server Service

Compliant with IEC 62541 Standard. Provides functions of Active Transmission, Transmission Security Encryption (SSL/TLS), User Authentication (X.509 Certificates / Account password), Communication Error Detection and Recovery, etc. to connect SCADA or OPC UA Clients. Allowed up to 8000 OPC UA tags and up to 20 sessions for the OPC UA Client connection.

### ■ Built-in MQTT Broker Service

MQTT Broker inside and compliant with MQTT V.3.1.1 protocol. Provides functions of IoT Active M2M Transmission, QoS (Quality of Service), Retain Mechanism, Identity Verification, Encryption, Last Will, MQTT Client Drivers, etc. The Broker can connect up to 400 MQTT Clients.

### ■ Save I/O Data Directly into Remote Database & Local Side LOG File

UA series can collect devices I/O status and then directly save into remote side SQL Database. UA series can also save I/O data into a CSV log file on the local side. Furthermore, users can set the time interval of which CSV file to generate and divide on the local side.

### ■ Support Logic Control IFTTT to Send Event Messages to LINE... APPs

UA can combine the IFTTT cloud platform functions and send messages over 460 Web APPs (such as Line, Twitter, etc.) when the special events occur. The device I/O change can be set to trigger the event of the IFTTT cloud service, and the IFTTT logic control (If This, Then That) will immediately let the pre-set Web Service (Such as LINE) send a message to one user or group to handle the event immediately.

### ■ Support Ethernet and Serial Communication Modules

- Under the Ethernet communication, UA supports Modbus TCP, MQTT and ICP DAS EtherNet/IP modules:
  - \* Up to 100 Modbus TCP Slave module connections
  - \* Up to 200 MQTT module connections
  - \* Up to 50 EtherNet/IP EIP-2000 module connections
- Under the Serial communication, UA provides 3 RS-232/RS-485 Serial ports to support Modbus RTU/ASCII modules:
  - \* Up to 32 Modbus RTU/ASCII Slave modules per COM
- Through the UA Web UI, users can quickly set up the modules and display the real-time status.

- **Provide Function Wizard Web UI for easily step-by-step setup**

The Web UI of UA provides a wizard-like “Step Box” in the Function Wizard area to guide user step-by-step to complete the project or function. It provides many items for setting the Communication Conversion, Azure Connecting, Function Configuration, PID Operation, Condition Trigger the APP Message Notification, and will be more. It will help users to set projects easily and quickly.

- **Support IoT Cloud Platforms Connection**

UA can actively connect to Amazon AWS, Microsoft Azure or other IoT Cloud platforms to send over the I/O data.

# 1.3 Specifications

## ● Hardware Specifications: UA-2200 Series

Model	UA-2241M	UA-2241MX-4GE	UA-2241MX-4GC
<b>Main Unit</b>			
CPU	ARM CPU, 1.0 GHz		
System Memory	DDR3 SDRAM 512 MB		
Non-Volatile Memory	FRAM 64 KB		
Storage	· Flash 512 MB · microSD socket with one 4 GB microSD card (support up to 32 GB microSDHC card)		
Real Time Clock	Provide second, minute, hour, date, day of week, month, year		
<b>Display</b>			
Signal	VGA (Analog RGB), reserved		
<b>LED Indicators</b>			
Status	PWR (Power), RUN (Running), L1, L2, L3	PWR (Power), RUN (Running), L1, L2, L3, 4G	PWR (Power), RUN (Running), L1, L2, L3, 4G
<b>COM Ports</b>			
Console Port	RS-232 (RxD, TxD and GND); Non-isolated		
ttyO2	RS-485 ( Data+, Data- ); 2500 VDC isolated		
ttyO4	RS-232 (RxD, TxD and GND); Non-isolated		
ttyO5	RS-485 ( Data+, Data- ); 2500 VDC isolated		
<b>Ethernet</b>			
Ports	2 x RJ-45, 10/100/1000 Based-TX ( Auto-negotiating, Auto MDI/MDI-X, LED indicators )		
<b>USB</b>			
Connector	2 x 2.0 host		
<b>HMI</b>			
Rotary Switch	1 x 10 Position (0 ~ 9)		
<b>Port Expansion</b>			
Type	1 Expansion Bus, for one optional XV-board (Support XV511i to expand 4 RS-485 ports)		



Model	UA-2241M	UA-2241MX-4GE	UA-2241MX-4GC
<b>Power</b>			
Input Range	+12 ~ +48 VDC		
Consumption	4.8 W	6.5 W	
<b>Mechanical</b>			
Casing	Metal		
Dimensions (mm)	35 x 167 x 119 (W x L x H)		
Installation	DIN-Rail		
<b>Environmental</b>			
Operating Temperature	-25 ~ +75°C		
Storage Temperature	-40 ~ +80°C		
Ambient Relative Humidity	10 ~ 90% RH (non-condensing)		
<b>3G System Wireless Communication</b>			
Data Transmission	-	<ul style="list-style-type: none"> <li>· DC-HSPA+ Download: Max. 42 Mbps; Upload: Max 5.76 Mbps</li> <li>· TD-SCDMA Download: Max. 4.2 Mbps; Upload: Max 2.2 Mbps</li> <li>· CDMA2000 EVDO Download: Max. 14.7 Mbps; Upload: Max 5.4 Mbps</li> </ul>	
Frequency Band	-	WCDMA 850/900/2100 MHz	WCDMA 900/2100 MHz TD-SCDMA 1900/2100 MHz CDMA2000 (BC0) 800 MHz
<b>4G System Wireless Communication</b>			
Data Transmission	-	Download Max 100 Mbps; Upload Max 50 Mbps	
Frequency Band	-	FDD LTE: B1/B3/B5/B7/B8/B20	FDD LTE: B1/B3/B8 TDD LTE: B38/B39/B40/B41

## ● Hardware Specifications: UA-5200 Series

Model	UA-5231	UA-5231M	UA-5231M-3GWA	UA-5231M-4Gx
<b>Main Unit</b>				
CPU	ARM CPU, 1.0 GHz			
System Memory	DDR3 SDRAM 512 MB			
Non-Volatile Memory	FRAM 64 KB			
Storage	Flash eMMC 8 GB			
	microSD socket with one 4 GB microSD card (support up to 32 GB microSDHC card)			
Real Time Clock	Provide second, minute, hour, date, day of week, month, year			
<b>Display</b>				
Signal	VGA (Analog RGB), reserved			
<b>LED Indicators</b>				
Status	PWR/RUN (Power/Running), L1, L2	PWR/RUN, L1, L2, 3G	PWR/RUN, L1, L2, 4G	
<b>COM Ports</b>				
Console Port	RS-232 (RxD, TxD and GND); Non-isolated			
ttyO2	RS-485 ( Data+, Data- ); 2500 VDC isolated			
ttyO4	RS-232 (RxD, TxD and GND); Non-isolated			
ttyO5	RS-485 ( Data+, Data- ); 2500 VDC isolated			
<b>Ethernet</b>				
Ports	1 x RJ-45, 10/100/1000 Based-TX ( Auto-negotiating, Auto MDI/MDI-X, LED indicators )			
<b>USB</b>				
Connector	1 x 2.0 host			
<b>HMI</b>				
Rotary Switch	1 x 10 Position (0 ~ 9)			
<b>Port Expansion</b>				
Type	-			
<b>Power</b>				

Model	UA-5231	UA-5231M	UA-5231M-3GWA	UA-5231M-4Gx
Input Range	+12 ~ +48 VDC			
Consumption	4.8 W		6.5 W	
<b>Mechanical</b>				
Casing	Plastic		Metal	
Dimensions (W x L x H) (mm)	91 x 132 x 52		117 x 126 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)			

Model	UA-5231M-3GWA	UA-5231M-4GE	UA-5231M-4GC
<b>3G System Wireless Communication</b>			
Data Transmission	<ul style="list-style-type: none"> <li>WCDMA / HSPA+ Download: Max. 14.4Mbps; Upload: Max 5.76Mbps</li> </ul>	<ul style="list-style-type: none"> <li>DC-HSPA+ Download: Max. 42 Mbps; Upload: Max 5.76 Mbps</li> <li>TD-SCDMA Download: Max. 4.2 Mbps; Upload: Max 2.2 Mbps</li> <li>CDMA2000 EVDO Download: Max. 14.7 Mbps; Upload: Max 5.4 Mbps</li> </ul>	
Frequency Band	<ul style="list-style-type: none"> <li>WCDMA 850/900/1900/2100 MHz</li> </ul>	<ul style="list-style-type: none"> <li>WCDMA 850/900/2100 MHz</li> </ul>	<ul style="list-style-type: none"> <li>WCDMA 900/2100 MHz</li> <li>TD-SCDMA 1900/2100 MHz</li> <li>CDMA2000 (BC0) 800 MHz</li> </ul>
<b>4G System Wireless Communication</b>			
Data Transmission	-	<ul style="list-style-type: none"> <li>Download Max 100 Mbps; Upload Max 50 Mbps</li> </ul>	
Frequency Band	-	<ul style="list-style-type: none"> <li>FDD LTE: B1/B3/B5/B7/B8/B20</li> </ul>	<ul style="list-style-type: none"> <li>FDD LTE: B1/B3/B8</li> <li>TDD LTE: B38/B39/B40/B41</li> </ul>

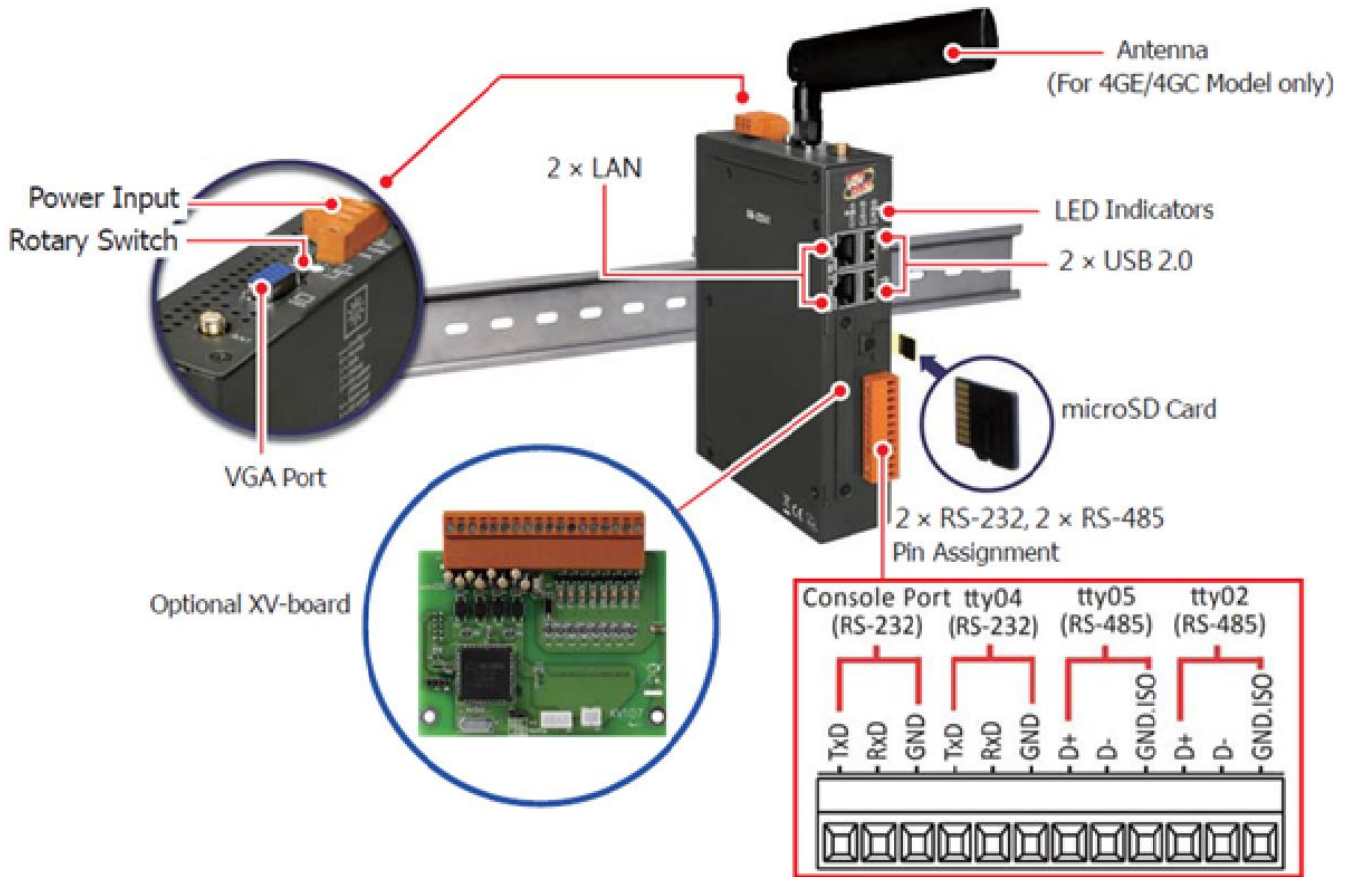
● **Software Specifications: UA-2200 and UA-5200 Series**

Model	UA-2200/UA-5200 Series Software Specifications
<b>OS</b>	
Linux	Linux Kernel 3.2.14
<b>Protocol</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>● UA-TCP UA-SC UA Binary</li> <li>● User Authentication: Username/Password, X.509 Certificate</li> <li>● Security Policy:                             <ul style="list-style-type: none"> <li>&gt; None</li> <li>&gt; Basic128Rsa15 (Sign / Sign &amp; Encrypt)</li> <li>&gt; Basic256 (Sign / Sign &amp; Encrypt)</li> </ul> </li> <li>● Recommend Max. 20 Client Sessions, and Max. 8000 Tags.</li> </ul>
MQTT Broker	Compliance with MQTT v3.1.1 protocol. Support MQTT message distribution management. Recommend Max. 400 Client Devices.
MQTT Client	Connect the MQTT Broker to read/control the devices supporting the MQTT protocol, or connect the MQTT Broker to externally read/control the devices supporting other protocols that linking with the UA series. (MQTT Ver. 3.1.1; TLS Ver. 1.2) Recommend Max. 200 Connections.
Modbus TCP Master	To read or control the devices that support standard Modbus TCP Slave protocol. Recommend Max. 100 modules.
Modbus RTU/ASCII Master	A max. of 3 ports: COM2, COM3, COM4 to connect other Modbus RTU Slave devices (e.g. M-7000). Recommend Max. 32 devices per port (32*3 port=96) for better communication quality.
EtherNet/IP Scanner	Support connect EIP-2000 series modules of ICP DAS. Recommend Max. 50 devices per UA.
<b>Service</b>	
Embedded Service	SFTP server, Web server, SSH
<b>Data Logger</b>	

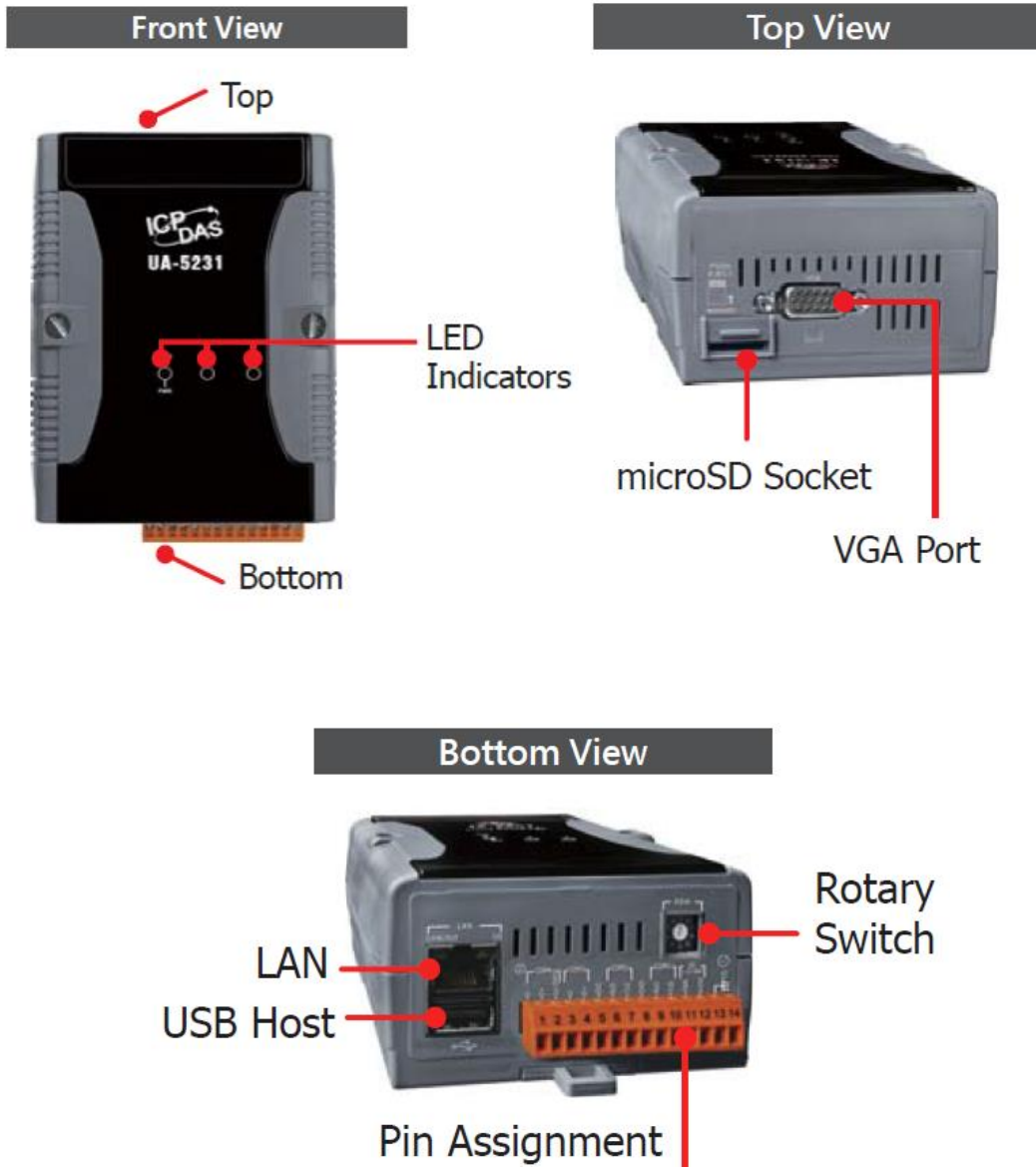
<b>Model</b>	<b>UA-2200/UA-5200 Series Software Specifications</b>
Local Data Logger	Record I/O data, and save to the local MicroSD card or SSD in CSV format.
Remote Database	Record I/O data, and send to the remote database of MS SQL / MySQL / MariaDB. Recommend Max. 1 Databases per Time, and Max. 1000 Tags.
<b>Function</b>	
PID Function	Combine the remote I/O devices for the PID logic control system.
<b>IoT Service Integration</b>	
Microsoft Azure	MQTT Service can connect to MS Azure IoT Hub for Cloud platform service.
Amazon Web Services	MQTT Service can connect to AWS IoT Core for Cloud platform service.
IBM Bluemix	MQTT Service can connect to IBM Bluemix for Cloud platform service.
IFTTT	Support Logic event sending to IFTTT Web platform. IFTTT Logic Trigger APP (Line, Twitter, Gmail ...)

# 1.4 Appearance

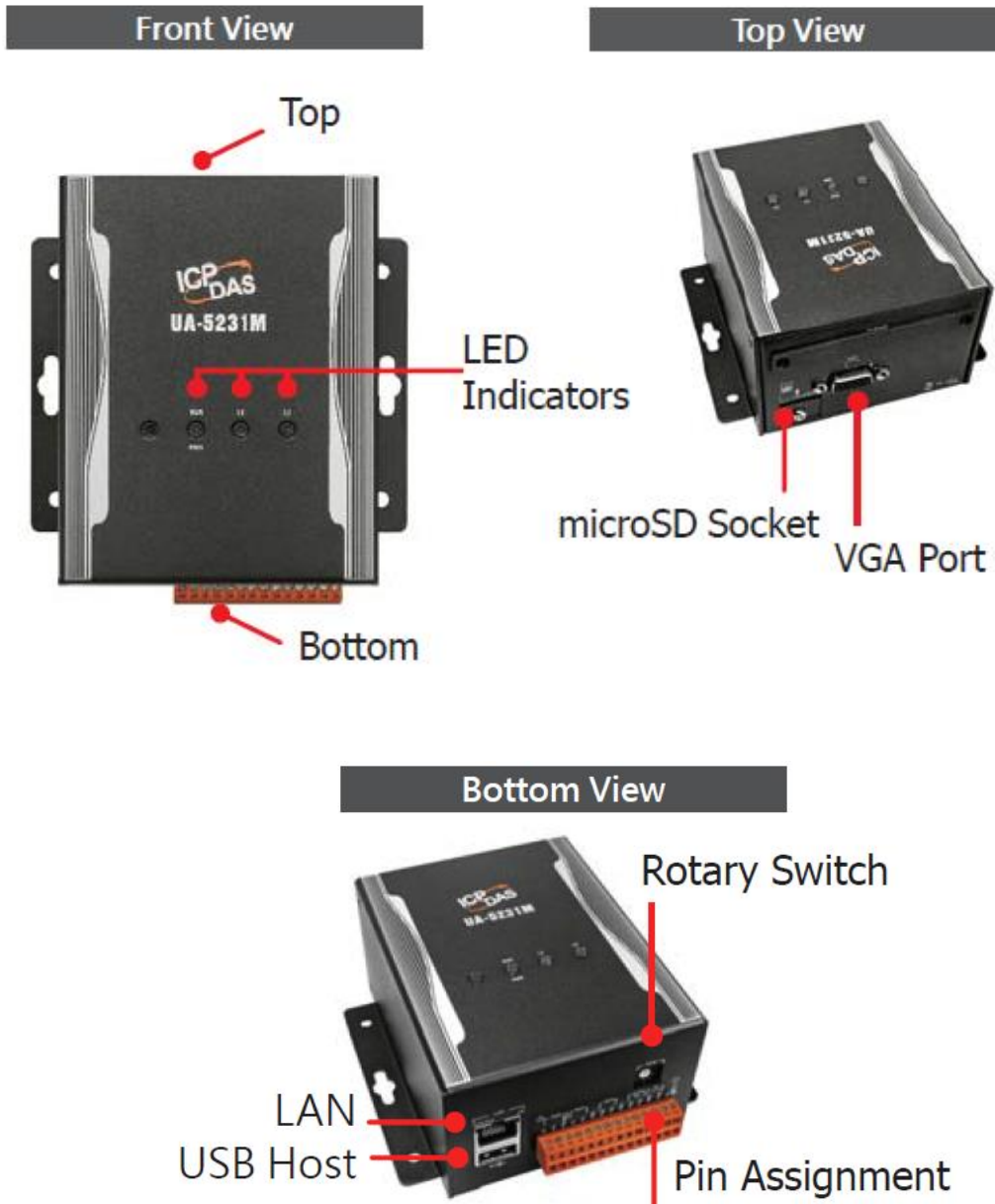
## UA-2241M / UA-2241MX-4GC / UA-2241MX-4GE



## UA-5231

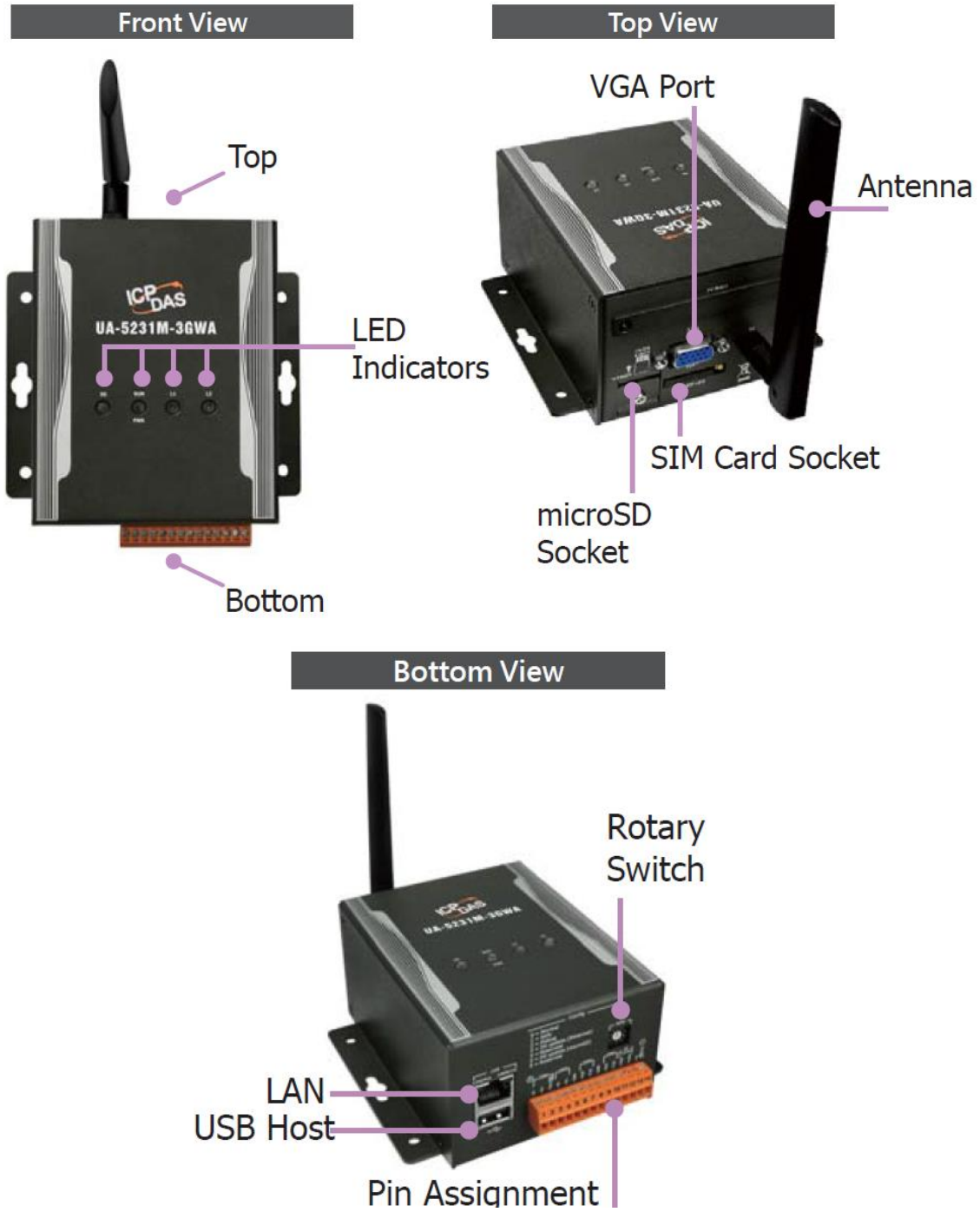


## UA-5231M





### UA-5231M-3GWA / UA-5231M-4GE / UA-5231M-4GC

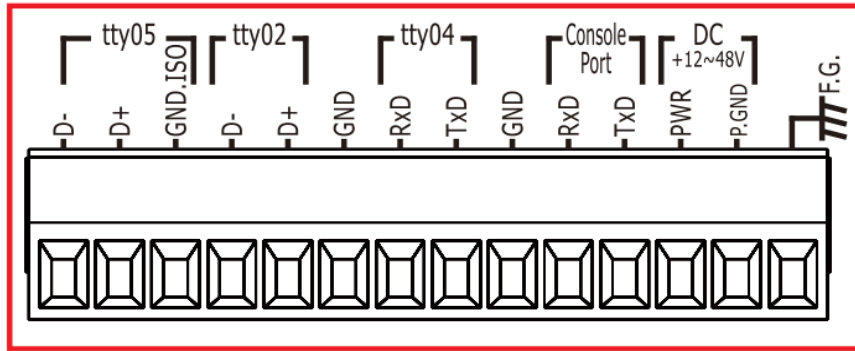


# 1.5 Pin Assignment

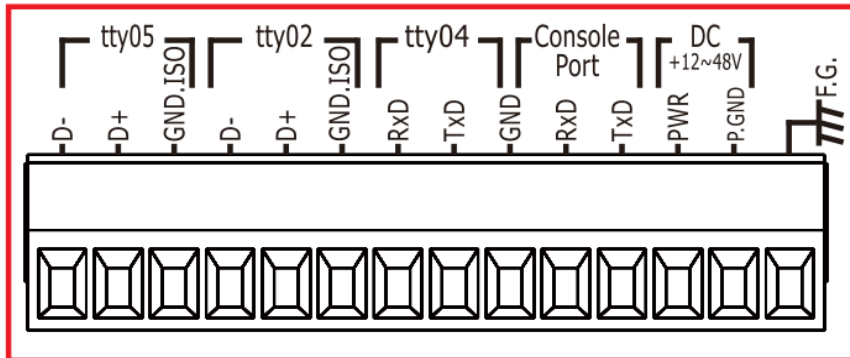
- **UA-5200 Series**

**UA-5231 / UA-5231M / UA-5231M-3GWA / UA-5231M-4GC / UA-5231M-4GE**

- **Versions shipped before 2021/09**

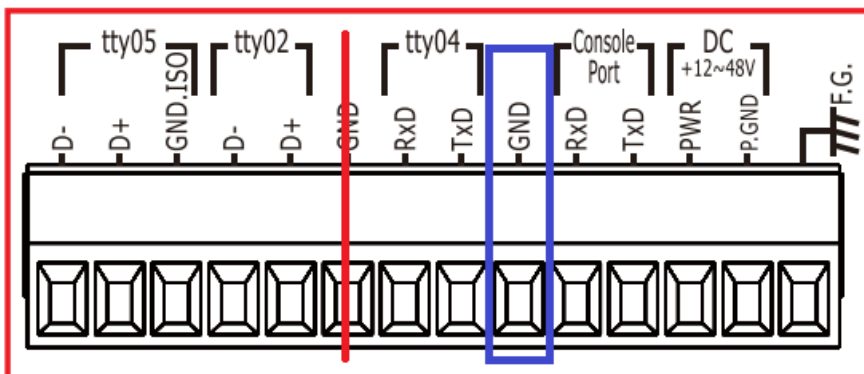


- **Versions shipped after 2021/09**



- **Versions shipped between 2021/05 ~ 2021/09 (Notice)**

The pin assignment diagram on the shell of the UA-5231 shipped between 2021/05 and 2021/09 (as shown below), the Ground (GND) with the red line is ineffective when connected to RS-232 (tty04, Console), please uses another Ground (GND in blue box)



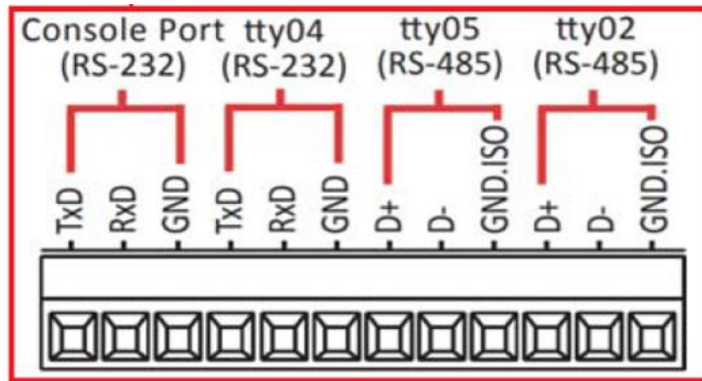
Because the shell of UA-5231 does not indicate the date of shipment or the hardware version, the users must log in to the UA Web User Interface and check the Version Information.

**When the [Flash Information] of [Version Information] is displayed as “eMMC”, the connection of RS-232, the Grund should be connected to the GND position in the blue box above.**

Version Information	
Firmware Version	Version 1.3.3.2
Main Program	Version 1.1.71
Web Interface	Version : 6.5.28 Date : 2021/08/12
Install Information	2021/09/02-09:27:11_WebUpdate_InstallSuccess
Flash Information	eMMC Flash

● **UA-2200 Series**

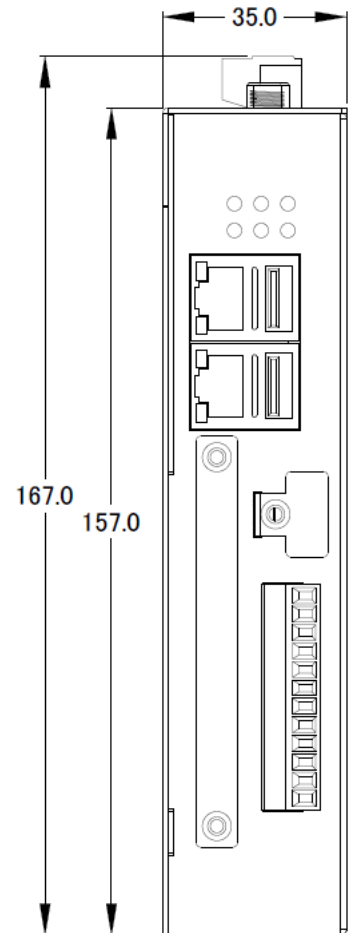
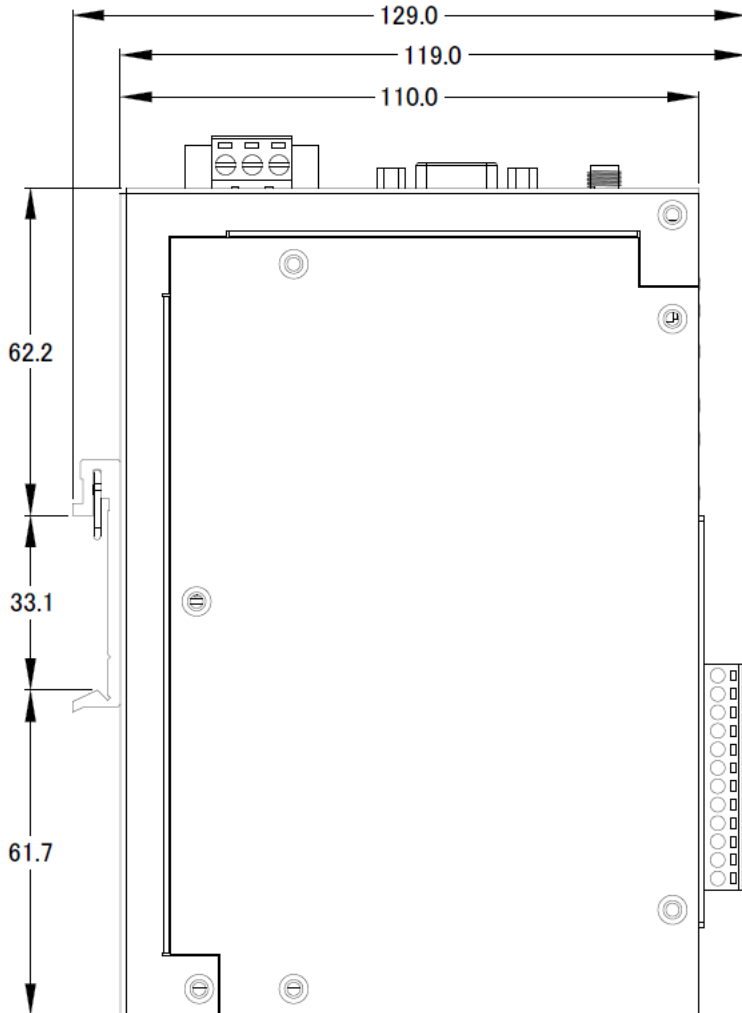
**UA-2241M / UA-2241MX-4GC / UA-2241MX-4GE**



# 1.6 Dimensions

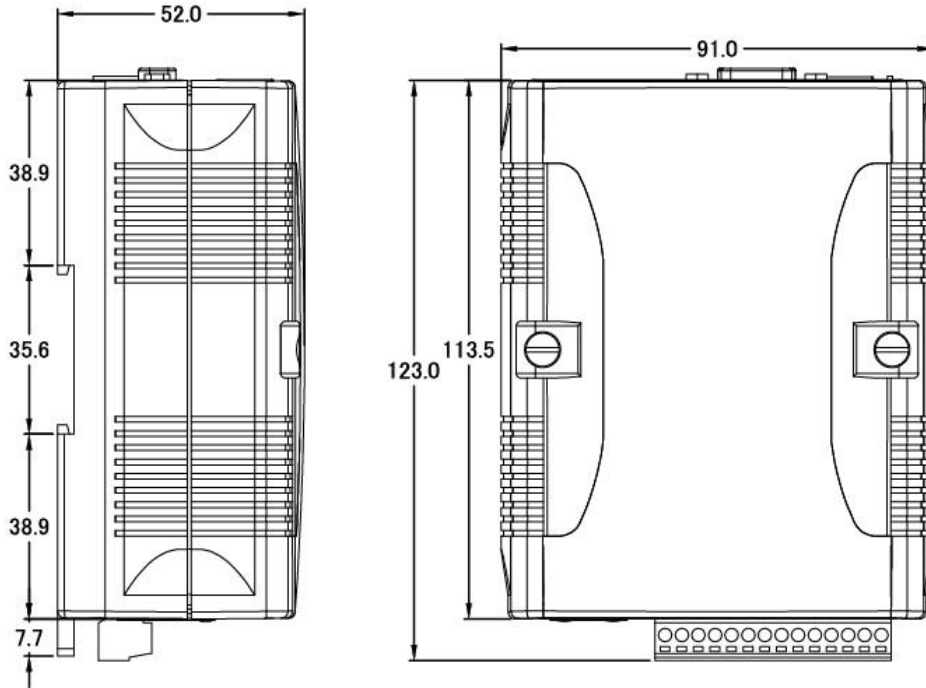
## UA-2241M / UA-2241MX-4GE / UA-2241MX-4GC

Unit: mm



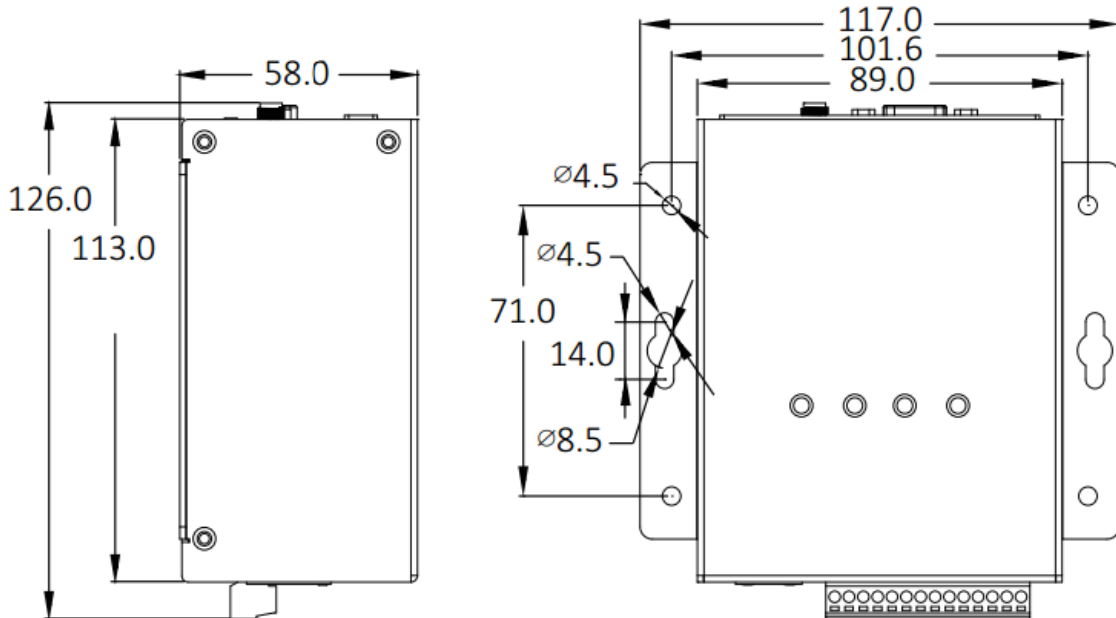
**UA-5231**

Unit: mm



**UA-5231M / UA-5231M-3GWA / UA-5231M-4GE / UA-5231M-4GC**

Unit: mm



## 2. Quick Start 1: Hardware/Network Connection

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

### 2.1 Hardware Connection

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

#### 2.1.1 Preparations for Devices

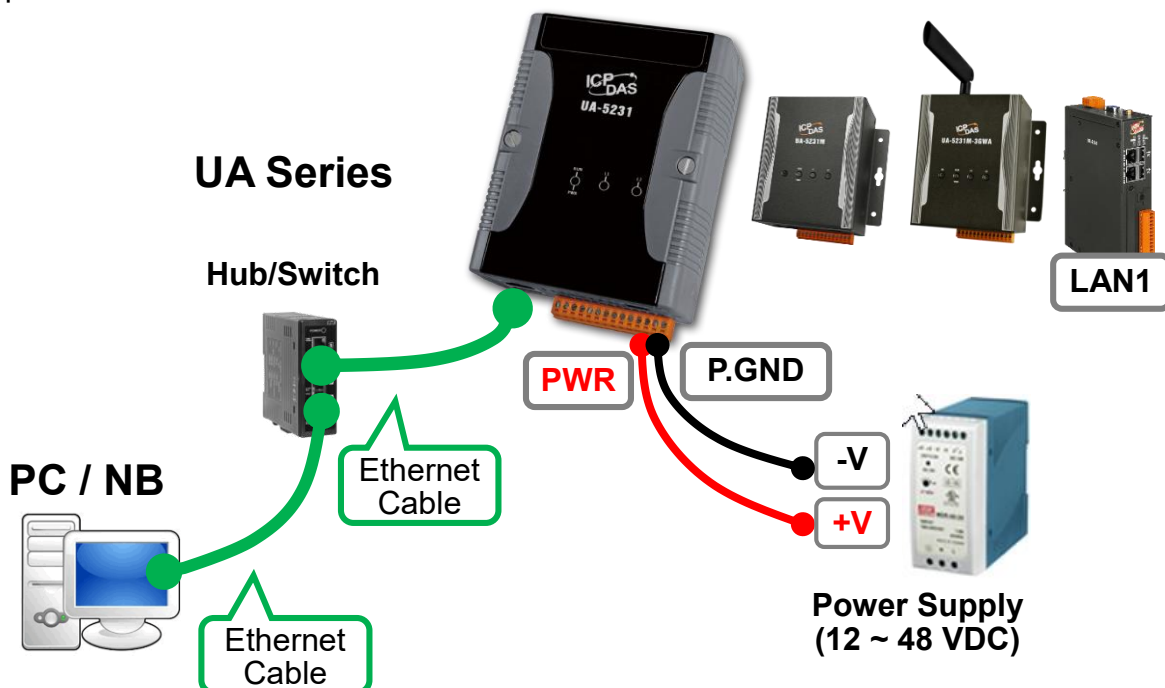
In addition to the UA series controllers (Ex: UA-5231), please prepare the following:

1. **PC/NB**: Can connect to the network and set the network
2. **Ethernet Hub or Switch** (Ex: NS-205)
3. **Power Supply: +12 ~ +48 VDC** (Ex: MDR-60-24)

#### 2.1.2 Hardware Wiring

Connect the UA with the RJ-45 Ethernet port LAN1 to an Ethernet hub/switch and PC. You can also link directly the UA to PC with an Ethernet cable.

After power is connected, please [ **wait 1 minute** ] for UA start-up procedure. When the "RUN/PWR" light ("RUN" "PWR" lights for UA-2200) starts flashing, it represents the boot is complete.



## 2.2 Network Connection

This section introduces how to connect to the UA Web User Interface (UA Web UI).

**Setting new UA or the new user please uses the method A in the Chapter 2.2.1** (The same method as the “UA Series Quick Start” manual).

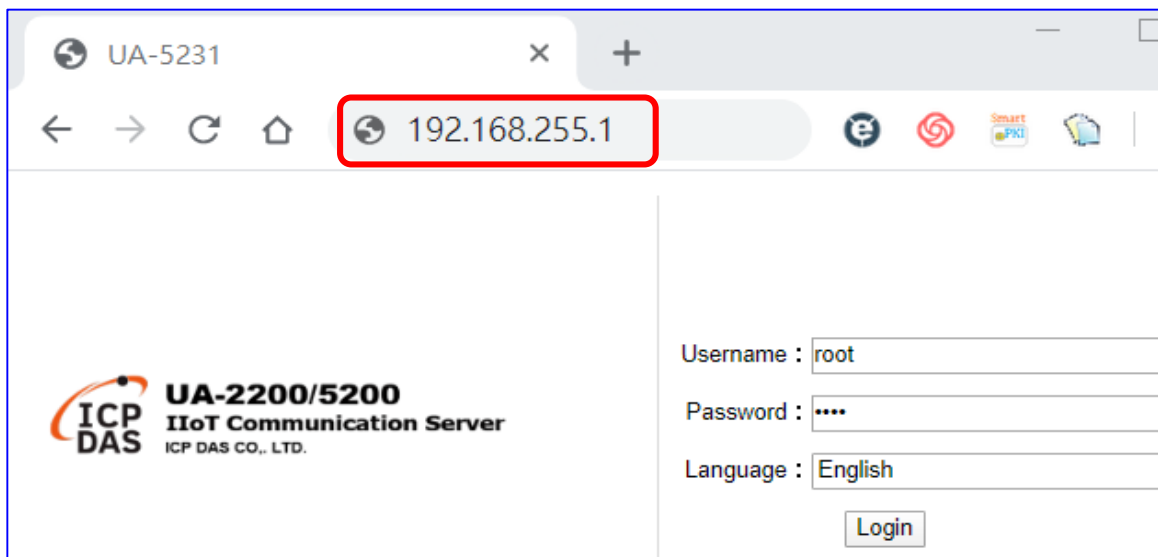
**Other** users please see the following introductions to choose **method B(Sec.2.2.2)** or **C**.

The methods to login the UA series Web UI:

**A. Using Factory Default Setting:** Suitable for setting a new UA controller and the PC network IP is not in the same domain with UA. 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:** Suitable for quick setting when many UA controllers are in the network but the IP are unknown. UA Series provides a free software utility for auto searching UA controllers in the network and can quick jump to the login web page of UA. (Refer [Section 2.2.2](#))

**C. Using IP Address:** Suitable for the UA has a fixed IP and in the same domain with the PC. If the UA has a fixed IP and in the same domain with 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.



After login the UA Web UI, then can set up the UA project.

### 2.2.1 Connection by Factory Default Settings (For New UA)

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

Factory Default Settings of UA Series			
Network	IP	LAN1: 192.168.255.1 LAN2: 10.0.0.1	Assign UA a new IP setting according to your case. <b>UA-2200</b> series uses <b>LAN1</b> to connect PC.
	Netmask	255.255.0.0	
	Gateway	LAN1: 192.168.1.1 LAN2: 10.168.1.1	
Web UI Account	Username	root	<b>After login, change the default username/password to use other functions.</b>
	Password	root	

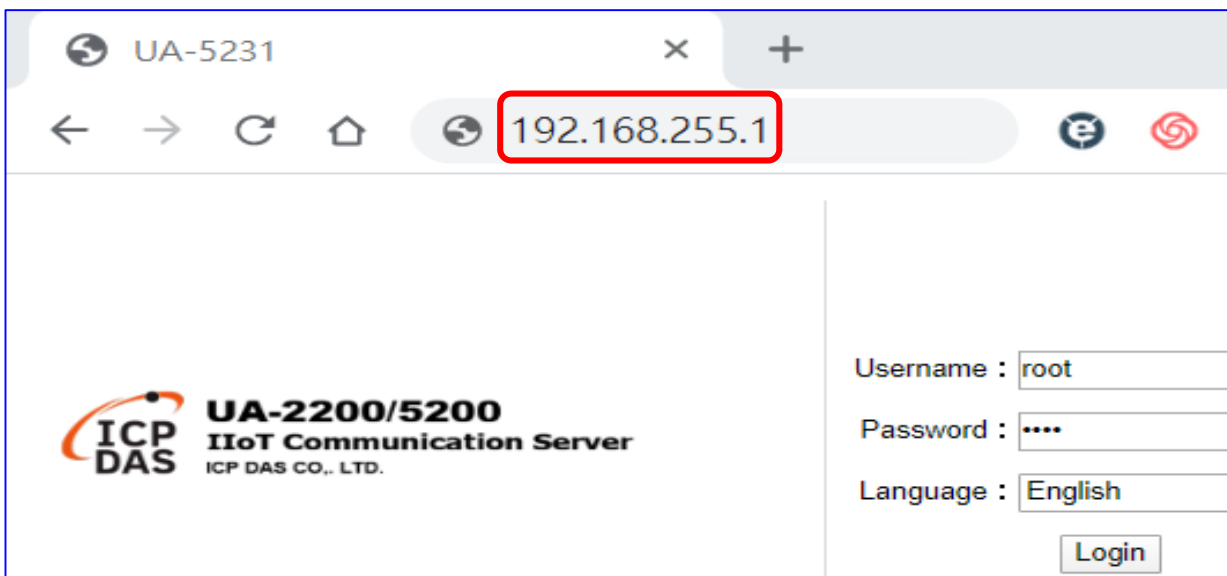
**[Steps]**

1. Change **PC's IP** setting to be in the same network with UA.  
**Note. Write down the PC original IP settings before modify.**

Ex:

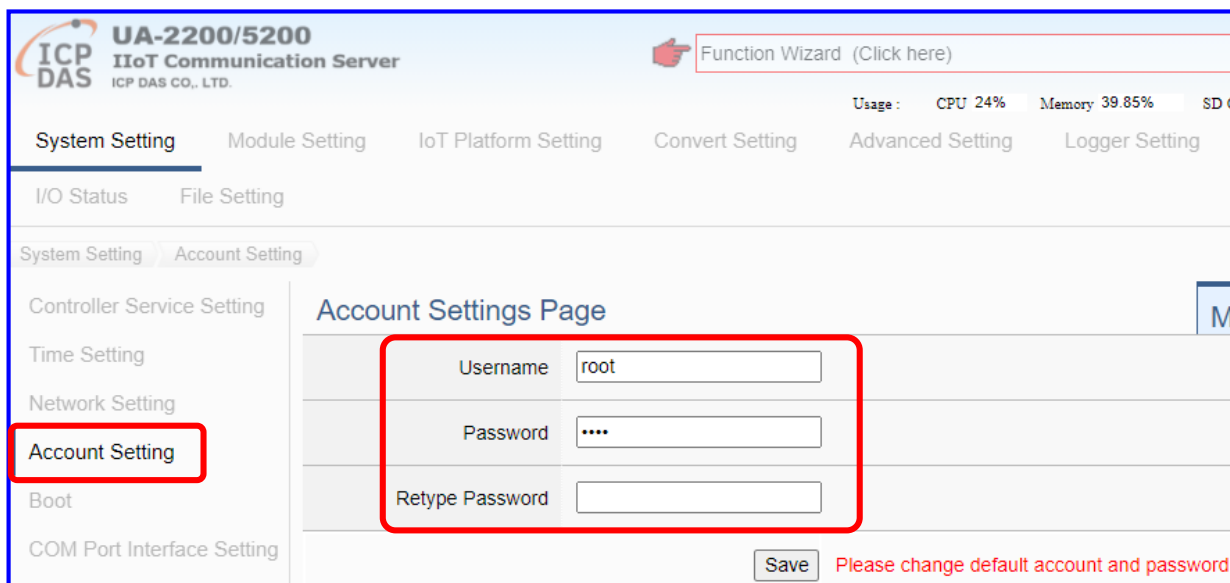
IP	192.168.255.1
Subnet mask	255.255.0.0
Gateway address	192.168.1.1

2. Make sure the PC and UA 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.

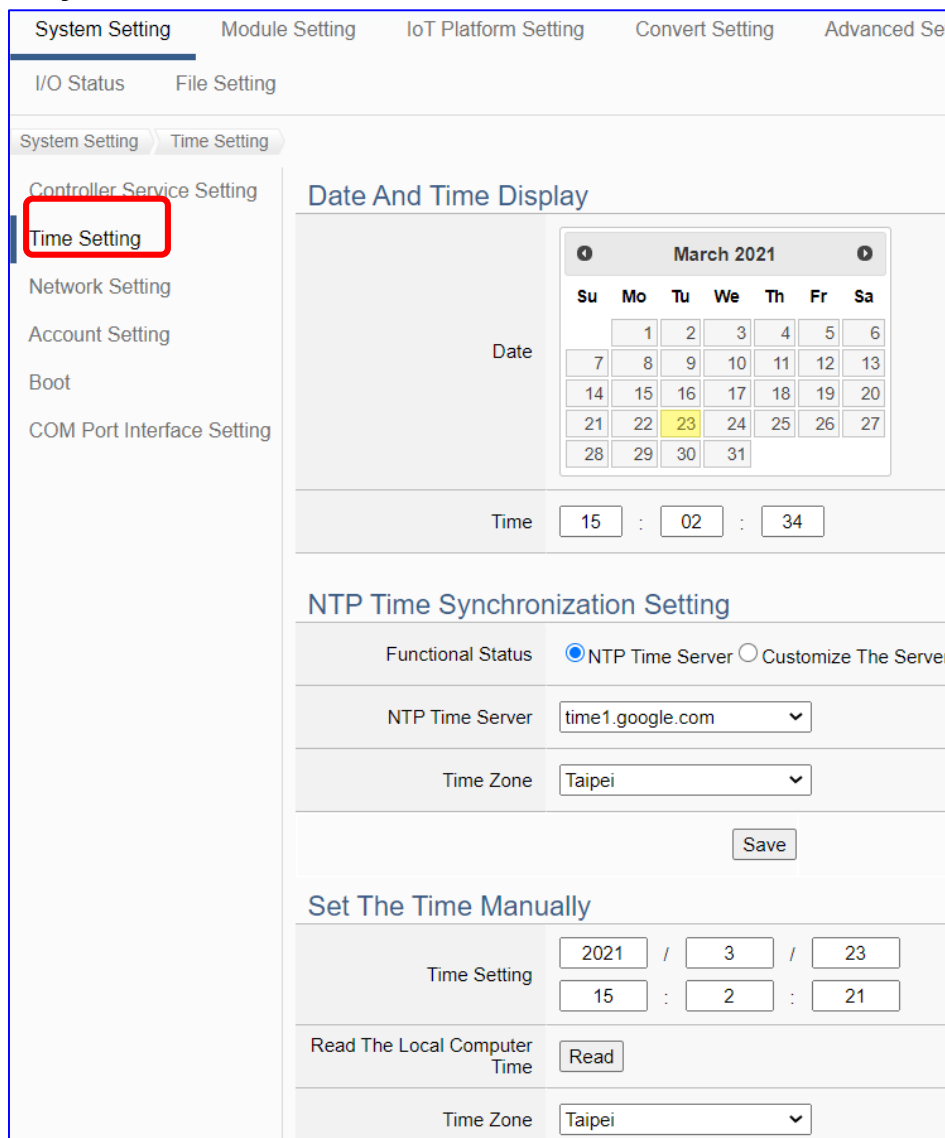




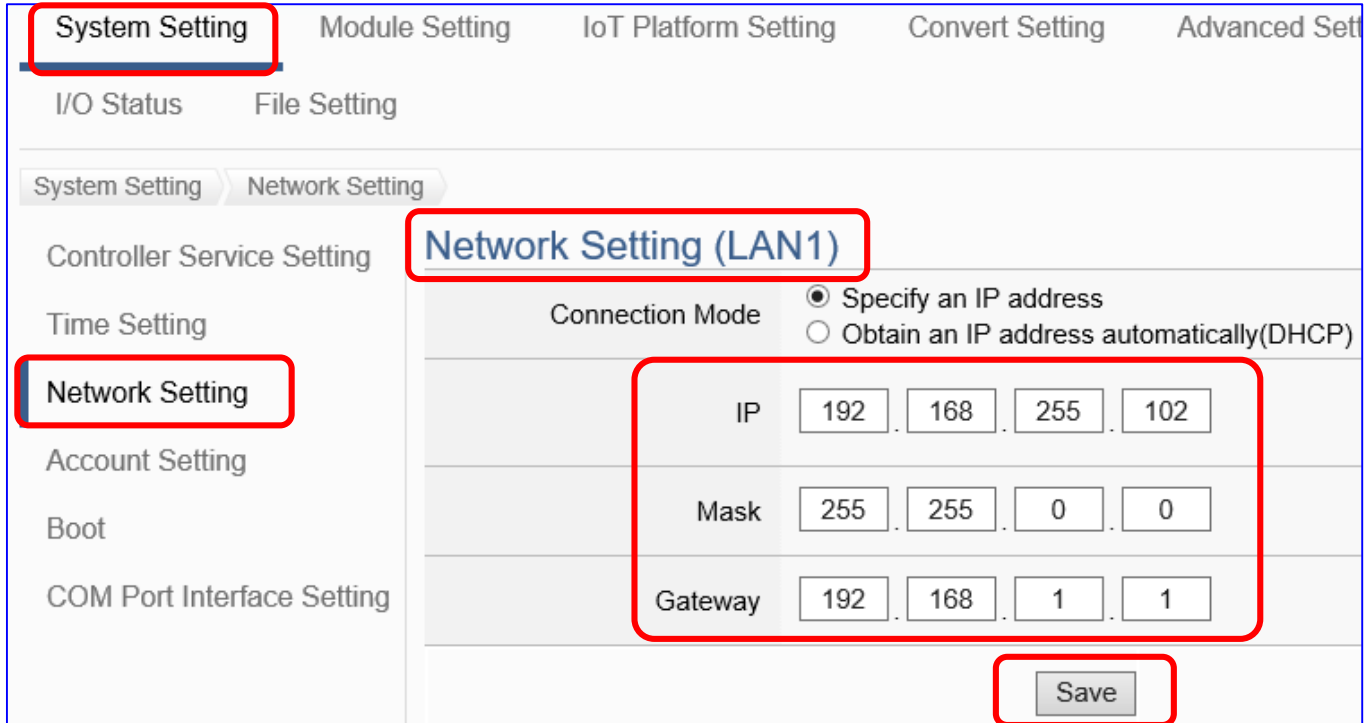
3. Click **【System Setting】** → **【Account Setting】** to change the **default Username / Password** to use other functions (Functions will be gray before changed).



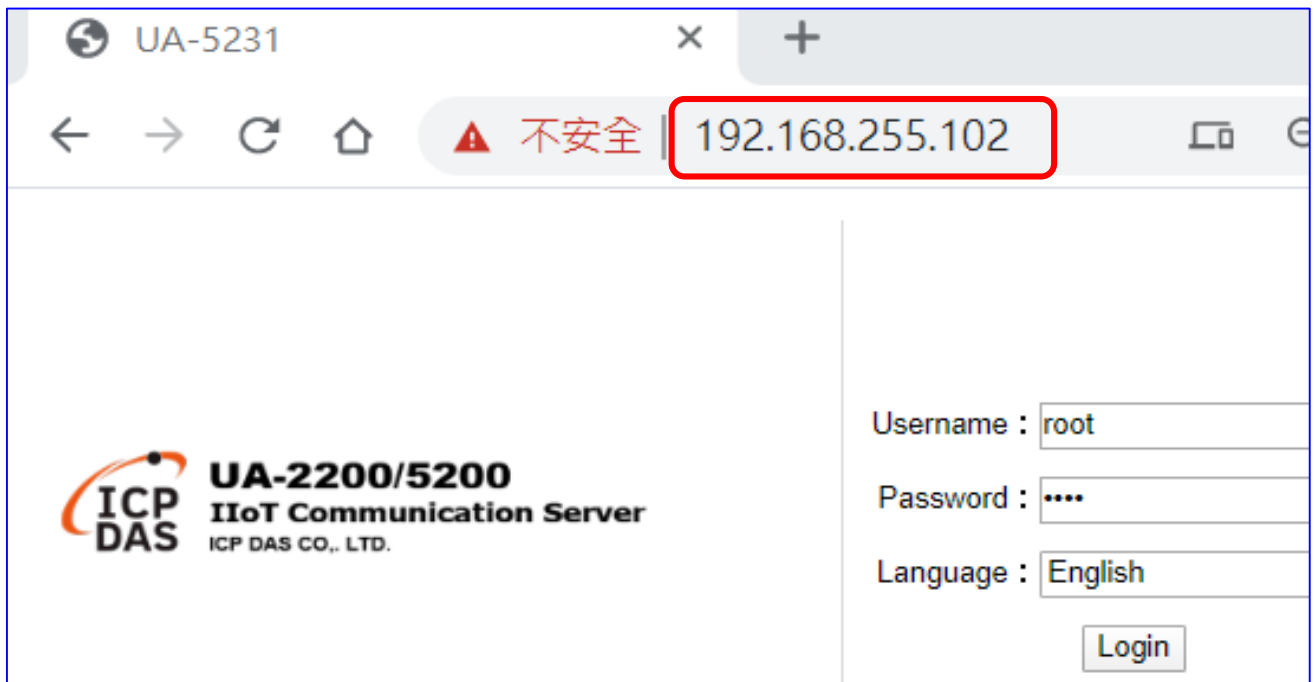
4. Click **【System Setting】** → **【Time Setting】**, check if the **UA time** is correct. If not, **modify** or set **Time Synchronization**.



- 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 to login the Web UI of UA series. Then configure user's UA project.



## 2.2.2 Connection by Utility Searching

Setting new UA or the new user please uses the method in the [Chapter 2.2.1](#). (A)


If the UA 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. (C)

This section introduces the 2nd method(B) that users use the UA Utility to search the Network IP. This method is suitable for connecting multiple UA series controllers to the Internet, but the IP addresses of UA are unknown or need to modify the UA quickly.

**UA Utility** is a free tool software to quickly search each UA series on the network and connect to its Web UI for setting UA series products and project.

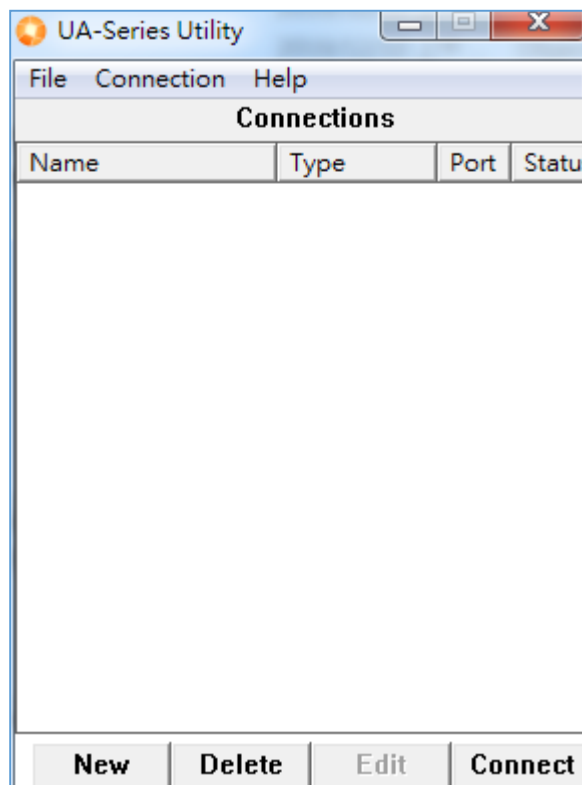
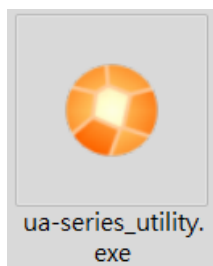
In the PC, install the **UA-Series Utility (.exe)**, and then run it to connect the device. Please download the utility program from the website:

<https://www.icpdas.com/en/download/index.php?nation=US&kind1=6&kind2=17&model=&kw=ua->

Utility & Tools				
FILE NAME	DESCRIPTION	MODEL	LAST UPDATE	DETAIL
UA-Series Utility	Utility For all UA-Series IIoT Communication Server & I/O modules		2020-05-22	

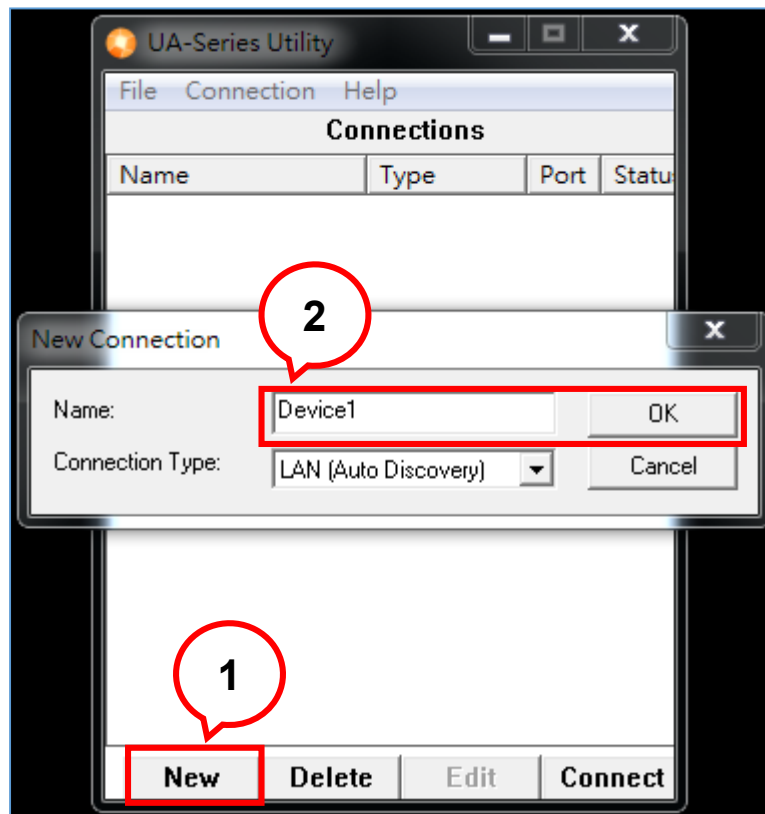
### 1. Install and execute the Utility

Run the **UA-Series Utility** (file name: **ua-series\_utility.exe**) to install the Utility program.



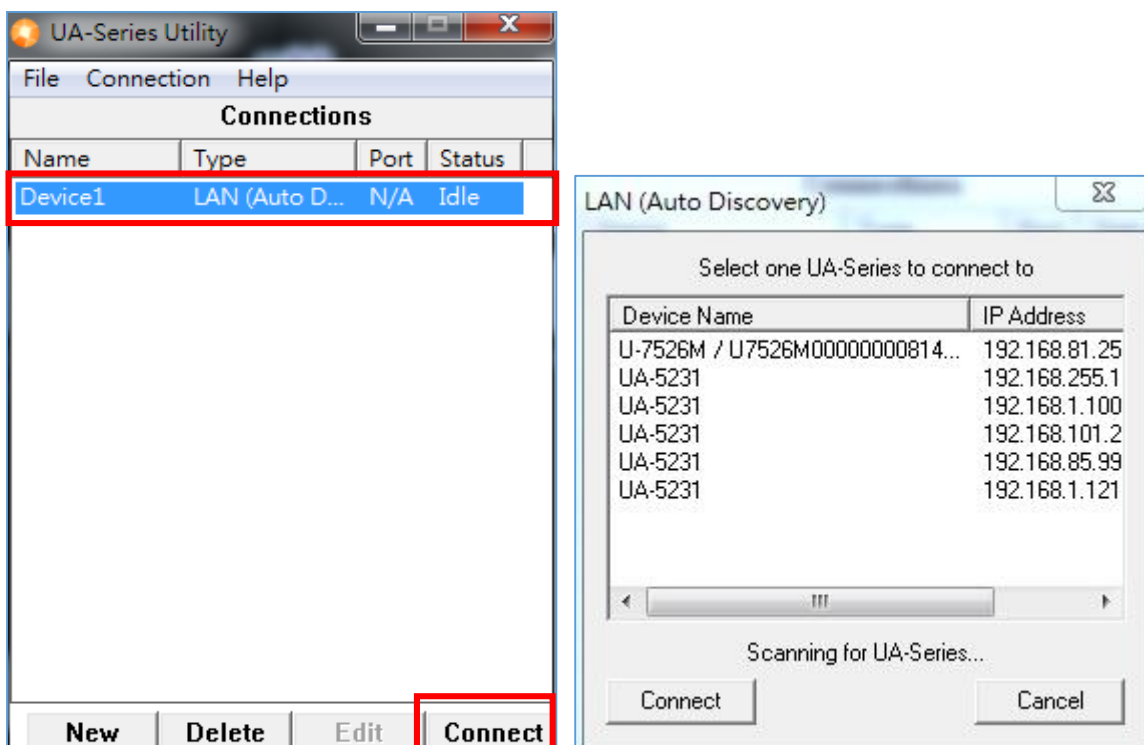
## 2. Create a new connection

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



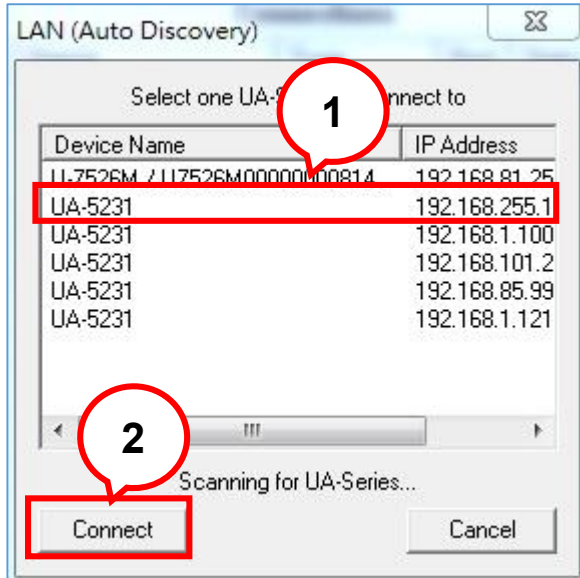
## 3. Search the UA controller

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 devices over the network.



#### 4. Connect to the UA Series

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

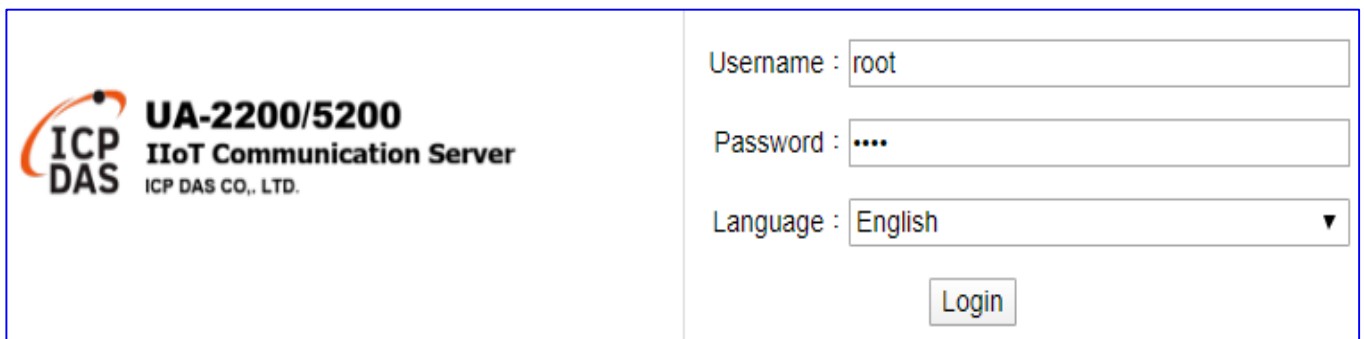


#### 5. Connection to the UA Web UI

The default web browser will be run and direct go to the UA login web site.

Please enter the username and password to login the UA series Web UI.

The factory default username: **root**. The factory default password: **root**. After login in, change the default Username/password first, or user cannot use any other function (New design for data security).



## 6. Login the Web UI of the UA Series

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

The screenshot displays the web interface for the UA-2200/5200 IoT Communication Server. The header includes the ICP DAS logo and the product name. A navigation menu at the top lists various settings categories. The main content area is divided into a sidebar with a 'System Setting' menu and a main panel showing 'Version Information' and 'System Setting' details.

Version Information	
Firmware Version	Version 1.3.1.9
Main Program	Version 1.1.58
Web Interface	Version : 6.5.19 Date : 2020/11/06
Install Information	2020/12/14-16:33:54_WebUpdate_InstallSuccess
Flash Information	NAND Flash

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.

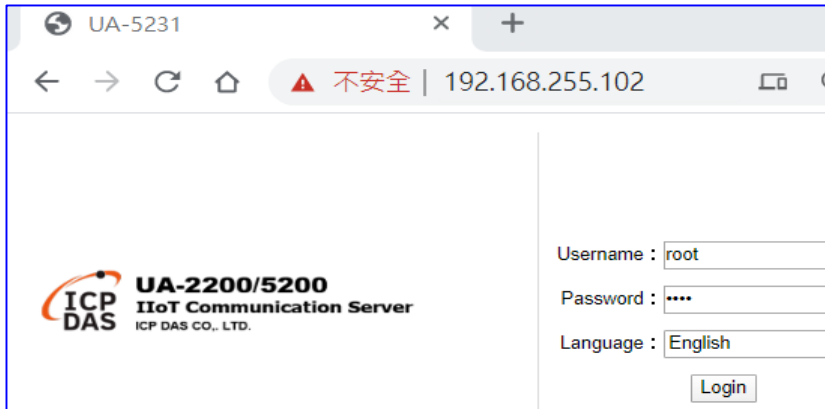
© ICP DAS Co., Ltd. All Rights Reserved

### 3. Quick Start 2: Web UI / Setting Steps

This chapter introduces the UA Web User Interface (UI), the steps for project / function / list settings, and a project example. For more project examples please see [Chapter 4](#). The detail parameters of the menus, functions, etc. will introduce in the next chapters [Chapter 5](#).

First, login the UA Web UI as below. (**Default username/password: root/root**)

If your UA controller is not connect to the network yet, please refer to [Chapter 2](#).



After log in the Web UI, users can see the version information, including the version of the install Firmware program, main program and Web Interface (and date).

Version Information	
Firmware Version	Version 1.3.1.9
Main Program	Version 1.1.58
Web Interface	Version : 6.5.19 Date : 2020/11/06
Install Information	2020/12/14-16:33:54_WebUpdate_InstallSuccess
Flash Information	NAND Flash

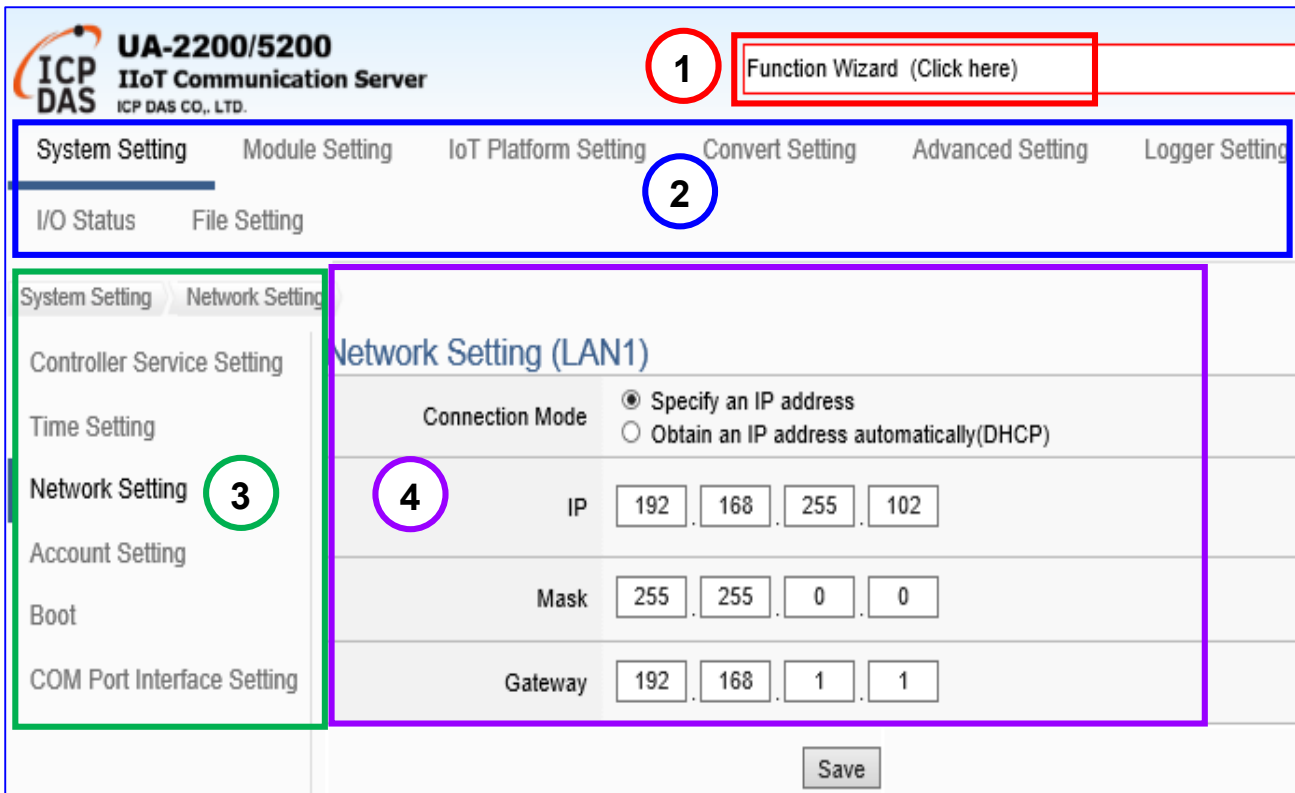
System Setting	
Controller Service Setting	Controller Service Setting provides the function to display and set the running status of the controller service about the project, MQTT broker and DDNS.
Time Setting	Time Setting provides the function to display and set the date, time and time zone of the controller. (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.

### 3.1 Web UI Environment Overview

**The function areas of the Web UI:**

This chapter will overview these areas. The following chapters will introduce the settings of the functions and parameters.

1. **Function Wizard:** A quick setup area for commonly used projects or functions. 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 to [Chapter 4](#))
2. **Main Menu Area:** The main menu contains all the setting functions that classified into several categories. Click the main menu item, the sub-menu will appear on the left of the page, and the function descriptions will appear under the main menu area. (Refer to [Chapter 5](#))
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 to [Chapter 5](#))
4. **Setting Area:** The setting area is for displaying and setting the functions and parameters of UA series controller. The content of this area will be vary according to the selected main menu and sub-menu.





## 3.2 Setting Steps for Project / List

- **【Project Setting Steps】 :**

The setting for UA series controller is to set up from the left to the right of the main menu functions. The “**Function Wizard**” even provides the “**Step Box**” for users to follow the steps and prevent from selecting the wrong function.

The Web UI screen of UA Series is as following; User can set up own UA project. If user has not login your UA Web UI, refer to [CH2 Quick Start 1](#).

**CH4 [Function Wizard]:** In the upper right corner of the web UI, to guide users quickly set up a project. It provides many quick setting option items. With the step-by-step guide of the function wizard, the project can be completed quickly. Please see [Chapter 4](#).

**CH5 [Main Menu]:** The user can also select the function of the main menu to set projects by himself. For the detailed function and parameter descriptions, please refer to [Chapter 5](#).

Version Information	
Firmware Version	Version 1.3.1.9
Main Program	Version 1.1.58
Web Interface	Version : 6.5.19 Date : 2020/11/06
Install Information	2020/12/14-16:33:54_WebUpdate_InstallSuccess
Flash Information	NAND Flash

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.

**For Example:** the setting steps of the Modbus communication conversion with the OPC UA protocol are as below:

**Steps for setting the Project:**

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

**Main Menu:**

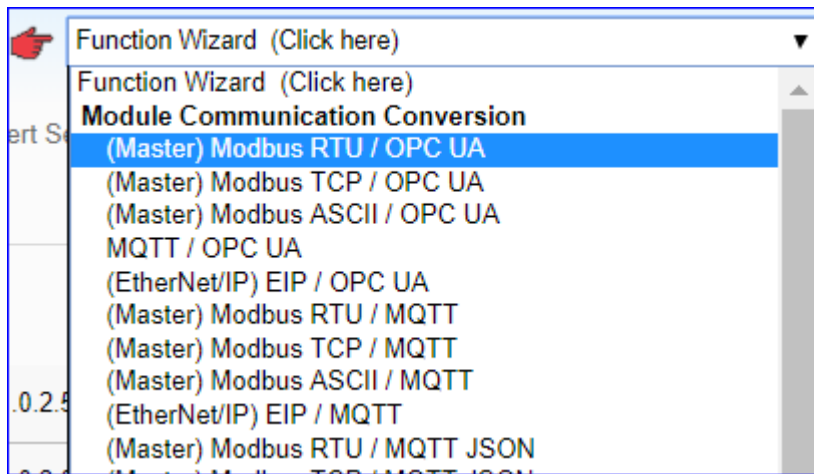
System Setting    Module Setting    IoT Platform Setting    Convert Setting    Advanced Setting    I/O Status    File Setting

**Function Wizard Step Box: (EX. Modbus RTU / OPC UA)**

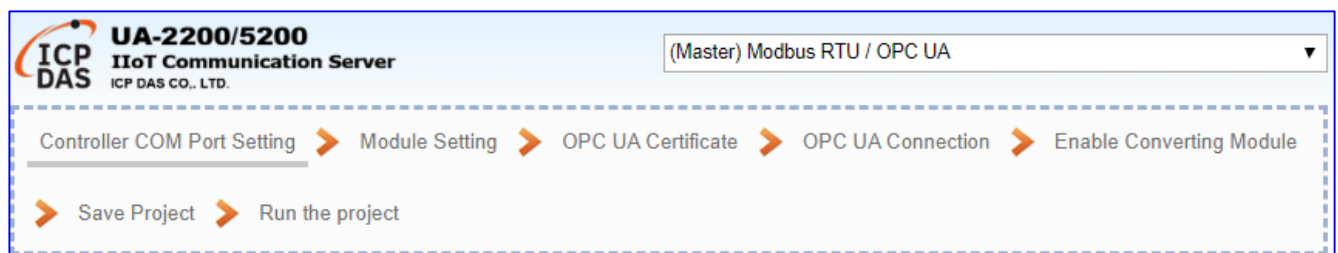
Controller COM Port Setting > Module Setting > OPC UA Connection > Enable Converting Module > Save Project >

Run the project

Click the **Function Wizard** and select the “**(Master) Modbus RTU / OPC UA**” item.



The Web UI will enable a Wizard guide 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.



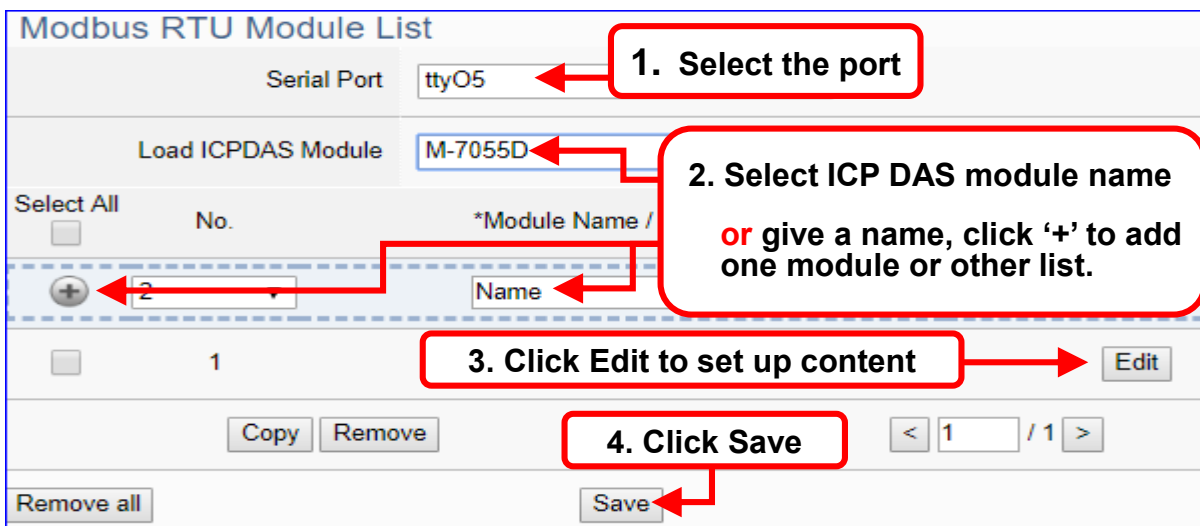
● **【 List Setting Steps 】 :**

In the steps of setting up the project, various lists need to be set up, including: modules, connections, I/O channels, variables, communication connections, database... etc. About the List setting of module, connection..., they have the similar steps as below.

**Steps for Setting List:**

1. Select the connection **port** for the module (or connection...)
2. Select ICP DAS module/list **name**  
 or give a name, then click the button [ + ] to add a list of module, connect...
3. Click the button [**Edit**] to enter the Content Setting page. Set up the list content.
4. Click [**Save**] to back, and then [Save] the list page.

**Steps for List:** e.g. Modbus RTU Module List.



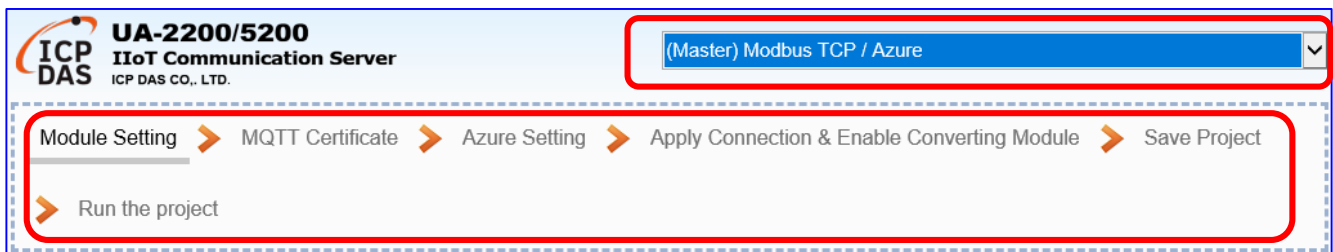
The next chapter [4.Function Wizard: Project Quick Setup & Examples](#) will provide various **examples** about setting the commonly projects and functions.

# 4. Function Wizard: Project Quick Setup & Examples

Chapter 4 is about the UA project setup. [Function Wizard] in the up-right corner of the Web UI provides an easy and quick setting “Step Box” that very suitable for the new users.



[Function Wizard Step Box] (abbreviation [Step Box]) is a step guide of the Function Wizard. When users select an item, the Web UI will enable a Wizard and show a “Step Box” for user to follow the “Step Box” step-by-step and then can complete the project easily & quickly.

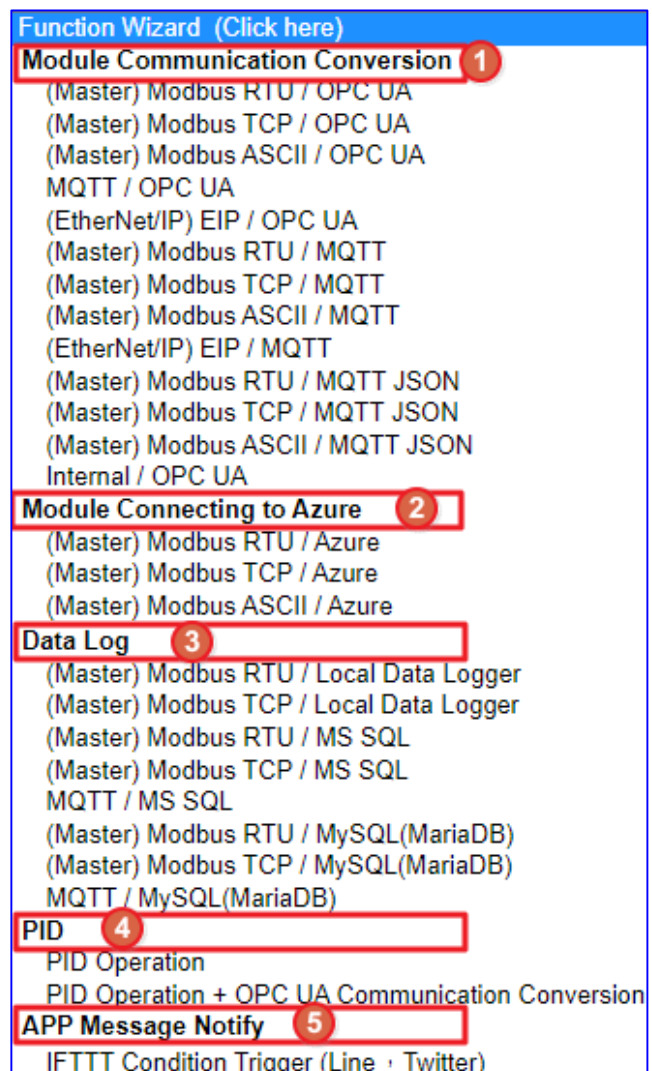


This chapter will focus on the setting steps of the projects/functions that are divided into several categories (in red box, they also are the Section titles) and will introduce the function step settings in each section. We will develop more items in the future.

The setting examples of most used Function Wizard items will be put on the UA series FAQ website. Please refer to ICP DAS homepage > Support > FAQ > IIoT > UA series FAQ. <https://www.icpdas.com/en/faq/index.php?kind=326>

When set up the project, please choose an item from a category and follow the steps.

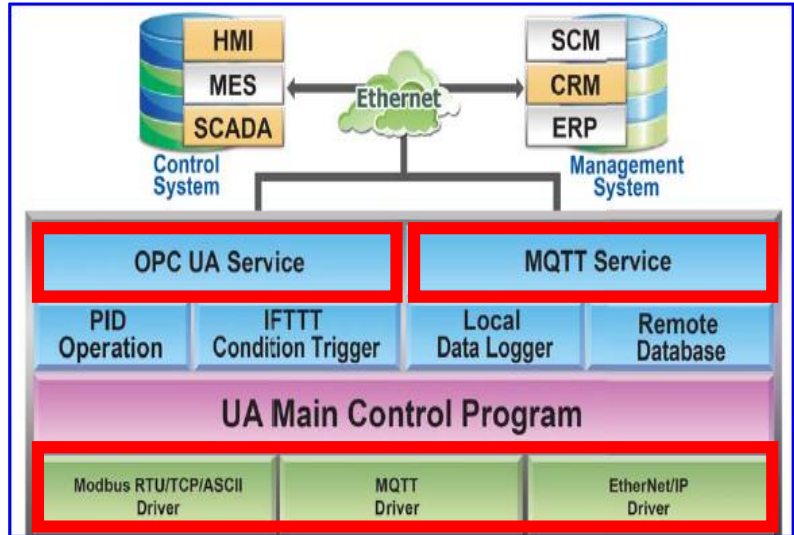
Please at least refer one example in this chapter to know the setting steps and tips.



## 4.1 Module Communication Conversion

“Module Communication Conversion” of UA series, a very commonly used function, can effectively 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”. In the category, there are several items that can be divided into the following protocol types and will introduce them in the sub-sections: OPC UA, MQTT, MQTT JSON, and Internal for virtual.

**Module Communication Conversion**  
 (Master) Modbus RTU / OPC UA  
 (Master) Modbus TCP / OPC UA  
 (Master) Modbus ASCII / OPC UA  
 MQTT / OPC UA  
 (EtherNet/IP) EIP / OPC UA  
 (Master) Modbus RTU / MQTT  
 (Master) Modbus TCP / MQTT  
 (Master) Modbus ASCII / MQTT  
 (EtherNet/IP) EIP / MQTT  
 (Master) Modbus RTU / MQTT JSON  
 (Master) Modbus TCP / MQTT JSON  
 (Master) Modbus ASCII / MQTT JSON  
 Internal / OPC UA



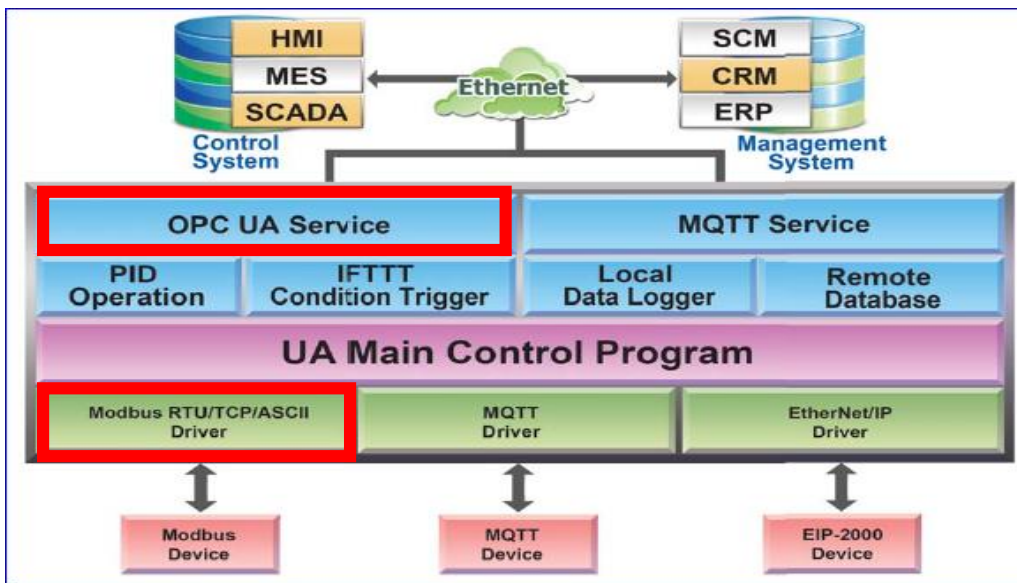
<p><b>Modbus RTU / OPC UA (4.1.1, Ex. M-7055D)</b>  <b>Modbus TCP / OPC UA</b>  <b>Modbus ASCII / OPC UA (4.1.1, Ex. M-7055D)</b></p>	<p>Using the <b>OPC UA Service</b> to convert with <b>Modbus RTU/TCP/ ASCII</b> protocols.</p>
<p><b>MQTT / OPC UA (4.1.2, Ex. MQ-7255M)</b></p>	<p>Using the <b>OPC UA Service</b> to convert with <b>EtherNet/IP</b> protocols.</p>
<p><b>EIP / OPC UA (4.1.3, Ex. EIP-2060)</b></p>	<p>Using the <b>OPC UA Service</b> to convert with <b>EtherNet/IP</b> protocols.</p>
<p><b>Modbus RTU / MQTT</b>  <b>Modbus TCP / MQTT (4.1.4, Ex. DL-302)</b>  <b>Modbus ASCII / MQTT</b></p>	<p>Using the <b>MQTT Service</b> to convert with <b>Modbus RTU/TCP/ ASCII</b> protocols.</p>
<p><b>EIP / MQTT (4.1.5, Ex. EIP-2060)</b></p>	<p>Using the <b>MQTT Service</b> function to convert with <b>EtherNet/IP</b> protocols.</p>
<p><b>Modbus RTU / MQTT JSON (4.1.6, Ex. M-7055D)</b>  <b>Modbus TCP / MQTT JSON</b>  <b>Modbus ASCII / MQTT JSON(4.1.6,Ex. M-7055D)</b></p>	<p>Using the <b>MQTT Service</b> function in-group of <b>JSON</b> format to convert with <b>Modbus RTU/TCP/ ASCII</b> protocols.</p>
<p><b>Internal / OPC UA</b></p>	<p>Using the <b>OPC UA Service</b> function to convert <b>Internal</b> to OPC UA (Server) protocols, or as an intermediary to provide <b>data exchange</b> for <b>OPC UA Client to Client</b>.</p>

### 4.1.1 Function Wizard: Modbus / OPC UA (RTU Example, Use M-7055D)

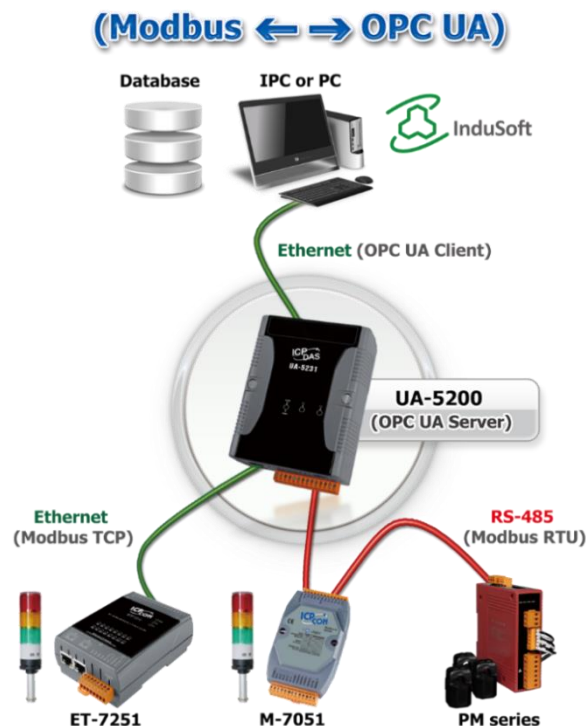
Modbus / OPC UA Conversion include the conversion of **OPC UA** and **Modbus RTU / TCP / ASCII** three protocols. With the OPC UA Service function, the OPC UA Server can read and write the Modbus RTU/TCP/ASCII devices that connected to the controller.

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

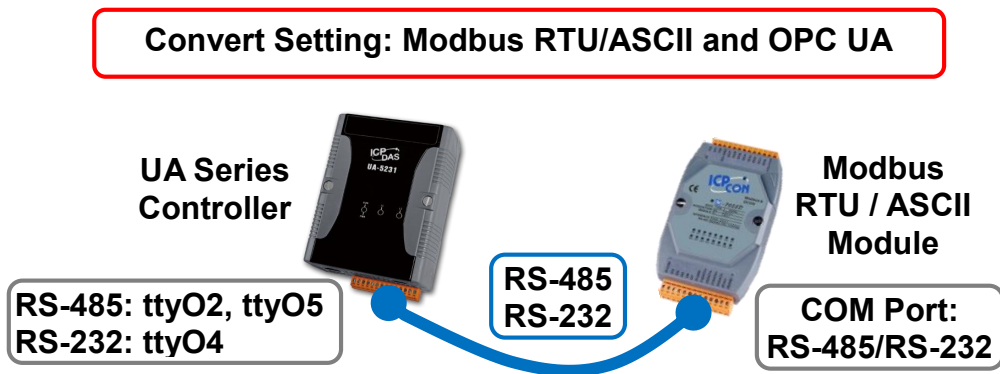
#### Modbus / OPC UA Function Diagram:



#### Application Solution:

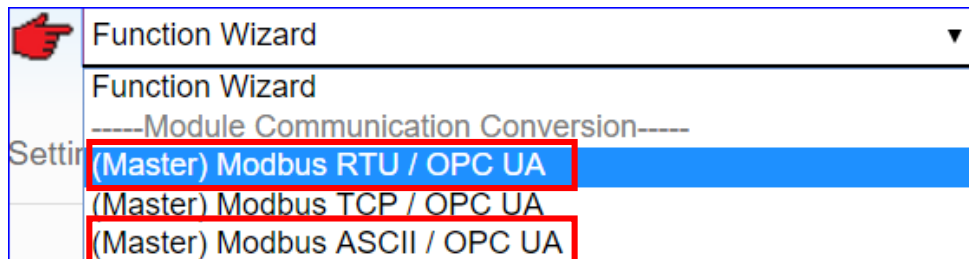


● **Convert Setting: Modbus RTU/ASCII and OPC UA**



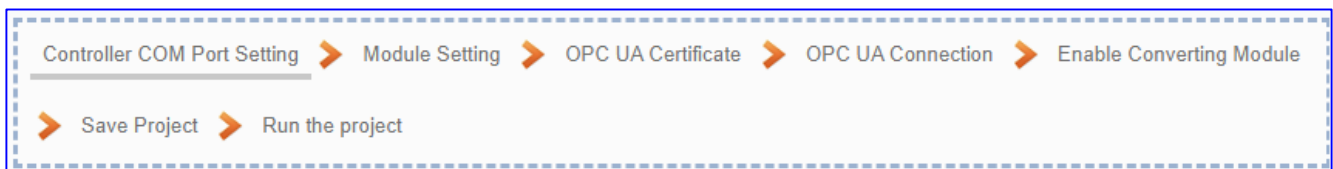
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

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 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.

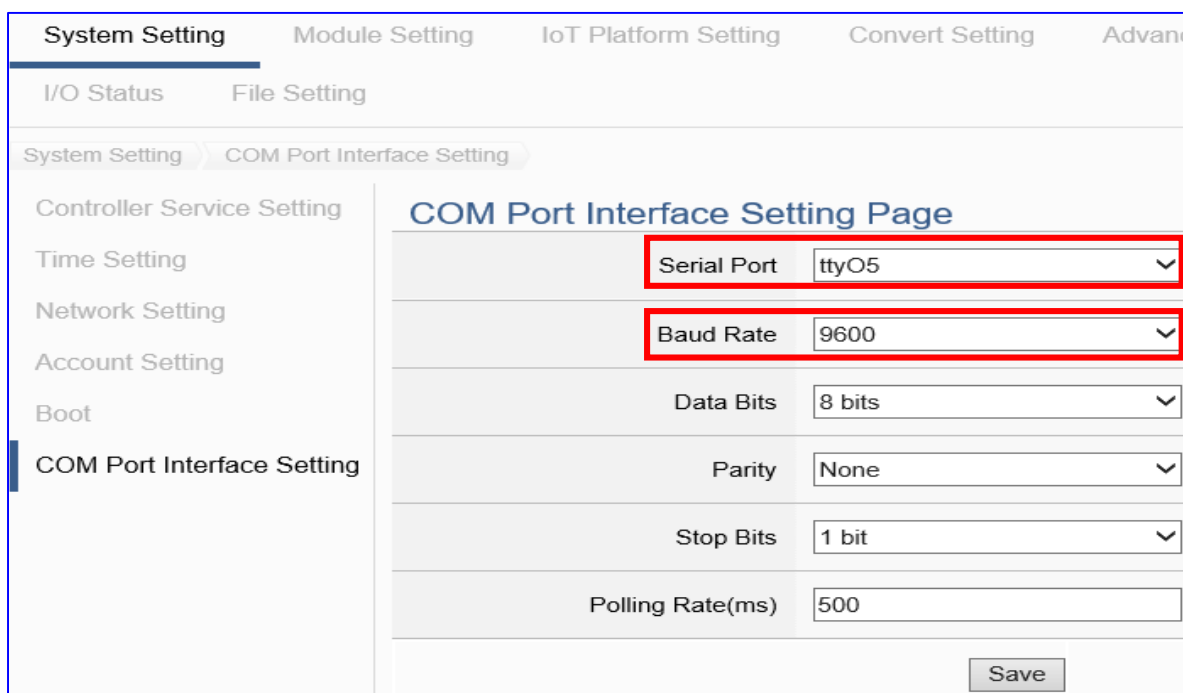


● **Step 1. Controller COM Port Setting**



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

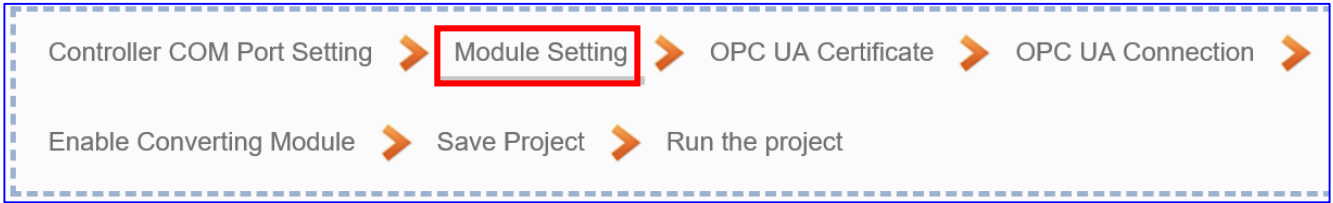
The user can find the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).



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.

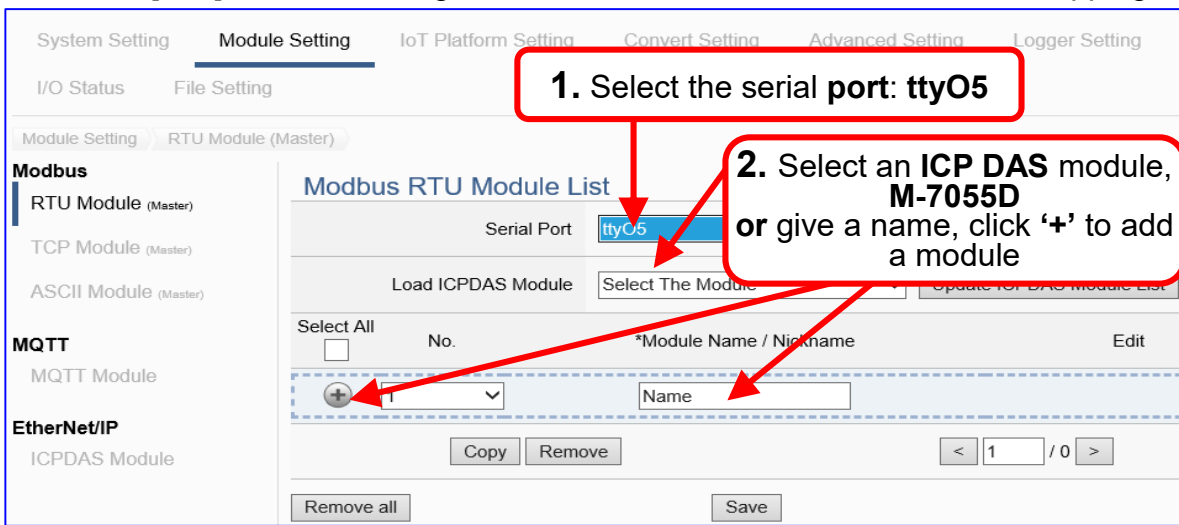


● **Step 2. Module Setting**

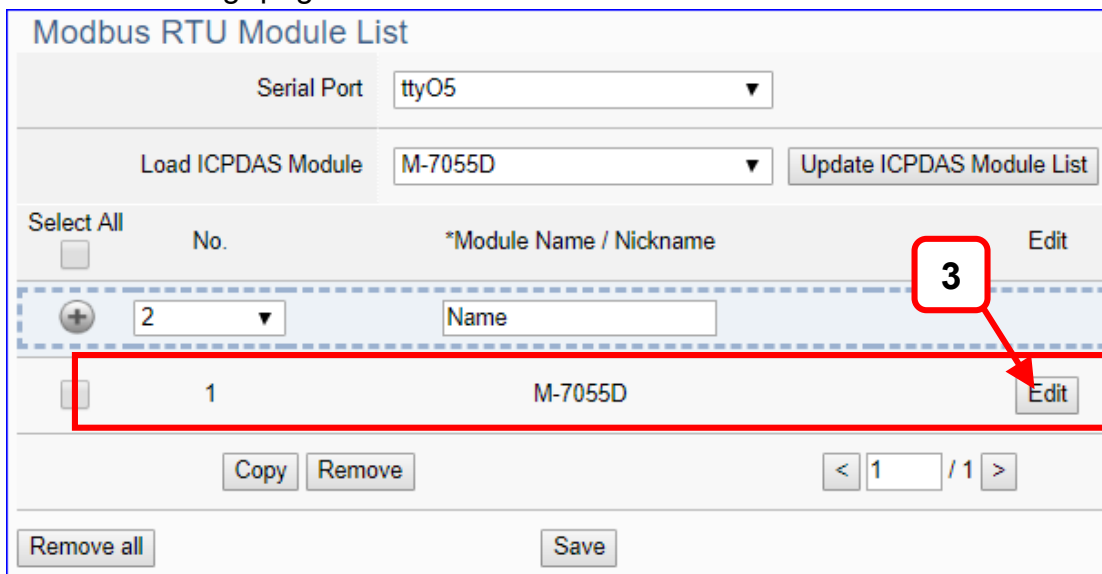


Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. If using ICP DAS module, user just need to select the model number, system will auto add and setup the module. If not, give a module name (Default: Name), click [ + ] button to add a new module, and then click [Edit] button to configure the module content and the Modbus mapping table.



Add a module (ex: No.: 1, Name: M-7055D) 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.		3
Module Name		M-7055D
Slave ID		2
Timeout(ms)		500
Write Retry	<input type="checkbox"/>	1

### Modbus Mapping Table Setting

<b>DO mapping 01</b>		Data Model		01 Coil Status
<b>UA start address: 0</b>		Start Address		0
<b>If DO x 8, enter 8</b>		Data Number		1
<b>Click [Add]</b>		Create Tables		<input type="button" value="Add"/>

If use ICP DAS module, system will auto setup the Modbus Mapping Table; if not, user needs to check the Modbus address or I/O number from the module user manual.

> Modbus Mapping Table Setting:

Set module in the order of Data Model, Start Address and Data Number, then click "Add".

Ex: M-7055D has 8 Data Models of "01 Coil Status (0x)" (Mapping: DO), so select Model "01", Start Add. "0", Number "8", and click "Add".

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

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. (Range: 1 ~ 247)				
Timeout	Set the timeout value for the module. Default: 500 ms				
Write Retry	Check to retry writing again when there is no response after the set time is up, and it can be set up to retry 3 times				
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 gray; padding: 2px; margin-top: 5px; width: fit-content;"> <table style="width: 100%; border-collapse: collapse;"> <tr style="background-color: #0070c0; color: white;"><td style="padding: 2px;">01 Coil Status(0x)</td></tr> <tr><td style="padding: 2px;">02 Input Status(1x)</td></tr> <tr><td style="padding: 2px;">03 Holding Registers(4x)</td></tr> <tr><td style="padding: 2px;">04 Input Registers(3x)</td></tr> </table> </div>	01 Coil Status(0x)	02 Input Status(1x)	03 Holding Registers(4x)	04 Input Registers(3x)
01 Coil Status(0x)					
02 Input Status(1x)					
03 Holding Registers(4x)					
04 Input Registers(3x)					
Start Address	The start address of the Modbus command. <b>Note:</b> Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.				

**If use ICP DAS module, system will auto-setup the module and its Modbus Mapping Table.** Such as this example, we select the module from “Load ICP DAS Module”, it auto-shows the M-7055D (DO x 8, DI x 8) module content and Modbus Mapping Table as below.

### Module Content Setting

No.	<input type="text" value="3"/>
Module Name	<input type="text" value="M-7055D"/>
Slave ID	<input type="text" value="2"/>
Timeout(ms)	<input type="text" value="500"/>
Write Retry	<input type="checkbox"/> <input type="text" value="1"/>

### 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"/>

### Modbus Mapping Table

		Address	Nickname	Scaling	Bitwise
		Coil Status(0x)	Input Status(1x)	Holdng 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"/>			

**If not use ICP DAS Module**, please check the module’s user manual to find out the module Modbus Address, and refer to the chapter “Module Setting” of the UA manual as below.

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)

**Note:** the start address of UA series is bass on “0”. Some modules start address are bass on “1”, but please note UA is follow the rule of start address “0”, and set enough Data Number for mapping to the I/O channels of the linking module.

In this example, M-7055D has 8 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 (left) and the [**Coil Status(0x)**] table after setting (right):

**Modbus Mapping Table Setting**

**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

Edit

M-7055D **8 DI** setting (left) and the [**Input Status(1x)**] table after setting (right):

**Modbus Mapping Table Setting**

**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

Edit

The Modbus Mapping table is showing as below. Click [OK] to save and exit.

Modbus Mapping Table		Address	Nickna
Coil Status(0x)			
Address	0		
Number	8		
Type	Bool		
Edit			
Input Status(1x)			
Address	0		
Number	8		
Type	Bool		
Edit			

For more setting item descriptions, please refer to chapter [Chapter 5.2. Module Setting](#).



**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table	Address	Nickname	Scaling	Bitwise
<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="DO0"/>	Bool	<input type="text" value="Light 01"/>	
1	<input type="text" value="DO1"/>	Bool	<input type="text"/>	
2	<input type="text" value="DO2"/>	Bool	<input type="text"/>	
3	<input type="text" value="DO3"/>	Bool	<input type="text"/>	
4	<input type="text" value="DO4"/>	Bool	<input type="text"/>	
5	<input type="text" value="DO5"/>	Bool	<input type="text"/>	
6	<input type="text" value="DO6"/>	Bool	<input type="text"/>	
7	<input type="text" value="DO7"/>	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="DI0"/>	Bool	<input type="text"/>	

Modbus Mapping Table – Nickname	
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.

**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated. The M-7055D has no AI/AO, so here uses the screen of DL-302 for an example.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

**Bitwise:**

**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

The M-7055D has no AI/AO, so here uses other module's setting screen as an example.

Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.



● **Step 3. OPC UA Certificate**



Click the next step, and enter the **Step 3 [OPC UA Certificate]** of the UI setting. This step is about setting the OPC UA Certificate for the security and encryption, e.g. upload, download, delete certificate. **If the user's project does not need to use the secure encryption connection, please skip this step and click the next step directly.**

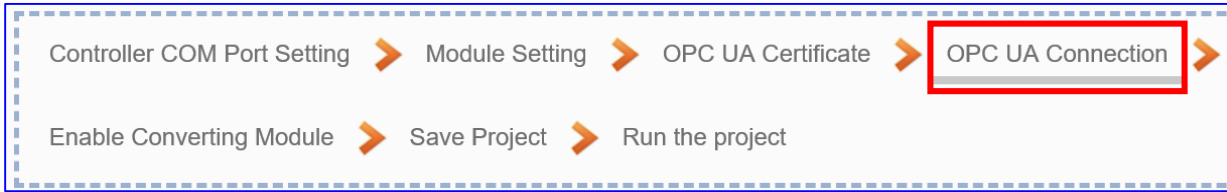
In the **[OPC UA Certificate]** step, users can add mutual credentials on both side's devices to strengthen security encryption.

- ① First, obtain the **OPC UA Client** trust certificate file of the device from the connected party, save it to the PC. In this step, select this file and upload it to the UA controller. (If there was an old certificate file in UA, remove it first.)
- ② The device of the other side needs the UA certificate also. In this step, download the **OPC UA Server** certificate file (**Certicate\_IPAddress\_.tar**) to the other party, so that they can decompress the file (**icpdasuaserver.der**) and upload to their device.

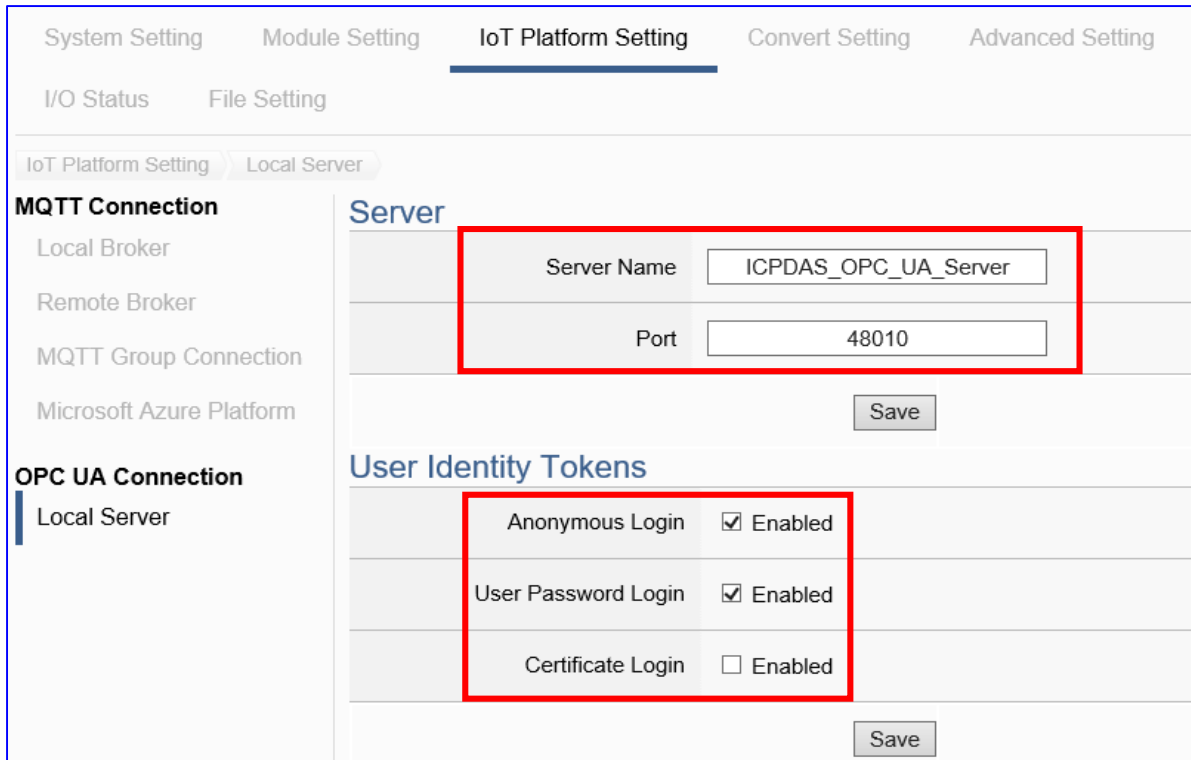


File Setting > OPC UA Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the OPC UA Trusted Certificate file in PC.</p> <p><b>Upload:</b> upload the Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>DER</b>. Extension name must be "<b>der / cer / crt</b>".</li> </ul> <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 5px 0;"> <span>Trusted Certificate</span> <span>Select File</span> <span>icpdasuaserver.der</span> <span>Upload</span> </div> <ul style="list-style-type: none"> <li>If select a wrong file, the system will show an error message.</li> </ul>
File Setting > OPC UA Certificate > Download the file from controller	
OPC UA Server Certificate	<p><b>Download:</b> Download the OPC UA Server Certificate file to the PC.</p> <ul style="list-style-type: none"> <li>File format: <b>DER</b>. File name: <b>Certicate_IP-address_.tar</b></li> </ul> <p>e.g.  <span style="border: 1px solid #ccc; padding: 2px;">Certicate_192.168.255.102.tar</span>. Before using, decompress to <b>icpdasuaserver.der</b>, as below.  <span style="border: 1px solid #ccc; padding: 2px;">icpdasuaserver.der</span></p>

● **Step 4. OPC UA Connection**



Click the next step, and enter the **Step 4 [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.



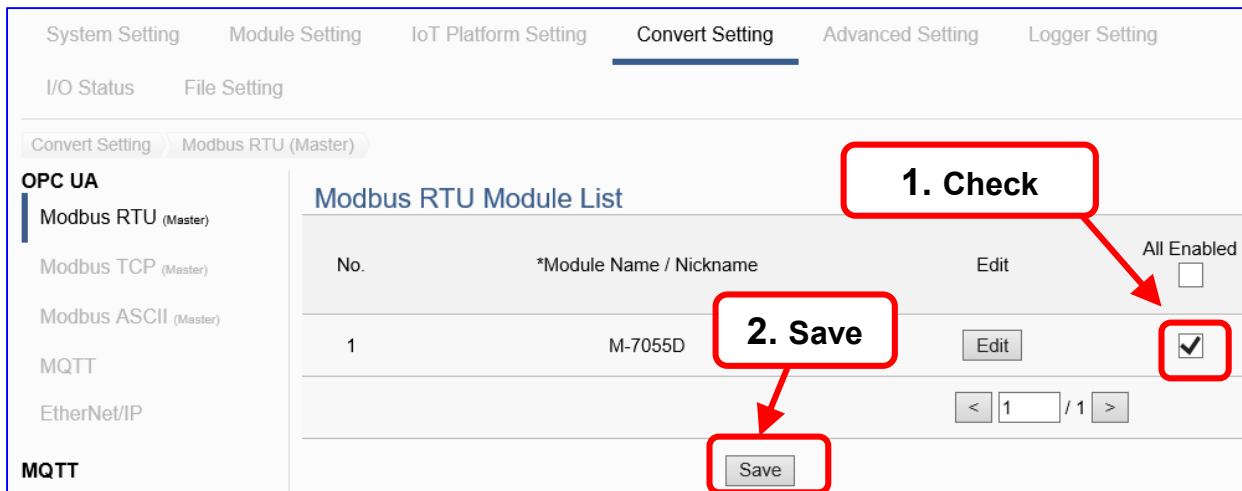
OPC UA Connection > Local Server Setting –Server	
Server Name	Display the active OPC UA Server name. Not editable. System values: ICPDAS_OPC_UA_Server
Port	The communication port number of the OPC UA Server. System Default: 48010.
Save	Click to save the settings of this item.
OPC UA Connection > Local 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 5. Enable Converting Module**



Click the next step, and enter the **Step 5 [Enable Converting Module]** UI setting  
 This step is for enabling the Modbus RTU (or ASCII) / OPC UA conversion.

In this step, user just need check the enabled box of the module. If user want to enable some I/O only, please click “Edit” to check the I/O one by one.



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.
< 1 / 1 >	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.

If users need to enable some channels only, please click [Edit] to enable individual channels.

If there had set up the Scaling or Bitwise in the step 2, the Scaling or Bitwise function will available only when the Enabled box is checked. So, in this step, be sure to check the Enabled box (as picture). For the setting method, please refer the Step 2.

Module Content Setting			
No.	3		
Module Name	M-7018Z		
Variable Table		I/O	Scaling
Name	Attribute	Data Type	Enabled
Scale_AI0	Read	Float	<input checked="" type="checkbox"/>
Scale_AI1	Read	Float	<input checked="" type="checkbox"/>
		OK	Cancel

More Descriptions:

### 1. Scaling:

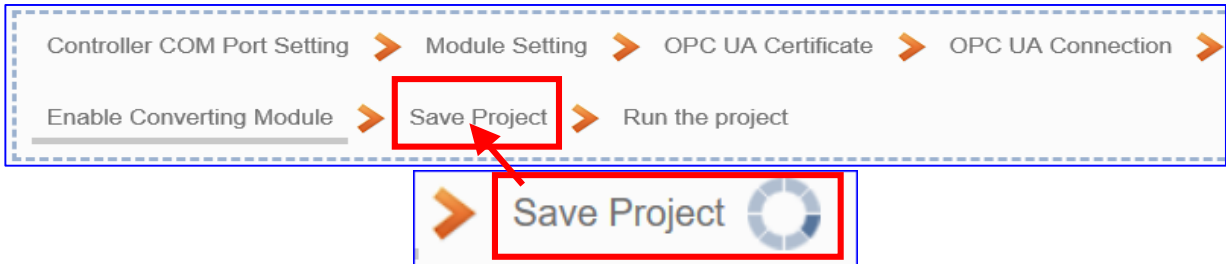
**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the "**Advanced Setting**" button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check "**Enable**" box, The Scaling conversion function will be activated.

### 2. Bitwise:

**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

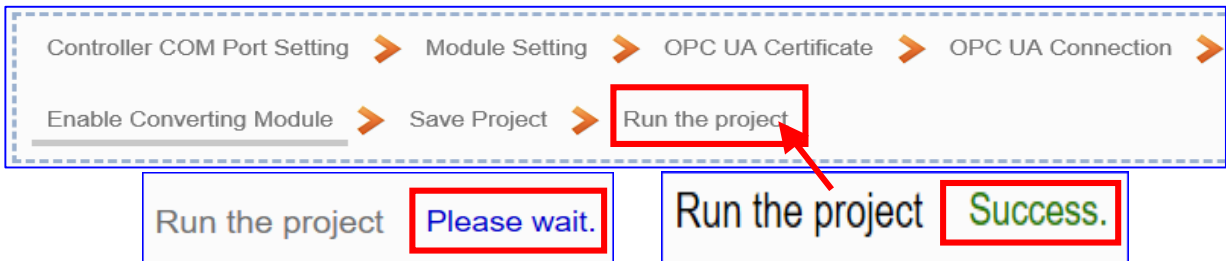
● **Step 6. Save Project**

The setting of this example is finished now. 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.



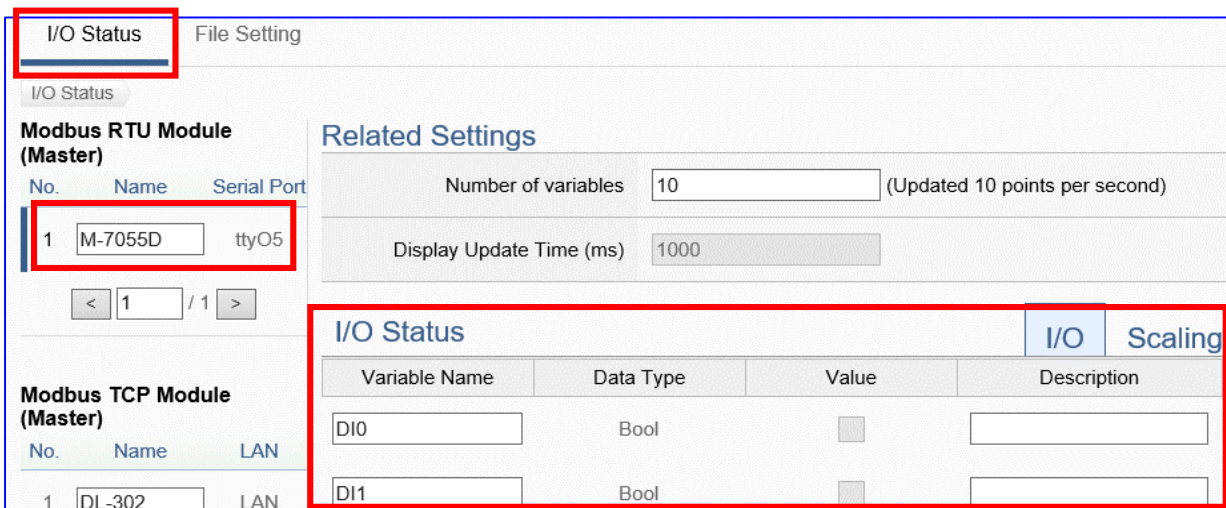
● **Step 7. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. Then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

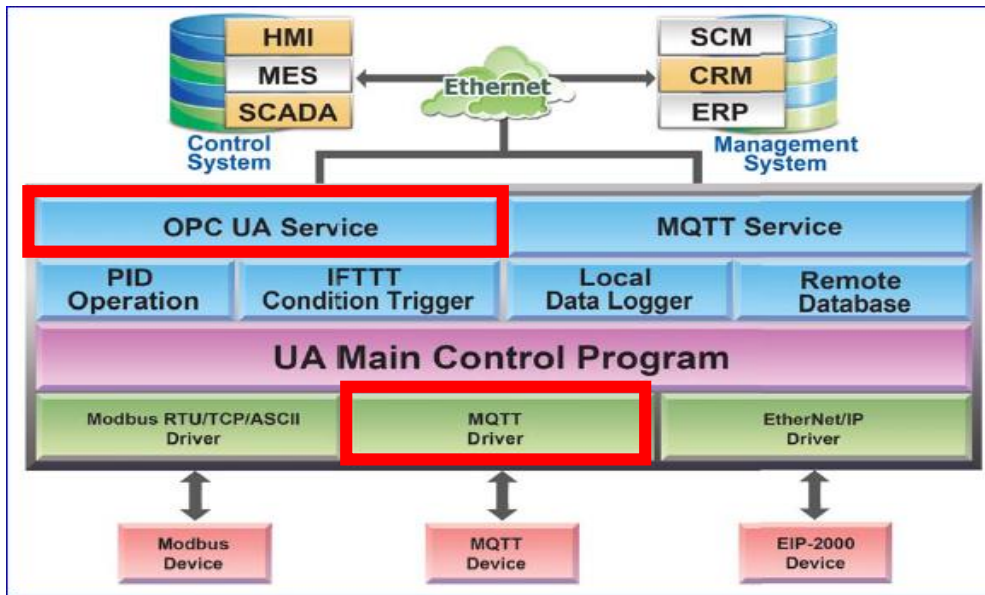
The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.



### 4.1.2 Function Wizard: MQTT / OPC UA (Use MQ-7255M)

MQTT / OPC UA Conversion include the conversion of OPC UA and MQTT protocols. With the **OPC UA Service** function, the OPC UA Server can read and write the **MQTT** device that connected to the controller.

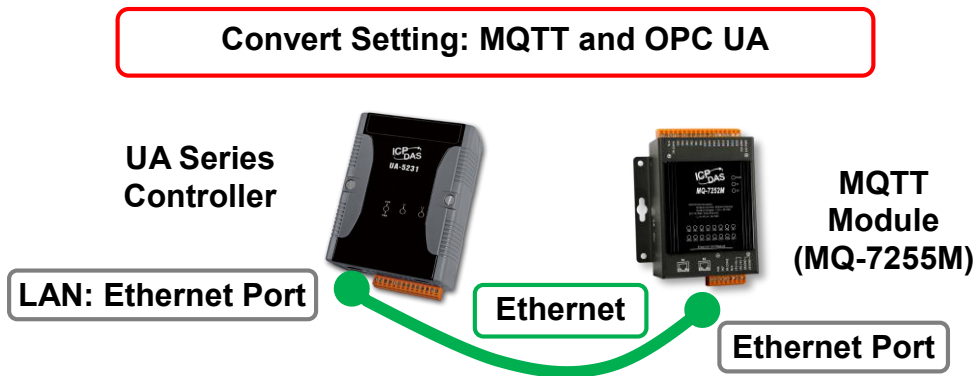
#### MQTT / OPC UA Function Diagram:



#### Application Solution:

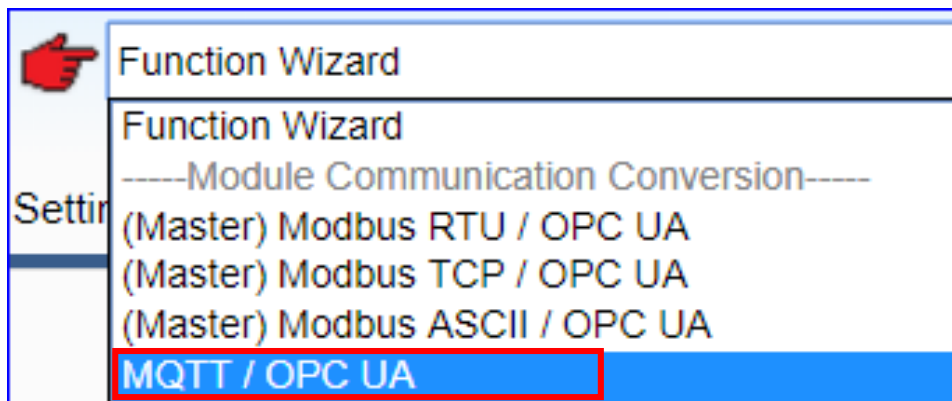


● **Convert Setting: MQTT and OPC UA**



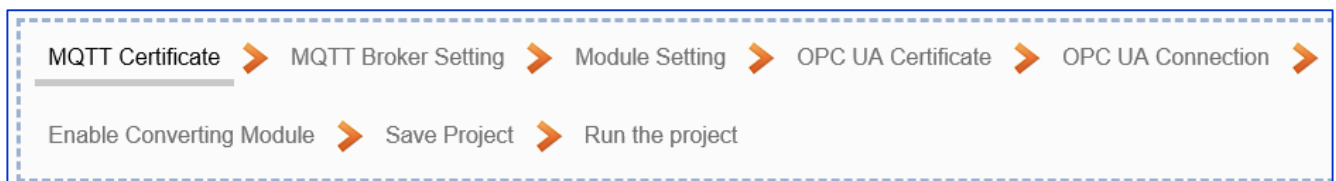
**Note:** The hardware/network connection methods please see the [Chapter 2](#) .

When UA series controller connects the MQTT module (via Ethernet, as MQ-7255M in the picture) and through the OPC UA server to read/write the I/O data of the MQTT module, user can choose the item **[MQTT / OPC UA]** of the “Module Communication Conversion” in the Function Wizard.

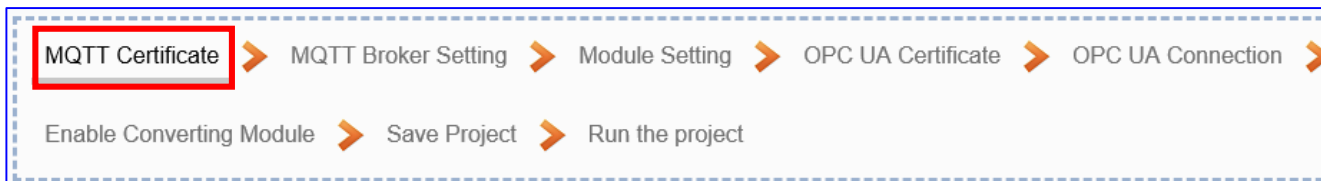


**[Step Box]:**

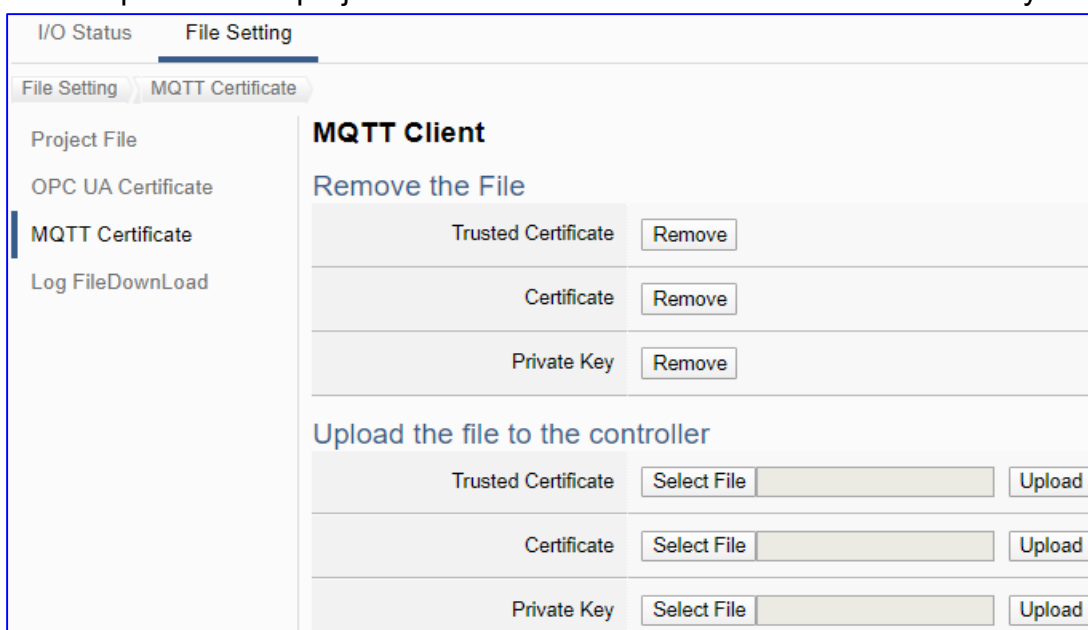
The Step Box of the **[MQTT / OPC UA]** has the steps as below. When enabling the Step Box, it auto-enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step-by-step and then can complete the project quickly and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



● **Step 1. MQTT Certificate**



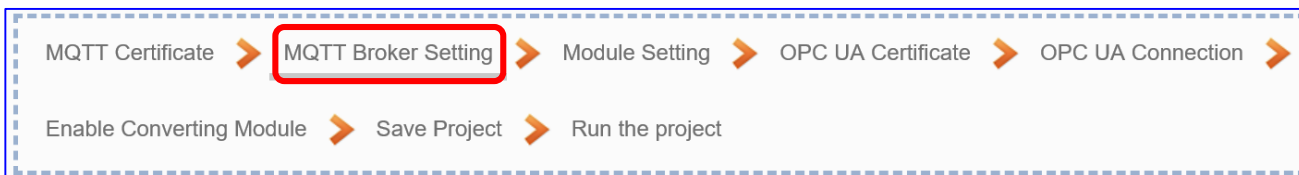
The [MQTT Certificate] is for setting up security communications to upload the **MQTT Trusted Certificate, Certificate and Private Key**. The users upload the file to the UA controller according to the type of obtained certificate. **If you want to perform Broker authentication, you need to upload the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



File Setting > MQTT Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the MQTT Trusted Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Certificate	<p><b>Select File:</b> select the MQTT Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Private Key	<p><b>Select File:</b> select the MQTT Private Key of the device.</p> <p><b>Upload:</b> upload the MQTT Private Key file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“.key”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>



● **Step 2. MQTT Broker Setting**

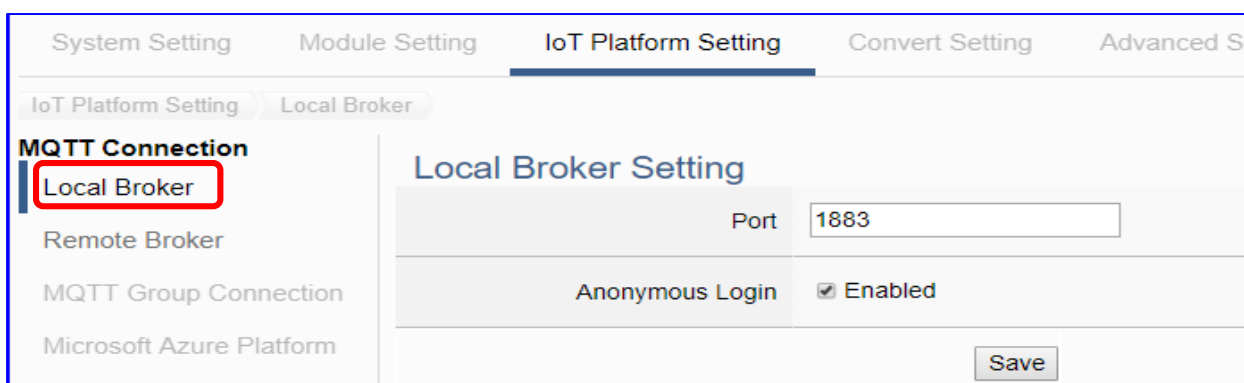


The **[MQTT Broker Setting]** is for setting the IoT platform and the MQTT Broker connection, e.g. the local or remote broker, port, login information, etc.

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

**The example uses local Broker.**

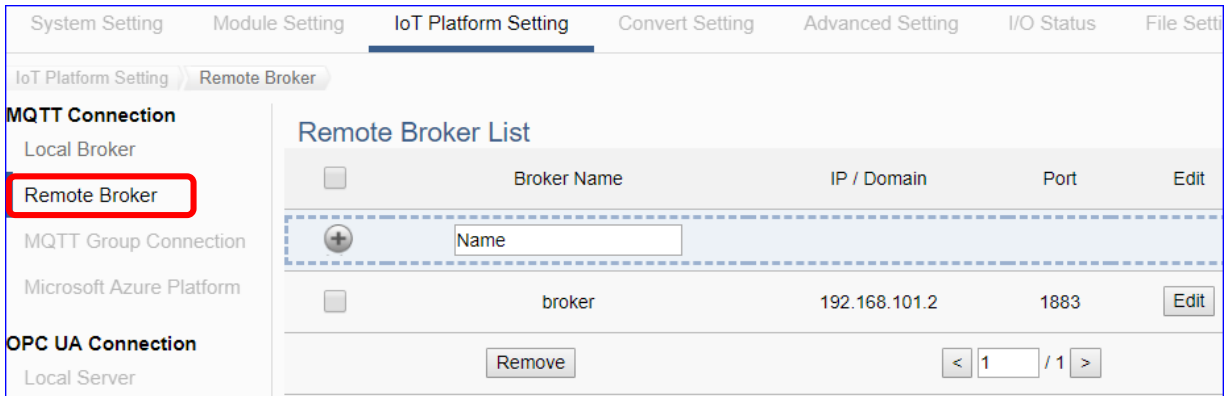
**Local Broker**



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.

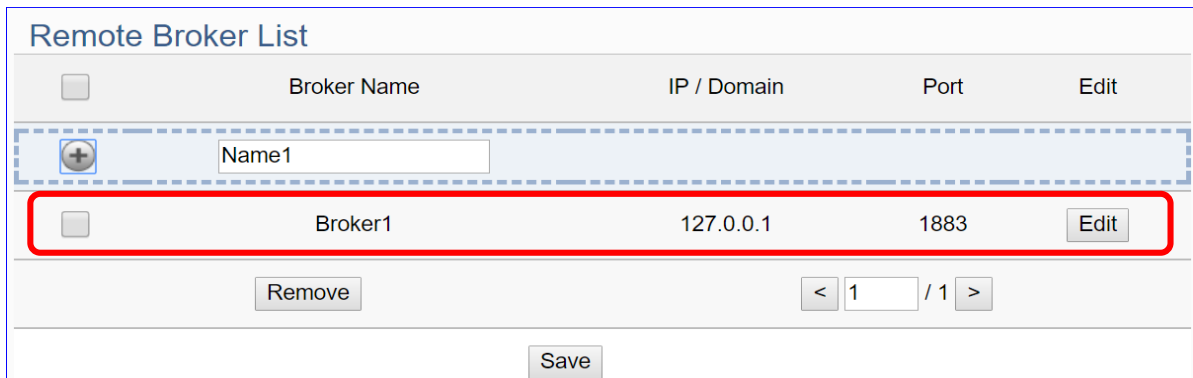
If users apply a remote Broker, the screen will as follow.

**Remote Broker:**



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):

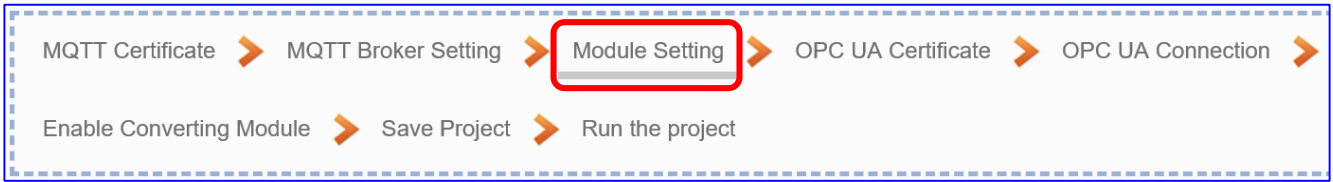


MQTT Connection > Remote Broker List	
Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Edit /	Click [Edit] can set the Broker.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

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

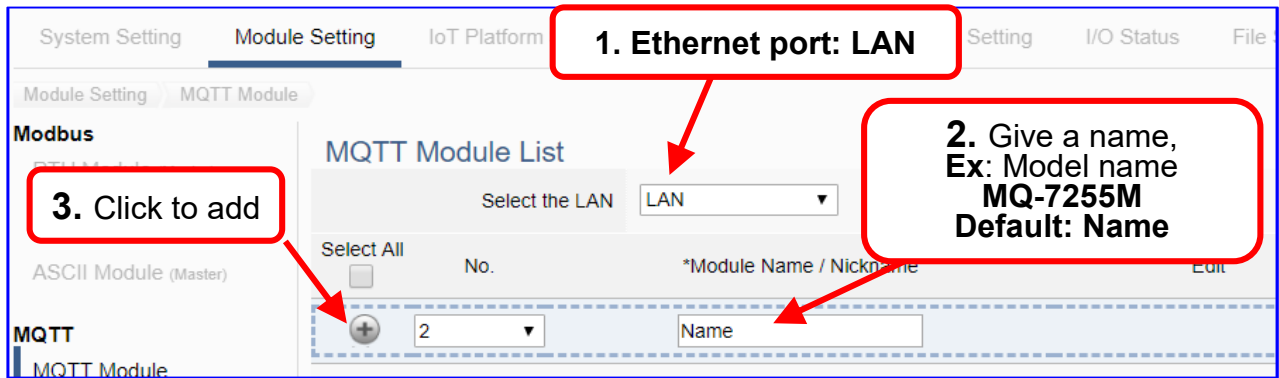
MQTT Connection > Remote Broker > Broker Content Settings	
Broker Name	The name of the remote MQTT Broker. (Editable)
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Keep Alive Time	The keep alive time. Default: 60 (second)
SSL/TLS	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. Module Setting**

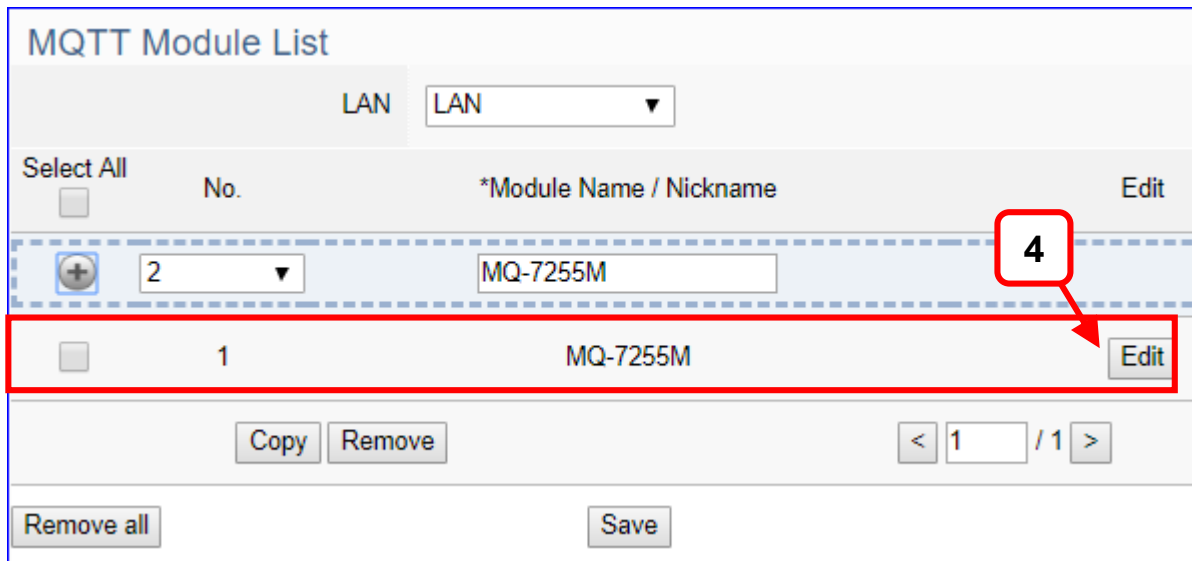


Click the next step, and enter the **Step 3 [Module Setting]**. This page is for setting the communication values of the connected modules.

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: MQ-7255M) 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.

[MQTT Client Setting] page:

**MQTT Client Setting**

No.	1
Module Name	MQ-7255M
MQTT Connection	<input checked="" type="radio"/> Broker (Local)

**MQTT Variable Setting**

Attribute	Read
Data Type	Bool
Data Number	1
Create Tables	<input type="button" value="Add"/>
Details	<input type="button" value="Show"/> <input type="button" value="Hide"/>

**> MQTT Variable Setting:**  
 Select attribute, data type and number of the module I/O, and click "Add".  
**Ex:** MQ-7255M, 8xDI, 8xDO  
 [DI] Attribute: Read, Type: Bool, Number: 8, click "Add"  
 [DO] Attribute: Read/Write, Type: Bool, Number: 8, click "Add"  
User can check the I/O data of module via user manual to set up the variable table.

MQTT Client 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.
MQTT Connection	The used Broker: Local Broker.
MQTT Variable Setting	
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the MQTT variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String.
Data Number	The number for the I/O variables of the module. Default: 1.
Create Tables	Click [Add] button, it will add a variable list in the MQTT Variable Table.
Details Show / Hide	Click [Show] to display all fields, click [Hide] to hide some fields. The hide fields: Subscribe QoS, Publish QoS, Retain.

Please create the variable table in the page. Select "Attribute", "Data Type" and "Data Number", and click "Add" button to create a variable table.

**Note:** The different "Attribute" variables need to create separately.

**[MQTT Variable Table] :**

MQTT Variable Table	
Details Show / Hide	Click [Show] to display all fields, click [Hide] to hide some fields. The hide fields: Subscribe QoS, Publish QoS and Retain.
Remove Table / Remove	Check the box in the left of the variable is to select that variable list, and click the “remove” on the box can delete that variable list. Click the “Remove” of the “Remove Table” will delete all lists.
Name	The name of the MQTT variable. Default: Tag#
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String
Subscribe Topic	The topic of receiving/subscribing data message. It can copy the Publish Topic of linked module, e.g. MQ-7255M in this example.
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. It can copy the Subscribe Topic of linked module, e.g. MQ-7255M in this example.
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.
Description	For users set up the description for the variables.
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.

MQTT Variable Table	
OK / Cancer	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.

The Subscribe / Publish Topic items must fill the related Topics of the connected MQTT module, e.g. UA connects with MQ-7255M in this case.

User can find the Topics from the MQTT setting of MQ-7200 Web page, and copy them to the UA setting:

Copy the Subscriptions I/O Topic of MQ-7200 to the I/O Publish Topic of UA, and copy the Publications I/O Topic of MQ-7200 to the I/O Subscribe Topic of UA.

The screenshot shows the ICP DAS MQTT configuration interface. The left sidebar contains a menu with 'MQTT' highlighted. The main content area is divided into two sections: 'Subscriptions' and 'Publications'.

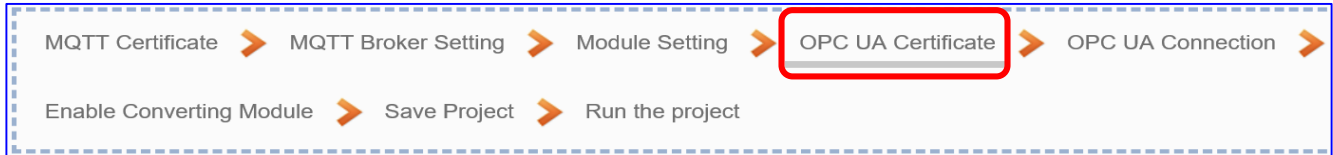
**Subscriptions Table:**

I/O	No.	Topic
Digital Output	0	MQ7255M_656660/SetValue/DO0
Digital Output	1	MQ7255M_656660/SetValue/DO1
Digital Output	2	MQ7255M_656660/SetValue/DO2
Digital Output	3	MQ7255M_656660/SetValue/DO3
Digital Output	4	MQ7255M_656660/SetValue/DO4
Digital Output	5	MQ7255M_656660/SetValue/DO5
Digital Output	6	MQ7255M_656660/SetValue/DO6
Digital Output	7	MQ7255M_656660/SetValue/DO7

**Publications Table:**

I/O	No.	Topic
Digital Output	0	MQ7255M_656660/GetValue/DO0
Digital Output	1	MQ7255M_656660/GetValue/DO1
Digital Output	2	MQ7255M_656660/GetValue/DO2
Digital Output	3	MQ7255M_656660/GetValue/DO3
Digital Output	4	MQ7255M_656660/GetValue/DO4
Digital Output	5	MQ7255M_656660/GetValue/DO5
Digital Output	6	MQ7255M_656660/GetValue/DO6
Digital Output	7	MQ7255M_656660/GetValue/DO7
Digital Input	0	MQ7255M_656660/GetValue/DI0
Digital Input	1	MQ7255M_656660/GetValue/DI1
Digital Input	2	MQ7255M_656660/GetValue/DI2
Digital Input	3	MQ7255M_656660/GetValue/DI3

● **Step 4. OPC UA Certificate**



Click the next step, and enter the **Step 4 [OPC UA Certificate]** of the UI setting. This step is about setting the OPC UA Certificate for the security and encryption, e.g. upload, download, delete certificate. **If the user's project does not need to use the secure encryption connection, please skip this step and click the next step directly.**

In the **[OPC UA Certificate]** step, users can add mutual credentials on both side's devices to strengthen security encryption.

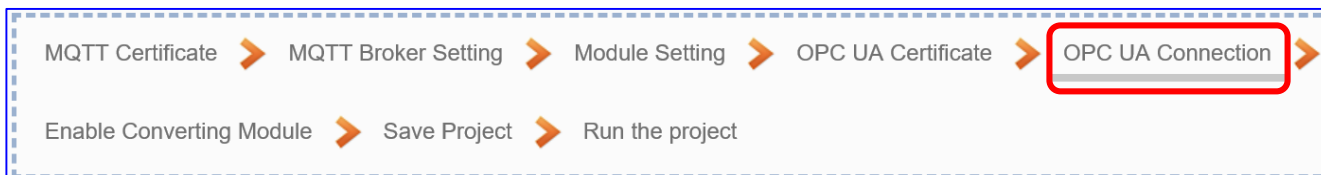
- ① First, obtain the **OPC UA Client** trust certificate file of the device from the connected party, save it to the PC. In this step, select this file and upload it to the UA controller. (If there was an old certificate file in UA, remove it first.)
- ② The device of the other side needs the UA certificate also. In this step, download the **OPC UA Server** certificate file (**Certicate\_IPAddress\_.tar**) to the other party, so that they can decompress the file (**icpdasuaserver.der**) and upload to their device.



File Setting > OPC UA Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the OPC UA Trusted Certificate file in PC.</p> <p><b>Upload:</b> upload the Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>• File format must be <b>DER</b>. Extension name must be "<b>der / cer / crt</b>".</li> </ul> <div style="border: 1px solid #ccc; padding: 2px; width: fit-content; margin: 5px 0;"> <span>Trusted Certificate</span> <span>Select File</span> <span>icpdasuaserver.der</span> <span>Upload</span> </div> <ul style="list-style-type: none"> <li>• If select a wrong file, the system will show an error message.</li> </ul>
File Setting > OPC UA Certificate > Download the file from controller	
OPC UA Server Certificate	<p><b>Download:</b> Download the OPC UA Server Certificate file to the PC.</p> <ul style="list-style-type: none"> <li>• File format: <b>DER</b>. File name: <b>Certicate_IP-address_.tar</b></li> </ul> <p>e.g.  <span style="border: 1px solid #ccc; padding: 2px;">Certificate_192.168.255.102.tar</span>. Before using, decompress to <b>icpdasuaserver.der</b>, as below.  <span style="border: 1px solid #ccc; padding: 2px;">icpdasuaserver.der</span></p>

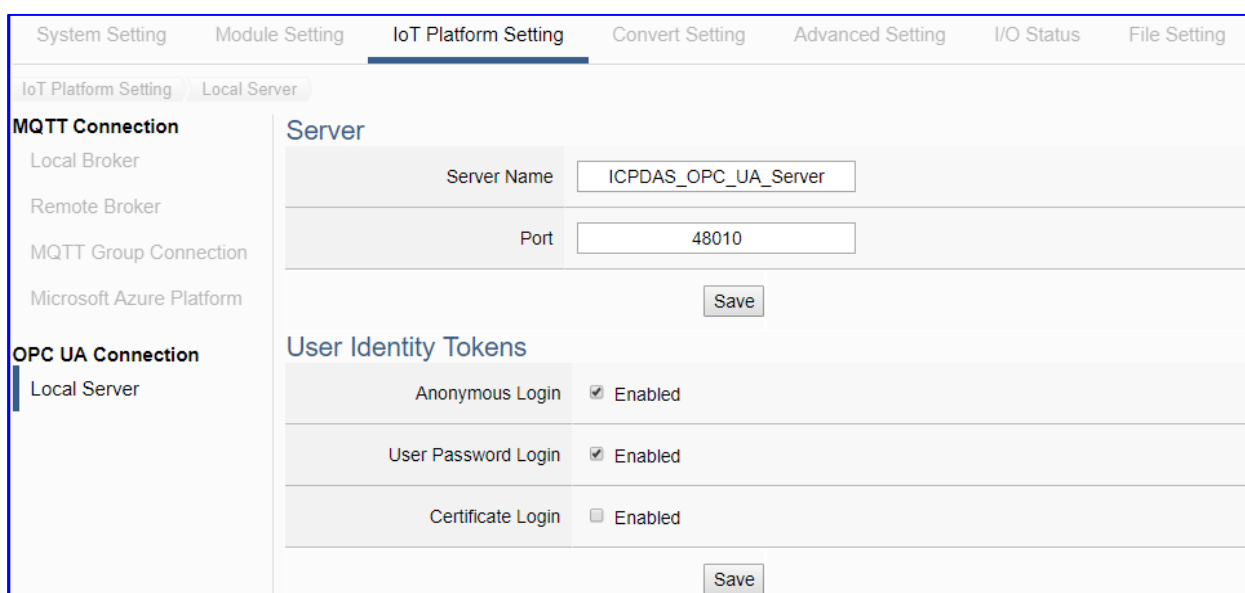


● **Step 5. OPC UA Connection**



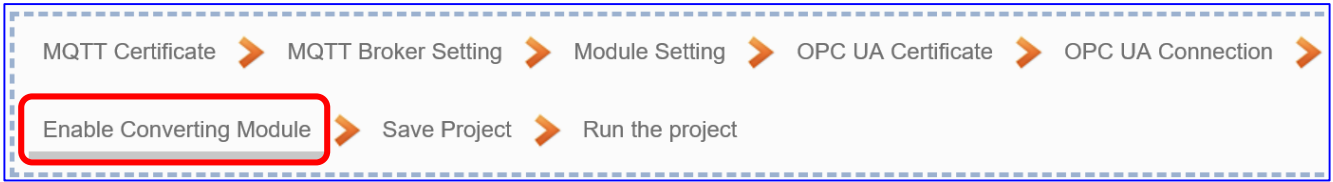
Click the next step, and enter the **Step 5 [OPC UA Connection]** of the UI setting. This page is for setting the IoT platform and the OPC UA connection, e.g. the server name, port, login identity information, etc.

We select the “MQTT / 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 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 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 6. Enable Converting Module**

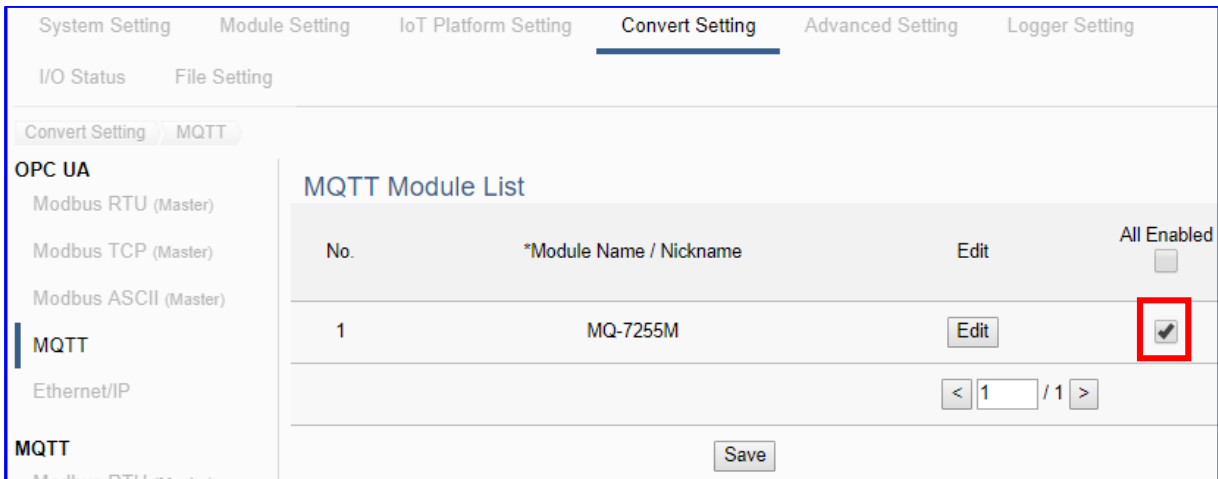


Click the next step, and enter the **Step 6 [Enable Converting Module]** UI setting

This step is for enabling the MQTT / OPC UA conversion.

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

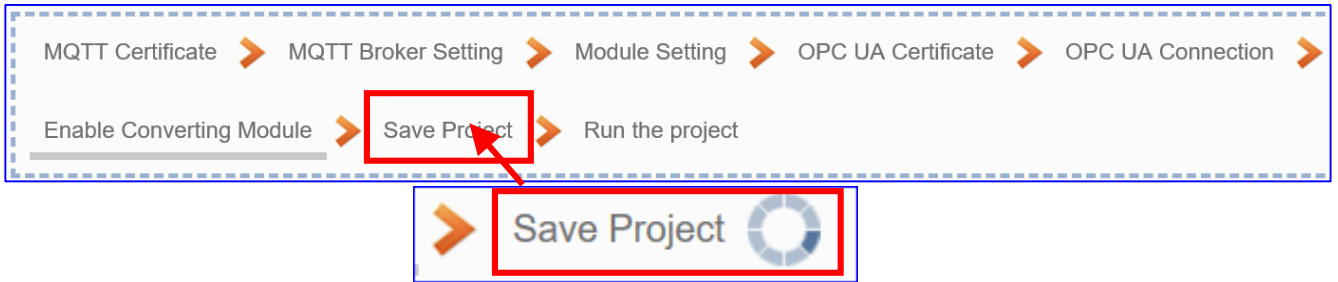
This step: Please check the box of the module to enable the converting.



Convert Setting > OPC UA > MQTT - MQTT 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. It is normal to set all channels as enabled, and the conversion will not affect the unconnected channels.
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.
< 1 / 1 >	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.

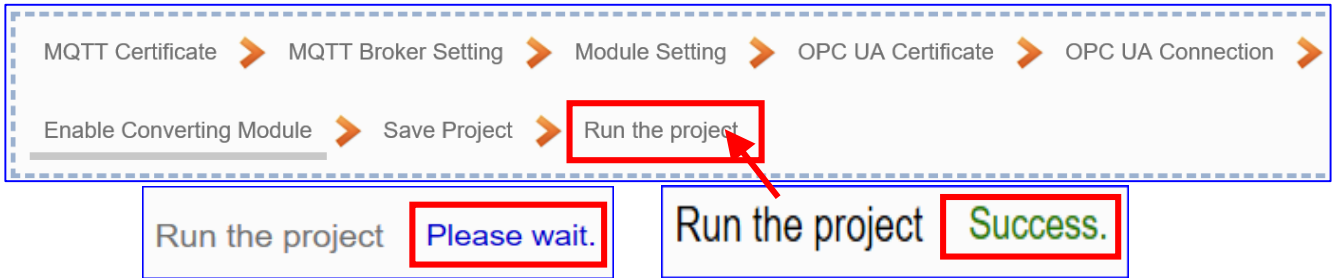
● **Step 7. Save Project**

The setting of this example is finished now. 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.



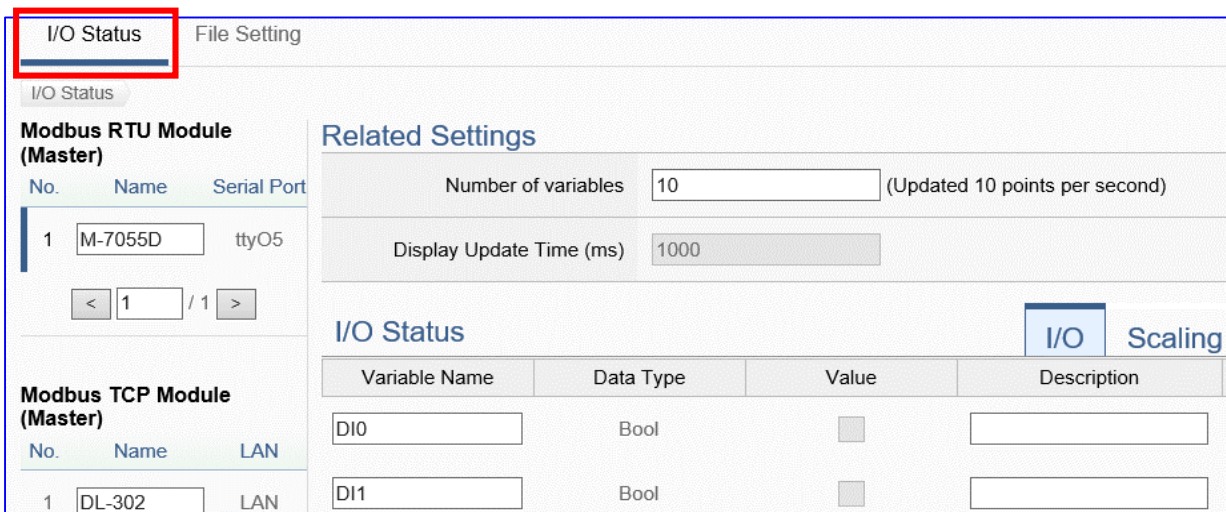
● **Step 8. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. Then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

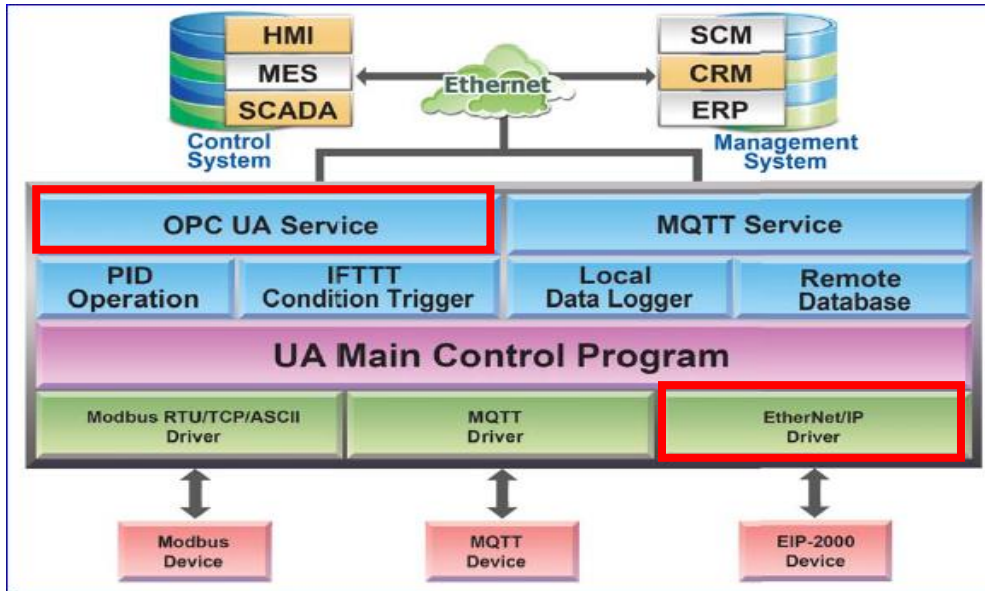
The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.



### 4.1.3 Function Wizard: EIP / OPC UA (Use EIP-2060)

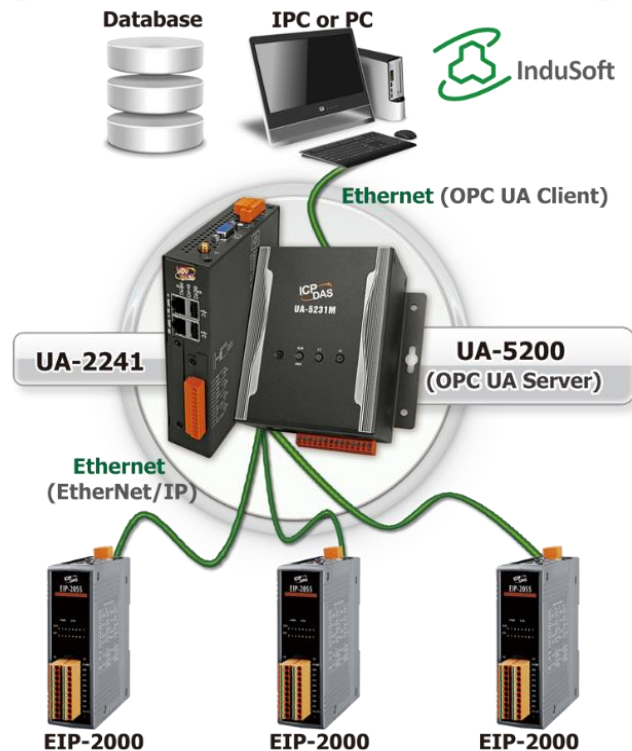
EIP / OPC UA Conversion include the conversion of OPC UA and **EtherNet/IP** protocols. With the **OPC UA Service** function, the OPC UA Server can read and write the EIP-2000 device that connected to the controller.

(EtherNet/IP) EIP / OPC UA Function Diagram:

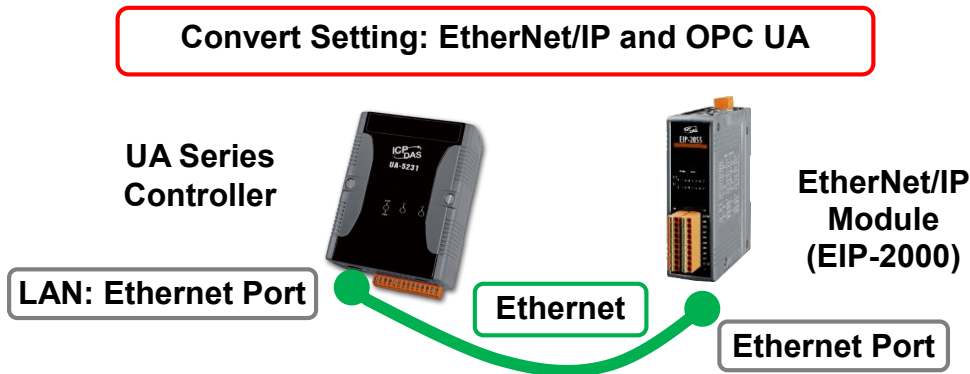


Application Solution:

### (EtherNet/IP ↔ OPC UA)

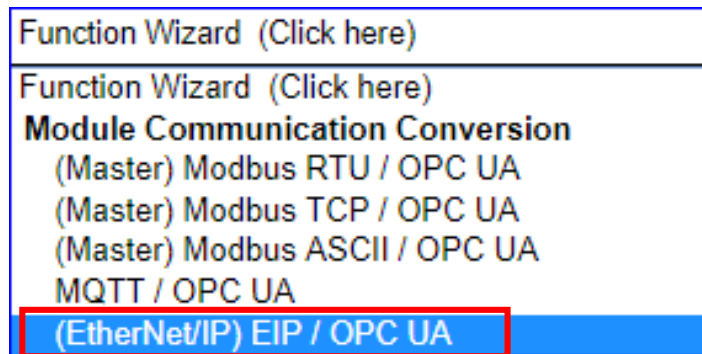


● **Convert Setting: EIP and OPC UA**



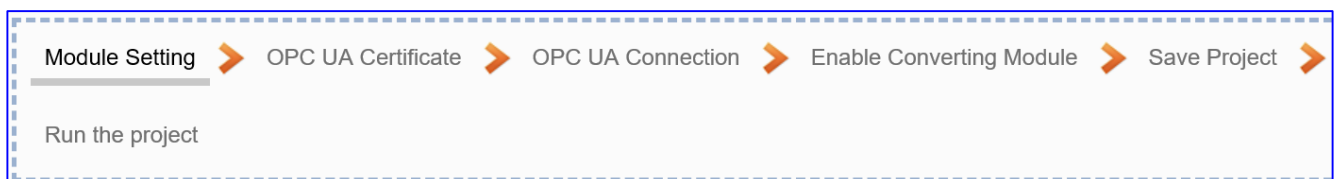
**Note:** The hardware/network connection methods please see the [Chapter 2](#) .

When UA series controller connects the EIP-2000 module (via Ethernet, as EIP-2060 in the picture) and through the OPC UA server to read/write the I/O data of the EIP-2000 module, user can choose the item **[(EtherNet/IP) EIP / OPC UA]** of the “Module Communication Conversion” in the Function Wizard.



**[Step Box]:**

The Step Box of the **[(EtherNet/IP) EIP / OPC UA]** has the 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.

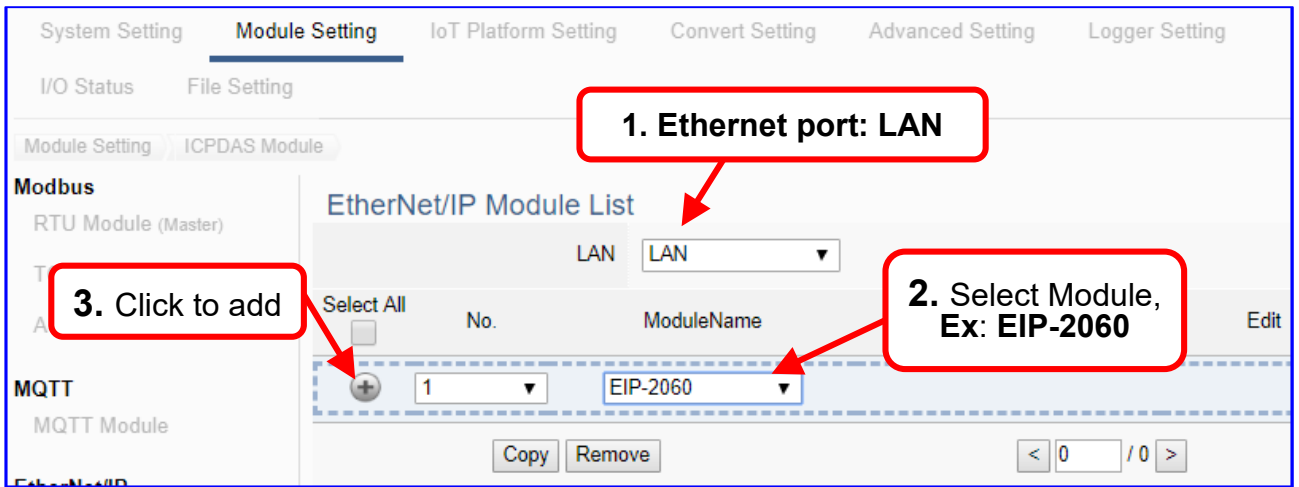


● **Step 1. Module Setting**

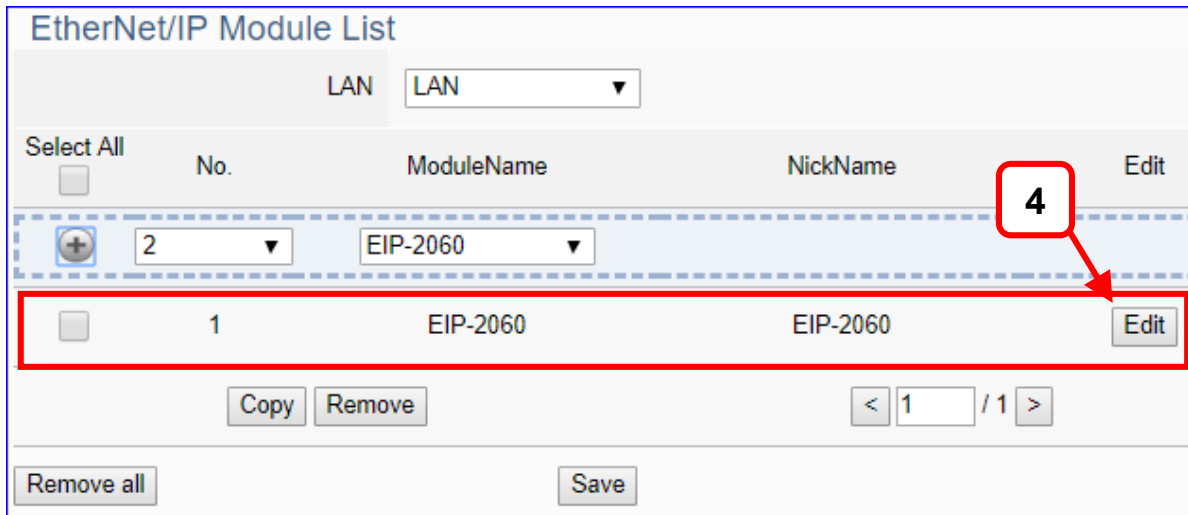


It auto-enters the first step of **Step 1 [Module Setting]**. This page is for setting the communication values of the connected modules.

The Ethernet port is LAN for connecting with the EtherNet/IP module. Select one EIP-2000 series model and click [ + ] button could add a new module.



Add a module (No.: 1, Name: EIP-2060) as below. 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:

Module Content Setting				
No.	<input type="text" value="1"/>			
Module Name	<input type="text" value="EIP-2060"/>			
NickName	<input type="text" value="EIP-2060"/>			
IP	<input type="text" value="192"/>	<input type="text" value="168"/>	<input type="text" value="13"/>	<input type="text" value="5"/>
ChannelNumber	<input type="text" value="12-ch(6DI+6DO)"/>			

User enters the real connected IP address.

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	The selected model number. (Not editable here)
NickName	User can give a nick name, default: selected model number
IP	Enter the IP address of the connected EIP module. This example: IP address of the EIP-2060 is 192.168.13.5
ChannelNumber	Select the number of the I/O channels.

The system will auto-display the selected I/O table by the order of Digital Input / Digital Output / Analogy Input / Analogy Output. This example: EIP-2060 have 6 DI and 6 DO.

Digital Input				
Channel	Name	Attributes	Data Type	Description
0	<input type="text" value="DI0"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
1	<input type="text" value="DI1"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
2	<input type="text" value="DI2"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
3	<input type="text" value="DI3"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
4	<input type="text" value="DI4"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
5	<input type="text" value="DI5"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>

Digital Output				
Channel	Name	Attributes	Data Type	Description
0	<input type="text" value="DO0"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>
1	<input type="text" value="DO1"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>
2	<input type="text" value="DO2"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>
3	<input type="text" value="DO3"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>
4	<input type="text" value="DO4"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>
5	<input type="text" value="DO5"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>

Analogy Input				
Channel	Name	Attributes	Data Type	Description

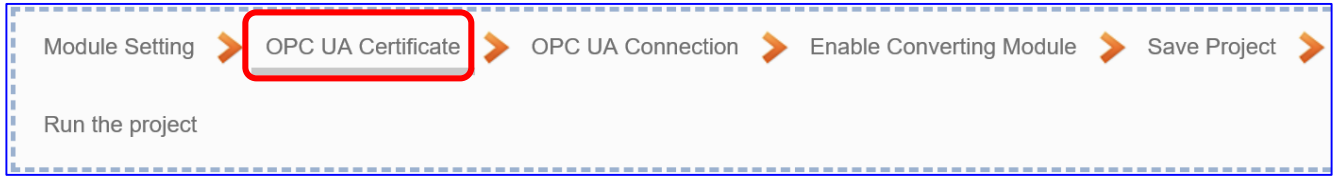
  

Analogy Output				
Channel	Name	Attributes	Data Type	Description

Digital Input / Digital Output / Analogy Input / Analogy Output	
Channel	Channel number will auto-display according to the model. (Not editable) Default: Number from 0.
Name	User can define the name. Default: DI#, DO#, AI#, AO# Available: number, English character, underline “_”, dash line “-”, cannot be a space, slash “/”, Chinese character, and other symbols.
Attributes	Display data attribute of the variable. (Not editable) Include Read, Read/Write...
Data Type	Display data type of the variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String
Description	For users set up the description for the channel.
OK / Cannel	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.



● **Step 2. OPC UA Certificate**



Click the next step, and enter the **Step 2 [OPC UA Certificate]** of the UI setting. This step is about setting the OPC UA Certificate for the security and encryption, e.g. upload, download, delete certificate. **If the user's project does not need to use the secure encryption connection, please skip this step and click the next step directly.**

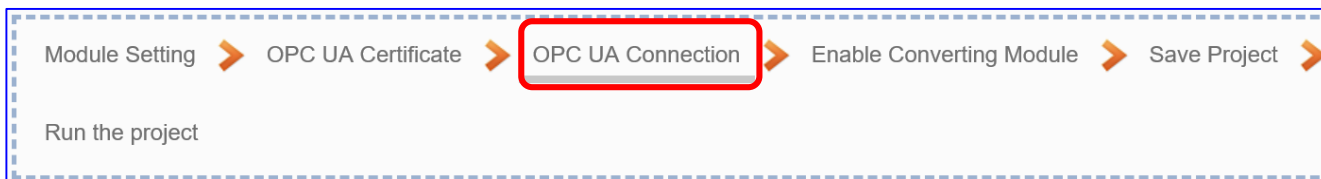
In the **[OPC UA Certificate]** step, users can add mutual credentials on both side's devices to strengthen security encryption.

- ① First, obtain the **OPC UA Client** trust certificate file of the device from the connected party, save it to the PC. In this step, select this file and upload it to the UA controller. (If there was an old certificate file in UA, remove it first.)
- ② The device of the other side needs the UA certificate also. In this step, download the **OPC UA Server** certificate file (**Certificate\_IPAddress\_.tar**) to the other party, so that they can decompress the file (**icpdasuaserver.der**) and upload to their device.



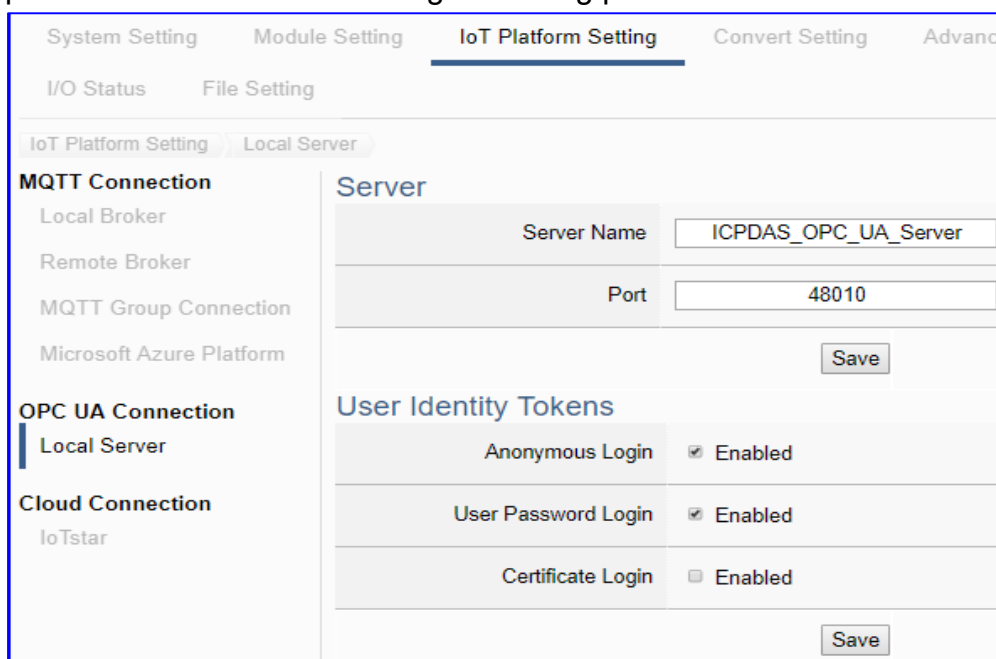
File Setting > OPC UA Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the OPC UA Trusted Certificate file in PC.</p> <p><b>Upload:</b> upload the Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>• File format must be <b>DER</b>. Extension name must be <b>“der / cer / crt”</b>.</li> </ul> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> <span>Trusted Certificate</span> <span>Select File</span> <span>icpdasuaserver.der</span> <span>Upload</span> </div> <ul style="list-style-type: none"> <li>• If select a wrong file, the system will show an error message.</li> </ul>
File Setting > OPC UA Certificate > Download the file from controller	
OPC UA Server Certificate	<p><b>Download:</b> Download the OPC UA Server Certificate file to the PC.</p> <ul style="list-style-type: none"> <li>• File format: <b>DER</b>. File name: <b>Certificate_IP-address_.tar</b></li> </ul> <p>e.g.  <span>Certificate_192.168.255.102.tar</span>. Before using, decompress to <b>icpdasuaserver.der</b>, as below.  <span>icpdasuaserver.der</span></p>

● **Step 3. OPC UA Connection**



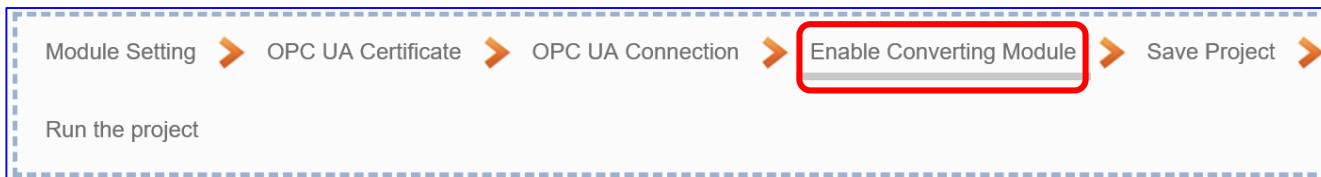
Click the next step, and enter the **Step 3 [OPC UA Connection]** of the UI setting. This page is for setting the IoT platform and the OPC UA connection, e.g. the server name, port, login identity information, etc.

We select the “EIP / 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.



<b>OPC UA Connection &gt; Local Server Setting –Server</b>	
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.
<b>OPC UA Connection &gt; Local Server Setting –User Identity Tokens</b>	
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.

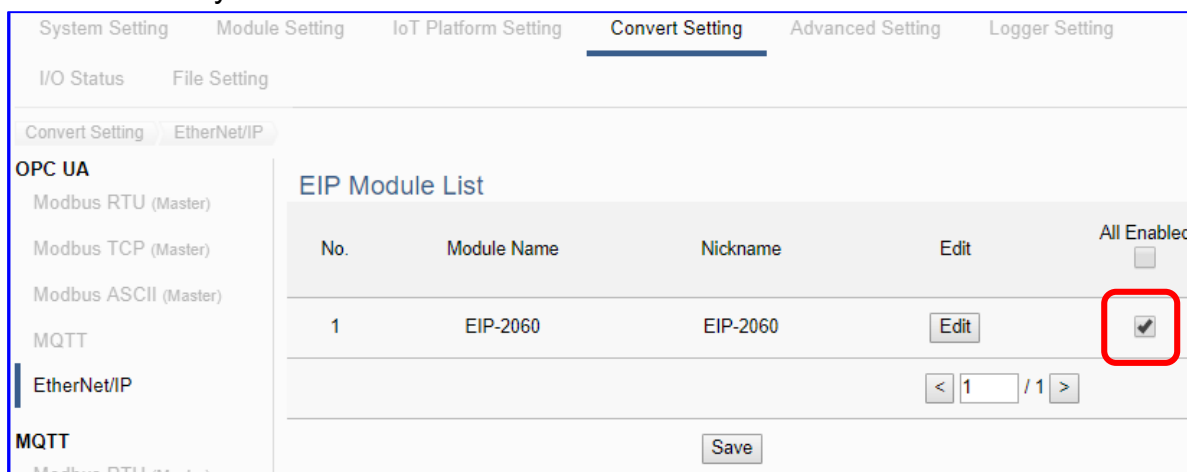
● **Step 4. Enable Converting Module**



Click the next step, and enter the **Step 4 [Enable Converting Module]** UI setting  
 This step is for enabling the EIP / OPC UA conversion.

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

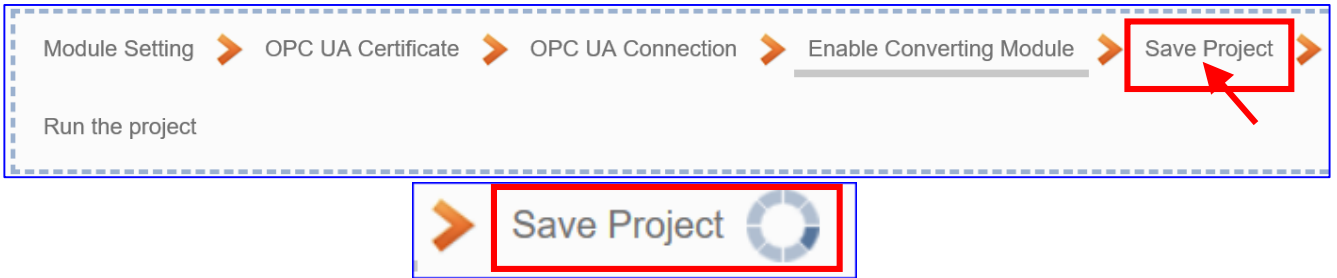
Check the box of the module to enable all I/O. If want to enable some I/O, click “Edit” to select I/O one by one.



Convert Setting > OPC UA > EtherNet/IP - EIP 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. It is normal to set all channels as enabled, and the conversion will not affect the unconnected channels.
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.
< 1 / 1 >	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.

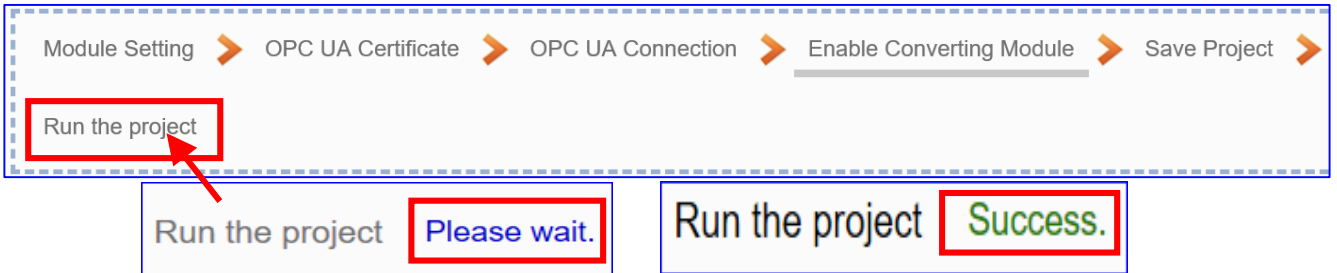
● **Step 5. Save Project**

The setting of this example is finished now. 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.



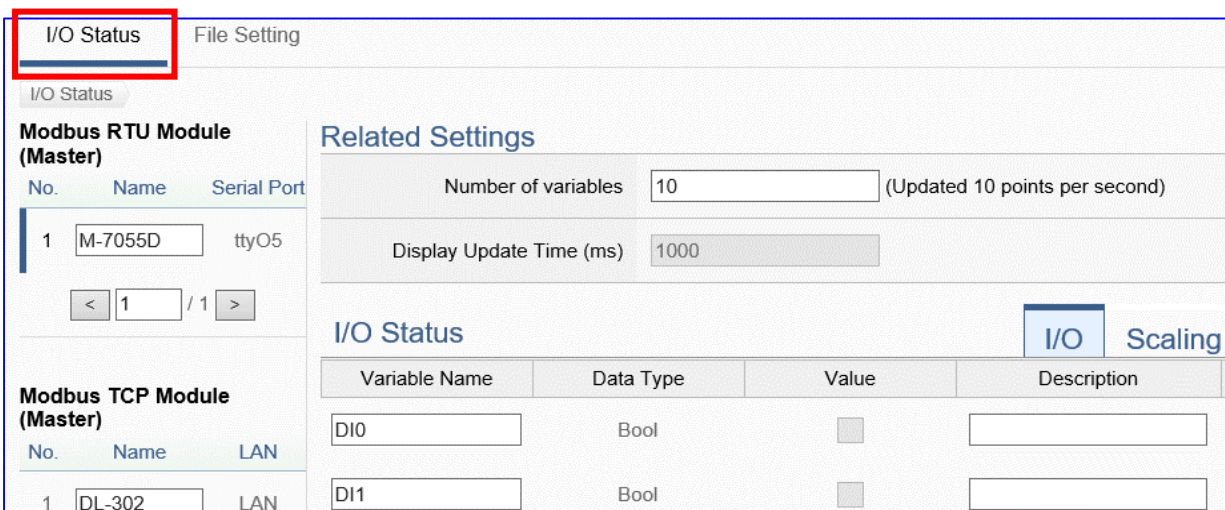
● **Step 6. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. Then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

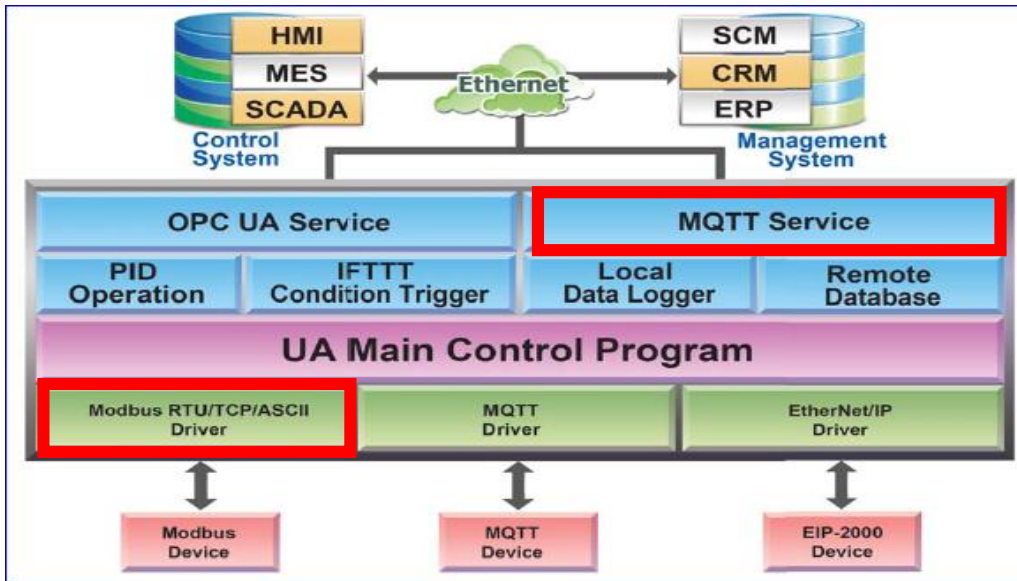
The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.



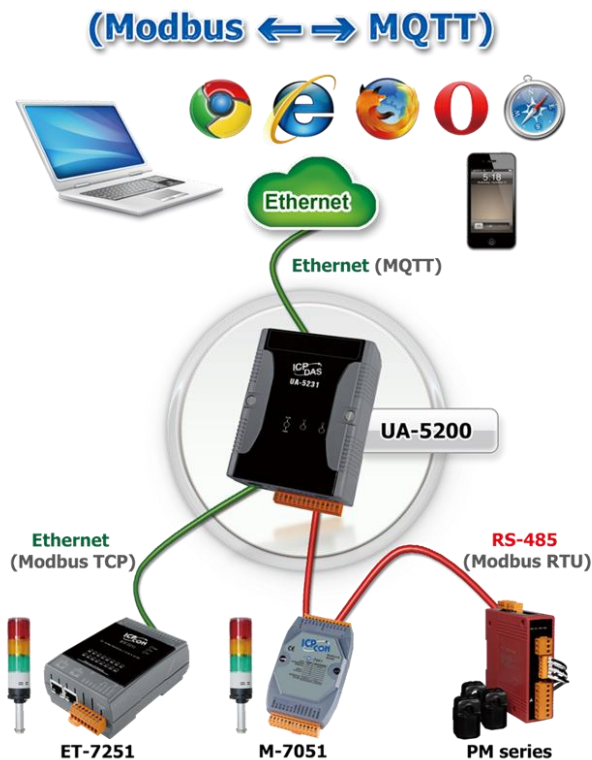
### 4.1.4 Function Wizard: Modbus / MQTT (TCP Example, Use DL-302)

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:

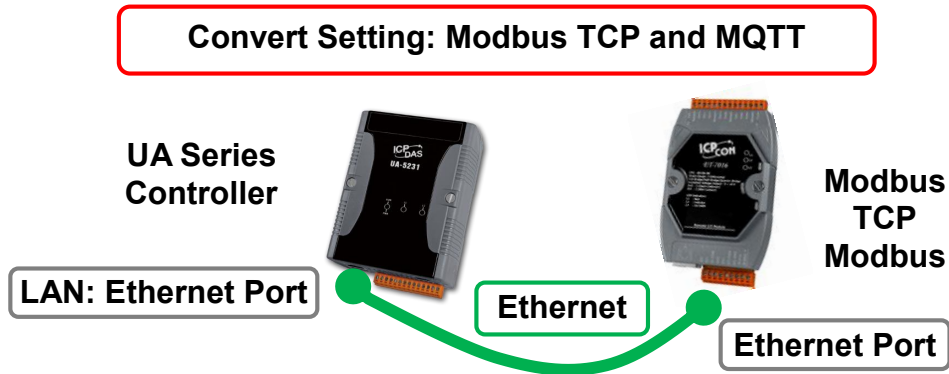


#### Application Solution:



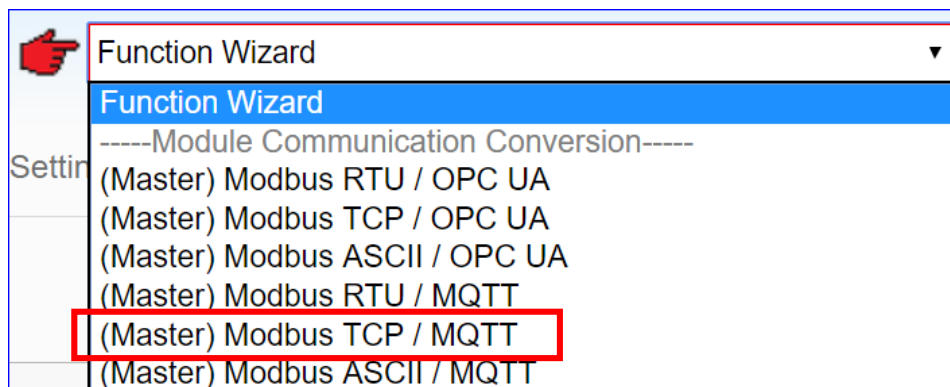
This section introduces the Modbus / MQTT conversion through the conversion of Modbus TCP and MQTT protocol.

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



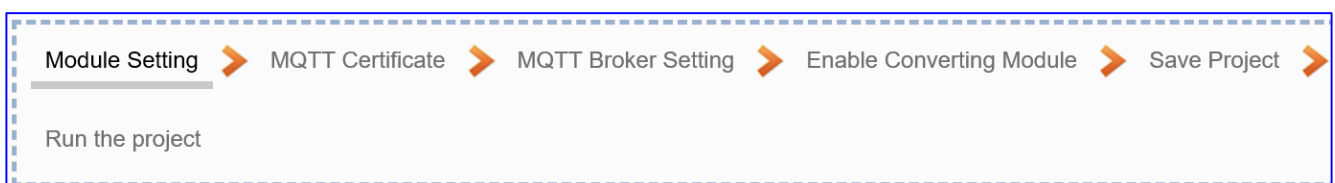
**Note:** The hardware/network connection methods please see the [Chapter 2](#) .

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.

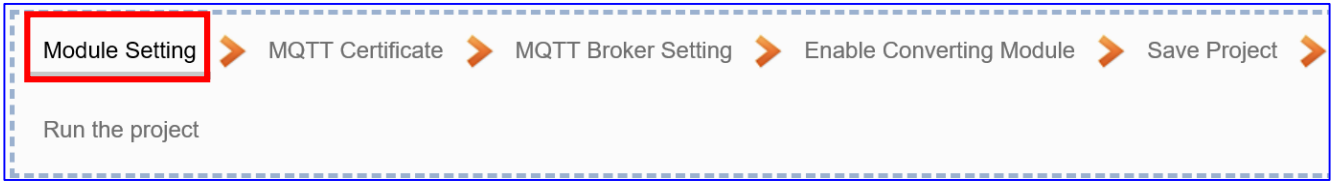


**[Step Box]:**

The Step Box of the [**Modbus TCP / MQTT**] has the 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.

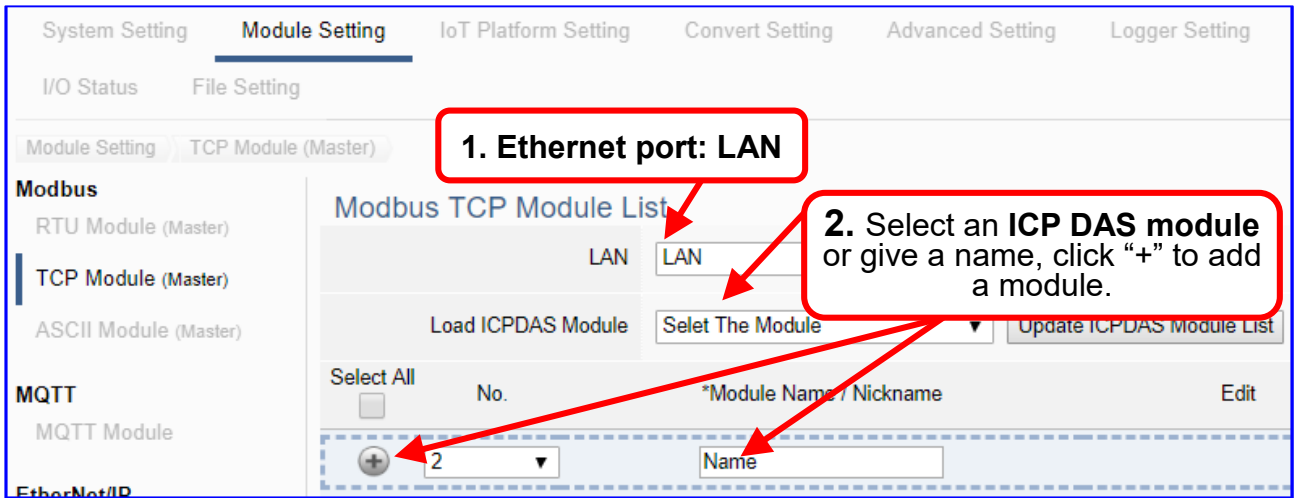


● **Step 1. Module Setting**

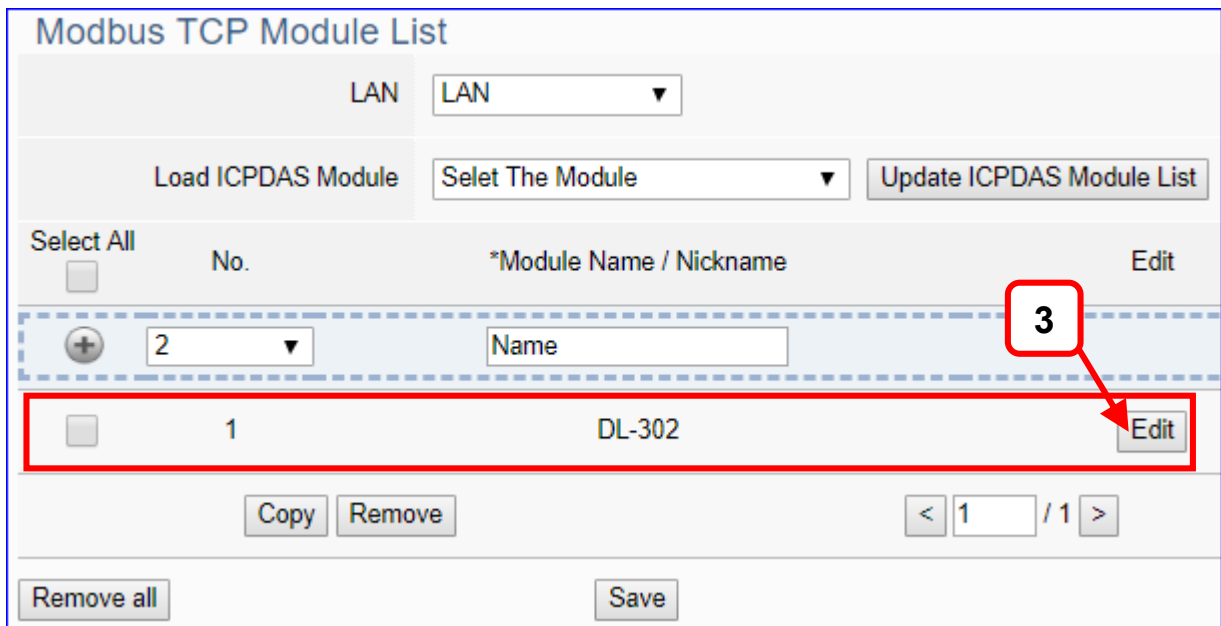


This page is for setting the communication values of the connected modules.

The Ethernet port is LAN for connecting with the TCP module. If using ICP DAS module, select the module and system will auto load the module data. If not, give a module name (Default: Name), click [ + ] button to add a new module.



Add a module (e.g. No.: 1, Name: DL-302) 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 to set up IP and the Modbus address mapping table.

Module Content Setting	
No.	1
Module Name	DL-302
IP	192 . 168 . 81 . 251
Port	502
Slave ID	1
Timeout(ms)	500
Polling Rate(ms)	500
Modbus Mapping Table Setting	
Data Model	01 Coil Status(0x)
Start Address	0
Data Number	1
Create Tables	<input type="button" value="Add"/>

**This Example: DL-302**

**[IP] 192.168.81.251 (by user case)**

**[Modbus Mapping Table Setting]**  
**Data Model: 04 Input Registers(3x)**  
**Start Address: 0**  
**Data Number: 6**  
**Type: 16-bit Short**  
**→ Click [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	Give 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. (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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise
Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)		
				Address	0
				Number	6
				Type	Short
				<input type="button" value="Edit"/>	

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>02 Input Status(1x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>03 Holding Registers(4x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
<b>04 Input Registers(3x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	CO2	Short	<input type="checkbox"/>					
1	Relative_humidity	Short	<input type="checkbox"/>					
2	Temperature_Celsius	Short	<input type="checkbox"/>					
3	Temperature_Fahrenheit	Short	<input type="checkbox"/>					
4	Dew_point_temperature	Short	<input type="checkbox"/>					

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.

**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

**Bitwise:**

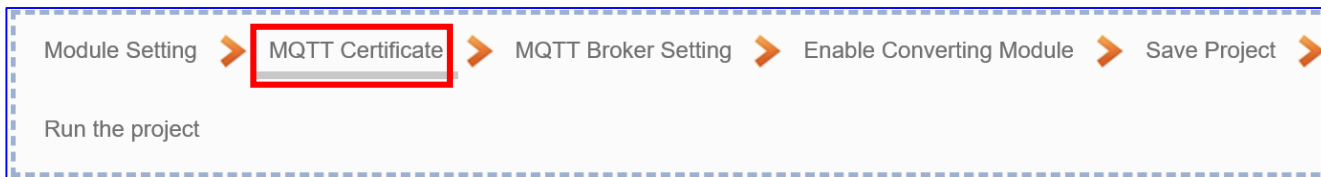
**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

The M-7055D has no AI/AO, so here uses other module's setting screen as an example.

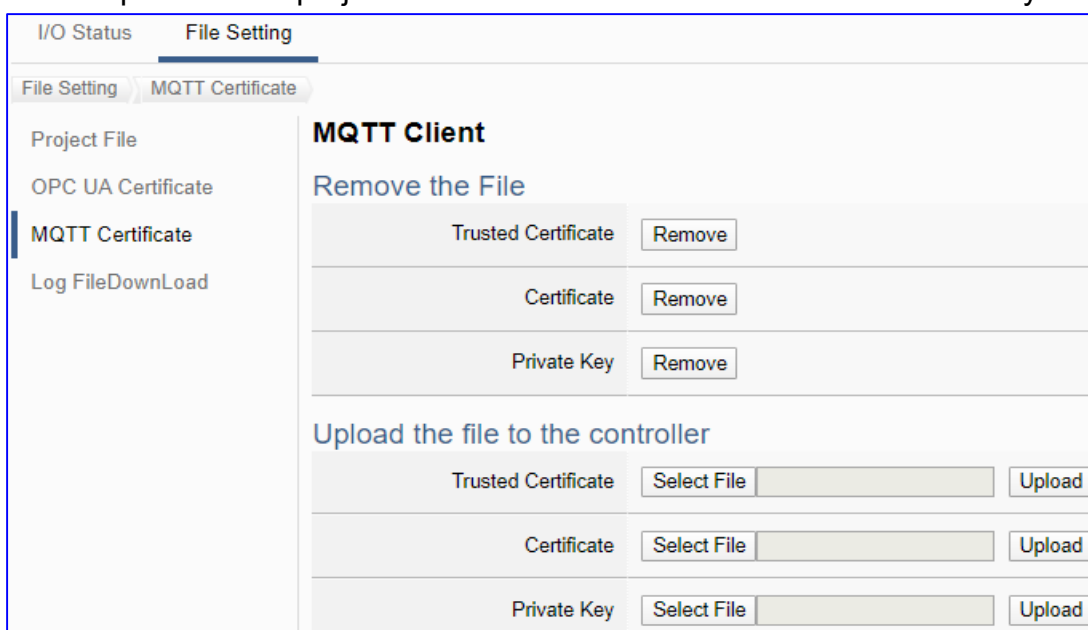
Modbus Mapping Table	Address	Nickname	Scaling	Bitwise
<b>03 Holding Registers(4x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference	Bitwise		
<b>04 Input Registers(3x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference	Bitwise		
0	CO2	<input type="button" value="Hide Detail"/>		
	Bit0	aa	Bit1	<input type="text"/>
	Bit2	bb	Bit3	<input type="text"/>
	Bit4	<input type="text"/>	Bit5	<input type="text"/>
	Bit6	<input type="text"/>	Bit7	<input type="text"/>
	Bit8	<input type="text"/>	Bit9	<input type="text"/>
	Bit10	<input type="text"/>	Bit11	<input type="text"/>
	Bit12	<input type="text"/>	Bit13	<input type="text"/>
	Bit14	<input type="text"/>	Bit15	<input type="text"/>

Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

● **Step 2. MQTT Certificate**

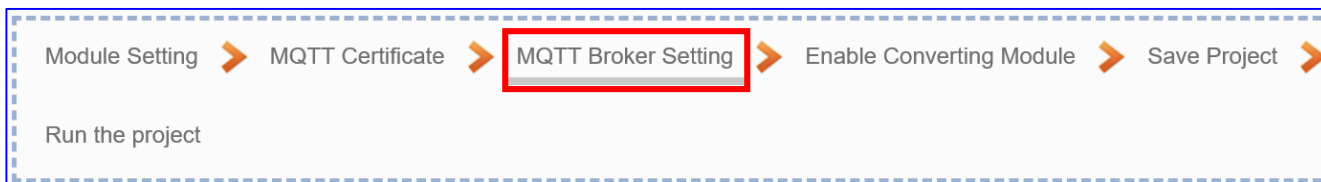


The [MQTT Certificate] is for setting up security communications to upload the **MQTT Trusted Certificate, Certificate and Private Key**. The users upload the file to the UA controller according to the type of obtained certificate. **If you want to perform Broker authentication, you need to upload the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



File Setting > MQTT Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the MQTT Trusted Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Certificate	<p><b>Select File:</b> select the MQTT Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Private Key	<p><b>Select File:</b> select the MQTT Private Key of the device.</p> <p><b>Upload:</b> upload the MQTT Private Key file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“.key”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>

● **Step 3. MQTT Broker Setting**



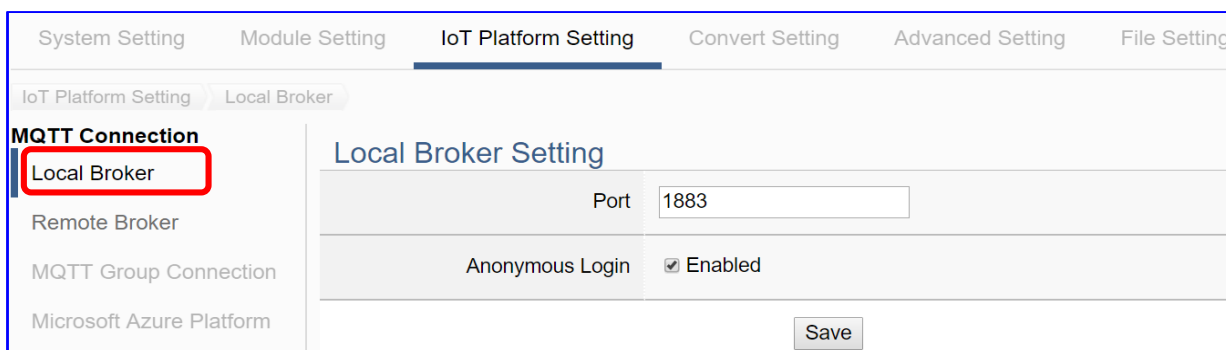
Click the next step, and enter the **Step 3 [MQTT Broker Setting]** of the UI setting.

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 / 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.

**The example uses local Broker.**

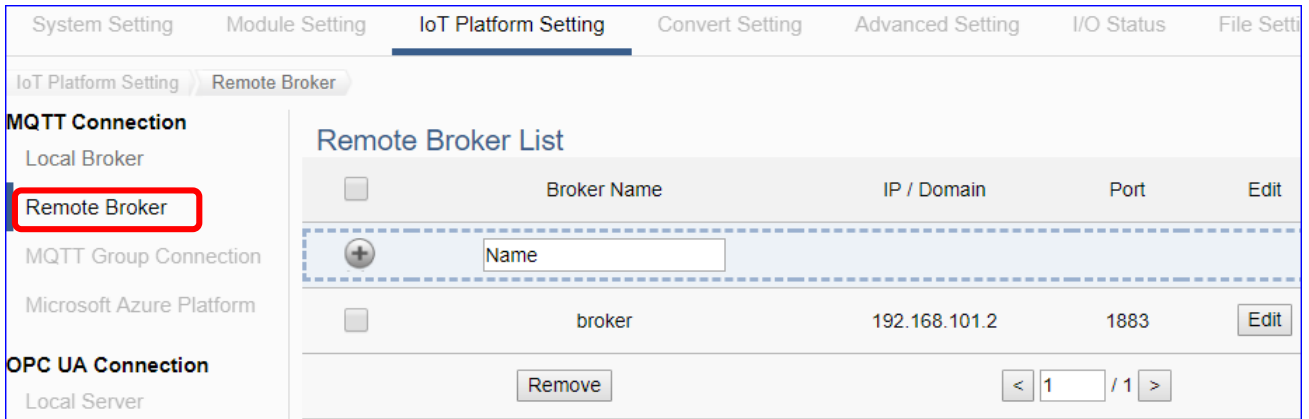
**Local Broker**



<b>MQTT Connection &gt; Local Broker Setting</b>	
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.

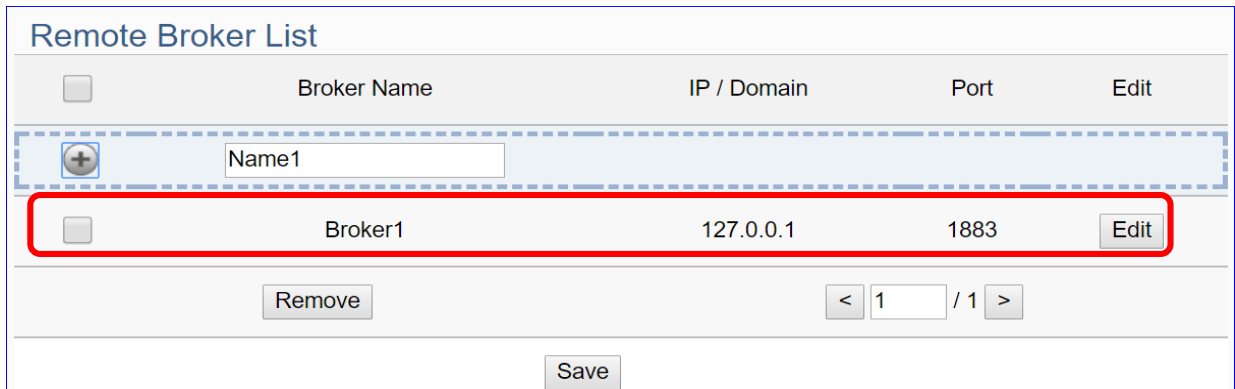
If users apply a remote Broker, the screen will as follow.

**Remote Broker:**



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):



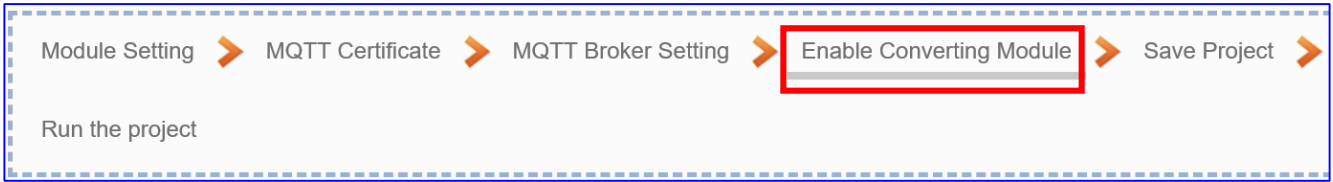
MQTT Connection > Remote Broker List	
Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Edit / Remove	Click [Edit] can set the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

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

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.

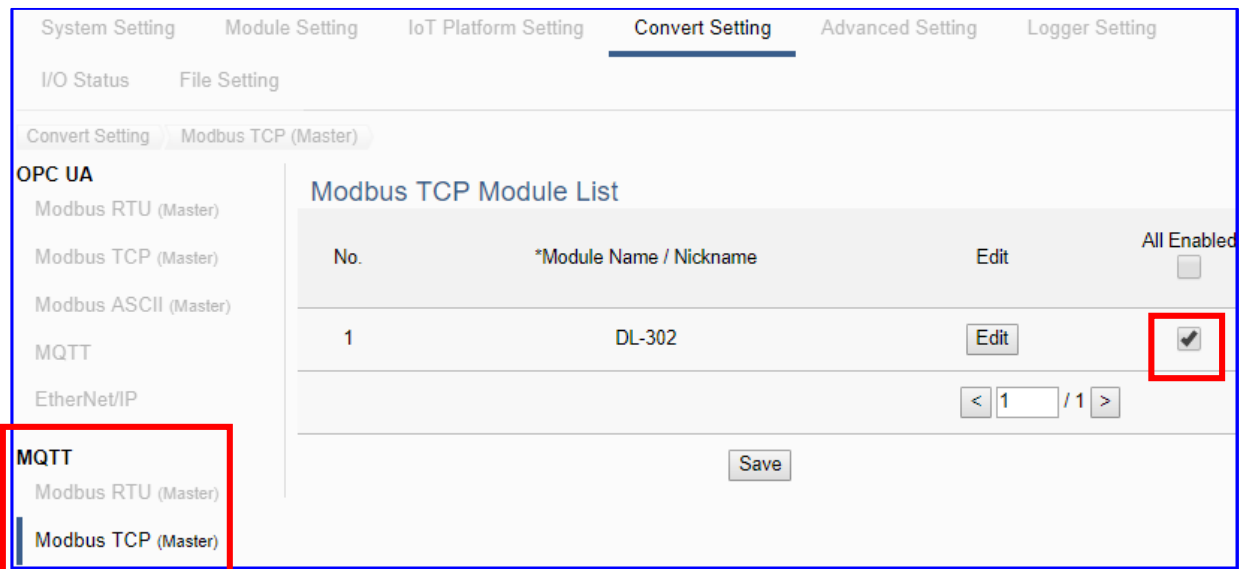


● **Step 4. Enable Converting Module**



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 ...
< 1 / 1 >	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="DL-302"/>
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"/> Name (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**

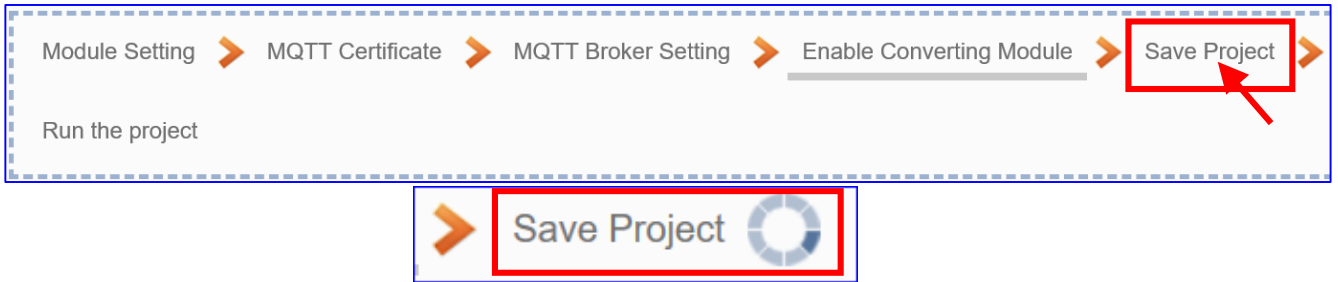
Details

Name	Attribute	Data Type	Subscribe Topic	Subscribe QoS	Publish Topic	Publish QoS	Retain	Enabled
Tag0	Read	Short		2	/MTCP_No.1_DL-302/Input_Registers/Tag0/Publish	2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Tag1	Read	Short		2	/MTCP_No.1_DL-302/Input_Registers/Tag1/Publish	2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Tag2	Read	Short		2	/MTCP_No.1_DL-302/Input_Registers/Tag2/Publish	2	<input type="checkbox"/>	<input checked="" 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.

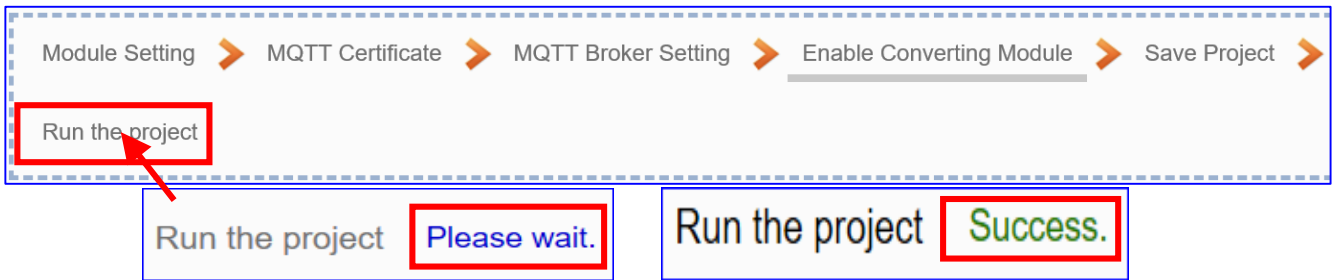
● **Step 5. Save Project**

The setting of this example is finished now. 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.



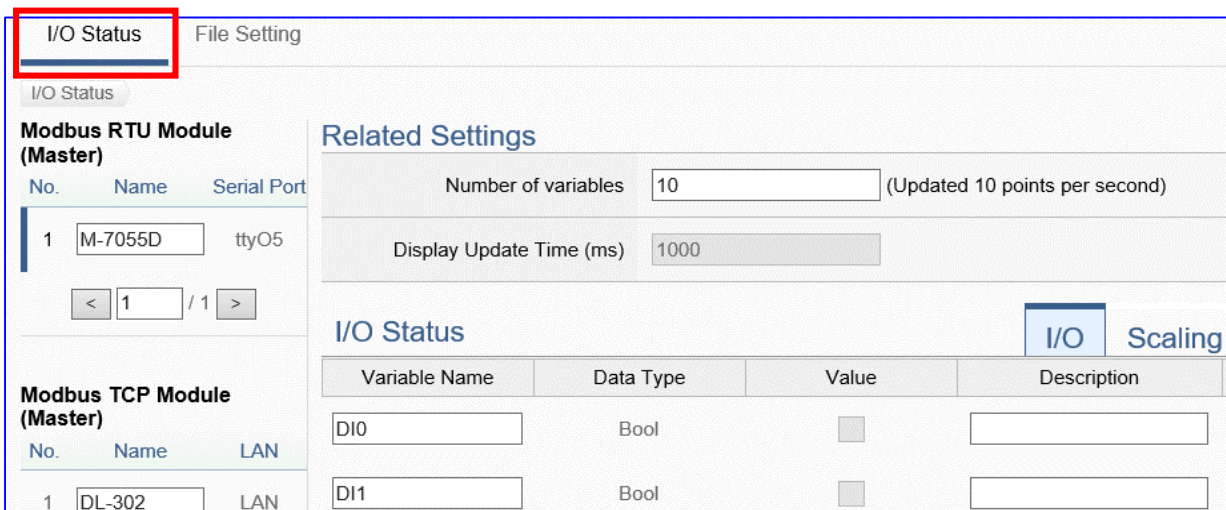
● **Step 6. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. Then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

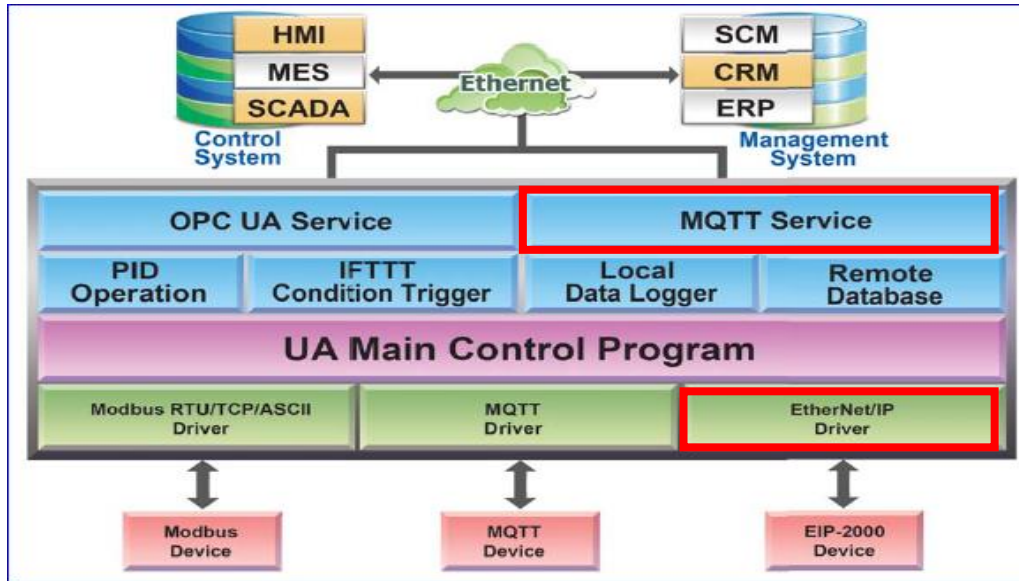
The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.



### 4.1.5 Function Wizard: EIP / MQTT (Use EIP-2060)

EIP / MQTT Conversion include the conversion of **MQTT** and **EtherNet/IP** protocol. 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 **EIP-2000** module that connected to the controller.

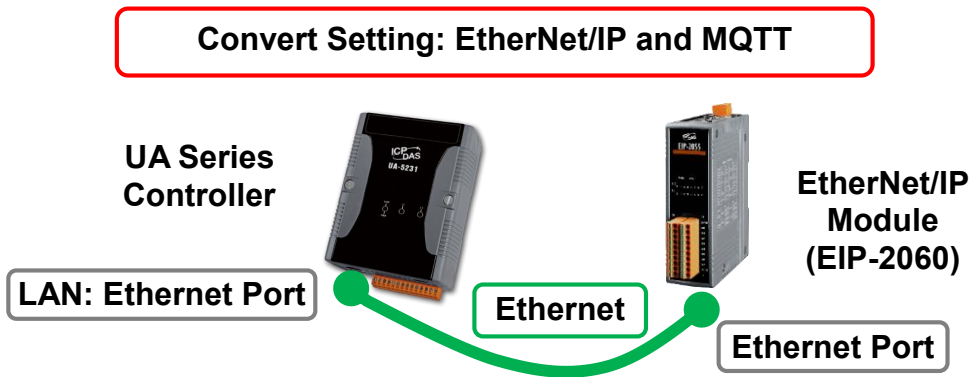
#### EIP / MQTT Function Diagram:



#### Application Solution:

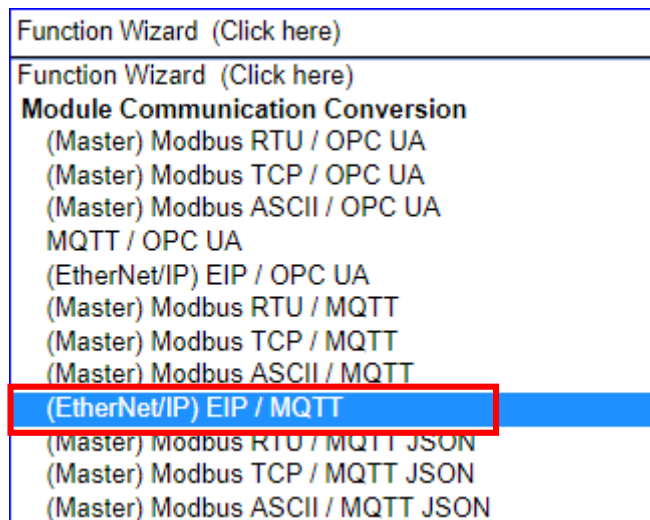


● **Convert Setting: EtherNet/IP and MQTT**



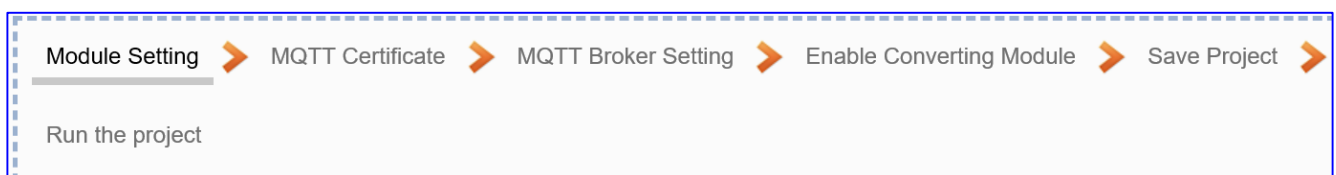
**Note:** The hardware/network connection methods please see the [Chapter 2](#) .

When UA series controller connects the EIP-2000 (via Ethernet, as the picture) and read/write the EIP-2000 I/O via MQTT Broker, user can choose the item [(EtherNet/IP) EIP / MQTT] of the “Module Communication Conversion” in the Function Wizard.

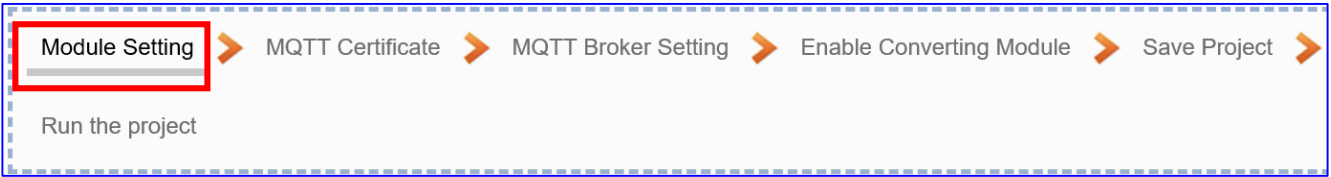


**[Step Box]:**

The Step Box of the [(EtherNet/IP) EIP / MQTT] has the 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.

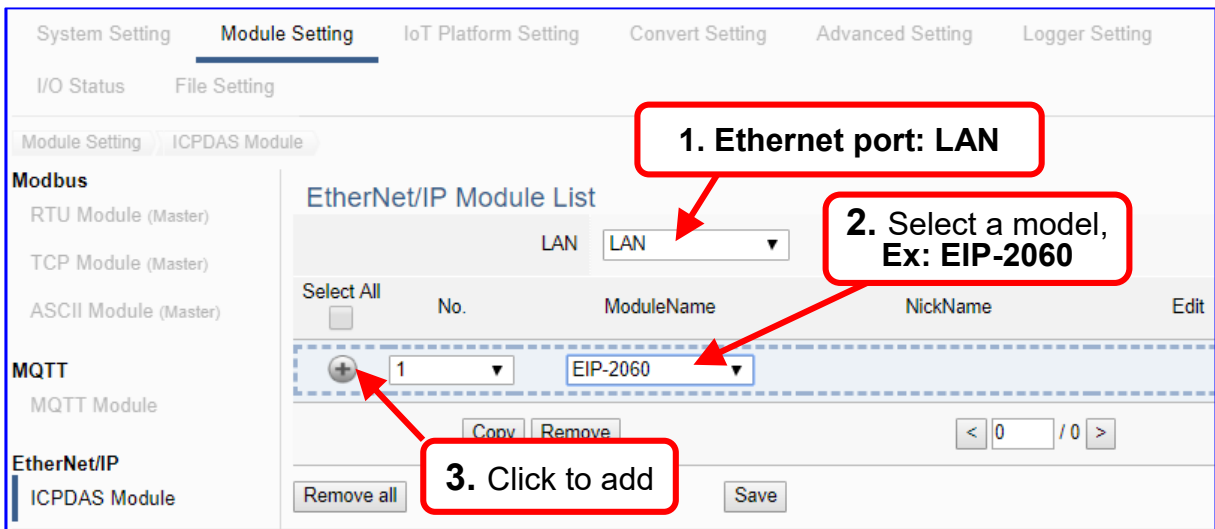


● **Step 1. Module Setting**

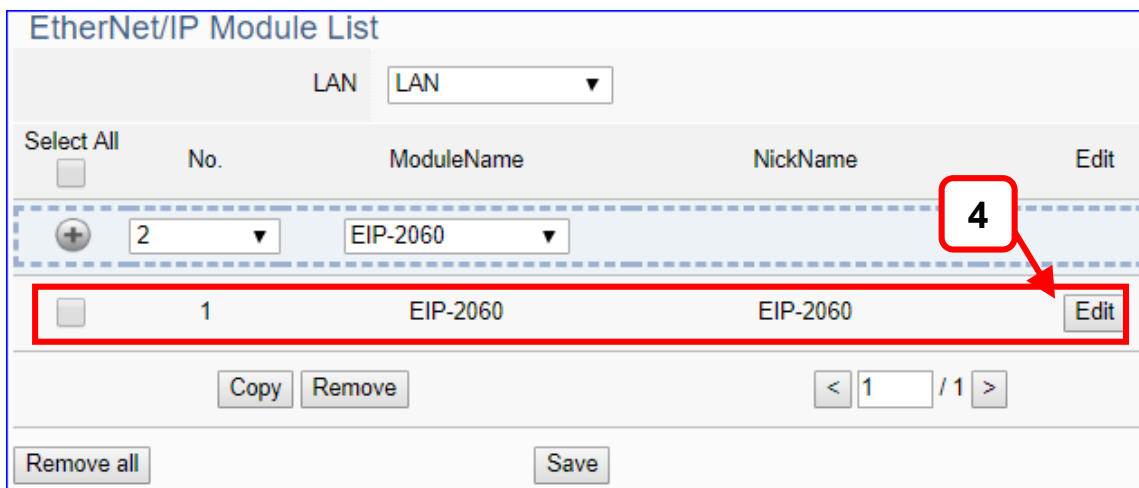


This page is for setting the communication values of the connected modules.

The Ethernet port is LAN for connecting with the EtherNet/IP module EIP-2000 Series by ICP DAS, and select the connected module (This example: EIP-2060). Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and I/O.



Add a module (No.: 1, Name: EIP-2060) as below.



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**] button to enter the “Module Content Setting” page.

[Module Content Setting] page:

Module Content Setting				
No.	<input type="text" value="1"/>			
Module Name	<input type="text" value="EIP-2060"/>			
NickName	<input type="text" value="EIP-2060"/>			
IP	<input type="text" value="192"/>	<input type="text" value="168"/>	<input type="text" value="13"/>	<input type="text" value="5"/>
ChannelNumber	<input type="text" value="12-ch(6DI+6DO)"/>			

**User enters the module connected IP address.**

Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	The selected model number. (Not editable here)
Nick Name	User can give a nick name, default: selected model number
IP	Enter the IP address of the connected EIP module. This example: IP address of the EIP-2060 is 192.168.13.5
Channel Number	Select the number of the I/O channels.

The system will auto-display the selected I/O table by the order of Digital Input / Digital Output / Analogy Input / Analogy Output. This example: EIP-2060 has 6 DI and 6 DO.

Digital Input				
Channel	Name	Attributes	Data Type	Description
0	<input type="text" value="DI0"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
1	<input type="text" value="DI1"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
2	<input type="text" value="DI2"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
3	<input type="text" value="DI3"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
4	<input type="text" value="DI4"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>
5	<input type="text" value="DI5"/>	<input type="text" value="Read"/>	Bool	<input type="text"/>



Digital Output				
Channel	Name	Attributes	Data Type	Description
0	<input type="text" value="DO0"/>	Read / Write ▼	Bool	<input type="text"/>
1	<input type="text" value="DO1"/>	Read / Write ▼	Bool	<input type="text"/>
2	<input type="text" value="DO2"/>	Read / Write ▼	Bool	<input type="text"/>
3	<input type="text" value="DO3"/>	Read / Write ▼	Bool	<input type="text"/>
4	<input type="text" value="DO4"/>	Read / Write ▼	Bool	<input type="text"/>
5	<input type="text" value="DO5"/>	Read / Write ▼	Bool	<input type="text"/>

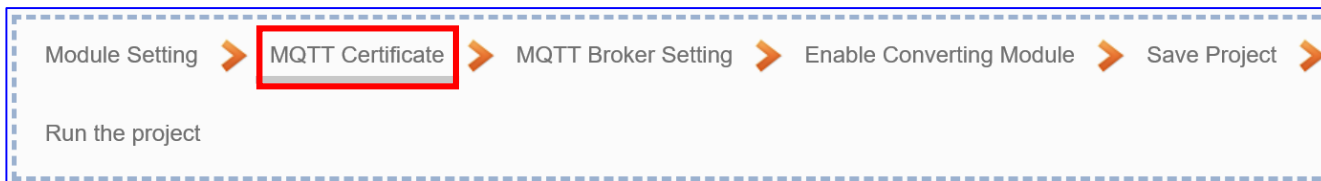
Analogy Input				
Channel	Name	Attributes	Data Type	Description

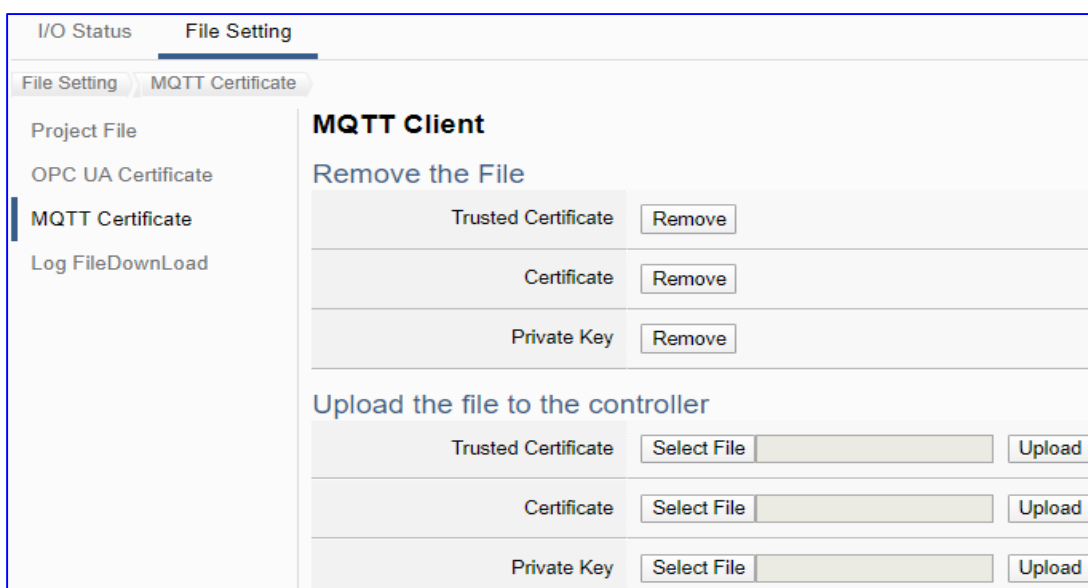
Analogy Output				
Channel	Name	Attributes	Data Type	Description

Digital Input / Digital Output / Analogy Input / Analogy Output	
Channel	Channel number will auto-display according to the model. (Not editable) Default: Number from 0.
Name	User can define the name. Default: DI#, DO#, AI#, AO# Available: number, English character, underline “_”, dash line “-”, cannot be a space, slash “/”, Chinese character, and other symbols.
Attributes	Display data attribute of the variable. (Not editable) Include Read, Read/Write...
Data Type	Display data type of the variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String
Description	For users set up the description for the channel.
OK / Cannel	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.

● **Step 2. MQTT Certificate**

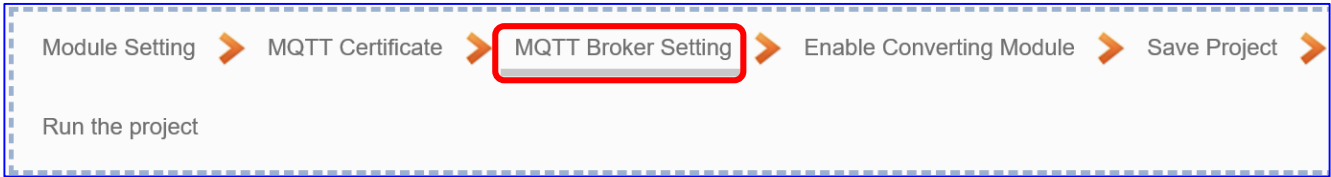


The [MQTT Certificate] is for setting up security communications to upload the **MQTT Trusted Certificate, Certificate and Private Key**. The users upload the file to the UA controller according to the type of obtained certificate. **If you want to perform Broker authentication, you need to upload the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



File Setting > MQTT Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the MQTT Trusted Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul> <p style="text-align: center;">Trusted Certificate    Select File    Certificate_192.168.255.10    Certificate type is wrong.    Upload</p>
Certificate	<p><b>Select File:</b> select the MQTT Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Private Key	<p><b>Select File:</b> select the MQTT Private Key of the device.</p> <p><b>Upload:</b> upload the MQTT Private Key file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“.key”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>

● **Step 3. MQTT Broker Setting**

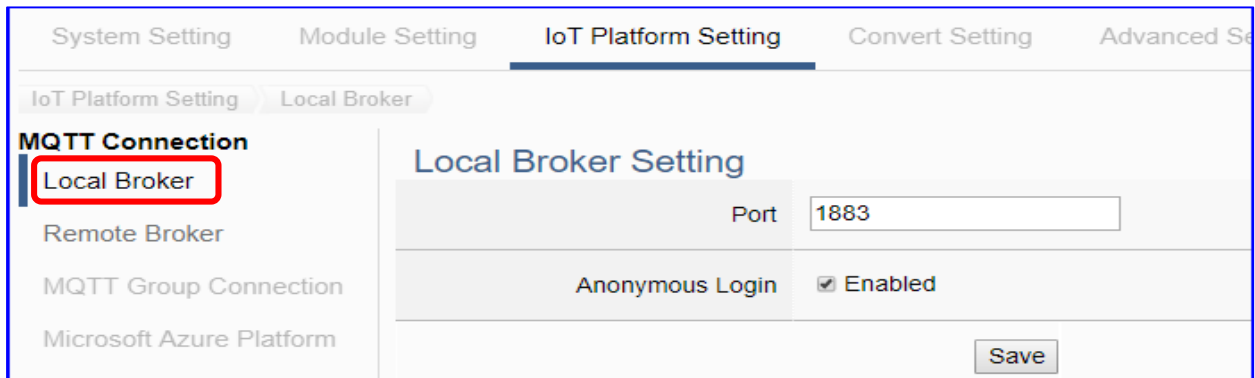


Click the next step, and enter the **Step 3 [MQTT Broker Setting]** of the UI setting. This page is for setting the IoT platform and the MQTT Broker connection, e.g. the local or remote broker, port, login information, etc.

We select the “EIP / 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.

**The example uses local Broker.**

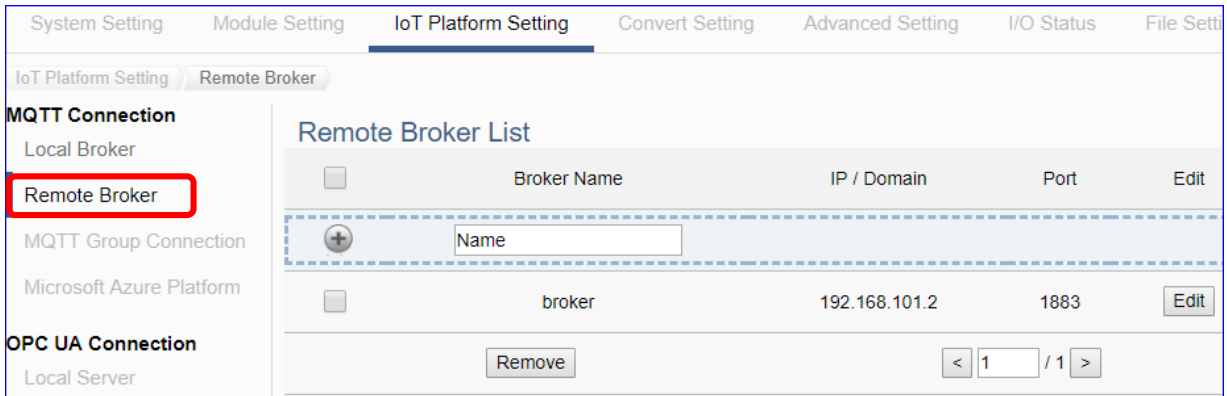
**Local Broker**



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.

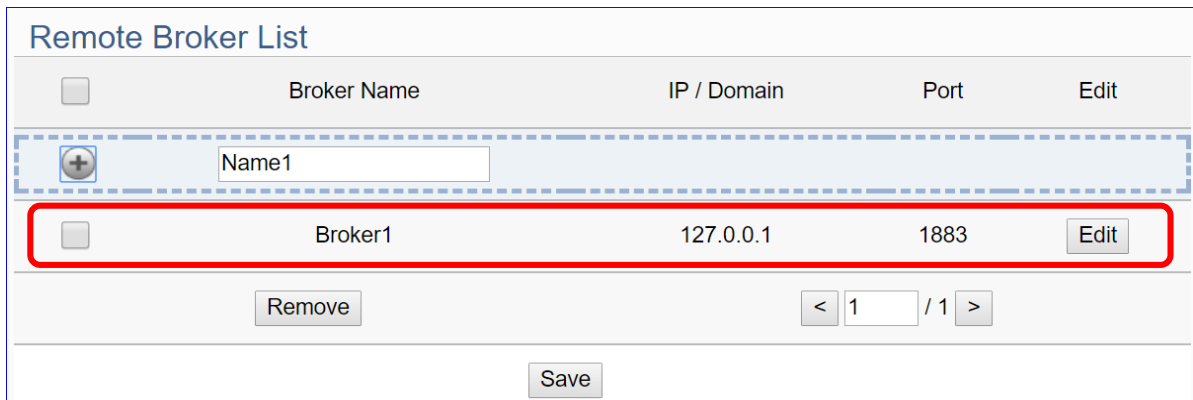
If user wants to use the remote Broker, please click the “Remote Broker” to set up.

**Remote Broker:**



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):

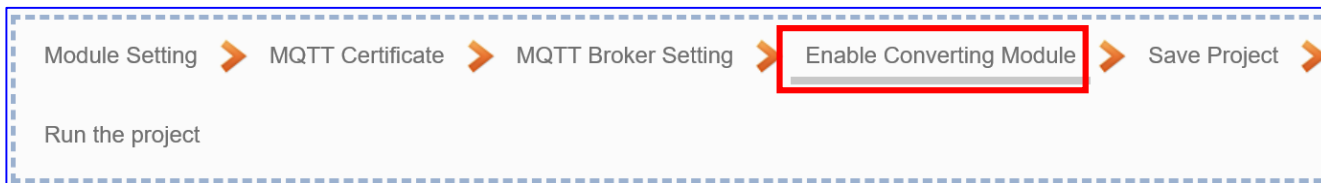


MQTT Connection > Remote Broker List	
Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
<b>Edit</b> / Remove	Click [Edit] can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

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

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.

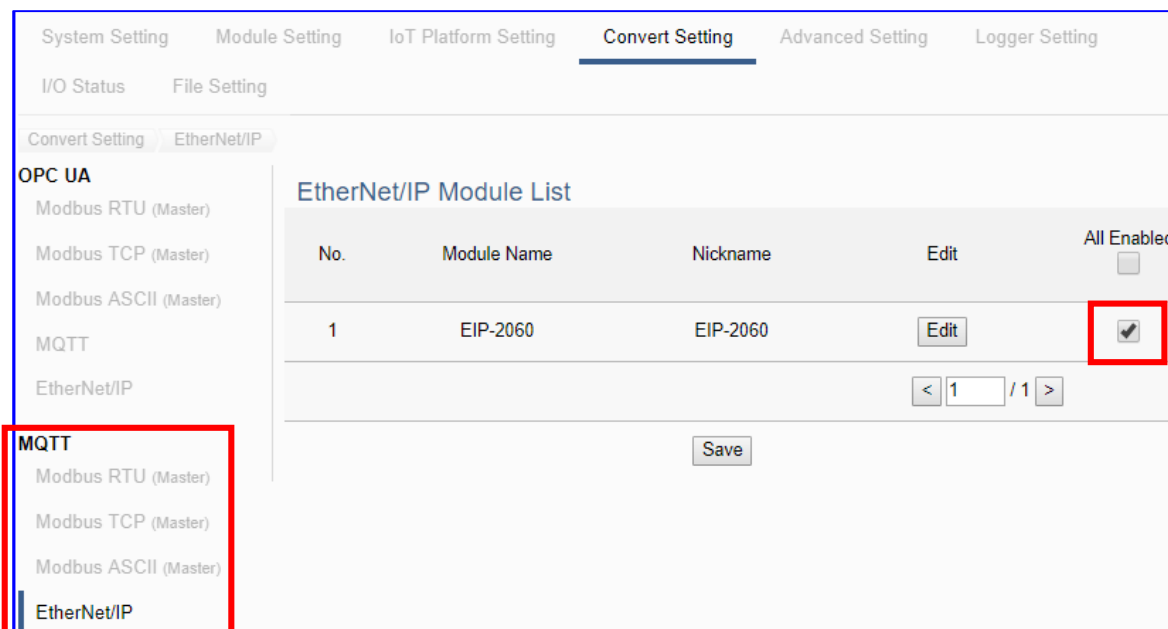
● **Step 4. Enable Converting Module**



Click the next step, and enter the **Step 4 [Enable Converting Module]** UI setting  
 This step is for enabling the module for the EtherNet/IP / MQTT conversion.

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

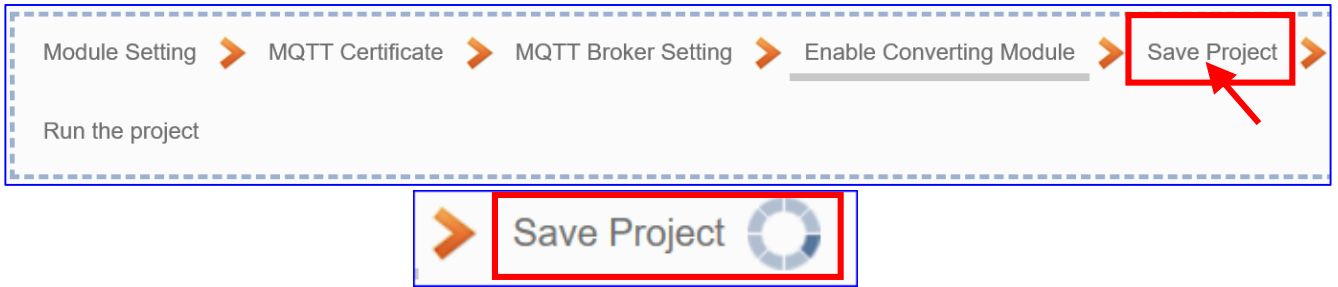
Please check the Enabled box of the module.



Convert Setting > MQTT > EtherNet/IP –EtherNet/IP Module List	
No.	The module number in the module list (Not editable here)
Module Name	The module user selected (Not editable here)
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 ...
<input type="button" value="1"/> / <input type="button" value="1"/>	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.

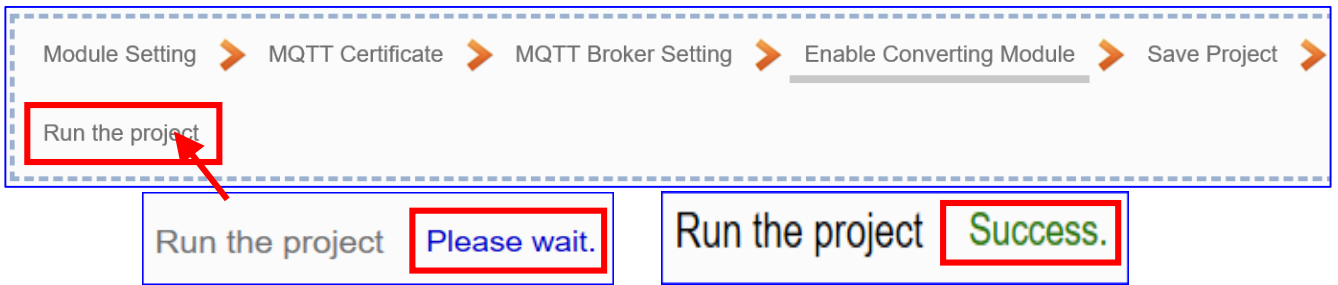
● **Step 5. Save Project**

The setting of this example is finished now. 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.



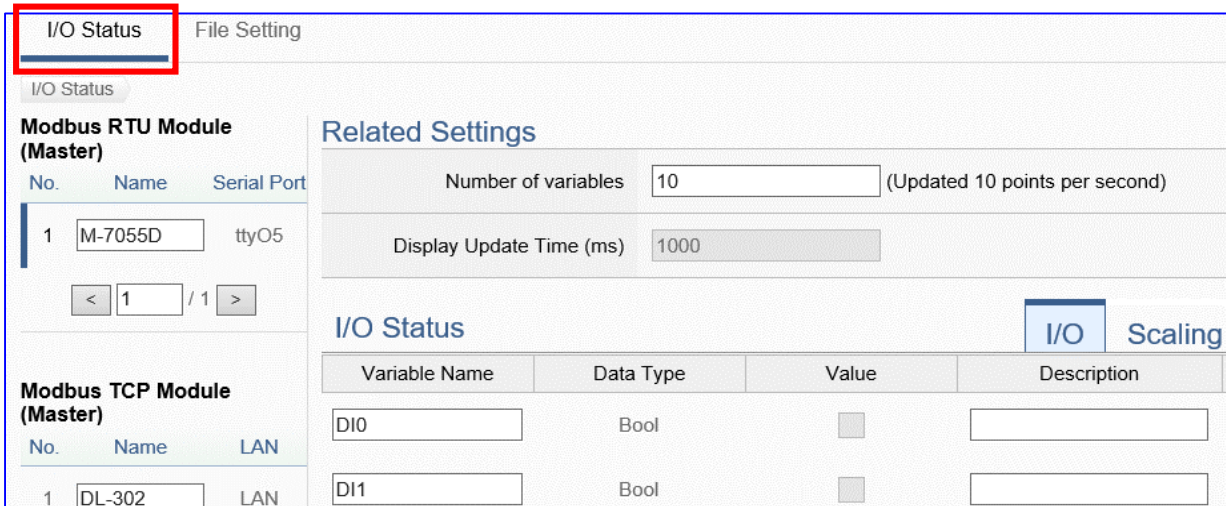
● **Step 6. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. Then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

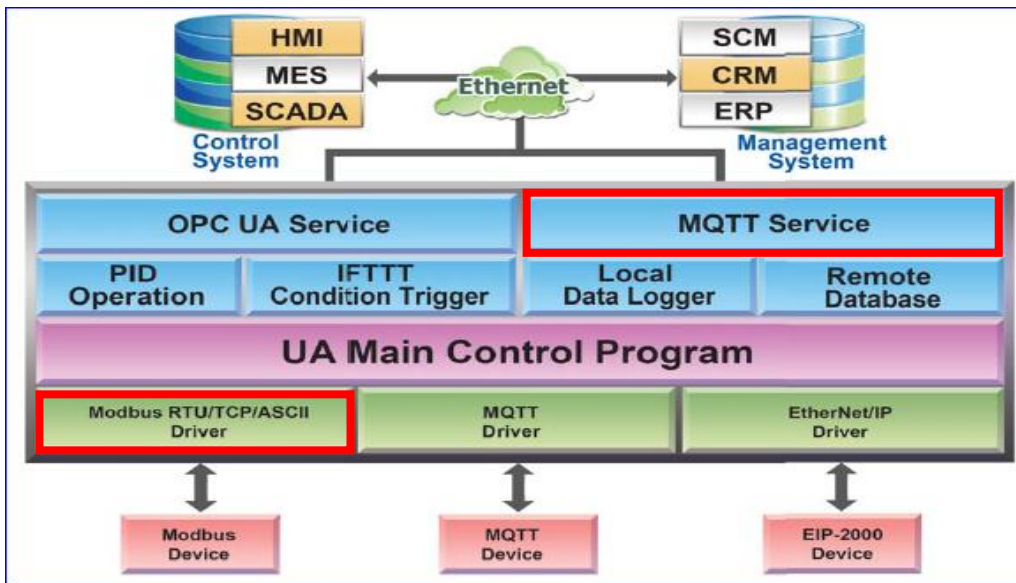
The new project now completes the setting, uploading and running in the UA controller and can process the conversion communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.



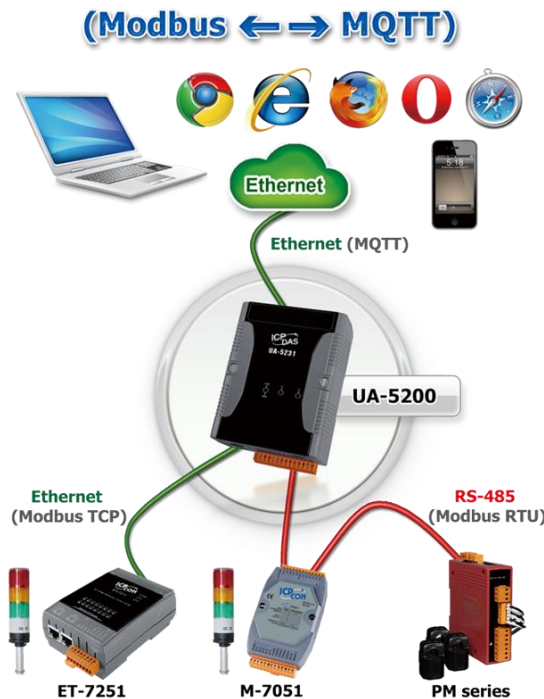
### 4.1.6 Function Wizard: Modbus / MQTT JSON (RTU Example, Use M-7055D)

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:



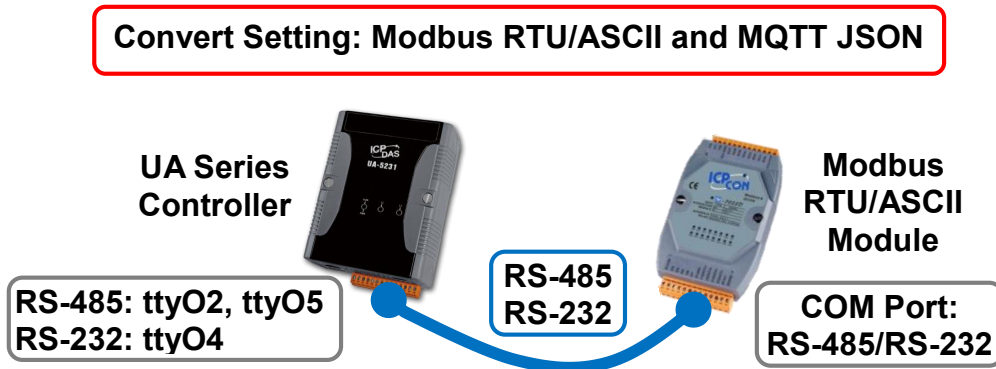
#### Application Solution:





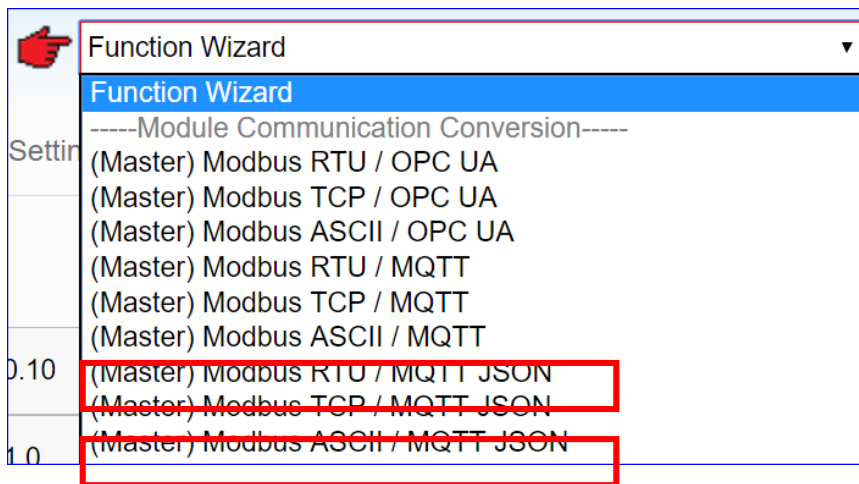
The settings of Modbus RTU/ASCII are the same. Here will introduce them together as a setting sample for Modbus / MQTT JSON conversion.

● **Convert Setting: Modbus RTU / ASCII and MQTT JSON**



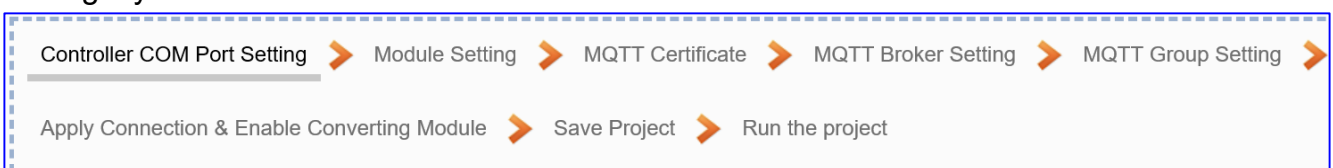
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

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.

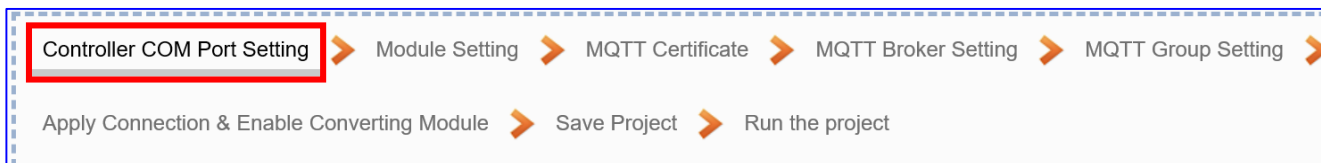


**[Step Box]:**

The Step Box of the **[Modbus RTU / MQTT JSON]** and **[Modbus ASCII / MQTT JSON]** has the same 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 default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).

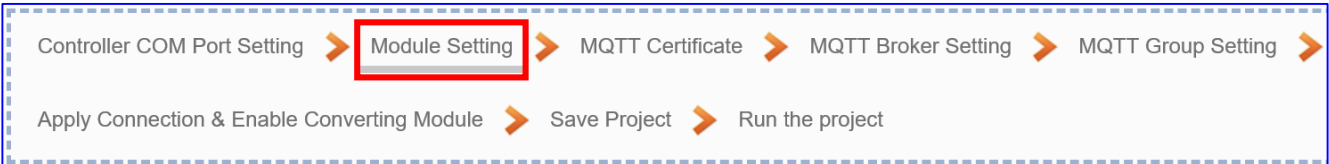
The screenshot shows a web interface with a sidebar on the left containing menu items: System Setting, COM Port Interface Setting, Controller Service Setting, Time Setting, Network Setting, Account Setting, Boot, and COM Port Interface Setting. The main content area is titled 'COM Port Interface Setting Page' and contains the following settings:

- Serial Port: ttyO2
- Baud Rate: 115200
- Data Bits: 8 bits
- Parity: None
- Stop Bits: 1 bit
- Polling Rate(ms): 500

A 'Save' button is located at the bottom right of the settings area.

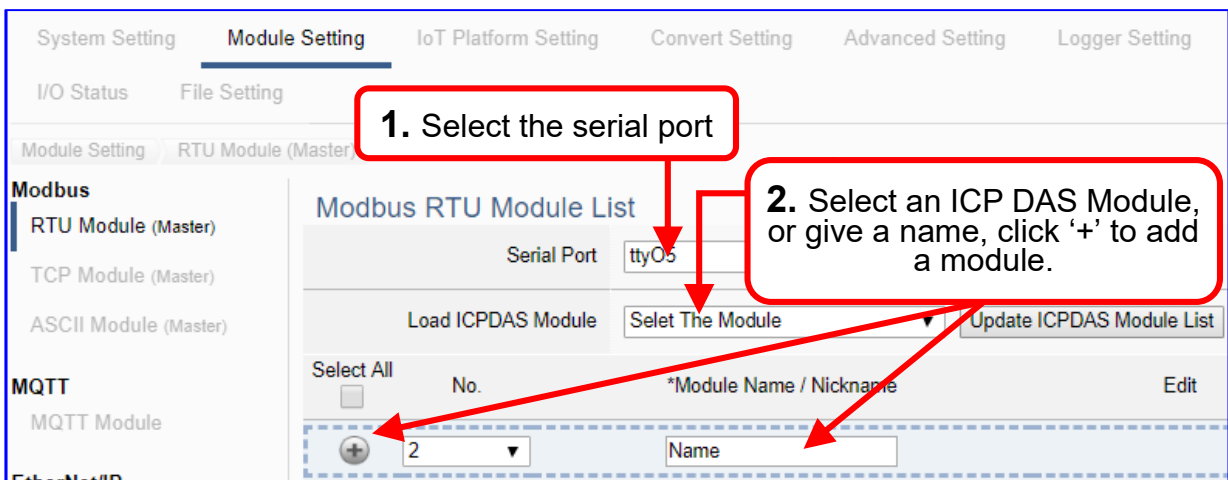
COM Port Interface Setting Page	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485 ; ttyO4: RS-232 ; ttyO5: RS-485
Baud Rate	Choose a baud rate to communicate with the module: 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. The UA controller and 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**

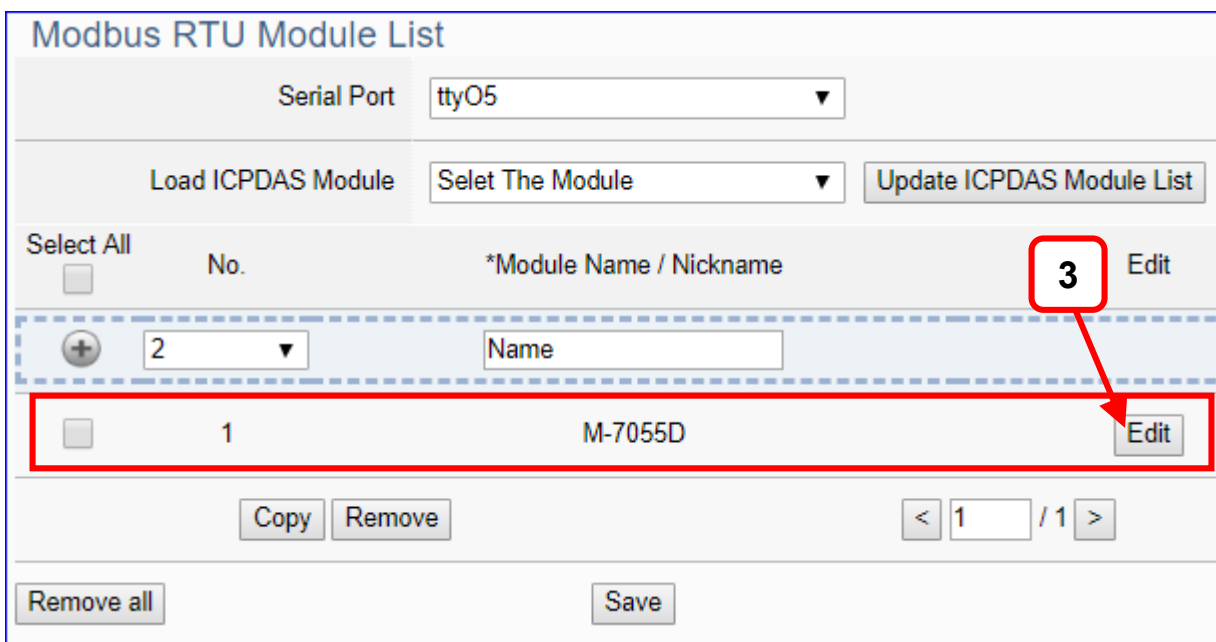


Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First, choose the connected port with the module. If using ICP DAS module, select the model to auto load and setup the module. If not, give a module name (Default: Name) and click [ + ] button to add a new module.



Add a module (No.: 1, Name: M-7055D) 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.	3
Module Name	M-7055D
Slave ID	2
Timeout(ms)	500
Write Retry	<input type="checkbox"/> 1

**Modbus Mapping Table Setting**

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

If select ICP DAS module, system will auto set up the Modbus Mapping Table, or user needs to check the module Modbus address or I/O number from the module user manual.

> **Modbus Mapping Table Setting:**  
 Set module in the order of Data Model, Start Address and Data Number, then click "Add".  
**Ex:** M-7055D has 8 Data Models of "01 Coil Status (0x)" (Mapping: DO), so select Model "01", Start Add. "0", Number "8", and click "Add".

Coil Status(0x)

Address	0
Number	8
Type	Bool
<input type="button" value="Edit"/>	

**DO mapping 01** →

**UA start address: 0** →

**If DO x 8, enter 8** →

**Click [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. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Write Retry	Check to retry writing again when there is no response after the set time is up, and it can be set up to retry 3 times
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 blue; padding: 5px; margin-top: 5px;"> <p style="background-color: #0070c0; color: white; padding: 2px;">01 Coil Status(0x)</p> <p style="padding: 2px;">02 Input Status(1x)</p> <p style="padding: 2px;">03 Holding Registers(4x)</p> <p style="padding: 2px;">04 Input Registers(3x)</p> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

### Address:

Display and edit the Modbus Mapping Table.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise		
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"/>					

If user selects ICP DAS module, the system will auto set up the Modbus Mapping Table. If not, user needs to check the module Modbus address or I/O number from the module user manual.

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table	Address	Nickname	Scaling	Bitwise
<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="DO0"/>	Bool	<input type="text" value="Light 01"/>	
1	<input type="text" value="DO1"/>	Bool	<input type="text"/>	
2	<input type="text" value="DO2"/>	Bool	<input type="text"/>	
3	<input type="text" value="DO3"/>	Bool	<input type="text"/>	
4	<input type="text" value="DO4"/>	Bool	<input type="text"/>	
5	<input type="text" value="DO5"/>	Bool	<input type="text"/>	
6	<input type="text" value="DO6"/>	Bool	<input type="text"/>	
7	<input type="text" value="DO7"/>	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="DI0"/>	Bool	<input type="text"/>	

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.

**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated. The M-7055D has no AI/AO, so here uses the screen of DL-302 for an example.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

**Bitwise:**

**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

The M-7055D has no AI/AO, so here uses other module's setting screen as an example.

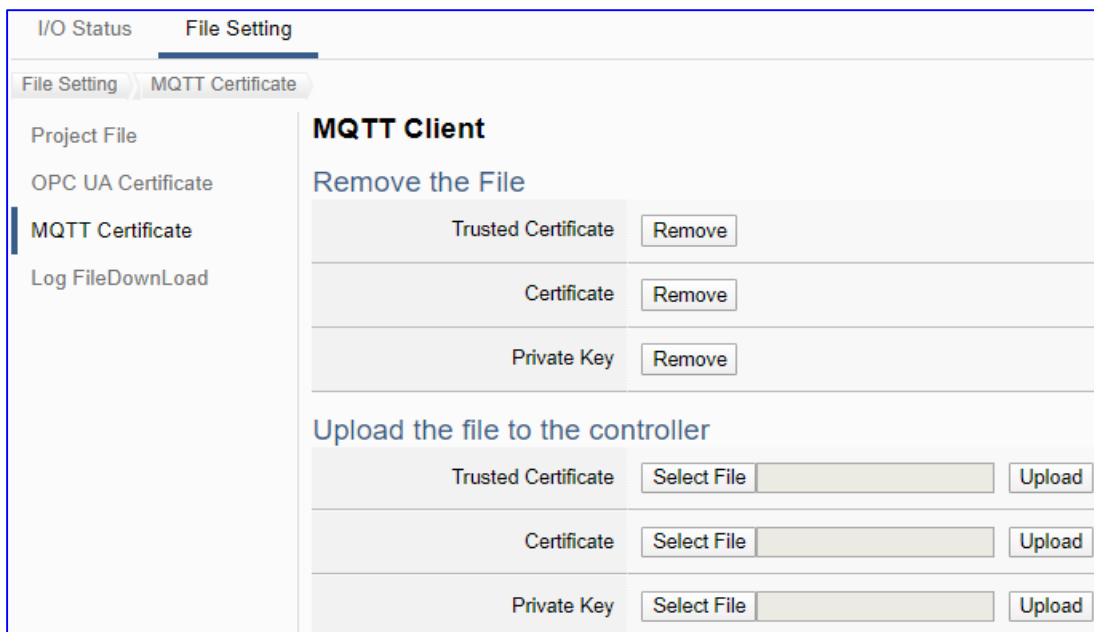
Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.



● **Step 3. MQTT Certificate**



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



File Setting > MQTT Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the MQTT Trusted Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>• File format must be <b>PEM</b>. Extension name must be "<b>pem / cer / crt</b>".</li> <li>• If select a wrong file, the system will show an error message.</li> </ul> <p>Trusted Certificate    <input type="button" value="Select File"/> Certificate_192.168.255.10: <span style="color: red;">Certificate type is wrong.</span>    <input type="button" value="Upload"/></p>
Certificate	<p><b>Select File:</b> select the MQTT Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>• File format must be <b>PEM</b>. Extension name must be "<b>pem / cer / crt</b>".</li> <li>• If select a wrong file, the system will show an error message.</li> </ul>
Private Key	<p><b>Select File:</b> select the MQTT Private Key of the device.</p> <p><b>Upload:</b> upload the MQTT Private Key file to the UA controller.</p> <ul style="list-style-type: none"> <li>• File format must be <b>PEM</b>. Extension name must be <b>".key"</b>.</li> <li>• If select a wrong file, the system will show an error message.</li> </ul>

● **Step 4. MQTT Broker Setting**



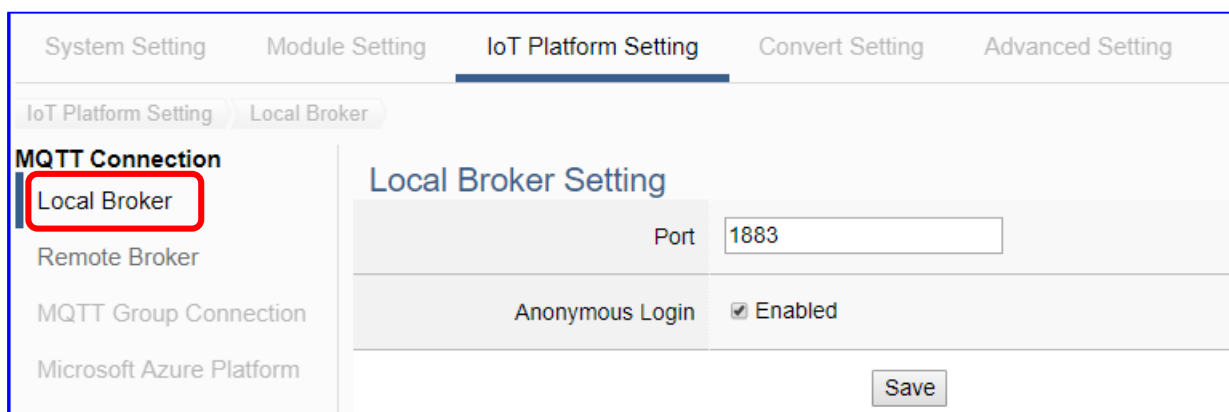
Click the next step, and enter the **Step 4 [MQTT Broker Setting]** of the UI setting.

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.

**The example uses local Broker.**

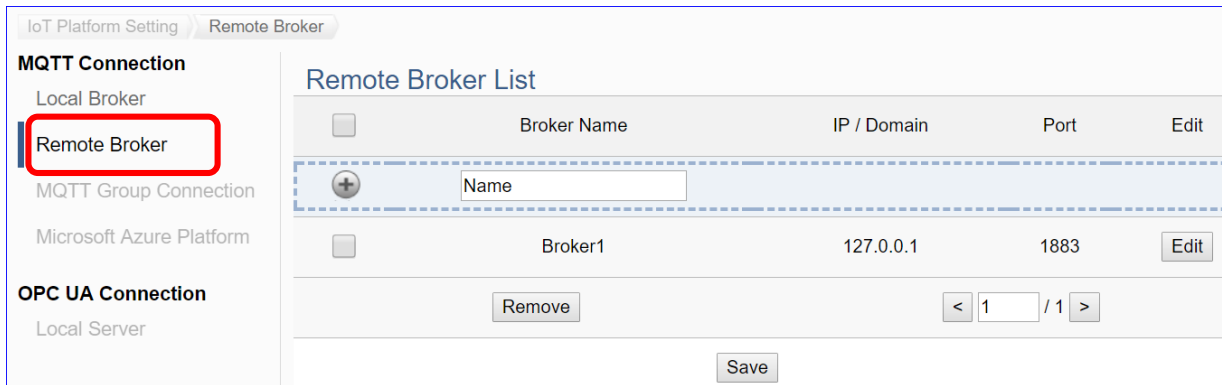
**Local Broker**



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.

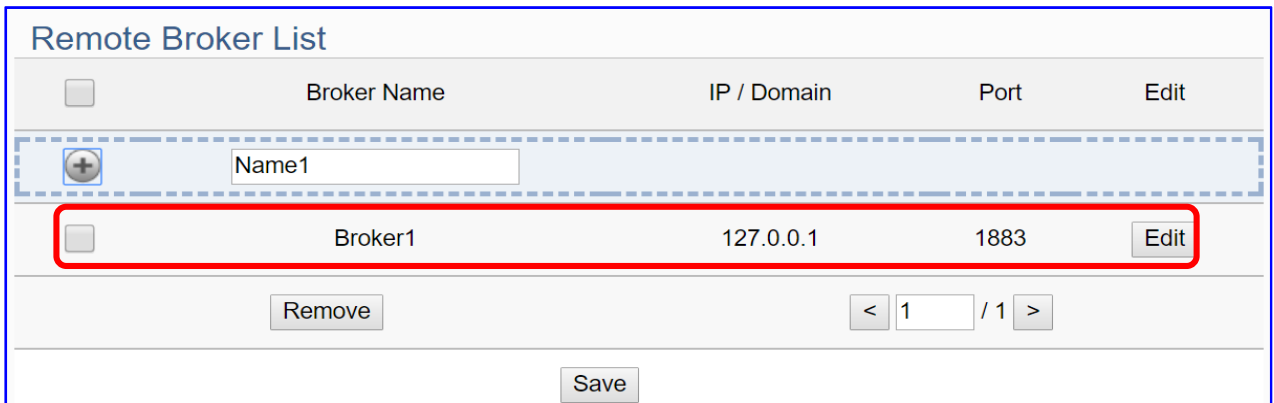
If users apply a remote Broker, the screen will as follow.

**Remote Broker:**



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) :

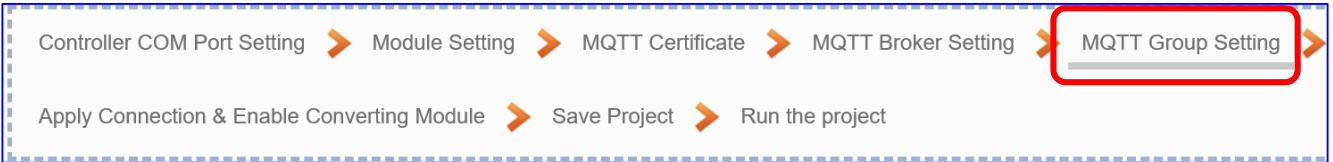


MQTT Connection > Remote Broker List	
Broker Name	The name of the remote MQTT Broker. User can define the name, e.g. Broker1. Default: Name.
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Edit /	Click [Edit] can set the Broker.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

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

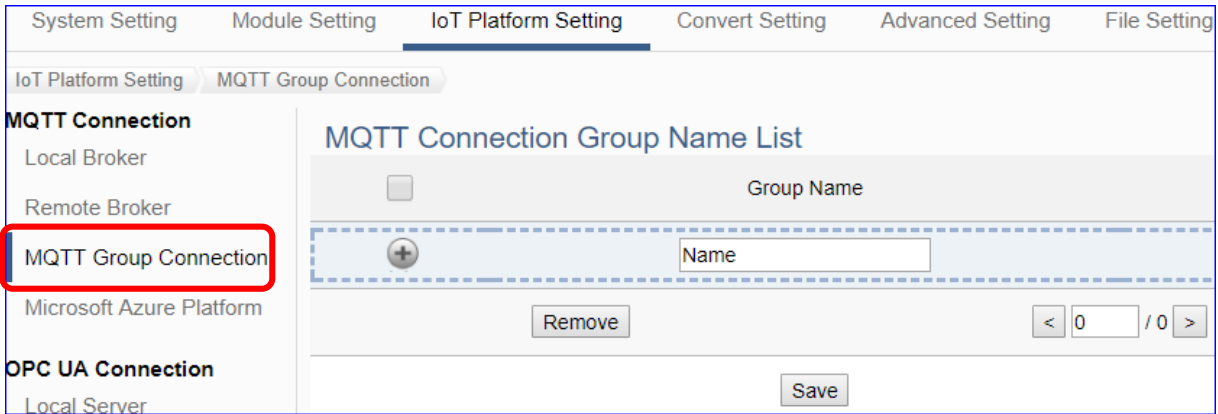
MQTT Connection > Remote Broker > Broker Content Settings	
Broker Name	The name of the remote MQTT Broker. (Editable)
IP / Domain	The IP address of the remote Broker. Default: 127.0.0.1
Port	The COM port of the remote Broker. Default: 1883
Keep Alive Time	The keep alive time. Default: 60 (second)
SSL/TLS	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 5. MQTT Group Setting**



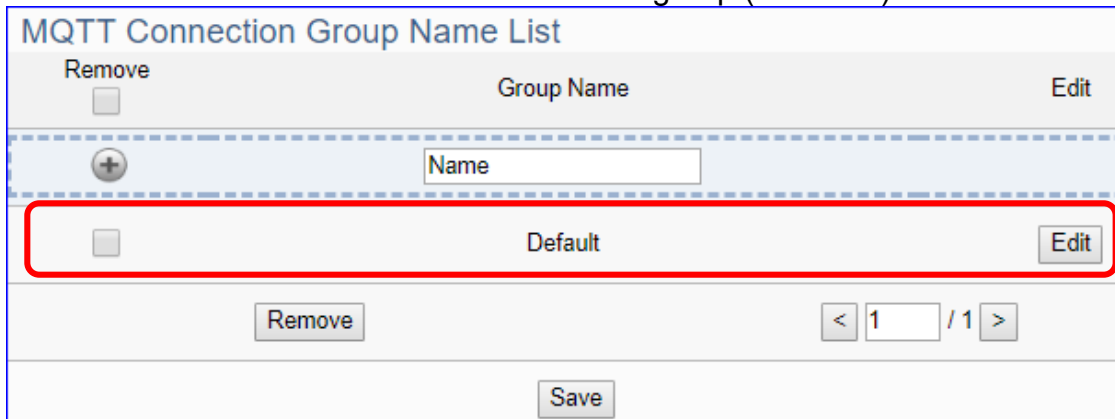
Click the next step, and enter the **Step 5 [MQTT Group Setting]** of the UI setting. This page is for setting the MQTT Group connection, setting with the MQTT JSON function in the Convert Transmission, It can make the I/O module messages in groups and then mapping to the user-defined publish and subscribe 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 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 add button 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.

Enter a name and click add button to create a new group (as below).



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

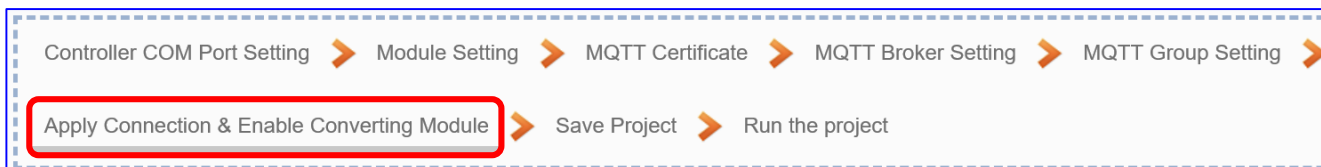
MQTT Client Setting	
No.	<input type="text" value="1"/>
Group Name	<input type="text" value="Default"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>
Will Topic	<input type="text"/>
Will	<input type="text"/>
MQTT Connection	<input checked="" type="checkbox"/>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

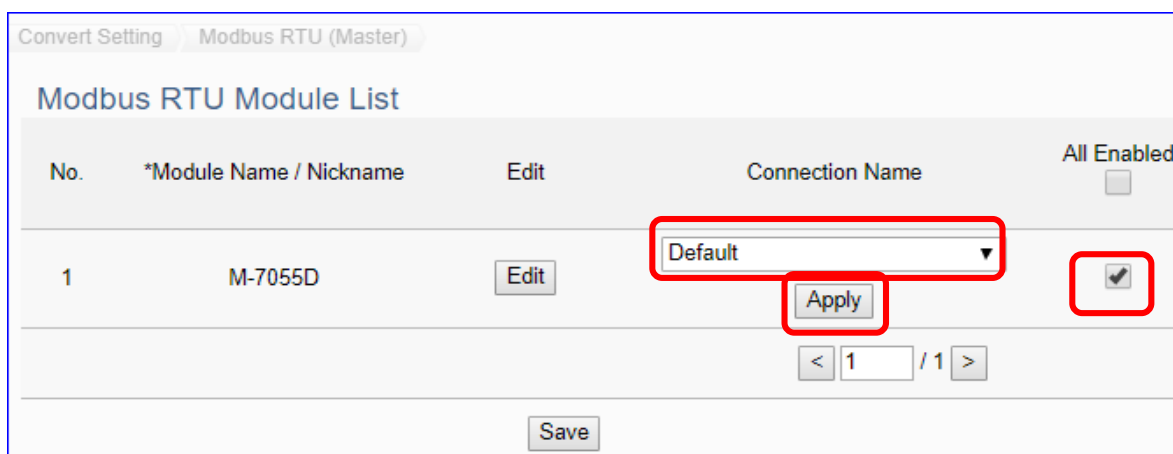
● **Step 6. Apply Connection & Enable Converting Module**



Click the next step, and enter the **Step 6 [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 Convert setting. The “Step Box” will prevent the user from selecting the wrong platform.

**Check the enabled box of the module, select the connection name and click “Apply”.**



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.
< 1 / 1 >	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.



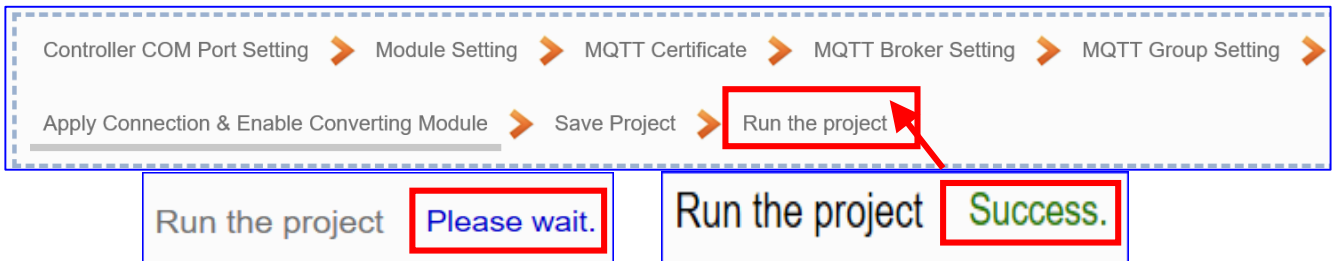
● **Step 7. Save Project**

The setting of this example is finished now. 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.



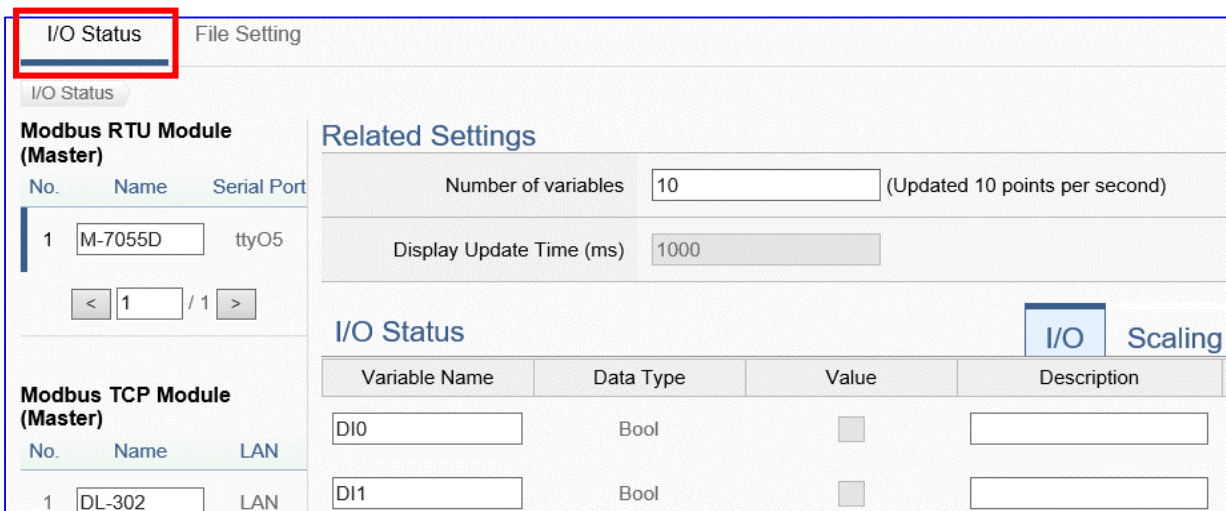
● **Step 8. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.



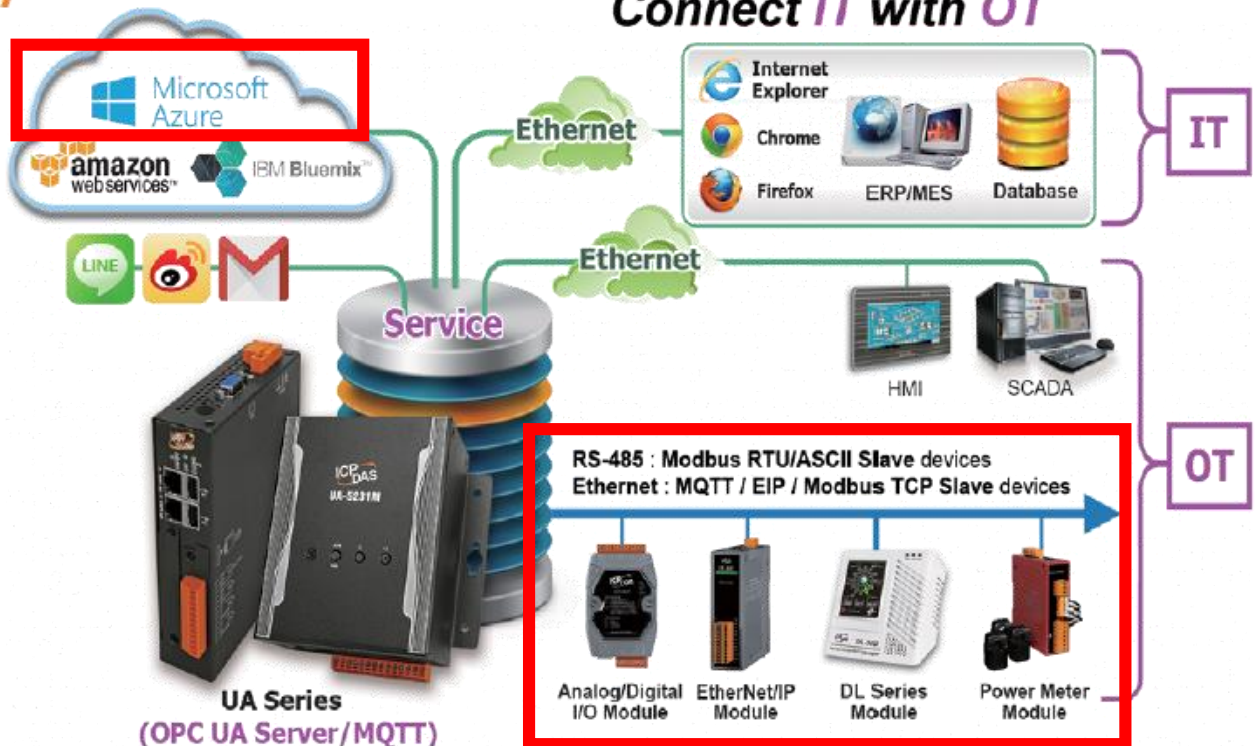
## 4.2 Module Connecting to Azure

"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> (4.2.1, Ex. DL-302)	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 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.

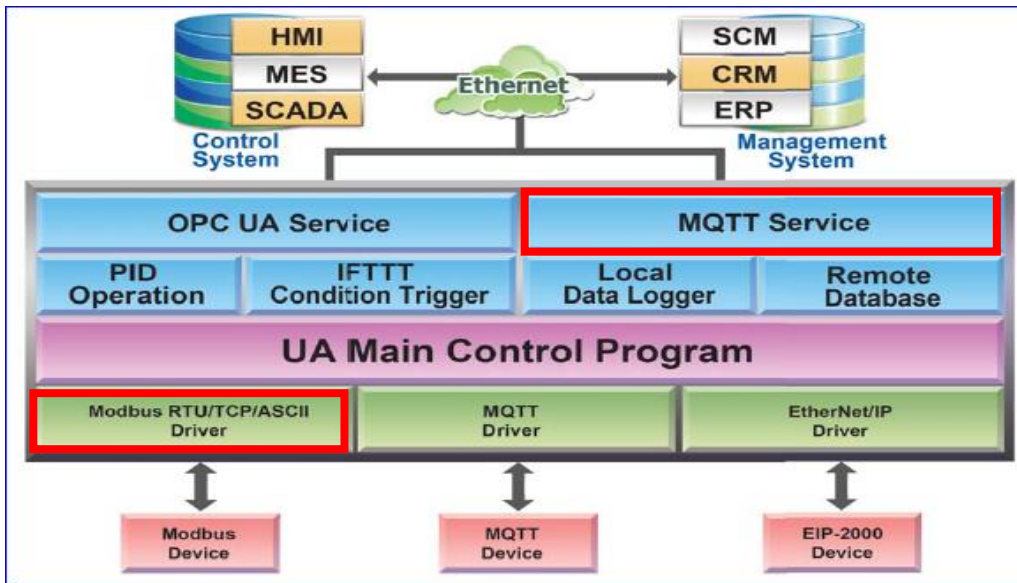
### System Architecture:



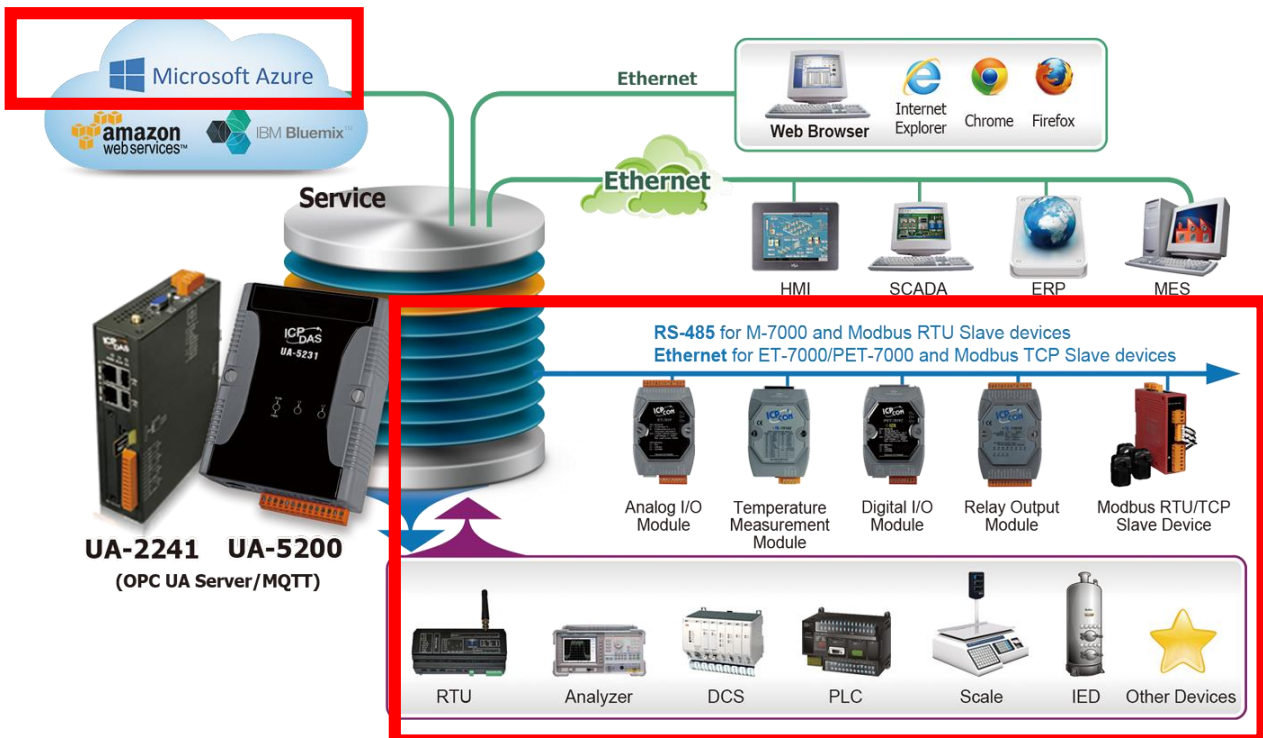
### 4.2.1 Function Wizard: Modbus TCP / Azure (Use DL-302)

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. **Before setting the Azure connection, user needs to apply user SAS Token and Root CA from Microsoft Azure.** This section will introduce the setting steps and the function parameters. There are 3 items about Azure function in the “Function Wizard”. All connections use the **MQTT Service**. Here will introduce the Modbus TCP / Azure.

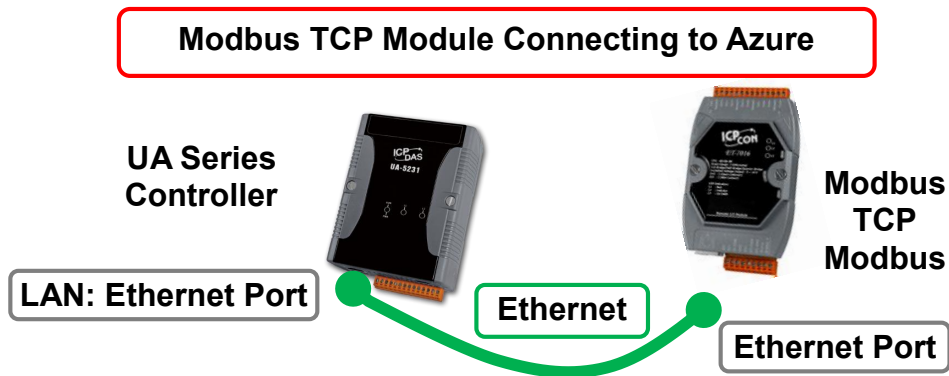
#### Function Diagram for Modbus TCP / Azure:



#### Application Solution:

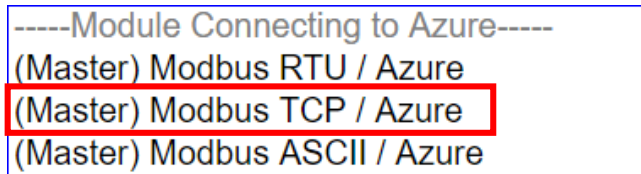


● **Modbus TCP Module Connecting to Azure**



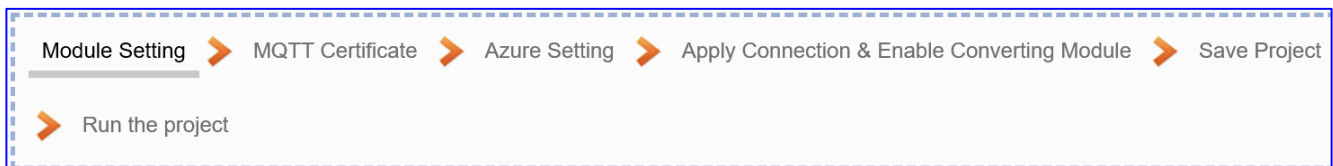
**Note:** The hardware/network connection methods please see the [Chapter 2](#) .

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.

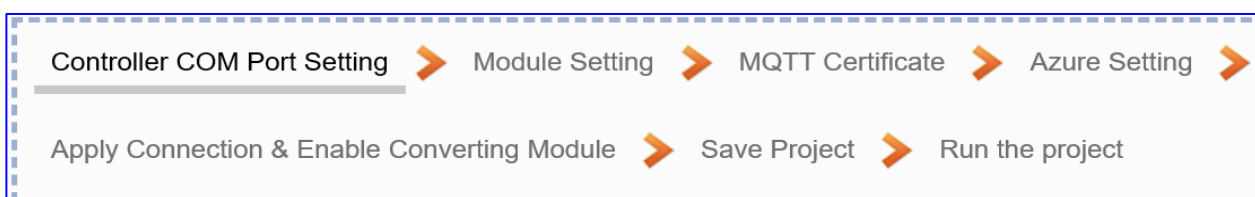


**[Step Box]:**

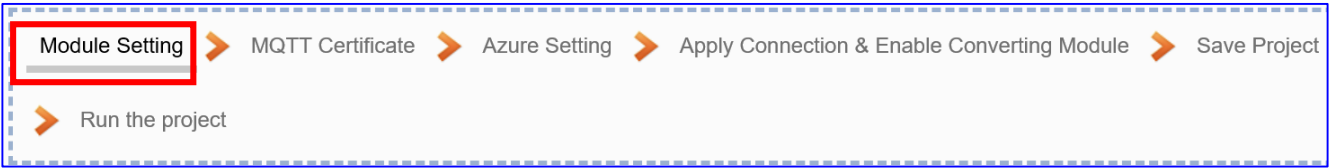
The Step Box of the [**Modbus TCP / Azure**] has the steps as below. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step-by-step and then can complete the project quickly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



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

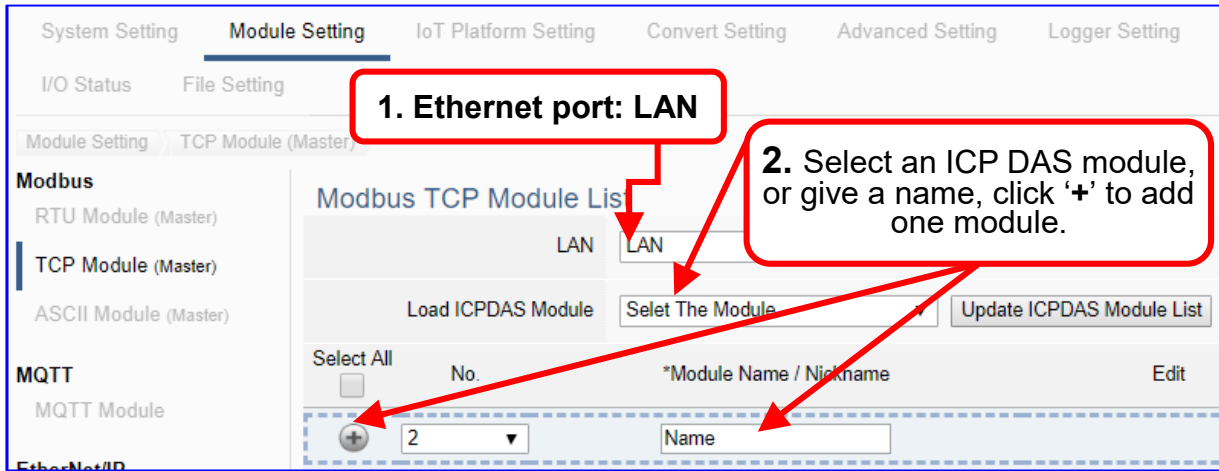


● **Step 1. Module Setting**

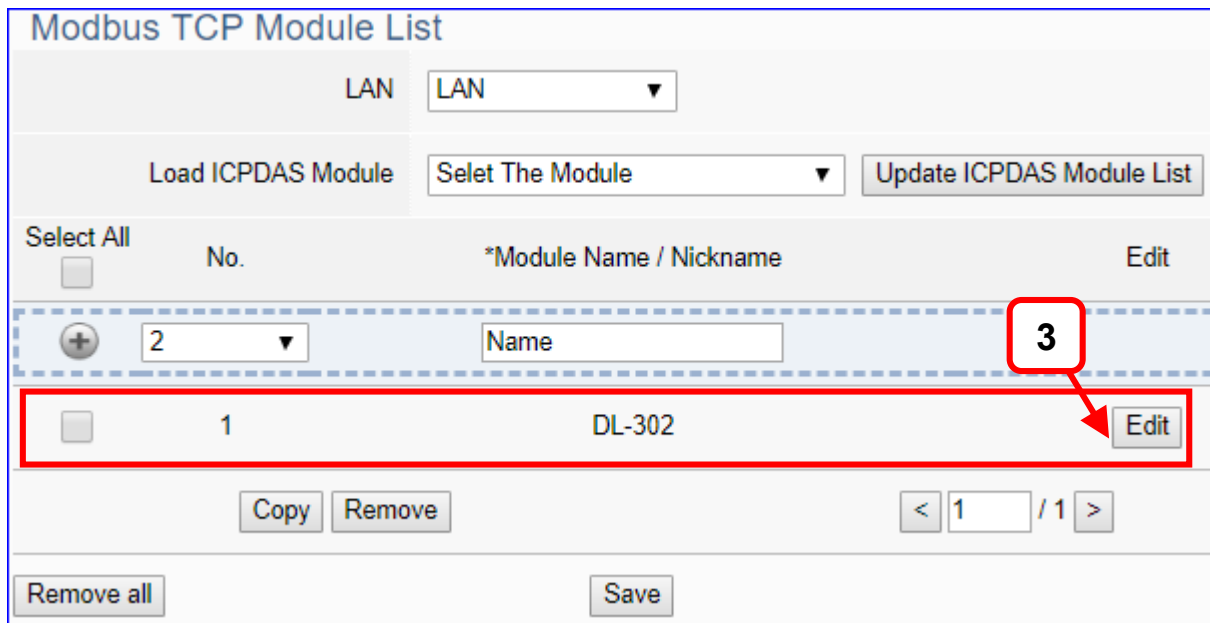


This page is for setting the communication values of the connected modules.

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: DL-302) 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] page:**

Module Content Setting	
No.	1
Module Name	DL-302
IP	192 . 168 . 81 . 251
Port	502
Slave ID	1
Timeout(ms)	500
Polling Rate(ms)	500
Modbus Mapping Table Setting	
Data Model	01 Coil Status(0x)
Start Address	0
Data Number	1
Create Tables	<input type="button" value="Add"/>

**Enter the IP by user case.**

**[Modbus Mapping Table Setting]**  
 Please check the module I/O & Modbus address via the module user manual.  
 Here use **DL-302** as an example. When select the DL-302 from the ICP DAS module list, the system auto set up the Modbus mapping table as below.  
 Data Model: 04 Input Registers (3x)  
 Start Address: 0  
 Data Number: 6  
 Type: 16-bit Short

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. (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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise
Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)		
				Address	0
				Number	6
				Type	Short
				<input type="button" value="Edit"/>	
<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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Variable name	Data Type	Description					
<b>02 Input Status(1x)</b>								
Table Display					<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Variable name	Data Type	Description					
<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				
<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="CO2"/>	Short	<input type="checkbox"/>	<input type="text" value="room1"/>				
1	<input type="text" value="Relative_humidity"/>	Short	<input type="checkbox"/>	<input type="text"/>				
2	<input type="text" value="Temperature_Celsius"/>	Short	<input type="checkbox"/>	<input type="text"/>				

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.



**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the "**Advanced Setting**" button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check "**Enable**" box, The Scaling conversion function will be activated.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

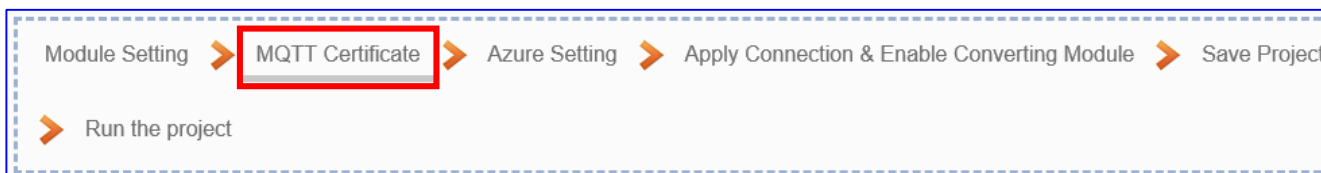
**Bitwise:**

**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

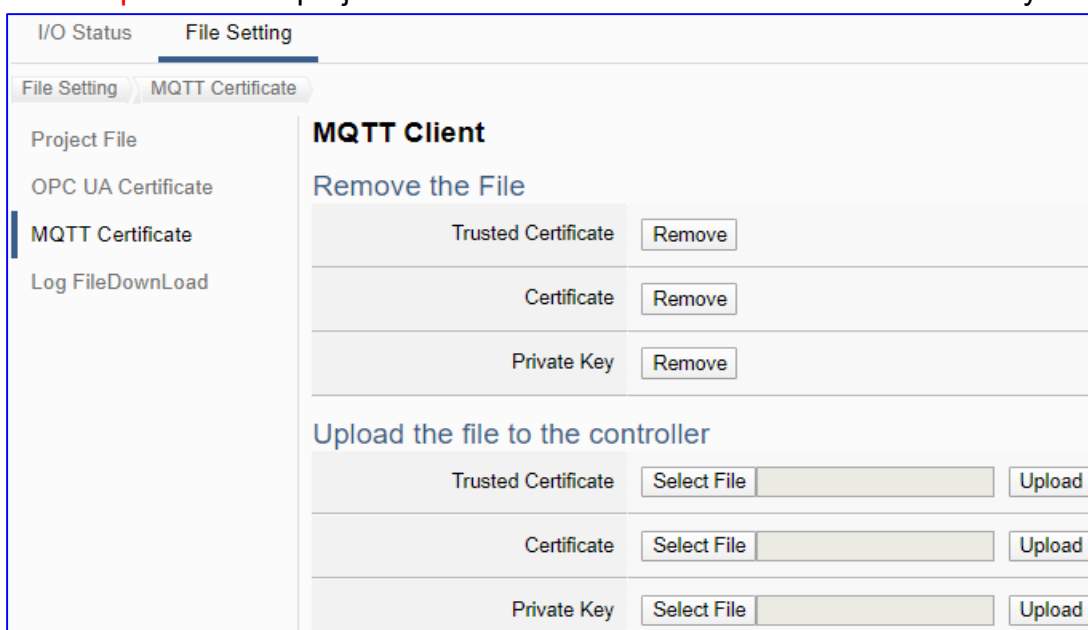
Modbus Mapping Table		Address	Nickname	Scaling	Bitwise
<b>03 Holding Registers(4x)</b>					
Table Display		<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Reference	Bitwise			
<b>04 Input Registers(3x)</b>					
Table Display		<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Reference	Bitwise			
0	CO2	<input type="button" value="Hide Detail"/>			
	Bit0	aa	Bit1	<input type="text"/>	
	Bit2	bb	Bit3	<input type="text"/>	
	Bit4	<input type="text"/>	Bit5	<input type="text"/>	
	Bit6	<input type="text"/>	Bit7	<input type="text"/>	
	Bit8	<input type="text"/>	Bit9	<input type="text"/>	
	Bit10	<input type="text"/>	Bit11	<input type="text"/>	
	Bit12	<input type="text"/>	Bit13	<input type="text"/>	
	Bit14	<input type="text"/>	Bit15	<input type="text"/>	

Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

● **Step 2. MQTT Certificate**

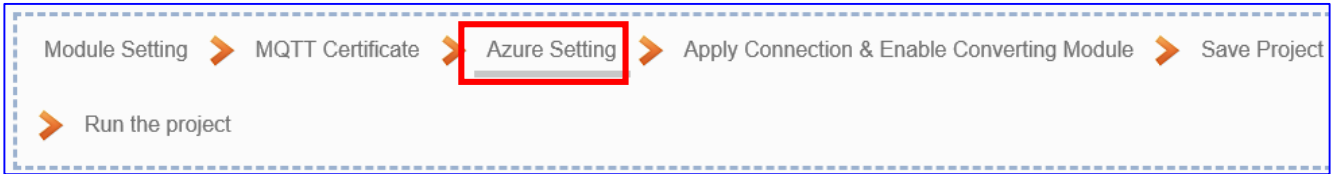


The [MQTT Certificate] is for setting up security communications to upload the **MQTT Trusted Certificate, Certificate and Private Key**. The users upload the file to the UA controller according to the type of obtained certificate. **If you want to perform Broker authentication, you need to upload the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



File Setting > MQTT Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the MQTT Trusted Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Certificate	<p><b>Select File:</b> select the MQTT Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“pem / cer / crt”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Private Key	<p><b>Select File:</b> select the MQTT Private Key of the device.</p> <p><b>Upload:</b> upload the MQTT Private Key file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be <b>“.key”</b>.</li> <li>If select a wrong file, the system will show an error message.</li> </ul>

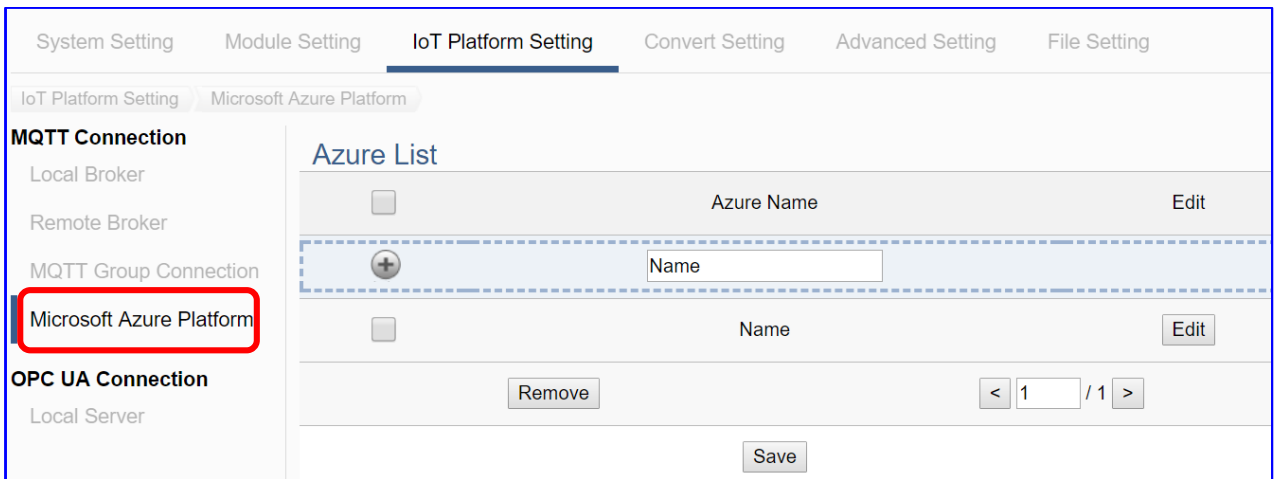
● **Step 3. Azure Setting**



Click the next step, and enter the **Step 3 [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 / Azure” 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.



<b>MQTT Connection &gt; Microsoft Azure Platform &gt; Azure List</b>	
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.

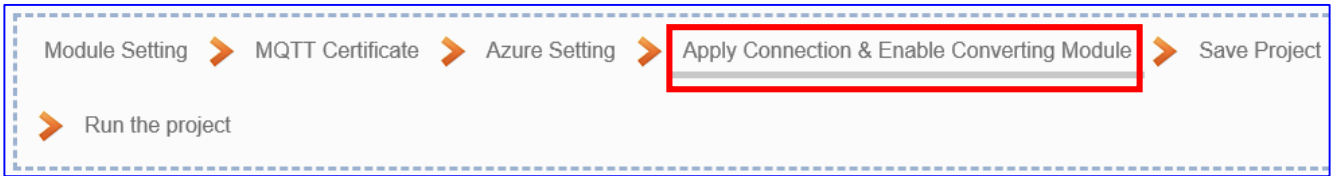
Click [Edit] button could enter the “**Azure Content Settings**” page:

### Azure Content Settings

Azure Name	<input type="text" value="Azure"/>
SAS Token	<pre>HostName=ICPDASIoTHub.azure-devices.net;DeviceId=UA-5231_1;SharedAccessSignature=SharedAccessSignature sr=ICPDASIoTHub.azure-devices.net%2Fdevices%2FUA-5231_1&amp;sig=9kUwQZc3OGLt8bMlrPWcfLSQT8AMOHqL0jhrVgqGZ6s%3D&amp;se=1575703053</pre>
Trusted Certificate	<input type="text" value="ca.crt"/>
Keep Alive Time(second)	<input type="text" value="60"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>

<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 user 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.
Trusted Certificate	Input the Root CA file name ( *.crt ) you previously downloaded for the UA controller from Microsoft Azure.
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
OK	Click to save and exit this page.

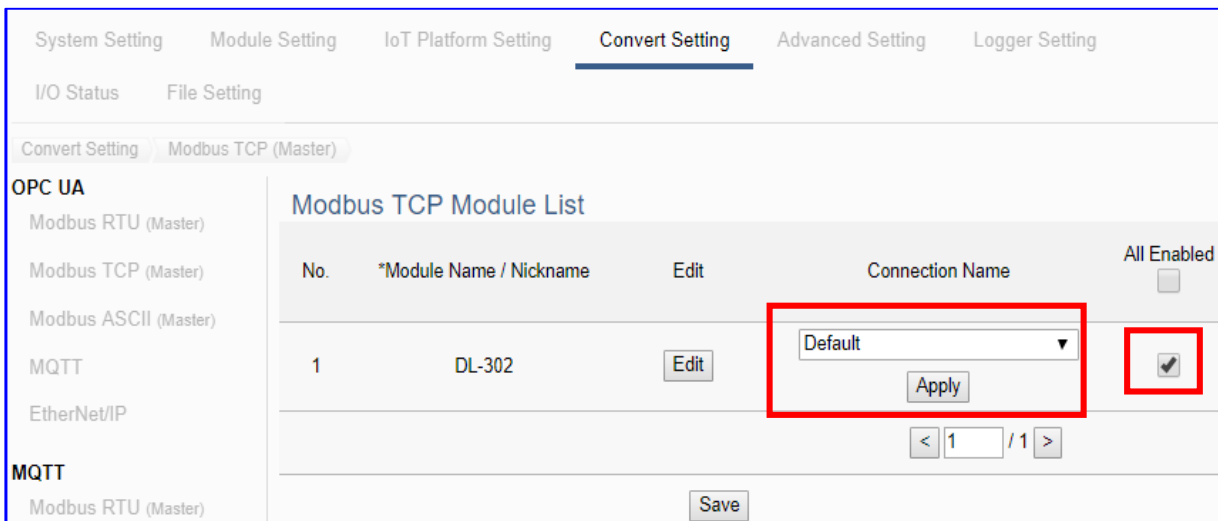
● **Step 4. Apply Connection & Enable Converting Module**



Click the next step, and enter the **Step 4 [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 through MQTT JSON group method, so this step will auto enter the **[Convert Setting > MQTT JSON - Modbus TCP (Master)]** page of Convert setting. The “Step Box” will prevent the user from selecting the wrong platform.

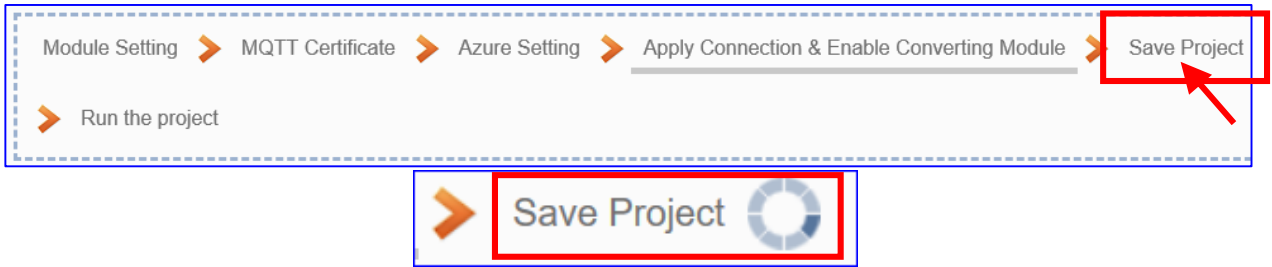
①**Select** and ②**apply** the connection name, ③**check** the module enabled box.



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 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.
< 1 / 1 >	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.

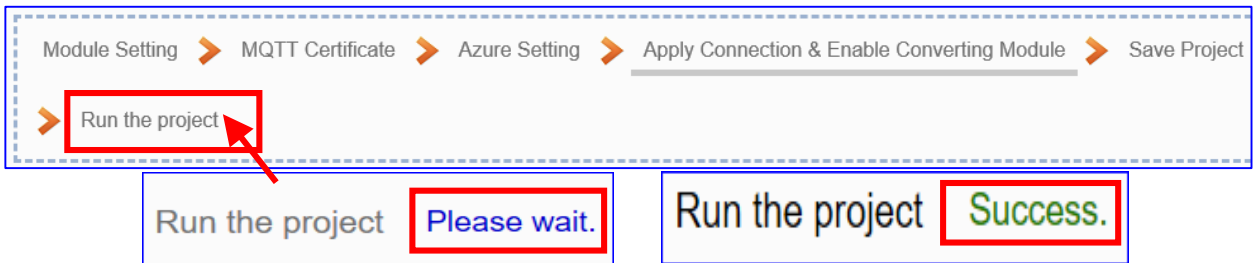
● **Step 5. Save Project**

The setting of this example is finished now. 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.



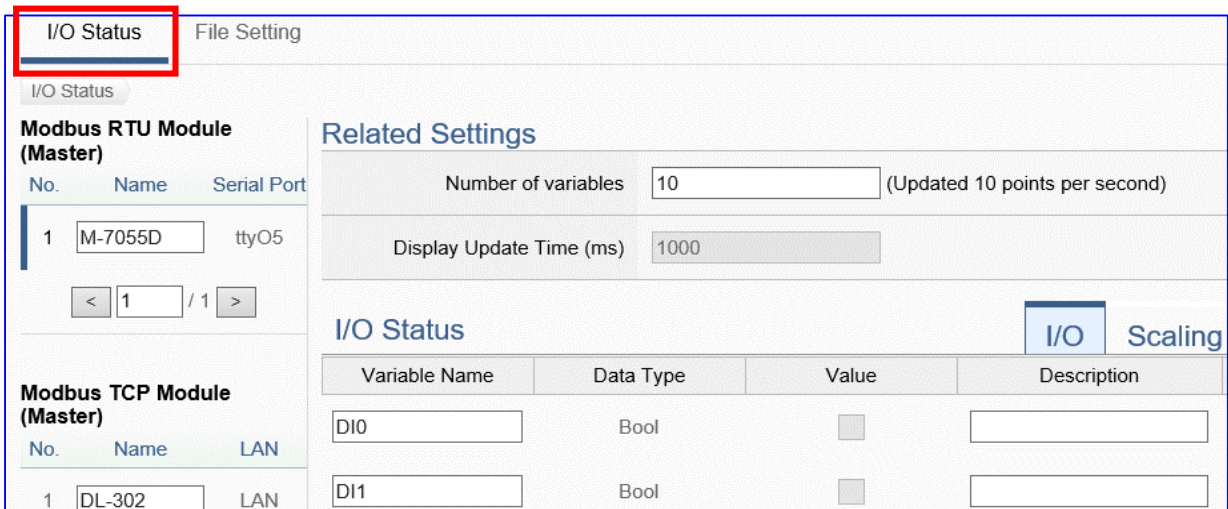
● **Step 6. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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 Azure communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.

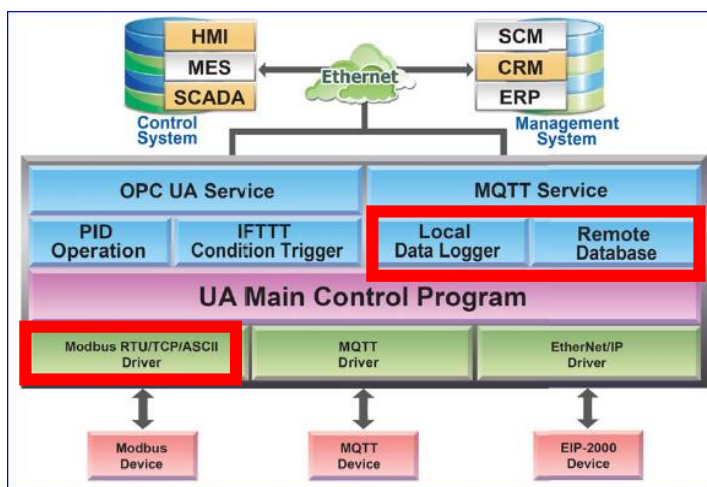


### 4.3 Data Log

UA series supports Data Logger function. Its Local Data Logger can save I/O data log to local CSV file, and record I/O status at the scheduled time. Furthermore, users can set the time interval of which CSV file to generate and divide on the local side. Its Remote Database can import I/O data collection directly into the remote SQL database, e.g. MS SQL, MySQL, MariaDB ..., for the Big Data analysis.

This section will introduce the setting steps and the function parameters of the “Data Log”. In the category, there are 6 items about Modbus RTU/TCP module for Local Data Logger or MS SQL, MySQL/MariaDB... Remote Database. This section will introduce the function items in 3 sub-sections.

- Data Log**
- (Master) Modbus RTU / Local Data Logger
  - (Master) Modbus TCP / Local Data Logger
  - (Master) Modbus RTU / MS SQL
  - (Master) Modbus TCP / MS SQL
  - MQTT / MS SQL
  - (Master) Modbus RTU / MySQL(MariaDB)
  - (Master) Modbus TCP / MySQL(MariaDB)
  - MQTT / MySQL(MariaDB)



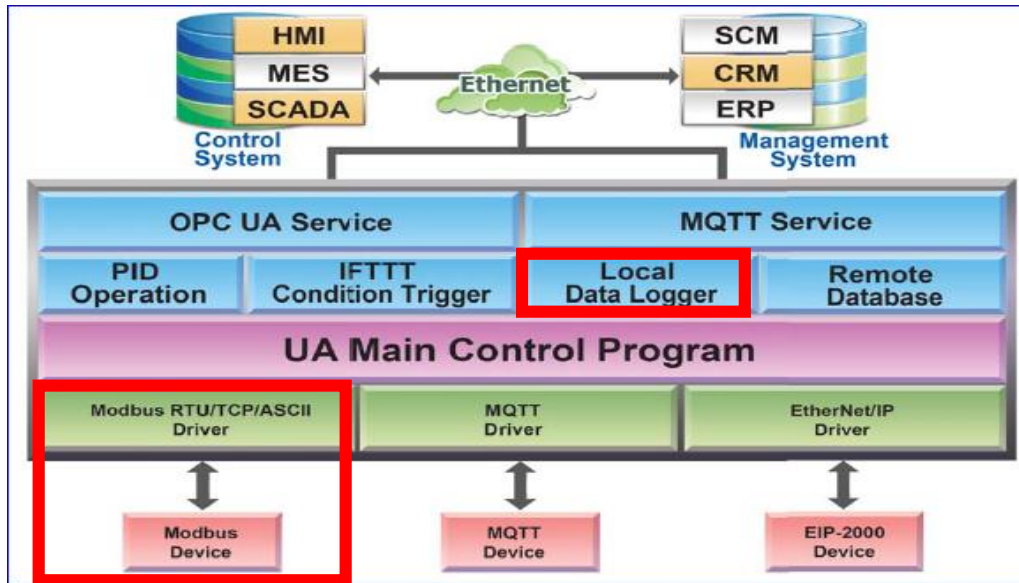
<p><b>Modbus RTU / Local Data Logger(4.3.1, Multi-module)</b>  <b>Modbus TCP / Local Data Logger(4.3.2, Use DL-302)</b></p>	<p>Provide users to record I/O data of Modbus RTU/TCP module to internal register.</p>
<p><b>Modbus RTU / MS SQL(4.3.3, Multi-module)</b>  <b>Modbus TCP / MS SQL (4.3.4, Use DL-302)</b>  <b>MQTT / MS SQL</b></p>	<p>Provide users to record I/O data of Modbus RTU/TCP module into remote database MS SQL.</p>
<p><b>Modbus RTU / MySQL(MariaDB) (4.3.5, M-7026)</b>  <b>Modbus TCP / MySQL(MariaDB) (4.3.6, Multi-module)</b>  <b>MQTT / MySQL(MariaDB)</b></p>	<p>Provide users to record I/O data of Modbus RTU/TCP module into MySQL/MariaDB remote database.</p>



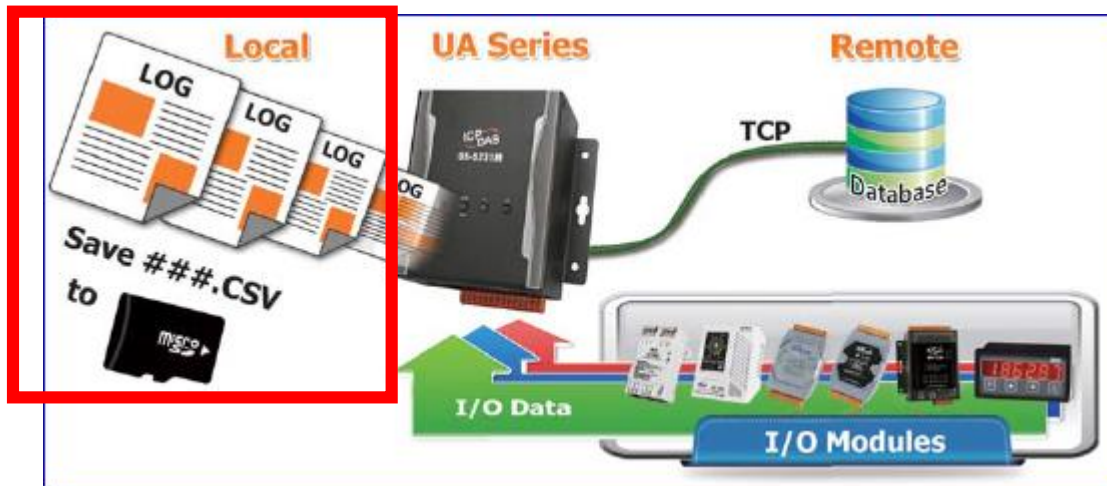
### 4.3.1 Function Wizard: Modbus / Local Data Logger (RTU Example, tM-AD4P2C2 + DL-302)

Local Data Logger supports to save I/O data log to Local CSV file in the SD card of the UA, and can record I/O status at the scheduled time. The Modbus / Local Data Logger settings include Modbus RTU and TCP. Here will introduce Modbus RTU and multi-module as the setting example.

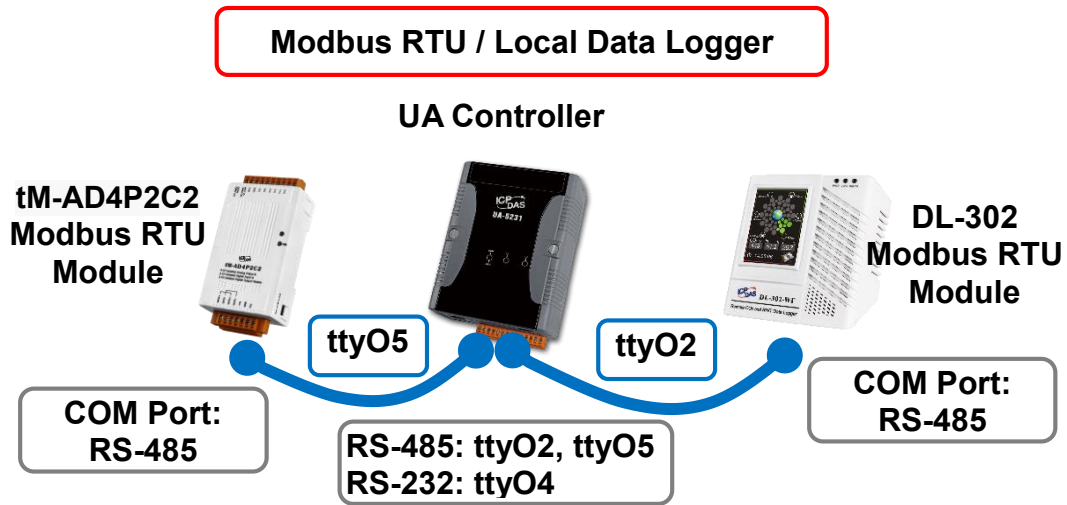
#### Modbus / Local Data Logger Function Diagram:



#### Application Solution:

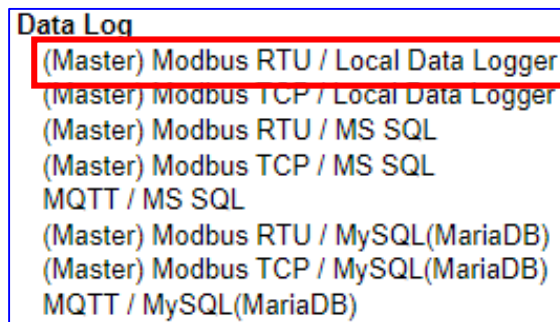


- **Modbus RTU / Local Data Logger: (Multi-module example: Use tM-AD4P2C2 + DL-302)**



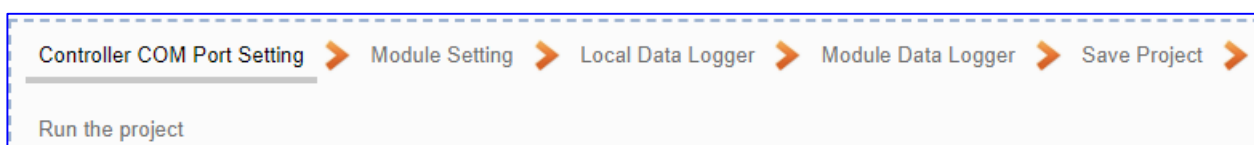
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus RTU multiple modules, e.g. **tM-AD4P2C2 (port: ttyO5)** and **DL-302 (port: ttyO2)**, as the picture. And save the data logger record into the CSV file of microSD card in the UA, user can choose the item [**Modbus RTU / Local Data Logger**] of the “Data Log” in the Function Wizard.

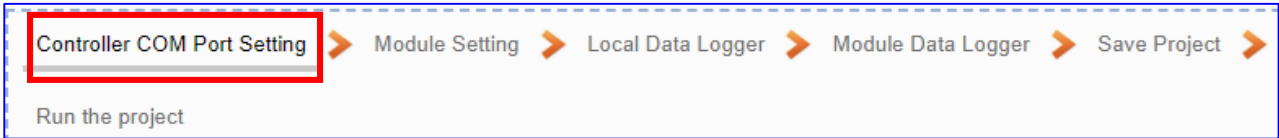


**[Step Box]:**

The Step Box of the [**Modbus RTU / Local Data Logger**] has the steps as the picture. 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**

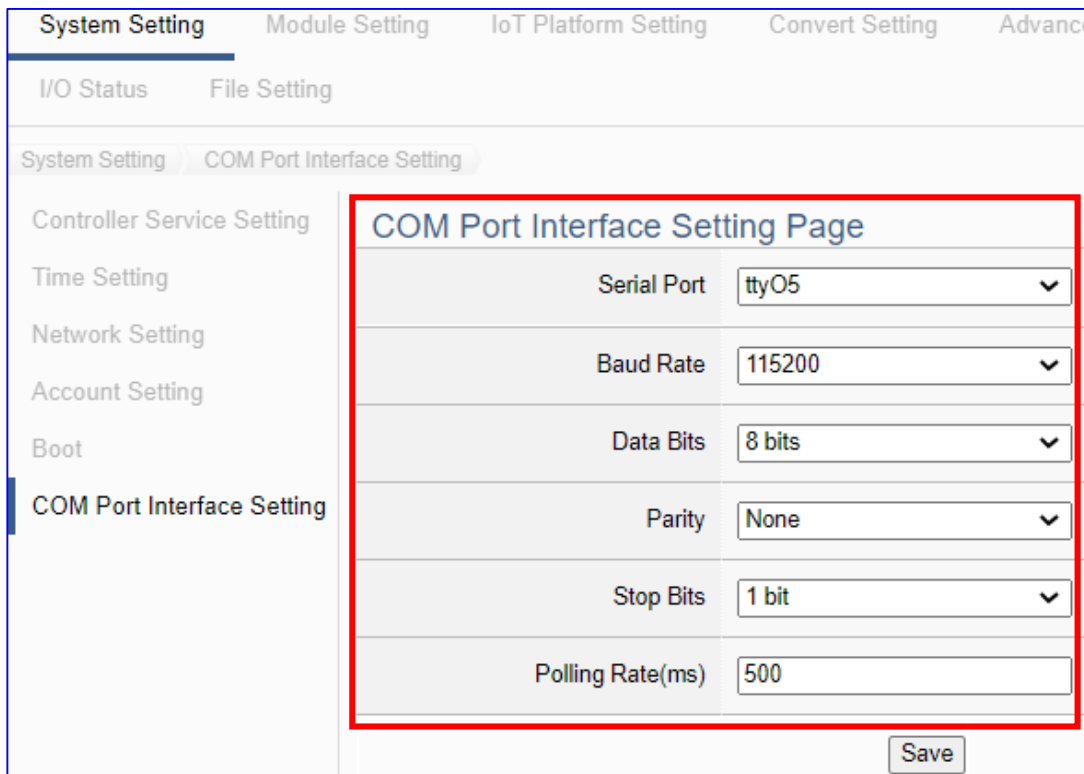


This page allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication. The user can find the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).

In this example: the communication data of module **tM-AD4P2C2** can be found in the product website <https://www.icpdas.com/en/product/tM-AD4P2C2>

COM Ports	
Ports	1 x RS-485
Baud Rate	1200 ~ 115200 bps
Data Format	(N, 8, 1), (N, 8, 2), (O, 8, 1), (E, 8, 1)
Protocol	DCON, Modbus/RTU, Modbus/ASCII

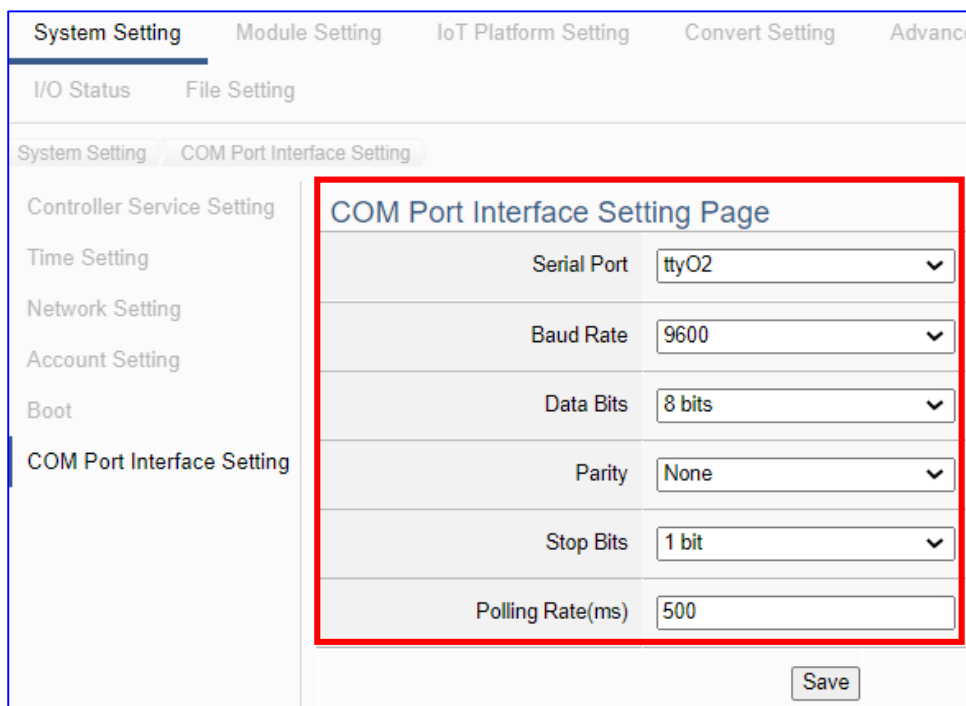
Setting as below. **Note:** This example uses **ttyO5** port to link tM-AD4P2C2 module. After settings, click “Save”.



In this example: the communication data of module **DL-302** can be found in the product user manual. <https://www.icpdas.com/en/product/DL-302> (as the following)

<b>Address:</b> Sets the address for a module.
Default: 1 Range: 0 ~ 255
<b>Protocol:</b> Sets the communication protocol.
- ModbusRTU (default) - DCON - DCONChkSum: uses DCON protocol and enables checksum validation feature
<b>Baud Rate</b>
Default: 9600 Support Baud Rate: 1200/ 2400/ 4800/ 9600/ 19200/ 38400/ 57600/ 115200 (unit: bps)
<b>Parity</b>
Default: N,8,1 Support format: N81, N82, E81, O81
<b>Response Delay (ms):</b> Sets the delay time between receiving the command and sending the data.
Default: 0 ms Range: 0 ~ 30 (unit: ms)
<b>Save:</b> Saves the modification and returns to the Settings menu. All the changes take effect immediately after saving changes.
<b>Skip:</b> Returns to the Settings menu without saving any changes.

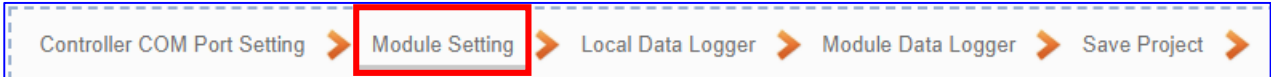
Setting as below. **Note:** This example uses **ttyO2** port to link DL-302 module. After settings, click “Save”.



The descriptions for the setting parameters:

<b>COM Port Interface Setting Page</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
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.

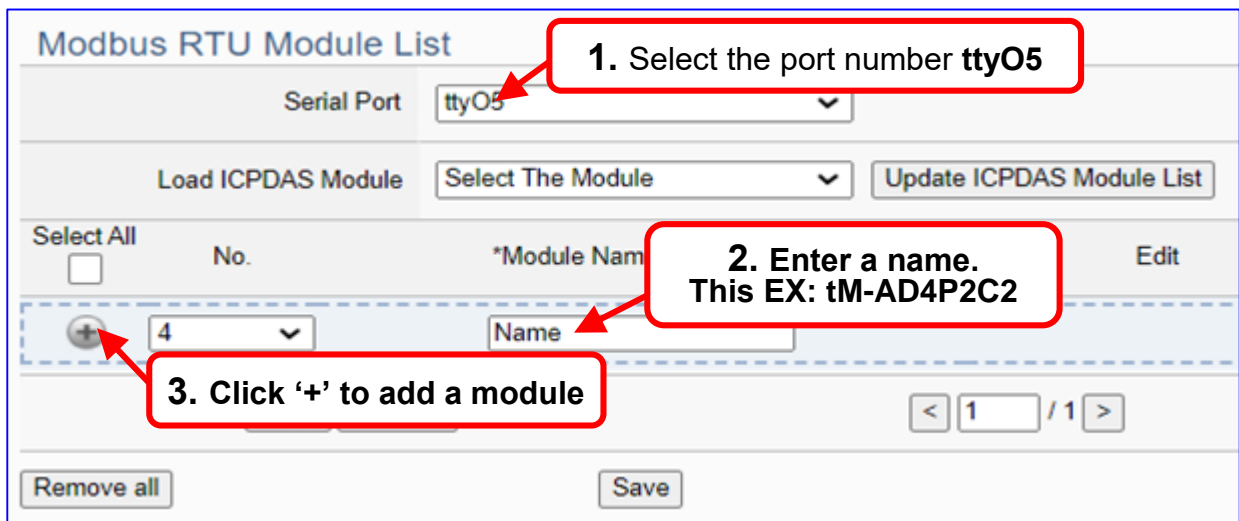
● **Step 2. Module Setting**



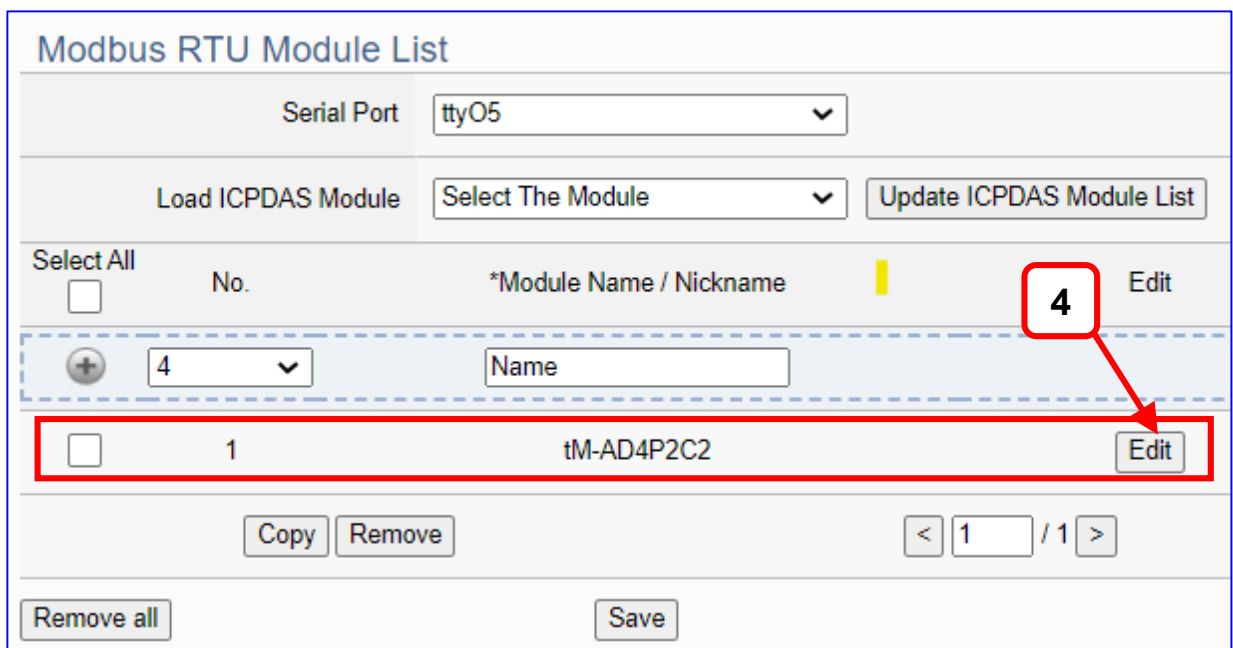
It auto-enter the first step, **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First check the 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.

**This example to set up tM-AD4P2C2 : (Port: ttyO5)**



The module (No.: 1, Name: **tM-AD4P2C2**) is as below, and then click [Edit] button to enter the “Module Content Setting” page.



[Module Content Setting] page can set the module and its Modbus mapping table.

**> Modbus Mapping Table Setting:**  
 Set module in the order of Data Model, Start Address and Data Number, then click “Add”.  
 p.s. If select from ICP DAS module list, system will auto setup the Modbus Mapping Table; if not, user can check the Modbus address or I/O number from the module manual or website.  
**[This Example] tM-AD4P2C2:**  
 Set up **AO x 2**  
**Data Model: 03** Holding Registers(4x)  
**Start Address: 0,**  
**Data Number: 2,**  
**Type: 16-bit Short,**  
 click **[Add]**, as the left picture.

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. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Write Retry	Check: If there is no response after the set time, it will retry to write again, max. 3 times.
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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

[**Module Mapping Table**] page can set up the Modbus mapping table.

If select from the ICP DAS module list, system will auto-setup the module and its Modbus Mapping Table. If not, user needs to check the module data and set up the module. Such as this example, we check the module manual and set up the tM-AD4P2C2 with 2 AO. The **Address** of the Modbus Mapping Table is as following.

**Address:**

Display and edit the Modbus Mapping Table.

Module Content Setting																								
No.	<input type="text" value="1"/>																							
Module Name	<input type="text" value="tM-AD4P2C2"/>																							
Slave ID	<input type="text" value="2"/>																							
Timeout(ms)	<input type="text" value="500"/>																							
Write Retry	<input type="checkbox"/> <input type="text" value="1"/>																							
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="2"/>																							
Type	<input type="text" value="16-bit Short"/> ▼																							
Create Tables	<input type="button" value="Add"/>																							
Modbus Mapping Table																								
<table border="1"> <thead> <tr> <th colspan="2">Address</th> <th>Nickname</th> <th>Scaling</th> <th>Bitwise</th> </tr> <tr> <th>Coil Status(0x)</th> <th>Input Status(1x)</th> <th>Holding Registers(4x)</th> <th>Input Registers(3x)</th> <th></th> </tr> </thead> <tbody> <tr> <td colspan="2"></td> <td> <table border="1"> <tr> <td>Address</td> <td>0</td> </tr> <tr> <td>Number</td> <td>2</td> </tr> <tr> <td>Type</td> <td>Short</td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Edit"/></td> </tr> </table> </td> <td></td> <td></td> </tr> </tbody> </table>		Address		Nickname	Scaling	Bitwise	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>Short</td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Edit"/></td> </tr> </table>	Address	0	Number	2	Type	Short	<input type="button" value="Edit"/>			
Address		Nickname	Scaling	Bitwise																				
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>Short</td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Edit"/></td> </tr> </table>	Address	0	Number	2	Type	Short	<input type="button" value="Edit"/>															
Address	0																							
Number	2																							
Type	Short																							
<input type="button" value="Edit"/>																								
<input type="button" value="OK"/> <input type="button" value="Cancel"/>																								



**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
<b>02 Input Status(1x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
<b>03 Holding Registers(4x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	Vin0	Short	<input type="checkbox"/>					
1	Vin1	Short	<input type="checkbox"/>					
<b>04 Input Registers(3x)</b>								
Table Display								
Address	Variable name	Data Type	Swap	Description				
<div style="border: 2px solid red; border-radius: 10px; padding: 5px; display: inline-block;"> <b>Can set up the Variable Nickname, Ex: Vin0, Vin1...</b> </div>								
					OK		Cancel	

Modbus Mapping Table – Nickname	
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.

**This example: use the same way to set up DL-302 (Serial Port: ttyO2)**

The screenshot shows the 'Modbus RTU Module List' interface. At the top, the 'Serial Port' is set to 'ttyO2'. Below this, there is a 'Load ICPDAS Module' section with a dropdown menu set to 'Select The Module' and an 'Update ICPDAS Module List' button. The main table has columns for 'No.', '\*Module Name / Nickname', and 'Edit'. A new row is being added, highlighted with a dashed blue border. A red callout '3. Click '+' to add' points to the '+' button in the 'No.' column. Another red callout '2. Enter the module name This Ex: DL-302' points to the 'Name' input field. A third red callout '1. Select the serial port ttyO2' points to the 'Serial Port' dropdown menu.

Add a module (as the picture, this example: **DL-302**), and then click the [Edit] of the module to enter the screen of "Module Content Setting".

The screenshot shows the 'Modbus RTU Module List' interface after a module has been added. The table now contains two rows. The first row is highlighted with a dashed blue border and has a '+' button in the 'No.' column and an empty 'Name' field. The second row has 'No.' 2, '\*Module Name / Nickname' 'DL-302', and an 'Edit' button. A red callout '4' points to the 'Edit' button. The 'Serial Port' is still 'ttyO2' and the 'Load ICPDAS Module' dropdown is still 'Select The Module'.

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 mapping table

Module Content Setting	
No.	<input type="text" value="2"/>
Module Name	<input type="text" value="DL-302"/>
Slave ID	<input type="text" value="1"/>
Timeout(ms)	<input type="text" value="500"/>
Write Retry	<input type="checkbox"/> <input type="text" value="1"/>
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="6"/>
Type	<input type="text" value="16-bit Short"/>
Create Tables	<input type="button" value="Add"/>

**This Example: DL-302  
Set up AO Short x 6**

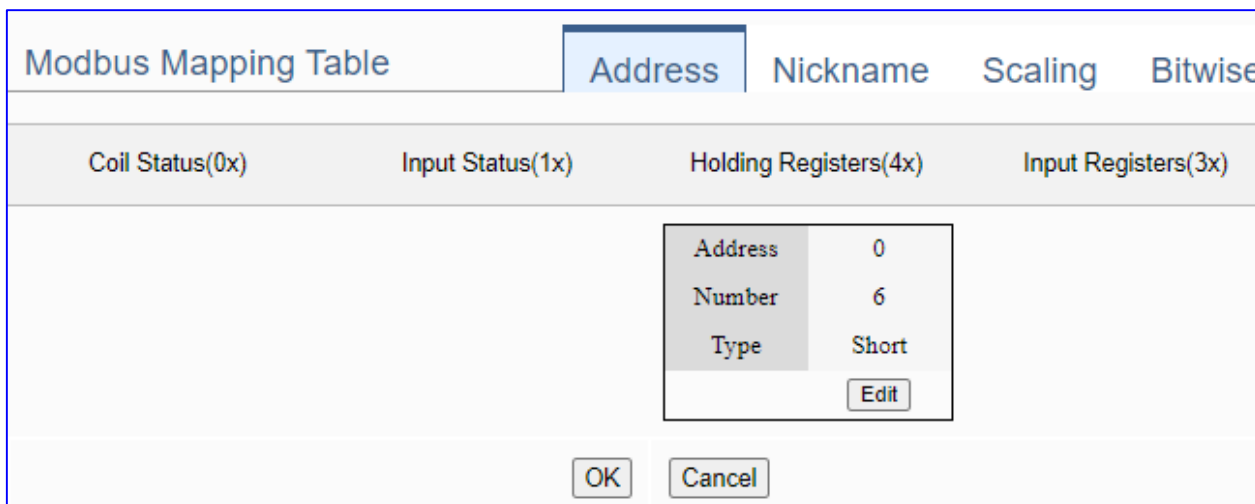
**[ Modbus Mapping Table Setting ]**  
**Data Models: 03 Holding Registers(4x)**  
**Start Address: 0**  
**Data Number : 6**  
**Type: 16-bit Short**  
**→ Click [Add]**

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. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Write Retry	Check: If there is no response after the set time, it will retry to write again, max. 3 times.
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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address:**

Display and edit the Modbus Mapping Table.



Modbus Mapping Table – Address	
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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

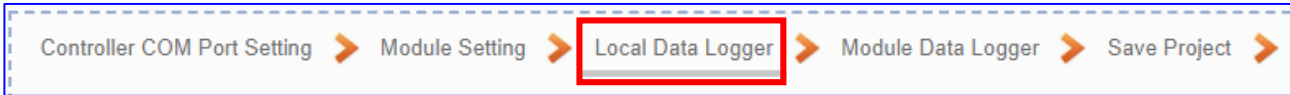
Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
01 Coil Status(0x)								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
02 Input Status(1x)								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
03 Holding Registers(4x)								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	CO2	Short	<input type="checkbox"/>					
1	RH							
2	TC							
3	TF	Short	<input type="checkbox"/>					
4	DC	Short	<input type="checkbox"/>					
5	DF	Short	<input type="checkbox"/>					
04 Input Registers(3x)								

**The user can define the nickname for the variables, as the picture.**

Modbus Mapping Table – Nickname	
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.

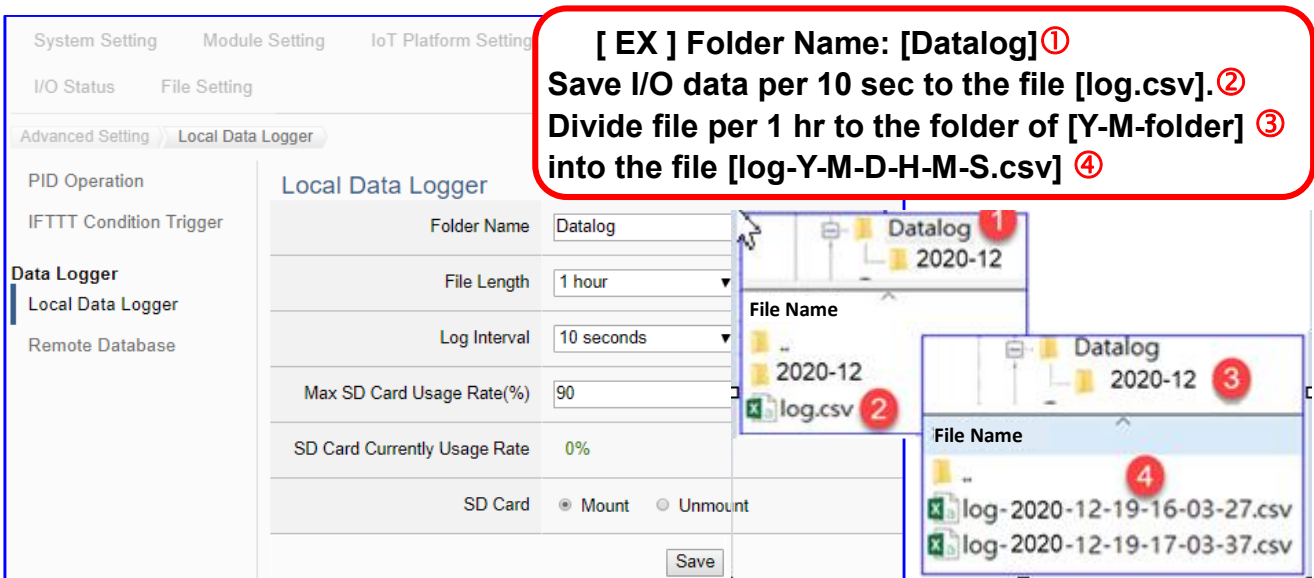
● **Step 3. Local Data Logger**



Click the next step, and enter the **Step 3 [Local Data Logger]** of the UI setting. This page is for setting the saving microSD card, e.g. the folder name, file length, log interval, card usage rate, and mount/unmount.

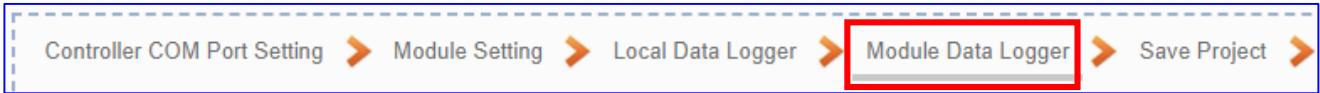
We select the “Modbus RTU / Local Data Logger” conversion at the beginning, so this step will auto enter the **[Data Logger > Local Data Logger]** page of Advanced Setting.

**Before setting the parameters, user can refer to the "Example of the Log file and fields for the local data log file" that after the step 6.**



Advanced Setting > Data Logger > Local Data Logger	
Folder Name	The folder name in microSD card of UA, user definable. The I/O data will save into the file “log.csv” under this folder.
File Length	Unit: hour. User can select per 1, 2, 3, ... 8, 12, or 24 hours to divide the log.csv into the file “log-Y-M-D-H-M-S.csv” under the folder “Y-M”. (e.g. 2020-12)
Log Interval	The interval to save I/O data per seconds, minutes or hours.
Max SD Card Usage Rate (%)	Set up the maximum usage rate (Unit: %) of UA microSD card. If the data current rate meet the max rate, the oldest data will be removed first.
SD Card Currently Usage Rate	Display the current usage rate of UA microSD card (show %).
SD Card	Mount: Click to mount microSD card and begin to record data. Unmount: Click to unmount microSD card and stop record data.
Save	Click to save the settings of this item.

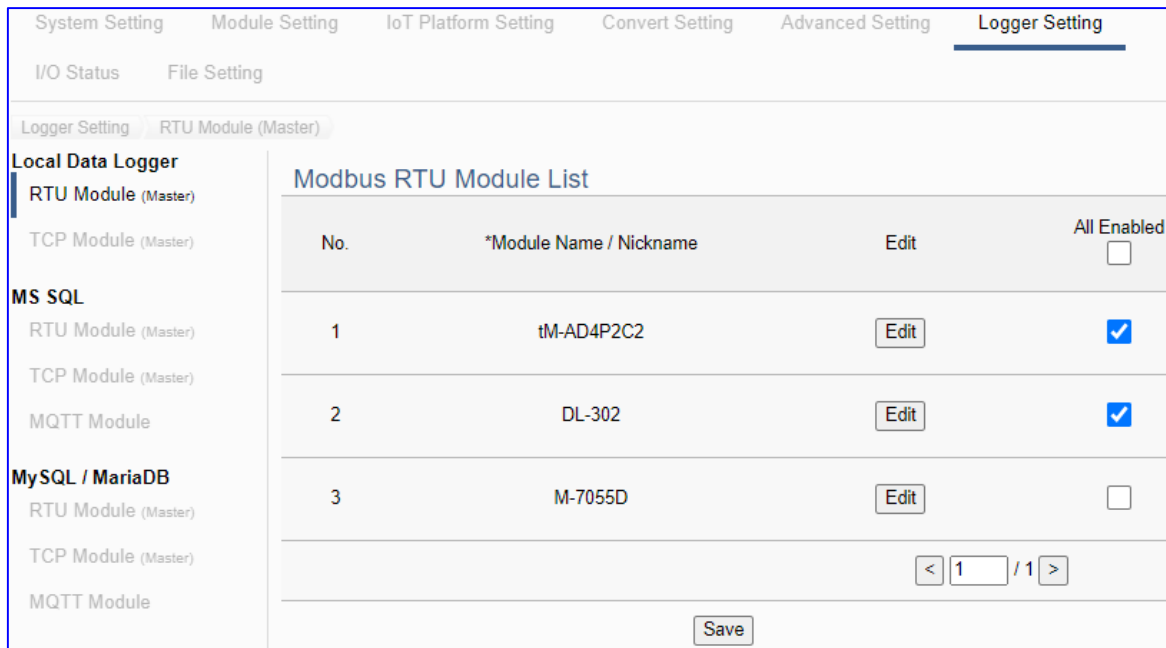
● **Step 4. Module Data logger**



Click the next step, and enter the **Step 4 [Module Data logger]** UI setting. This step is for enabling the Modbus TCP module for Local Data Logger.

We select the “Modbus RTU / Local data logger” of Data Log at the beginning, so this step will auto enter the [**Local Data Logger > RTU Module (Master)**] page of Logger Setting. The “Step Box” will prevent the user from selecting the wrong platform.

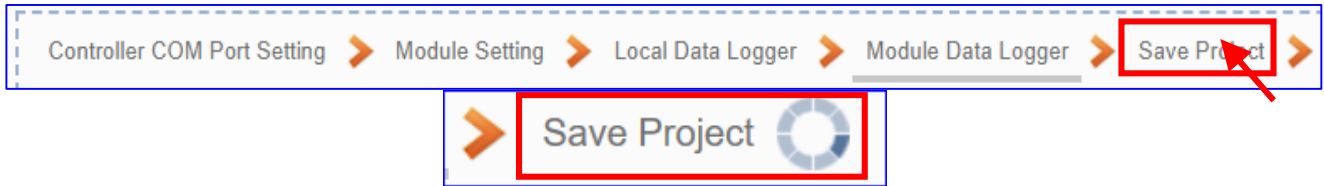
Check the box of the module user wants to do the data logger, e.g. tM-AD4P2C2 and DL-302



Logger Setting > Local Data Logger > TCP Module – Modbus TCP 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 <input checked="" type="checkbox"/> / <input type="checkbox"/>	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.
Edit	If user wants to enable some I/O channels for data logger, click [Edit] of that module to enter the “Content Setting”. It is normal to set all channels as enabled, and the function will not affect the unconnected channels.
<input type="button" value="&lt;"/> 1 / 1 <input type="button" value="&gt;"/>	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.

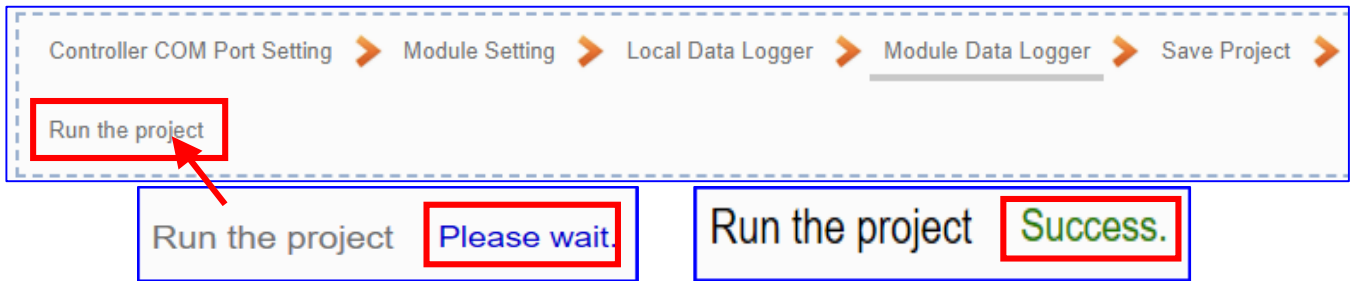
● **Step 5. Save Project**

The setting of this example is finished now. 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**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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 Data Logger function. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.

**I/O Status** File Setting

I/O Status

**Modbus RTU Module (Master)**

No.	Name	Serial Port
2	DL-302	ttyO2
3	M-7055D	ttyO2
1	tM-AD4P2C2	ttyO5

< 1 / 1 >

**Modbus TCP Module (Master)**

No.	Name	LAN
-----	------	-----

< 1 / 1 >

**Related Settings**

Number of variables: 10 (Updated 10 points per second)

Display Update Time (ms): 1000

**I/O Status** I/O Scaling Bitwise

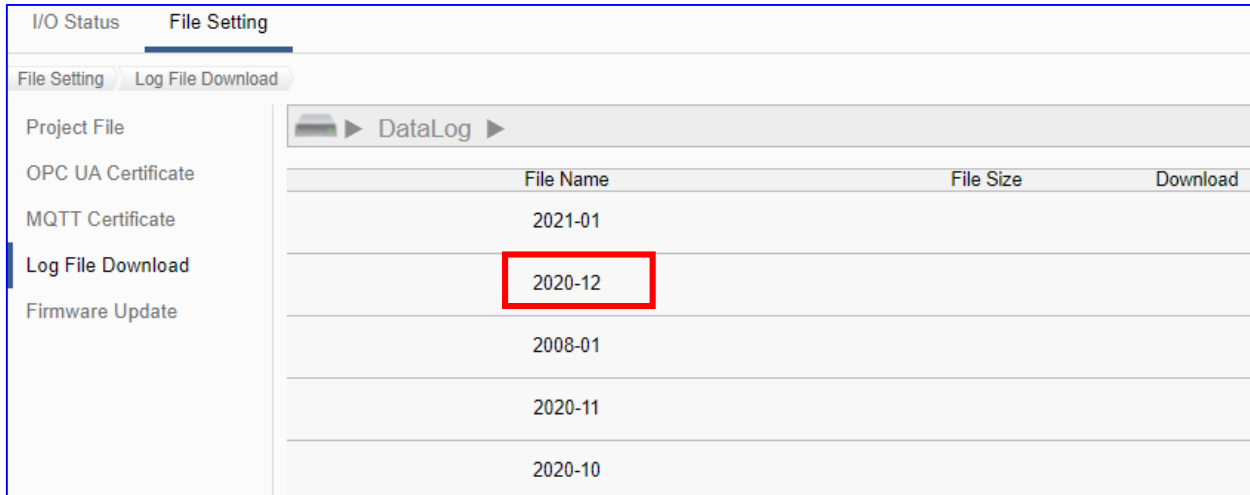
Variable Name	Data Type	Value	Description	Status
Vin0	Short	2378		Good
Vin1	Short	1284		Good

< 1 / 1 >



● **Get the Log File of the Local Data Logger:**

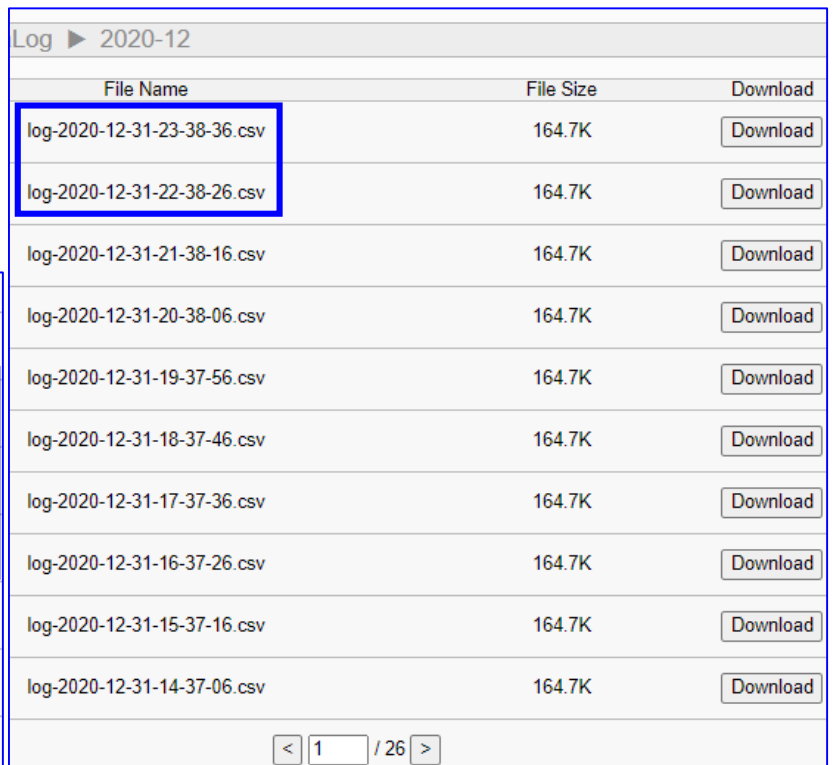
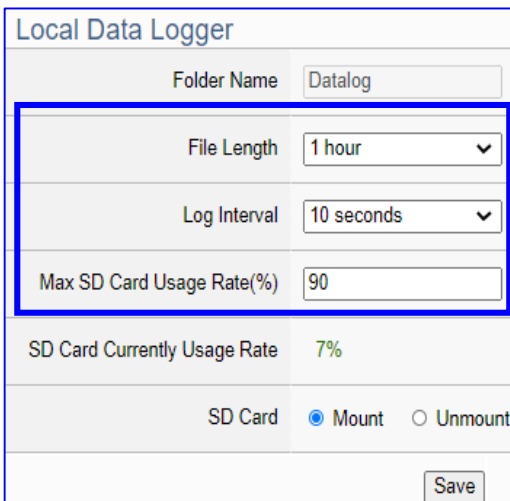
1. Enter the menu [**File Setting > Log File Download**], click the folder name, ex: [2020-12].



2. Download the Log file you need. The relationship of file name and the time is as below:  
 The log file is divided according to the **"File Length"** set by the user and stored in the file under the **"year-month"** folder. The file name/example is as follows:

**"log-20xx(year)-xx(month)-xx(day)-xx(hour)-xx(minute)- xx(seconds).csv"**

**[Example]:** Set **"File Length"** as **1 hr.**, **"Log interval"** as **10 Sec.**, and data recording will be performed every 10 seconds, and the file generation time is **every 1 hour + 10 seconds**. When the system time comes to these specific times, the file will automatically end and save the file to the storage. The system will generate a new file to continue recording the log data of the next 1 hour + 10 seconds, and so on. When all files are saved to over the setting of **"Max SD Card Usage Rate(%)"**, the oldest file will be overwritten.



● **Example of the CVS file and fields for the local data log file:**

- ✧ The Log record will be stored to the microSD card in the UA PAC, and the default name is the **folder "Datalog"** which can be customized by the user.
- ✧ I/O data records will be stored in the **file "log.csv"** under this folder.
- ✧ The log data file is divided every 1, 2, 3... 8, 12 or 24 hours according to user settings, and saved to the **file "log-YYYY-MM-DD-HH-MM-SS.csv"** under the **folder "YYYY-MM"**.
- ✧ Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

The example file is shown as the following.

	A	B	C	D
1	# Log file created/rotated Wednesday	9 Dec 20 04:46:29 GMT		
2	Timestamp	Name	Value	Status
3	2020-12-09-12-46-29-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
4	2020-12-09-12-46-29-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1133	Good
5	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.CO2	699	Good
6	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.RH	7089	Good
7	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.TC	2225	Good
8	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.TF	7205	Good
9	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.DC	1671	Good
10	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.DF	6207	Good
11	2020-12-09-12-46-39-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
12	2020-12-09-12-46-39-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1152	Good
13	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.CO2	699	Good
14	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.RH	7089	Good
15	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.TC	2225	Good
16	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.TF	7205	Good
17	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.DC	1671	Good
18	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.DF	6207	Good
19	2020-12-09-12-46-49-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
20	2020-12-09-12-46-49-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1172	Good
21	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.CO2	699	Good
22	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.RH	7085	Good
23	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.TC	2225	Good
24	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.TF	7205	Good
25	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.DC	1670	Good
26	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.DF	6206	Good
27	2020-12-09-12-46-59-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
28	2020-12-09-12-46-59-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1193	Good
29	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.CO2	698	Good
30	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.RH	7089	Good
31	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.TC	2223	Good
32	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.TF	7201	Good
33	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.DC	1668	Good
34	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.DF	6202	Good
35	2020-12-09-12-47-09-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
36	2020-12-09-12-47-09-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1213	Good

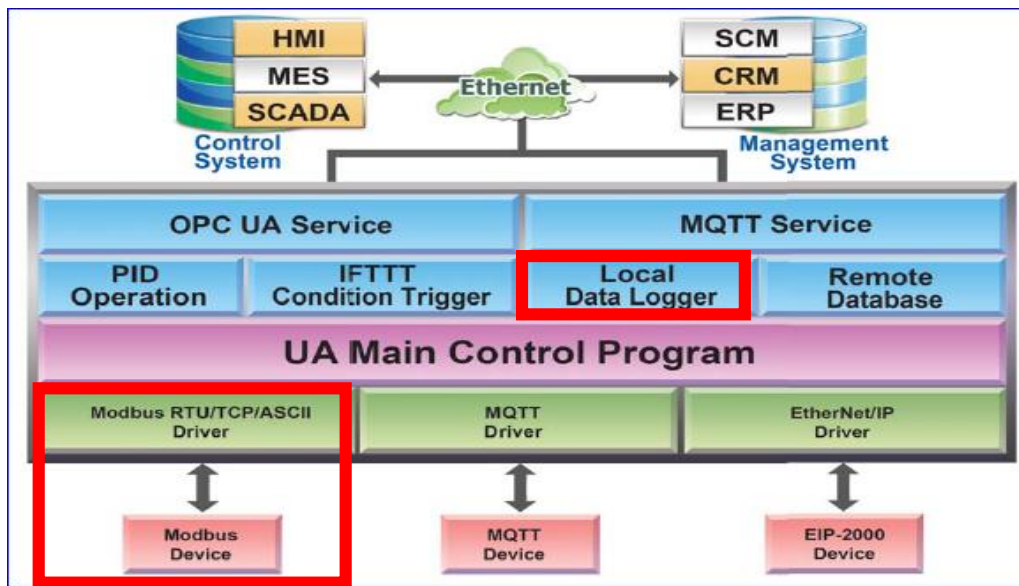
◀ ▶
log-2020-12-09-12-46-29
+

### 4.3.2 Function Wizard: Modbus / Local Data Logger (TCP Example, DL-302)

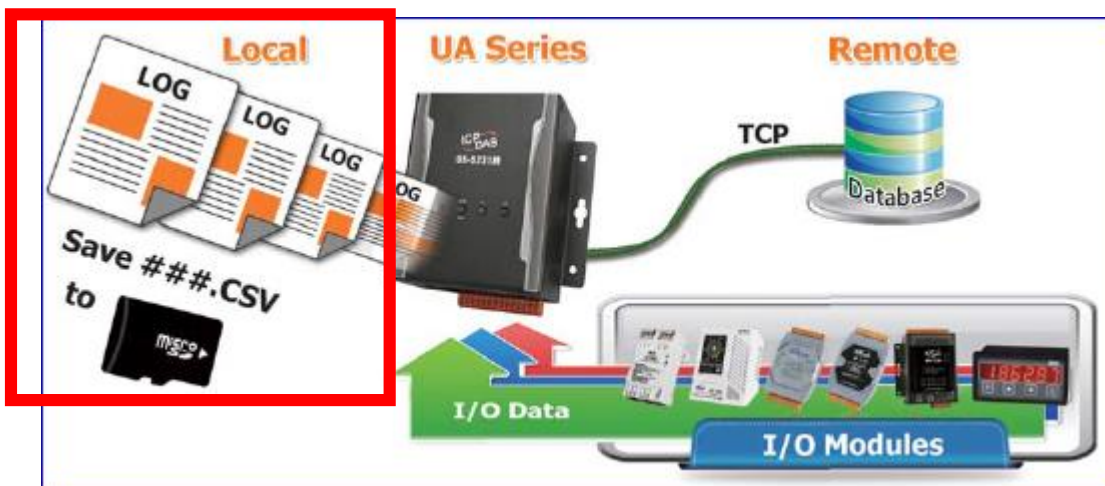
Local Data Logger supports to save I/O data log to Local CSV file in the SD card of the UA, and can record I/O status at the scheduled time. Furthermore, users can set the time interval of which CSV file to generate and divide on the local side.

The Modbus / Local Data Logger settings include Modbus RTU and TCP. Here will introduce Modbus TCP as the setting sample.

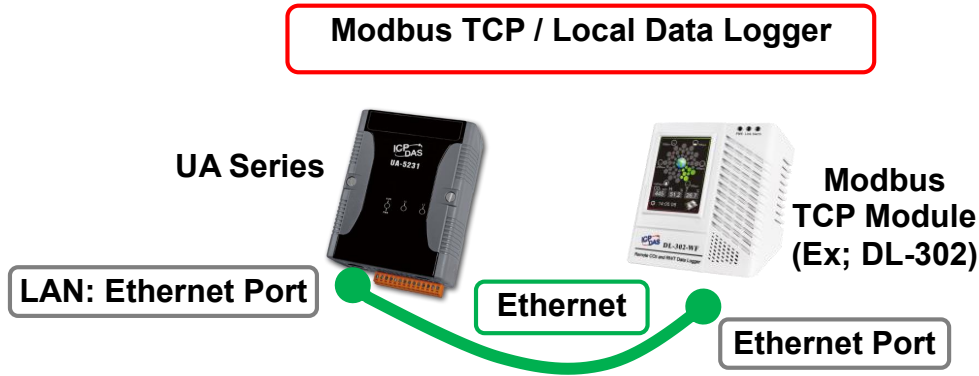
#### Modbus / Local Data Logger Function Diagram:



#### Application Solution:

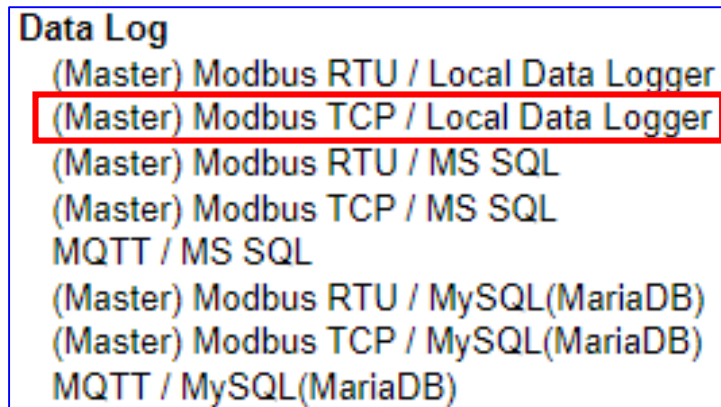


● **Modbus TCP / Local Data Logger**



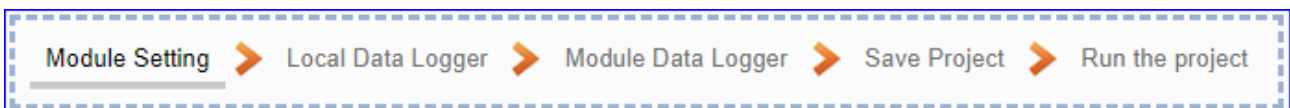
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus TCP module (via Ethernet, as the picture) and save the data logger record to the microSD card in the UA, user can choose the item [**Modbus TCP / Local Data Logger**] of the “Data Log” in the Function Wizard.

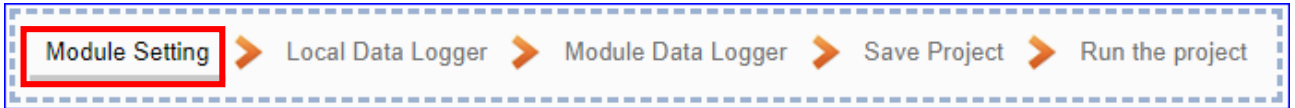


**[Step Box]:**

The Step Box of the [**Modbus TCP / Local Data Logger**] has 5 steps. 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.

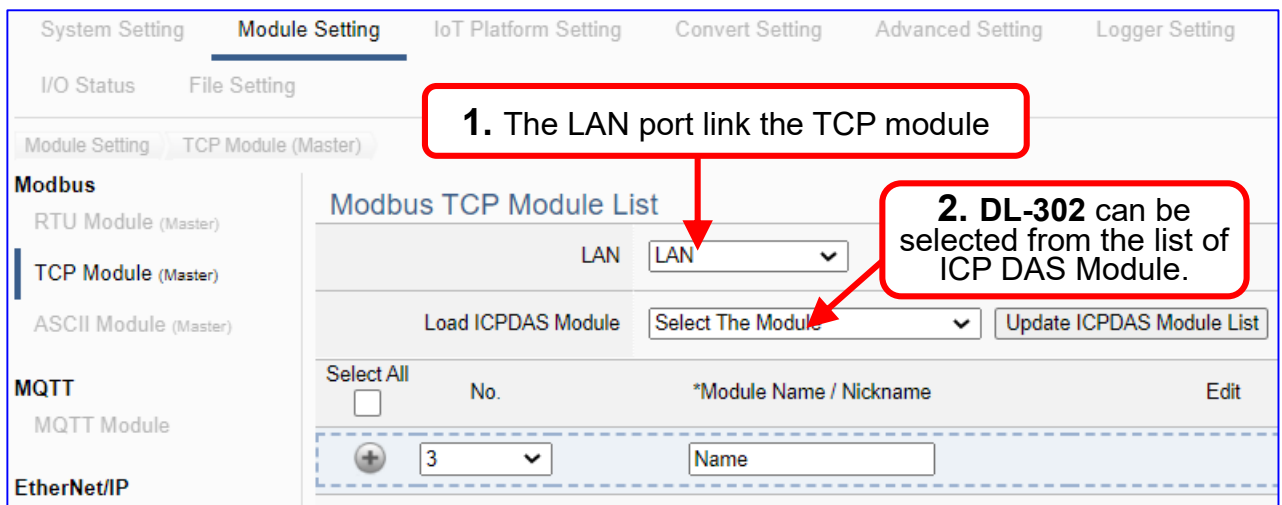


● **Step 1. Module Setting**

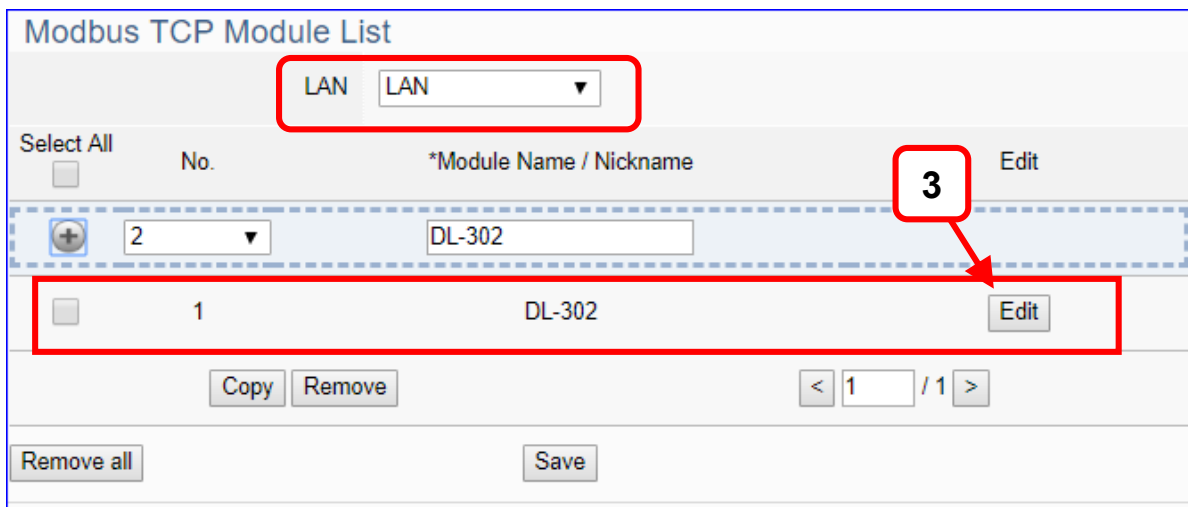


It auto-enter the first step, **Step 1 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First check the LAN port that connected with the module. If you use the module in the ICP DAS module list, you can select it and system will auto add and set up the module. Or you can give a name (Default name: Name). Click [ + ] button to 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: **DL-302**) 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] can set up the module and the Modbus mapping table:

Module Content Setting	
No.	1
Module Name	DL-302
IP	192 . 168 . 81 . 251
Port	502
Slave ID	1
Timeout(ms)	500
Polling Rate(ms)	500
Modbus Mapping Table Setting	
Data Model	04 Input Registers(3x)
Start Address	0
Data Number	3
Type	16-bit Short
Create Tables	<input type="button" value="Add"/> Success.

**This Example: DL-302**

**[IP] 192.168.81.251 (by user case)**

**DL-302 is in the ICP DAS Module list, when select it from the list the system will auto-add module and set up its Modbus Mapping Table. If not in the list, user has to set up the [Modbus Mapping Table] as below:**

**Data Model: 04 Input Registers(3x)**  
**Start Address: 0**  
**Data Number: 6**  
**Type: 16-bit Short**  
**→ Click [ 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. (Range: 1 ~ 247)				
Timeout(ms)	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) <div style="float: right; border: 1px solid black; padding: 2px;"> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="background-color: #007bff; color: white;">01 Coil Status(0x)</td></tr> <tr><td>02 Input Status(1x)</td></tr> <tr><td>03 Holding Registers(4x)</td></tr> <tr><td>04 Input Registers(3x)</td></tr> </table> </div>	01 Coil Status(0x)	02 Input Status(1x)	03 Holding Registers(4x)	04 Input Registers(3x)
01 Coil Status(0x)					
02 Input Status(1x)					
03 Holding Registers(4x)					
04 Input Registers(3x)					
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address:**

Display and edit the Modbus Mapping Table (in the order of mapping DO, DI, AO & AI).

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Variable name	Data Type	Description					
<b>02 Input Status(1x)</b>								
Table Display					<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Variable name	Data Type	Description					
<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				
<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="CO2"/>	Short	<input type="checkbox"/>	<input type="text" value="room1"/>				
1	<input type="text" value="Relative_humidity"/>	Short	<input type="checkbox"/>	<input type="text"/>				
2	<input type="text" value="Temperature_Celsius"/>	Short	<input type="checkbox"/>	<input type="text"/>				
3	<input type="text" value="Temperature_Fahrenheit"/>	Short	<input type="checkbox"/>	<input type="text"/>				
4	<input type="text" value="Dew_point_temperature_"/>	Short	<input type="checkbox"/>	<input type="text"/>				

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.



**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated.

**Note: this setting page is only displayed for AI/AO of the Modbus RTU/TCP.**

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

**Bitwise:**

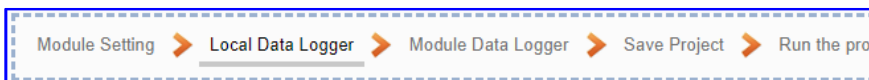
**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

**Note: this setting page is only displayed for AI/AO of the Modbus RTU/TCP.**

Modbus Mapping Table	Address	Nickname	Scaling	Bitwise
<b>03 Holding Registers(4x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference	Bitwise		
<b>04 Input Registers(3x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference	Bitwise		
0	<input type="text" value="CO2"/> <input type="button" value="Bit0"/> <input type="text" value="aa"/> <input type="button" value="Bit2"/> <input type="text" value="bb"/> <input type="button" value="Bit4"/> <input type="text"/> <input type="button" value="Bit6"/> <input type="text"/> <input type="button" value="Bit8"/> <input type="text"/> <input type="button" value="Bit10"/> <input type="text"/> <input type="button" value="Bit12"/> <input type="text"/> <input type="button" value="Bit14"/> <input type="text"/>	<input type="button" value="Hide Detail"/> <input type="button" value="Bit1"/> <input type="text"/> <input type="button" value="Bit3"/> <input type="text"/> <input type="button" value="Bit5"/> <input type="text"/> <input type="button" value="Bit7"/> <input type="text"/> <input type="button" value="Bit9"/> <input type="text"/> <input type="button" value="Bit11"/> <input type="text"/> <input type="button" value="Bit13"/> <input type="text"/> <input type="button" value="Bit15"/> <input type="text"/>		

Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

● **Step 2. Local Data Logger**



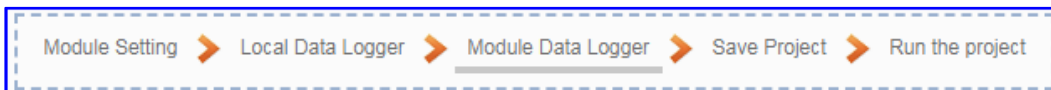
Click the next step, and enter the **Step 2 [Local Data Logger]** of the UI setting. This page is for setting the saving microSD card, e.g. the folder name, file length, log interval, card usage rate, and mount/unmount.

We select the “Modbus TCP / Local Data Logger” conversion at the beginning, so this step will auto enter the **[Data Logger > Local Data Logger]** page of Advanced Setting. The “Step Box” will prevent the user from selecting the wrong platform.

**Before setting the parameters, user can refer to the "Example of the Log file and fields for the local data log file" that after the step 5.**

Advanced Setting > Data Logger > Local Data Logger	
Folder Name	The folder name in microSD card of UA, user definable. The I/O data will save into the file “log.csv” under this folder.
File Length	Unit: hour. User can select per 1, 2, 3, ... 8, 12, or 24 hours to divide the log.csv into the file “log-Y-M-D-H-M-S.csv” under the folder “Y-M”.
Log Interval	The interval to save I/O data per seconds, minutes or hours.
Max SD Card Usage Rate (%)	Set up the maximum usage rate (Unit: %) of UA microSD card. If the data current rate meet the max rate, the oldest data will be removed first.
SD Card Currently Usage Rate	Display the current usage rate of UA microSD card (show %).
SD Card	Mount: Click to mount microSD card and begin to record data. Unmount: Click to unmount microSD card and stop record data.
Save	Click to save the settings of this item.

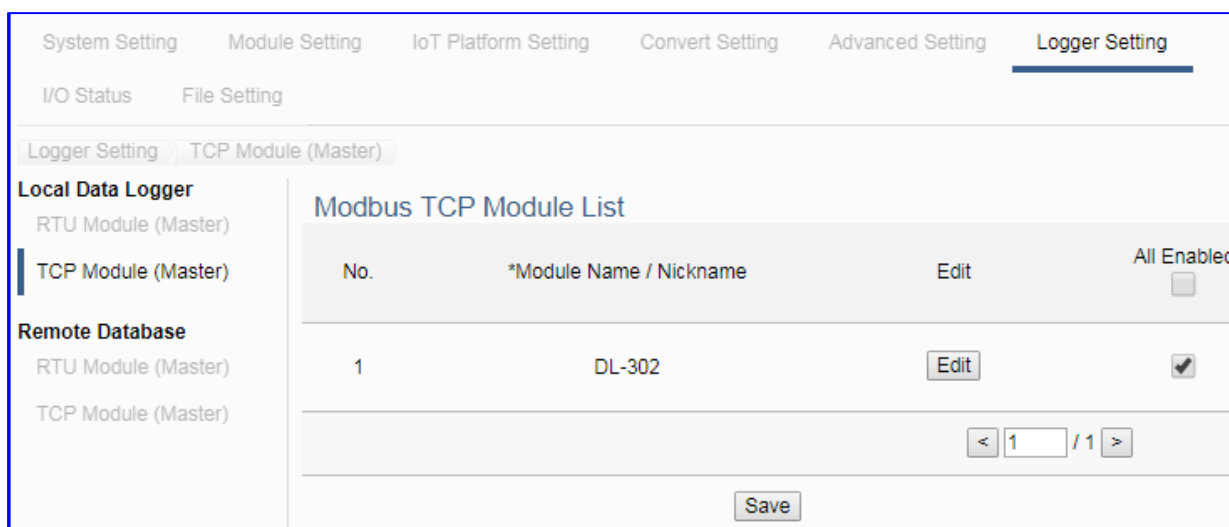
● **Step 3. Module Data logger**




Click the next step, and enter the **Step 3 [Module Data logger]** UI setting. This step is for enabling the Modbus TCP module for Local Data Logger.

We select the “Modbus TCP / Local data logger” of Data Log at the beginning, so this step will auto enter the **[Local Data Logger > TCP Module (Master)]** page of Logger Setting. The “Step Box” will prevent the user from selecting the wrong platform.

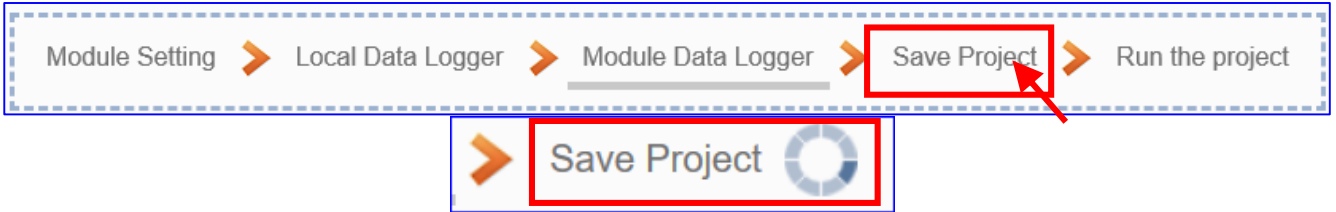
Please check the box of the module user wants to do the data logger, e.g. DL-302.



Logger Setting > Local Data Logger > TCP Module – Modbus TCP 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 data logger, click [Edit] of that module to enter the “Content Setting”. It is normal to set all channels as enabled, and the function 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.

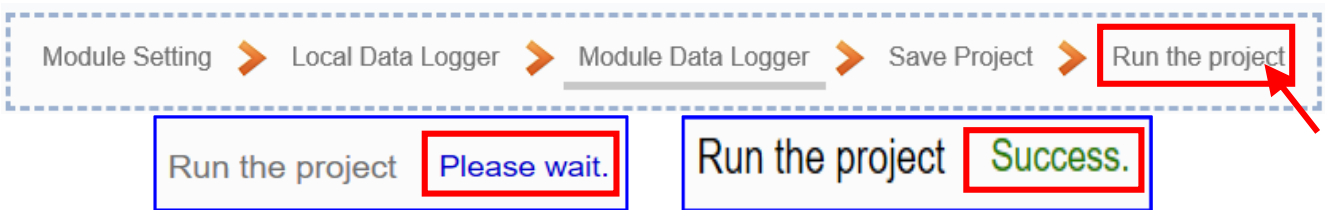
● **Step 4. Save Project**

The setting of this example is finished now. 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.



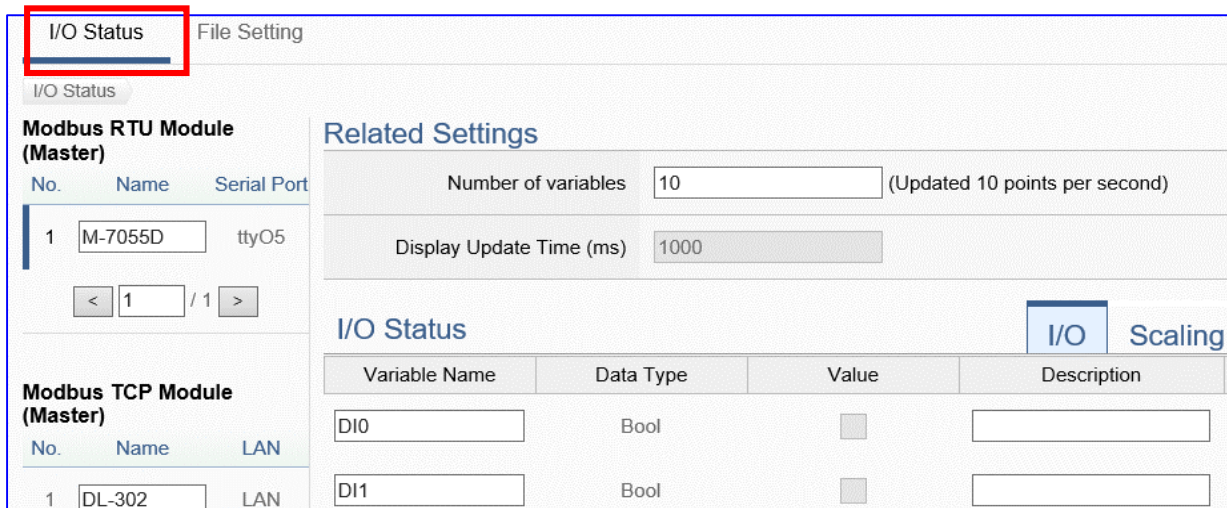
● **Step 5. Run the Project**

The project, after saving, needs to be executed. Click the next step **[Run the Project]**. This step can also via the **[System Setting > Controller Service Setting > Run Project]** to Stop and Run the project.



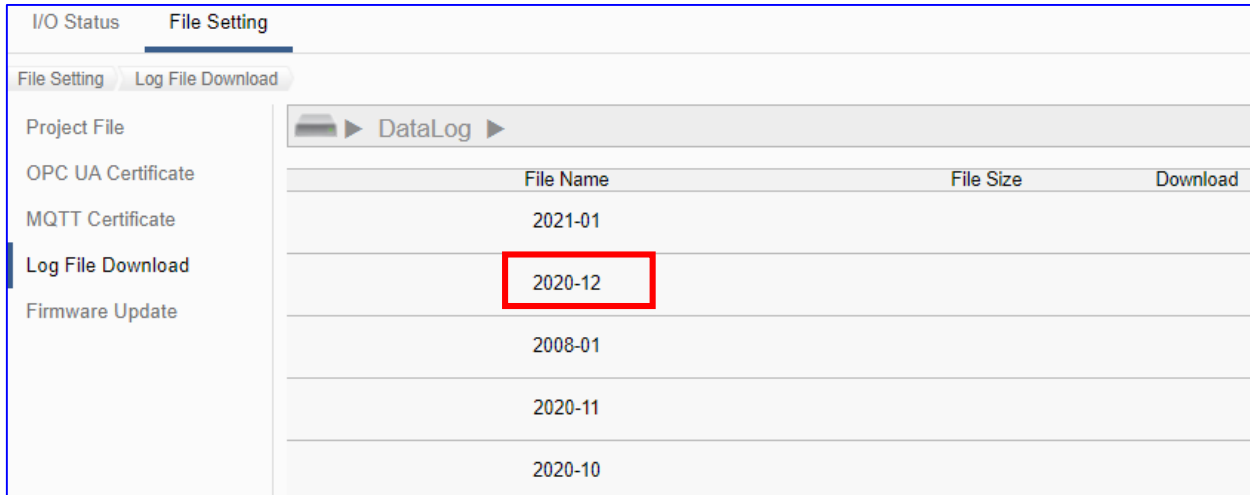
When the words **“Please wait”** disappears, the new words **“Success”** appears, that means the UA controller is running new project successfully. Then the Step Box will disappear automatically now, and back to the first screen view of the Web UI.

The new project now completes the setting, uploading and running in the UA controller and can process the Data Logger function. Users can see the I/O status from the menu **[I/O Status]**. For more about the Web UI settings, please refer to CH4 and CH5.



● **Get the Log File of the Local Data Logger:**

1. Enter the menu [File Setting > Log File Download], click the folder name, ex: [2020-12].



2. Download the Log file you need. The relationship of file name and the time is as below:  
 The log file is divided according to the "File Length" set by the user and stored in the file under the "year-month" folder. The file name/example is as follows:

"log-20xx(year)-xx(month)-xx(day)-xx(hour)-xx(minute)- xx(seconds).csv"

**[Example]:** Set "File Length" as 1 hr., "Log interval" as 10 Sec., and data recording will be performed every 10 seconds, and the file generation time is every 1 hour + 10 seconds. When the system time comes to these specific times, the file will automatically end and save the file to the storage. The system will generate a new file to continue recording the log data of the next 1 hour + 10 seconds, and so on. When all files are saved to over the setting of "Max SD Card Usage Rate(%)", the oldest file will be overwritten.

**Local Data Logger**

Folder Name: Datalog

File Length: 1 hour

Log Interval: 10 seconds

Max SD Card Usage Rate(%): 90

SD Card Currently Usage Rate: 7%

SD Card:  Mount  Unmount

Save

Log ▶ 2020-12

File Name	File Size	Download
log-2020-12-31-23-38-36.csv	164.7K	Download
log-2020-12-31-22-38-26.csv	164.7K	Download
log-2020-12-31-21-38-16.csv	164.7K	Download
log-2020-12-31-20-38-06.csv	164.7K	Download
log-2020-12-31-19-37-56.csv	164.7K	Download
log-2020-12-31-18-37-46.csv	164.7K	Download
log-2020-12-31-17-37-36.csv	164.7K	Download
log-2020-12-31-16-37-26.csv	164.7K	Download
log-2020-12-31-15-37-16.csv	164.7K	Download
log-2020-12-31-14-37-06.csv	164.7K	Download

< 1 / 26 >

● **Example of the CVS file and fields for the local data log file:**

- ✧ The Log record will be stored to the microSD card in the UA PAC, and the default name is the **folder "Datalog"** which can be customized by the user.
- ✧ I/O data records will be stored in the **file "log.csv"** under this folder.
- ✧ The log data file is divided every 1, 2, 3... 8, 12 or 24 hours according to user settings, and saved to the **file "log-YYYY-MM-DD-HH-MM-SS.csv"** under the **folder "YYYY-MM"**.
- ✧ Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

The example file is shown as the following.

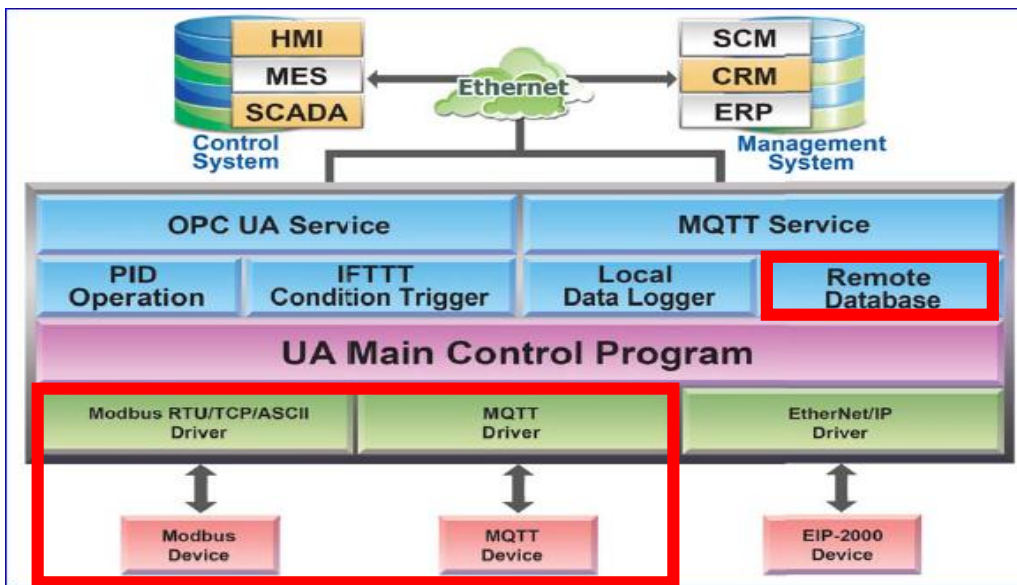
	A	B	C	D
1	# Log file created/rotated Wednesday	9 Dec 20 04:46:29 GMT		
2	Timestamp	Name	Value	Status
3	2020-12-09-12-46-29-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
4	2020-12-09-12-46-29-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1133	Good
5	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.CO2	699	Good
6	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.RH	7089	Good
7	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.TC	2225	Good
8	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.TF	7205	Good
9	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.DC	1671	Good
10	2020-12-09-12-46-29-619	MRTU_No.2_DL-302_AO.DF	6207	Good
11	2020-12-09-12-46-39-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
12	2020-12-09-12-46-39-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1152	Good
13	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.CO2	699	Good
14	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.RH	7089	Good
15	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.TC	2225	Good
16	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.TF	7205	Good
17	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.DC	1671	Good
18	2020-12-09-12-46-39-619	MRTU_No.2_DL-302_AO.DF	6207	Good
19	2020-12-09-12-46-49-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
20	2020-12-09-12-46-49-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1172	Good
21	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.CO2	699	Good
22	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.RH	7085	Good
23	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.TC	2225	Good
24	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.TF	7205	Good
25	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.DC	1670	Good
26	2020-12-09-12-46-49-619	MRTU_No.2_DL-302_AO.DF	6206	Good
27	2020-12-09-12-46-59-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
28	2020-12-09-12-46-59-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1193	Good
29	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.CO2	698	Good
30	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.RH	7089	Good
31	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.TC	2223	Good
32	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.TF	7201	Good
33	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.DC	1668	Good
34	2020-12-09-12-46-59-619	MRTU_No.2_DL-302_AO.DF	6202	Good
35	2020-12-09-12-47-09-619	MRTU_No.1_tM-AD4P2C2_AO.Vin0	2278	Good
36	2020-12-09-12-47-09-619	MRTU_No.1_tM-AD4P2C2_AO.Vin1	1213	Good

### 4.3.3 Function Wizard: Modbus / MS SQL (RTU Example, tM-AD4P2C2 + DL-302)

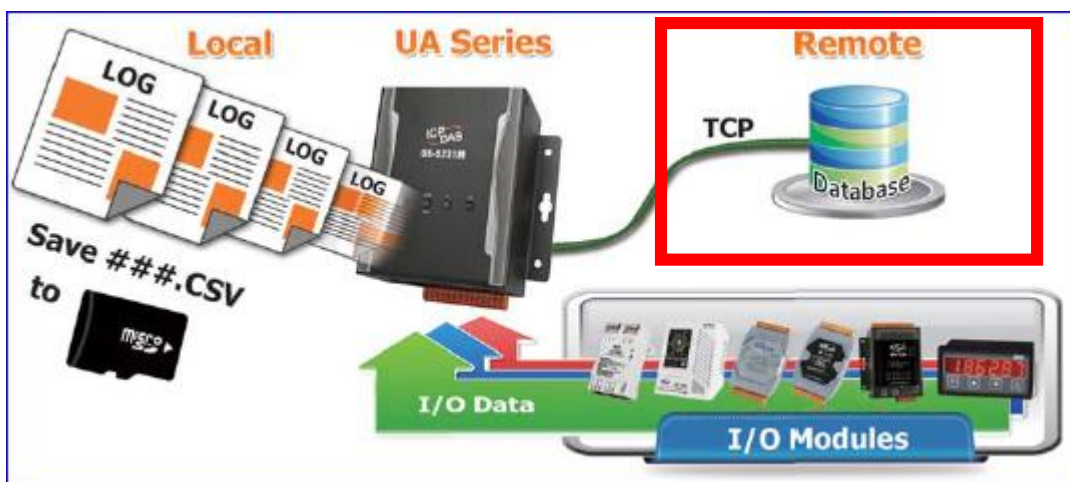
UA Data Logger supports to collect devices I/O status and then directly write into remote side MS SQL Database for the Big Data analysis.

The Modbus / Remote Database settings include Modbus RTU and TCP. Here will introduce **Modbus RTU and multiple modules** as the setting sample.

#### Modbus / Remote Database Function Diagram:

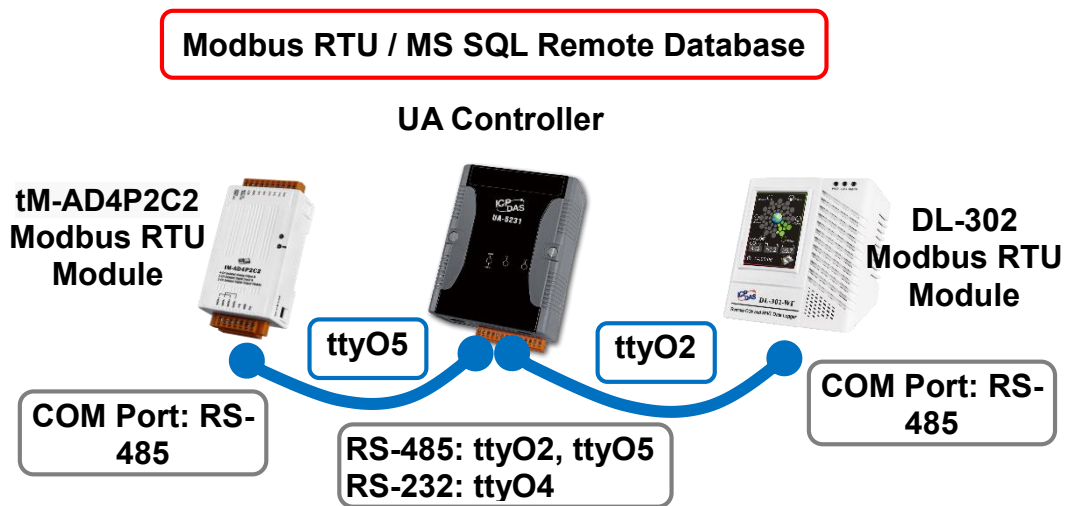


#### Application Solution:



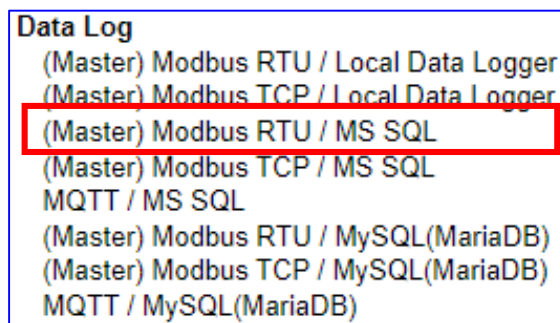


● **Modbus RTU / MS SQL: (Multi-module Example: use tM-AD4P2C2 + DL-302)**



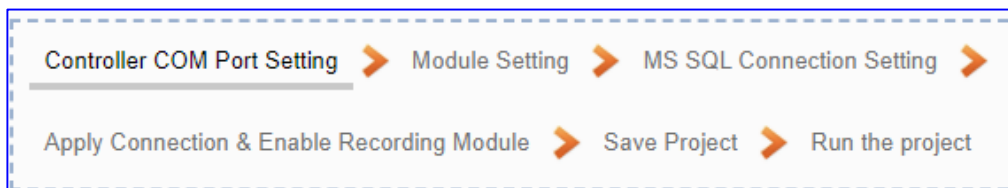
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus RTU multiple modules, e.g. **tM-AD4P2C2 (port: ttyO5)** and **DL-302 (port: ttyO2)**, as the picture. And save the I/O data into the **MS SQL remote database**, user can choose the item **[Modbus RTU / MS SQL]** of the “Data Log” in the Function Wizard.

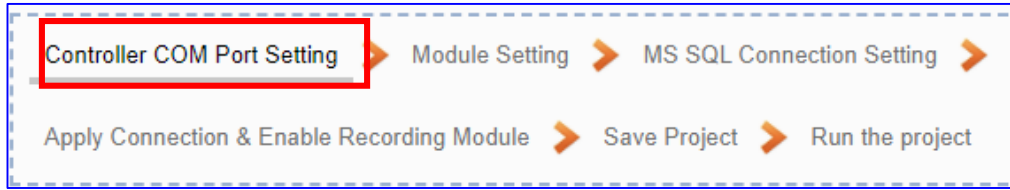


**[Step Box]:**

The Step Box of the **[Modbus RTU / MS SQL]** has the following steps. 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**

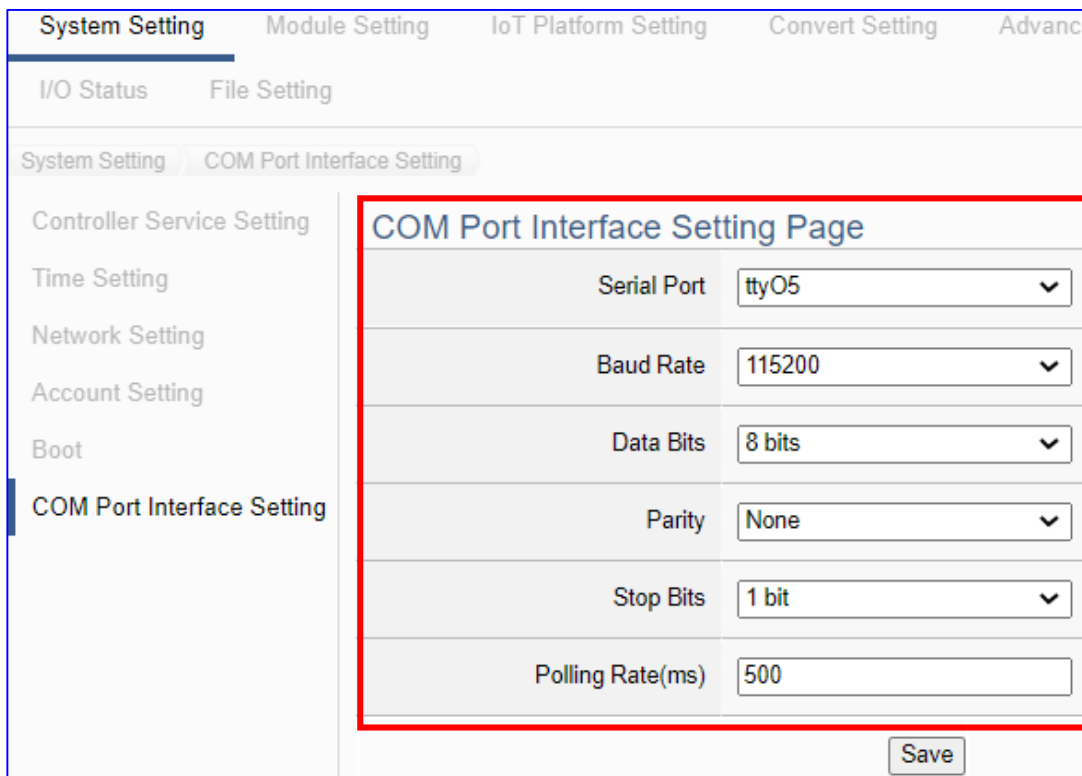


This page allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication. The user can find the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).

In this example: the communication data of module **tM-AD4P2C2** can be found in the product website <https://www.icpdas.com/en/product/tM-AD4P2C2>

COM Ports	
Ports	1 x RS-485
Baud Rate	1200 ~ 115200 bps
Data Format	(N, 8, 1), (N, 8, 2), (O, 8, 1), (E, 8, 1)
Protocol	DCON, Modbus/RTU, Modbus/ASCII

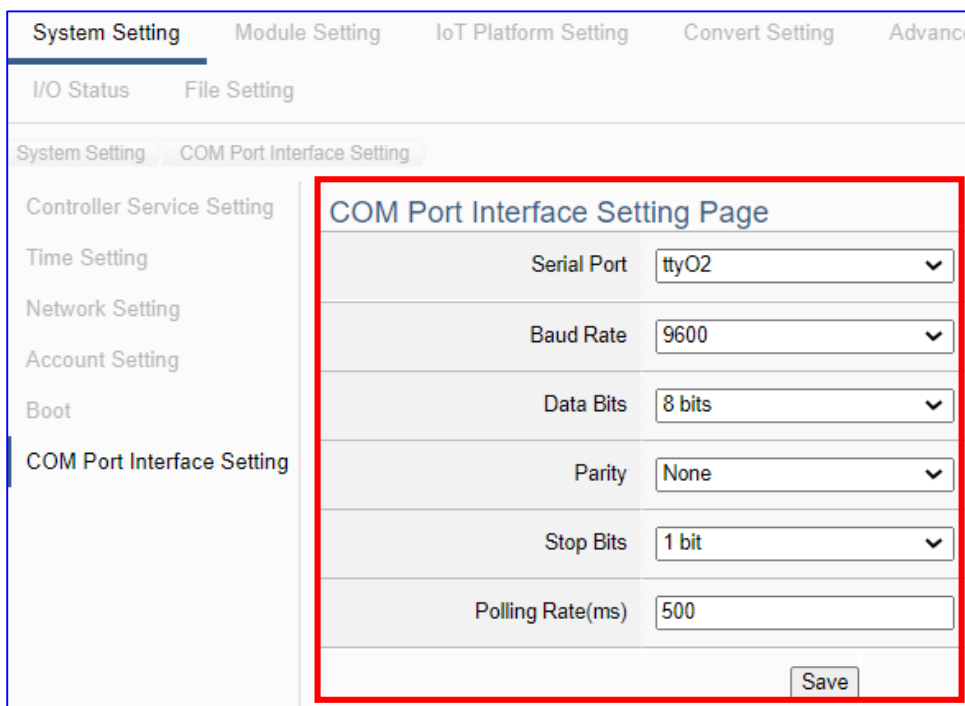
Setting as below. **Note: This example uses **ttyO5** port to link tM-AD4P2C2 module.** After settings, click “Save”.



In this example: the communication data of module **DL-302** can be found in the product user manual. <https://www.icpdas.com/en/product/DL-302> (as the following)

<b>Address:</b> Sets the address for a module.
Default: 1 Range: 0 ~ 255
<b>Protocol:</b> Sets the communication protocol.
- ModbusRTU (default) - DCON - DCONChkSum: uses DCON protocol and enables checksum validation feature
<b>Baud Rate</b>
Default: 9600 Support Baud Rate: 1200/ 2400/ 4800/ 9600/ 19200/ 38400/ 57600/ 115200 (unit: bps)
<b>Parity</b>
Default: N,8,1 Support format: N81, N82, E81, O81
<b>Response Delay (ms):</b> Sets the delay time between receiving the command and sending the data.
Default: 0 ms Range: 0 ~ 30 (unit: ms)
<b>Save:</b> Saves the modification and returns to the Settings menu. All the changes take effect immediately after saving changes.
<b>Skip:</b> Returns to the Settings menu without saving any changes.

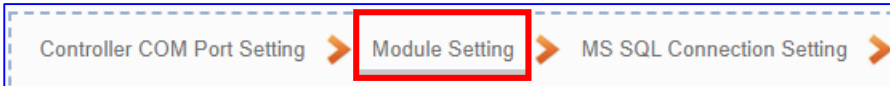
Setting as below. **Note:** This example uses **ttyO2** port to link DL-302 module. After settings, click “Save”.



The descriptions for the setting parameters:

<b>COM Port Interface Setting Page</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
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.

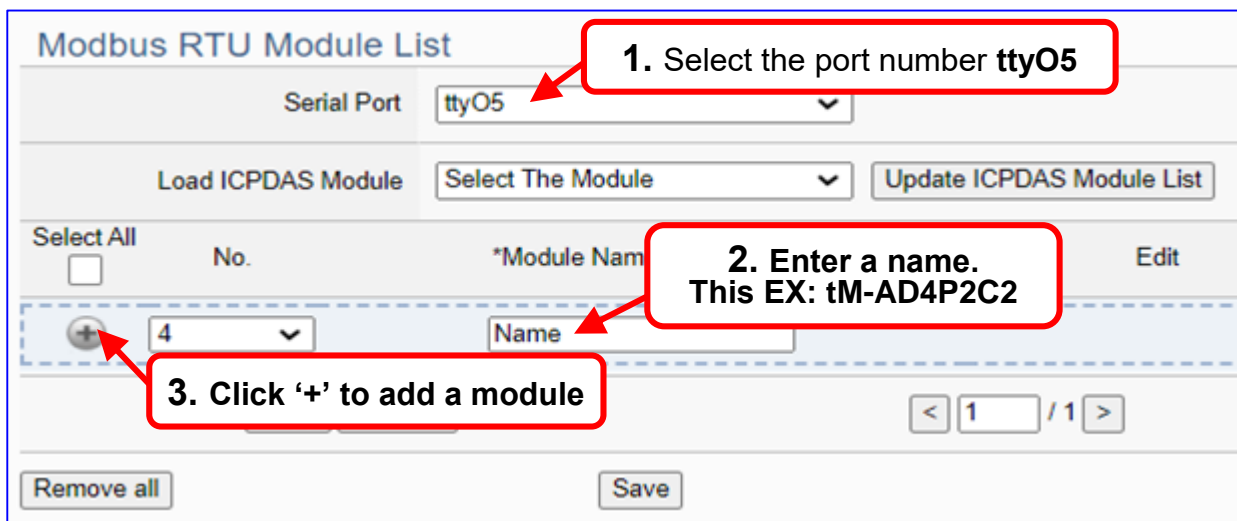
● **Step 2. Module Setting**



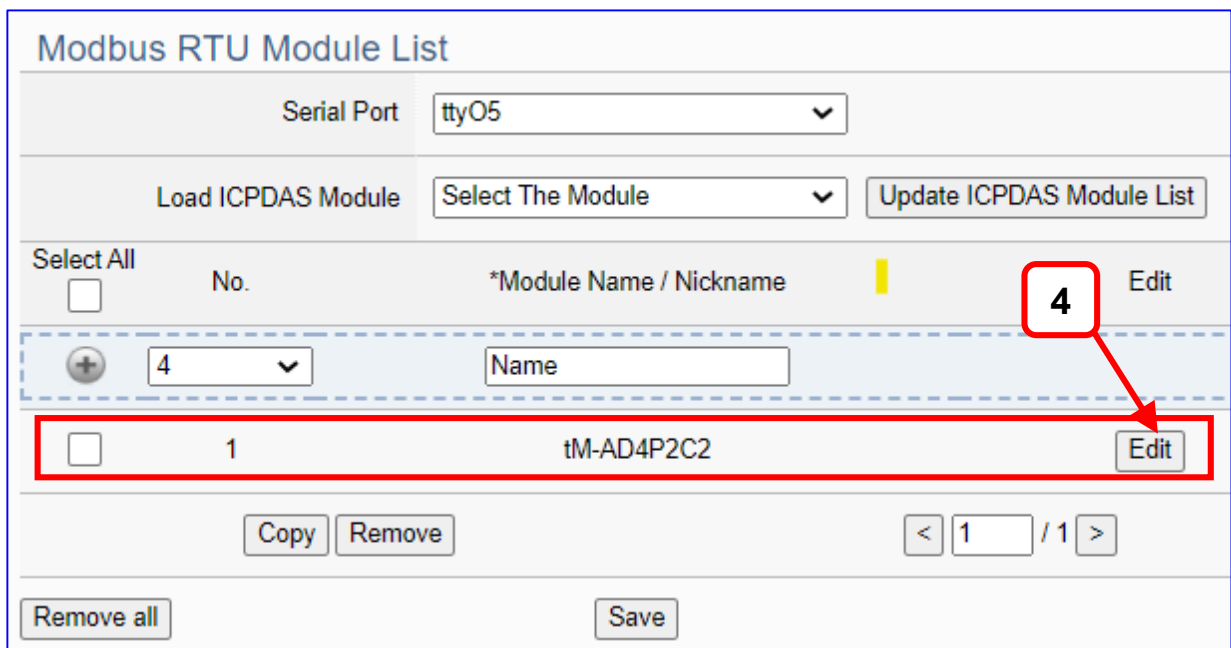
It auto-enter the first step, **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First check the 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.

**This example to set up tM-AD4P2C2 : (Port: ttyO5)**



The module (No.: 1, Name: **tM-AD4P2C2**) is as below, and then click [Edit] button to enter the “Module Content Setting” page.



[Module Content Setting] page can set the module and its Modbus mapping table.

**> Modbus Mapping Table Setting:**  
 Set module in the order of Data Model, Start Address and Data Number, then click “Add”.  
 p.s. If select from ICP DAS module list, system will auto setup the Modbus Mapping Table; if not, user can check the Modbus address or I/O number from the module manual or website.  
**[This Example] tM-AD4P2C2:**  
 Set up **AO x 2**  
**Data Model: 03** Holding Registers(4x)  
**Start Address: 0,**  
**Data Number: 2,**  
**Type: 16-bit Short,**  
 click **[Add]**, as the left picture.

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. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Write Retry	Check: If there is no response after the set time, it will retry to write again, max. 3 times.
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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

[**Module Mapping Table**] page can set up the Modbus mapping table.

If select from the ICP DAS module list, system will auto-setup the module and its Modbus Mapping Table. If not, user needs to check the module data and set up the module. Such as this example, we check the module manual and set up the tM-AD4P2C2 with 2 AO. The **Address** of the Modbus Mapping Table is as following.

**Address:**

Display and edit the Modbus Mapping Table.

Module Content Setting									
No.	<input type="text" value="1"/>								
Module Name	<input type="text" value="tM-AD4P2C2"/>								
Slave ID	<input type="text" value="2"/>								
Timeout(ms)	<input type="text" value="500"/>								
Write Retry	<input type="checkbox"/> <input type="text" value="1"/>								
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="2"/>								
Type	<input type="text" value="16-bit Short"/> ▼								
Create Tables	<input type="button" value="Add"/>								
Modbus Mapping Table									
<div style="display: flex; justify-content: space-around;"> <span>Address</span> <span>Nickname</span> <span>Scaling</span> <span>Bitwise</span> </div>									
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>Short</td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Edit"/></td> </tr> </table>		Address	0	Number	2	Type	Short	<input type="button" value="Edit"/>	
Address	0								
Number	2								
Type	Short								
<input type="button" value="Edit"/>									
<input type="button" value="OK"/> <input type="button" value="Cancel"/>									

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
<b>02 Input Status(1x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
<b>03 Holding Registers(4x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	Vin0	Short	<input type="checkbox"/>					
1	Vin1	Short	<input type="checkbox"/>					
<b>04 Input Registers(3x)</b>								
Table Display								
Address	Variable name	Data Type	Swap	Description				
<div style="border: 2px solid red; padding: 5px; display: inline-block;"> <b>Can set up the Variable Nickname, Ex: Vin0, Vin1...</b> </div>								
					OK		Cancel	

Modbus Mapping Table – Nickname	
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.



**This example: use the same way to set up DL-302 (Serial Port: ttyO2)**

**1. Select the serial port ttyO2**

**2. Enter the module name  
This Ex: DL-302**

**3. Click '+' to add**

Select All	No.	*Name	Edit
<input type="checkbox"/>		Name	

Add a module (as the picture, this example: **DL-302**), and then click the [Edit] of the module to enter the screen of "Module Content Setting".

**4**

Select All	No.	*Module Name / Nickname	Edit
<input type="checkbox"/>	4	Name	
<input type="checkbox"/>	2	DL-302	Edit

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 mapping table

Module Content Setting	
No.	<input type="text" value="2"/>
Module Name	<input type="text" value="DL-302"/>
Slave ID	<input type="text" value="1"/>
Timeout(ms)	<input type="text" value="500"/>
Write Retry	<input type="checkbox"/> <input type="text" value="1"/>
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="6"/>
Type	<input type="text" value="16-bit Short"/>
Create Tables	<input type="button" value="Add"/>

**This Example: DL-302  
Set up AO Short x 6**

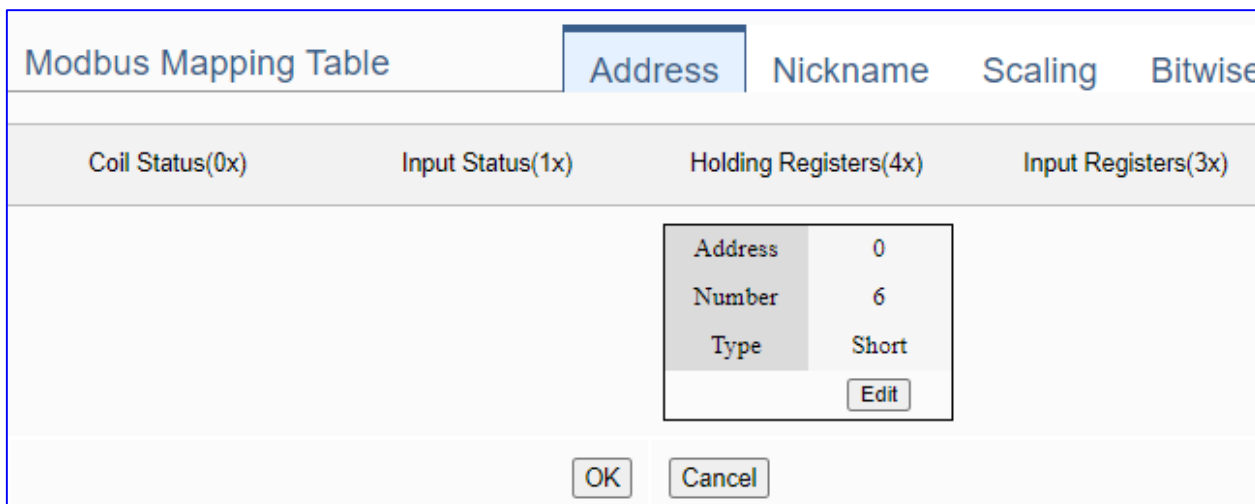
**[ Modbus Mapping Table Setting ]**  
**Data Models: 03 Holding Registers(4x)**  
**Start Address: 0**  
**Data Number : 6**  
**Type: 16-bit Short**  
**→ Click [Add]**

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. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Write Retry	Check: If there is no response after the set time, it will retry to write again, max. 3 times.
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; width: fit-content;"> <p>01 Coil Status(0x)</p> <p>02 Input Status(1x)</p> <p>03 Holding Registers(4x)</p> <p>04 Input Registers(3x)</p> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address:**

Display and edit the Modbus Mapping Table.



Modbus Mapping Table – Address	
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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

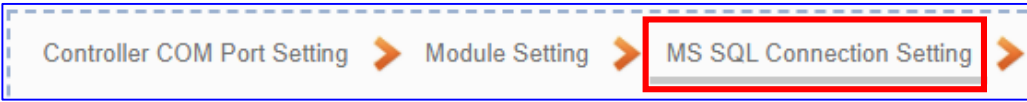
Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
01 Coil Status(0x)								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
02 Input Status(1x)								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
03 Holding Registers(4x)								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	CO2	Short	<input type="checkbox"/>					
1	RH							
2	TC							
3	TF	Short	<input type="checkbox"/>					
4	DC	Short	<input type="checkbox"/>					
5	DF	Short	<input type="checkbox"/>					
04 Input Registers(3x)								

**The user can define the nickname for the variables, as the picture.**

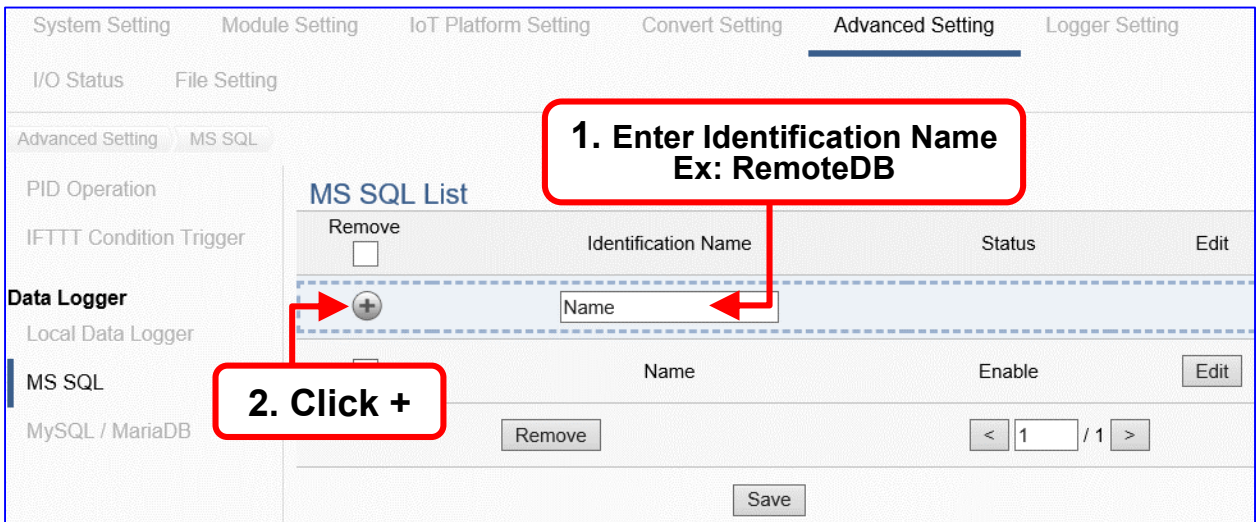
Modbus Mapping Table – Nickname	
Modbus Mapping Table	Coil Status(0x): Mapping to DO Modbus address Input Status(1x): Mapping to DI Modbus address Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address
Table Display	Click [Show] to display all fields, click [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. MS SQL Connection Setting**

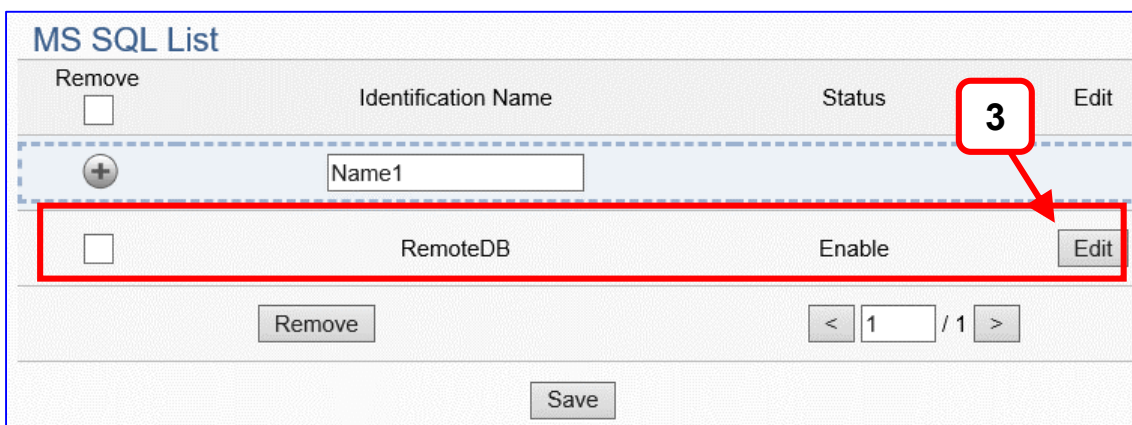


Click the next step, and enter the **Step 3 [MS SQL Connection Setting]** of the UI setting. This page is for setting the connecting remote database.

We select the “Modbus RTU / MS SQL” at the beginning, so this step will auto enter the [Advanced Setting > Data Logger > MS SQL] Setting. The “Step Box” will prevent the user from selecting the wrong platform.



Add a database identification name (Ex: RemoteDB) as below, and then click [Edit] button to enter the “MS SQL 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.

[MS SQL Content Setting] can set up the database relational setting.

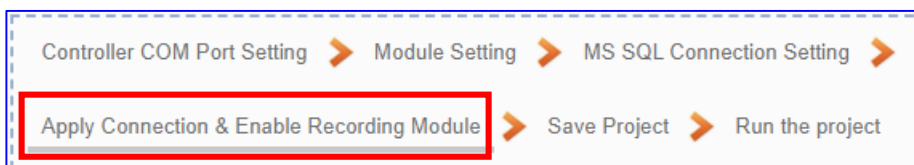
MS SQL content settings

Identification Name	ICPDAS_Cycle_DateTime
Database Name	ICPDAS
Table Name	Module_All_DateTime
Server Name	192.168.85.11\ICPDAS
Port	1433
Account	root
Password	....
Log Mode	Cycle
Interval Seconds	5
Date Time Format	[yyyy-MM-dd HH:mm:ss]
Enable	<input checked="" type="checkbox"/>
Test Connection	Connection

OK Cancel

Advanced Setting > Data Logger > MS SQL – Content Setting	
Identification Name	User defined name to identify the database.
Database Name	The name of the remote database. If not exist, It will create one.
Table Name	The table name of the remote DB. If not exist, It will create one.
Server Name	The Server IP and name of the remote database.
Port	The port to link with database. Default: 1433 (for MS SQL)
Account	The login name of the remote database.
Password	The login password of the remote database.
Log Mode	<b>Cycle:</b> Record one log data at the interval time set below. <b>Data Change:</b> Only record when the data has changed.
Interval Seconds	Set up the interval time to save the I/O data to the remote database. Unit: Second.
Date Time Format	Select to separate the date and time into two [Columns] or combine the date and time in one [Column].
Enable	Enable the data logger to the remote database. Default: check.
Test Connection	Click to test the connection to the remote database. Result: Success or Failure.
OK / Cancel	Click “OK” to save the settings of this page. Click “Cancel” to exit the setting page without saving.

**Step 4. Apply Connection & Enable Recording Module**



Click the next step, and enter the **Step 4 [Apply Connection & Enable Recording Module]** UI setting. This step is to enable the Modbus TCP module and connection.

We select the “Modbus RTU / MS SQL” of “Data Log” at the beginning, so this step will auto enter the **[Logger Setting > MS SQL > RTU Module (Master)]** setting page.

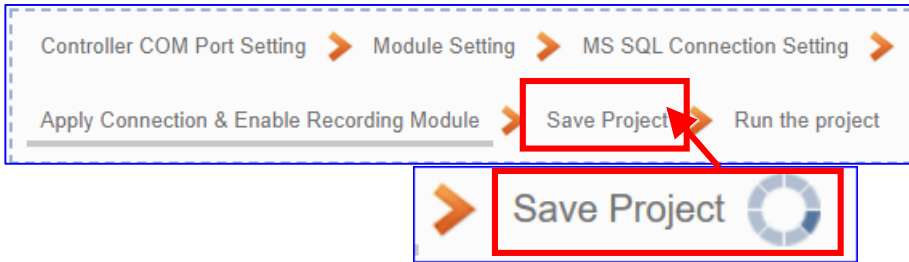
**Follow the steps: 1. Select the Database Name, 2. Click apply, and enable the Modules.**

Modbus RTU Module List				
No.	*Module Name / Nickname	Edit	Database Name	All Enabled
1	tM-AD4P2C2	<input type="button" value="Edit"/>	ICPDAS_Cycle_DateTime (Rei <input type="button" value="Apply"/>	<input checked="" type="checkbox"/>
2	DL-302	<input type="button" value="Edit"/>	ICPDAS_Cycle_DateTime (Rei <input type="button" value="Apply"/>	<input checked="" type="checkbox"/>

Logger Setting > MS SQL > RTU Module (Master) > Modbus RTU 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 data logger, click [Edit] of that module to enter the “Content Setting”. It is normal to set all channels as enabled, and the function will not affect the unconnected channels.
Database Name	Select and apply the recording remote database name.
All Enabled	Check [All Enabled] box to enable all modules in list for data logger. Default: Uncheck. Check the “box” of each module can enable just that module for data logger.
<input type="button" value=" &lt; 1 / 1 &gt;"/>	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.

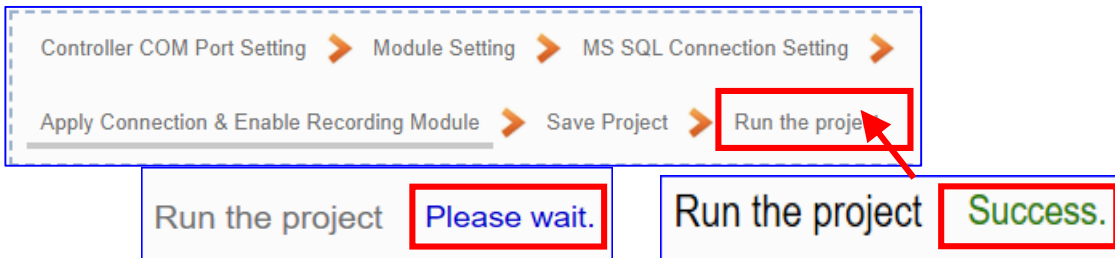
● **Step 5. Save Project**

The setting of this example is finished now. 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**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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 project communication. Users can see the I/O status from the menu [**I/O Status**]. For more LAN about the Web UI settings, please refer to CH4 and CH5.

I/O Status
File Setting

**Modbus RTU Module (Master)**

No.	Name	Serial Port
2	DL-302	ttyO2
3	M-7055D	ttyO2
1	tM-AD4P2C2	ttyO5

< 1 / 1 >

**Modbus TCP Module (Master)**

No.	Name	LAN
1	TPD-703	LAN

< 1 / 1 >

**Modbus ASCII Module (Master)**

No.	Name	Serial Port
-----	------	-------------

< 1 / 1 >

**Related Settings**

Number of variables:  (Updated 10 points per second)

Display Update Time (ms):

**I/O Status**

Variable Name	Data Type	Value	Description	Status
CO2	Short	776		Good
RH	Short	6676		Good
TC	Short	2212		Good
TF	Short	7181		Good
DC	Short	1564		Good
DF	Short	6015		Good

< 1 / 1 >



● **MS SQL Remote Database Example Descriptions:**

Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

For database operation, please refer to **FAQ-001** of the **UA series FAQ list**:

**FAQ-001** How to save the UA collected data into SQL and then show trend chart in InduSoft?  
 (Take MS SQL 2017 Express as an example)

The connection screen view of the MS SQL Remote Database.

**1. MS SQL database screen view: Date/Time column separated**

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. On the left, the Object Explorer shows the server structure for '192.168.81.5\SQLEXPRESS (SQL...)'. The main window shows a query window titled 'SQLQuery1.sql - 1...eName (root (55))' containing the following SQL query:

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Date]
, [Time]
, [Name]
, [Attribute]
, [DataType]
, [Value]
, [Status]
FROM [DatabaseName].[dbo].[TableName]
    
```

Below the query, the 'Results' pane shows the following data:

	Date	Time	Name	Attribute	DataType	Value	Status
1	2020-10-29	17:55:54	MRTU_No.1_IN-AD4P2C2_AO.VinU	Read & Write	Short	11979	GOOD
2	2020-10-29	17:55:54	MRTU_No.1_IN-AD4P2C2_AO.Vin1	Read & Write	Short	6155	GOOD
3	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.CO2	Read & Write	Short	694	GOOD
4	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.RH	Read & Write	Short	6351	GOOD
5	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.TC	Read & Write	Short	2650	GOOD
6	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.TF	Read & Write	Short	7970	GOOD
7	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.DC	Read & Write	Short	1901	GOOD
8	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.DF	Read & Write	Short	6621	GOOD
9	2020-10-29	17:55:59	MRTU_No.1_IN-AD4P2C2_AO.Vin0	Read & Write	Short	11980	GOOD
10	2020-10-29	17:55:59	MRTU_No.1_IN-AD4P2C2_AO.Vin1	Read & Write	Short	6002	GOOD
11	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.CO2	Read & Write	Short	693	GOOD
12	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.RH	Read & Write	Short	6353	GOOD
13	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.TC	Read & Write	Short	2650	GOOD
14	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.TF	Read & Write	Short	7970	GOOD
15	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.DC	Read & Write	Short	1901	GOOD
16	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.DF	Read & Write	Short	6621	GOOD

## 2. MS SQL database screen view: Date/Time column combined

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. On the left, the Object Explorer shows the server structure for '192.168.81.5\SQLEXPRESS (SQL S...)', including Databases, System Databases, Database Snapshots, DatabaseName, Database Diagrams, Tables, Views, External Resources, Synonyms, Programmability, Service Broker, Storage, Security, Server Objects, Replication, and PolyBase.

The main window shows a SQL query in the 'SQLQuery2.sql' window:

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [DateTime]
, [Name]
, [Attribute]
, [DataType]
, [Value]
, [Status]
FROM [ICPDAS].[dbo].[Module_All_DateTime]
    
```

Below the query editor, the 'Results' pane shows the output of the query as a table with 16 rows and 6 columns: Date/Time, Name, Attribute, DateTime, Value, and Status. The data is as follows:

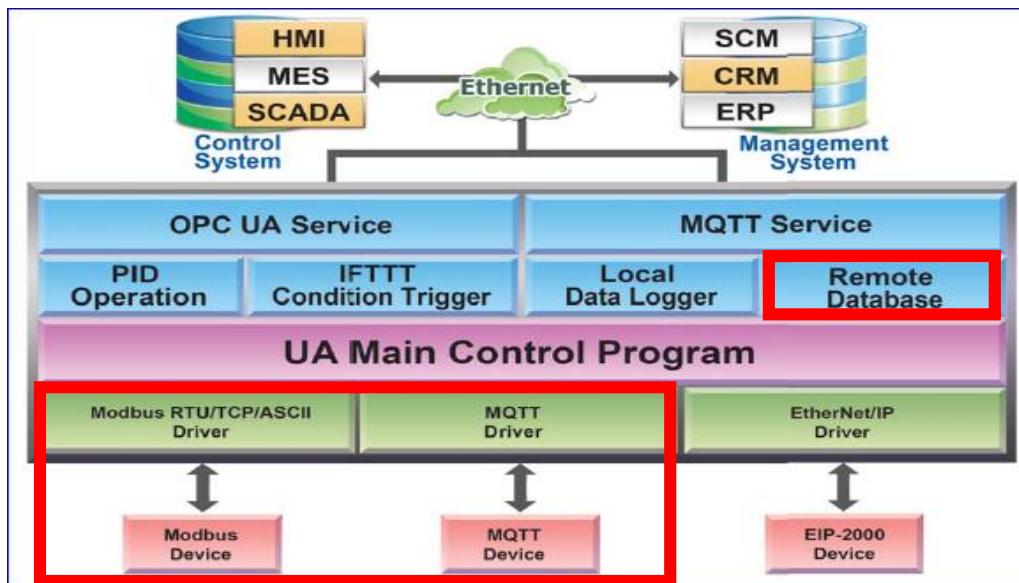
	Date/Time	Name	Attribute	DateTime	Value	Status
1	2020-10-29 10:12:21	MR.TU_No.1_0M-AD4F2C2_AO.Vin0	Read & Write	Shoot	11979	GOOD
2	2020-10-29 10:12:21	MR.TU_No.1_0M-AD4F2C2_AO.Vin1	Read & Write	Shoot	5042	GOOD
3	2020-10-29 10:12:21	MR.TU_No.2_DL-302_AO.CO0	Read & Write	Shoot	662	GOOD
4	2020-10-29 10:12:21	MR.TU_No.2_DL-302_AO.RH	Read & Write	Shoot	6627	GOOD
5	2020-10-29 10:12:21	MR.TU_No.2_DL-302_AO.TC	Read & Write	Shoot	2659	GOOD
6	2020-10-29 10:12:21	MR.TU_No.2_DL-302_AO.TF	Read & Write	Shoot	7996	GOOD
7	2020-10-29 10:12:21	MR.TU_No.2_DL-302_AO.DC	Read & Write	Shoot	1977	GOOD
8	2020-10-29 10:12:21	MR.TU_No.2_DL-302_AO.DF	Read & Write	Shoot	6750	GOOD
9	2020-10-29 10:12:26	MR.TU_No.1_0M-AD4F2C2_AO.Vin0	Read & Write	Shoot	11979	GOOD
10	2020-10-29 10:12:26	MR.TU_No.1_0M-AD4F2C2_AO.Vin1	Read & Write	Shoot	5169	GOOD
11	2020-10-29 10:12:26	MR.TU_No.2_DL-302_AO.CO0	Read & Write	Shoot	662	GOOD
12	2020-10-29 10:12:26	MR.TU_No.2_DL-302_AO.RH	Read & Write	Shoot	6627	GOOD
13	2020-10-29 10:12:26	MR.TU_No.2_DL-302_AO.TC	Read & Write	Shoot	2650	GOOD
14	2020-10-29 10:12:26	MR.TU_No.2_DL-302_AO.TF	Read & Write	Shoot	7994	GOOD
15	2020-10-29 10:12:26	MR.TU_No.2_DL-302_AO.DC	Read & Write	Shoot	1977	GOOD
16	2020-10-29 10:12:26	MR.TU_No.2_DL-302_AO.DF	Read & Write	Shoot	6750	GOOD

### 4.3.4 Function Wizard: Modbus / MS SQL (TCP Example, Use DL-302)

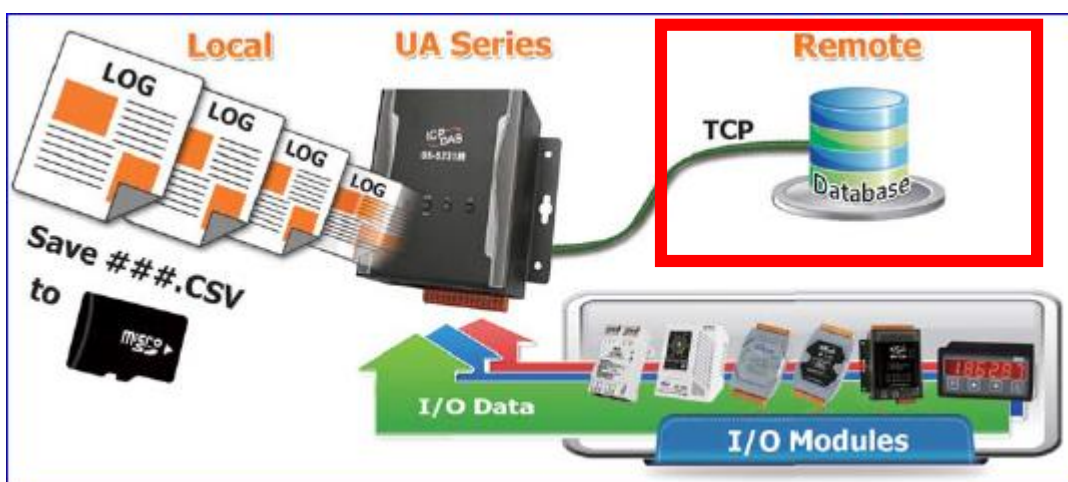
UA Data Logger supports to collect devices I/O status and then directly write into remote side MS SQL Database for the Big Data analysis.

The Modbus / Remote Database settings include Modbus RTU / TCP and MQTT devices. Here will introduce Modbus TCP as the setting sample.

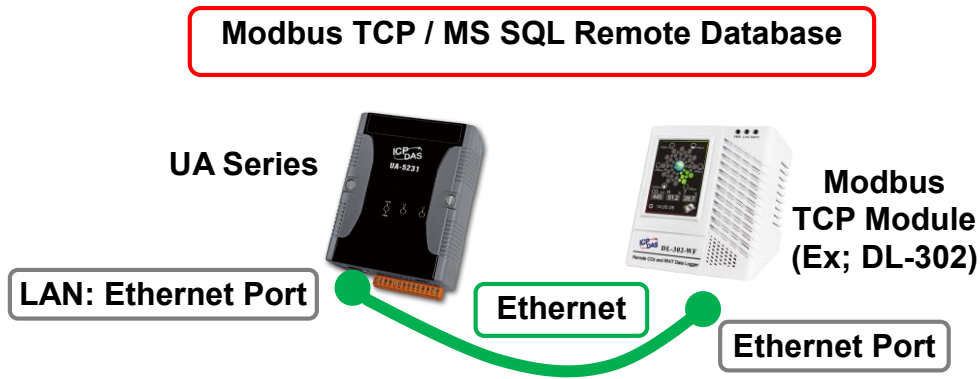
#### Modbus / Remote Database Function Diagram:



#### Application Solution:

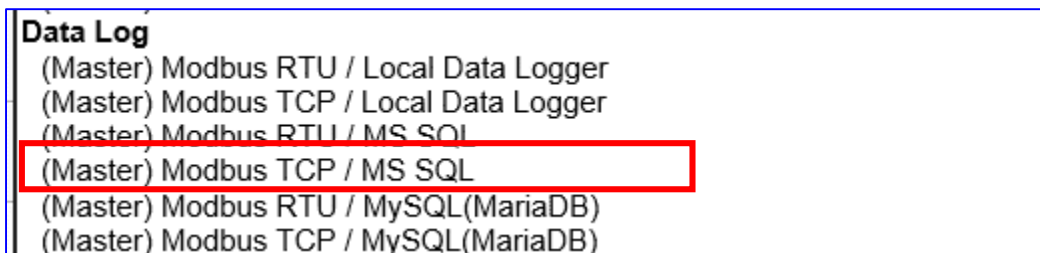


● **Modbus TCP / MS SQL**



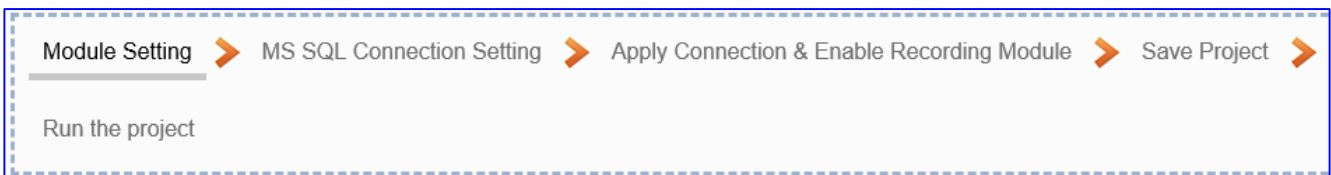
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus TCP module (via Ethernet, as the picture), user can choose the item [**Modbus TCP / MS SQL**] of the “Data Log” in the Function Wizard.

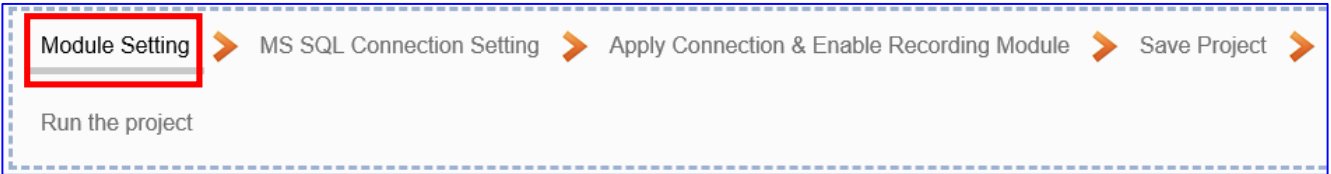


**[Step Box]:**

The Step Box of the [**Modbus TCP / MS SQL**] has 5 steps. 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.



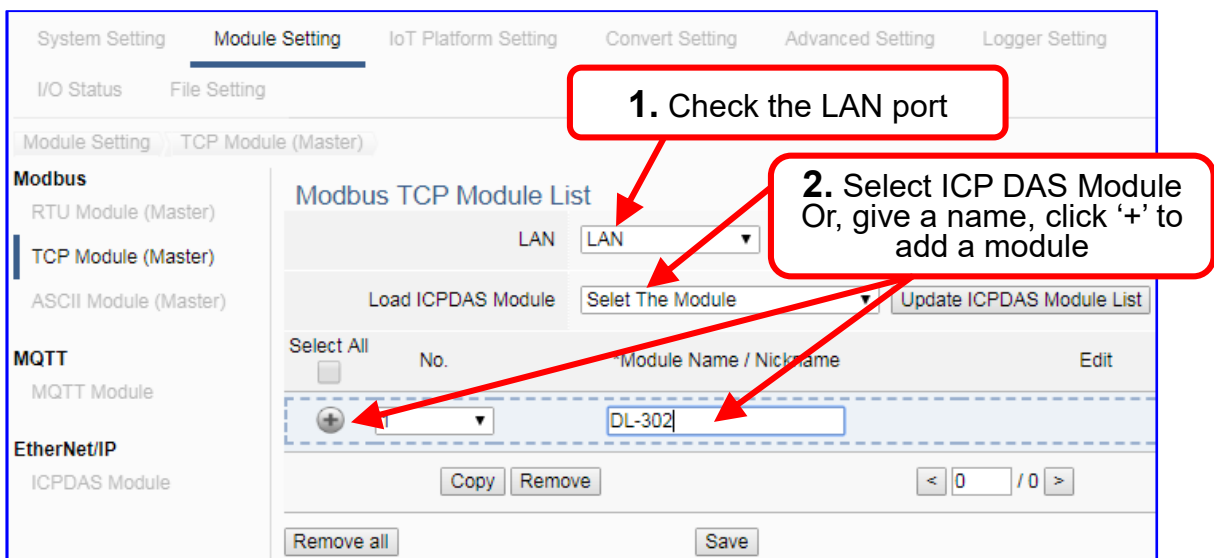
● **Step 1. Module Setting**



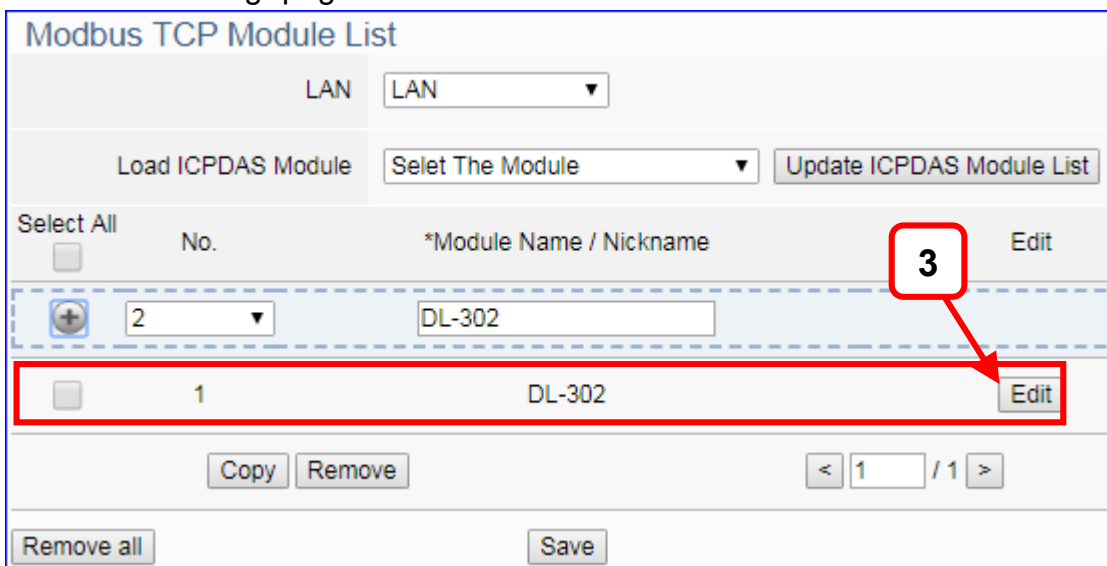
It auto-enter the first step, **Step 1 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First check the 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: DL-302) 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 mapping table:

Module Content Setting	
No.	1
Module Name	DL-302
IP	192 . 168 . 81 . 251
Port	502
Slave ID	1
Timeout(ms)	500
Polling Rate(ms)	500
Modbus Mapping Table Setting	
Data Model	04 Input Registers(3x)
Start Address	0
Data Number	3
Type	16-bit Short
Create Tables	<input type="button" value="Add"/> Success.

**Example: DL-302**

**[IP] 192.168.81.251 (by user case)**

**[ Modbus Mapping Table Setting ]**

**Data Model: 04 Input Registers(3x)**

**Start Address: 0**

**Data Number: 6**

**Type: 16-bit Short**

**→ Click [ 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. (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)
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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 mapping DO, DI, AO & AI.

**Address:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise
Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)		
				Address	0
				Number	6
				Type	Short
				<input type="button" value="Edit"/>	

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>02 Input Status(1x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>03 Holding Registers(4x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
<b>04 Input Registers(3x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	<input type="text" value="CO2"/>	Short	<input type="checkbox"/>	<input type="text"/>				
1	<input type="text" value="Relative_humidity"/>	Short	<input type="checkbox"/>	<input type="text"/>				
2	<input type="text" value="Temperature_Celsius"/>	Short	<input type="checkbox"/>	<input type="text"/>				
3	<input type="text" value="Temperature_Fahrenheit"/>	Short	<input type="checkbox"/>	<input type="text"/>				
4	<input type="text" value="Dew_point_temperature_"/>	Short	<input type="checkbox"/>	<input type="text"/>				

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.



**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the "**Advanced Setting**" button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check "**Enable**" box, The Scaling conversion function will be activated.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

**Bitwise:**

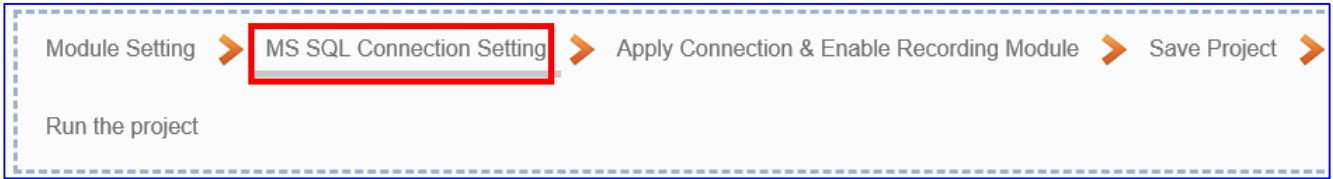
**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

The M-7055D has no AI/AO, so here uses other module's setting screen as an example.

Modbus Mapping Table	Address	Nickname	Scaling	Bitwise
<b>03 Holding Registers(4x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference	Bitwise		
<b>04 Input Registers(3x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference	Bitwise		
0	CO2	<input type="button" value="Hide Detail"/>		
	Bit0	aa	Bit1	<input type="text"/>
	Bit2	bb	Bit3	<input type="text"/>
	Bit4	<input type="text"/>	Bit5	<input type="text"/>
	Bit6	<input type="text"/>	Bit7	<input type="text"/>
	Bit8	<input type="text"/>	Bit9	<input type="text"/>
	Bit10	<input type="text"/>	Bit11	<input type="text"/>
	Bit12	<input type="text"/>	Bit13	<input type="text"/>
	Bit14	<input type="text"/>	Bit15	<input type="text"/>

Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

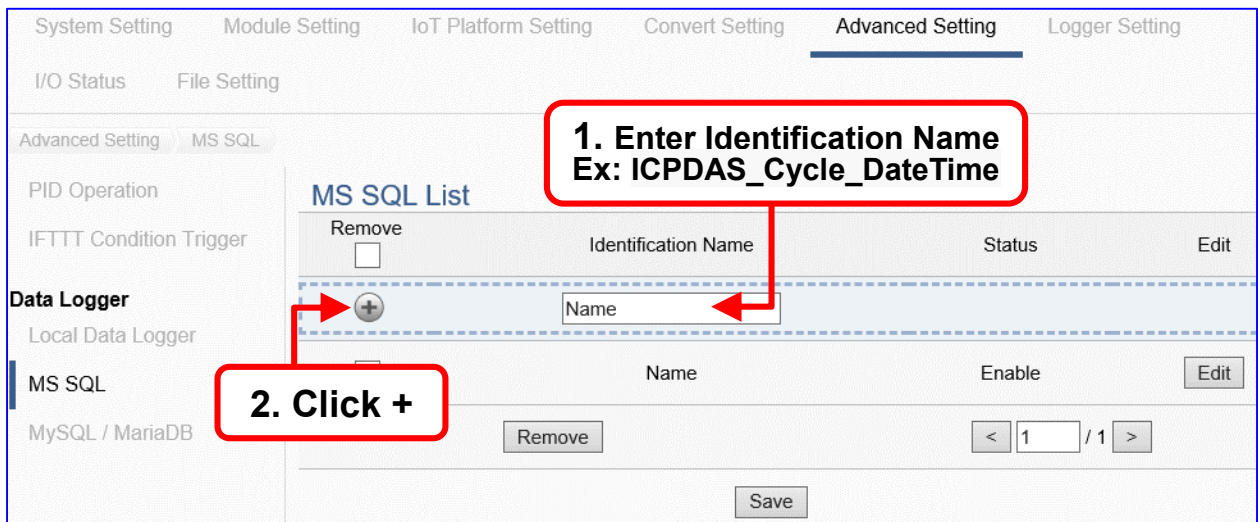
● **Step 2. MS SQL Connection Setting**



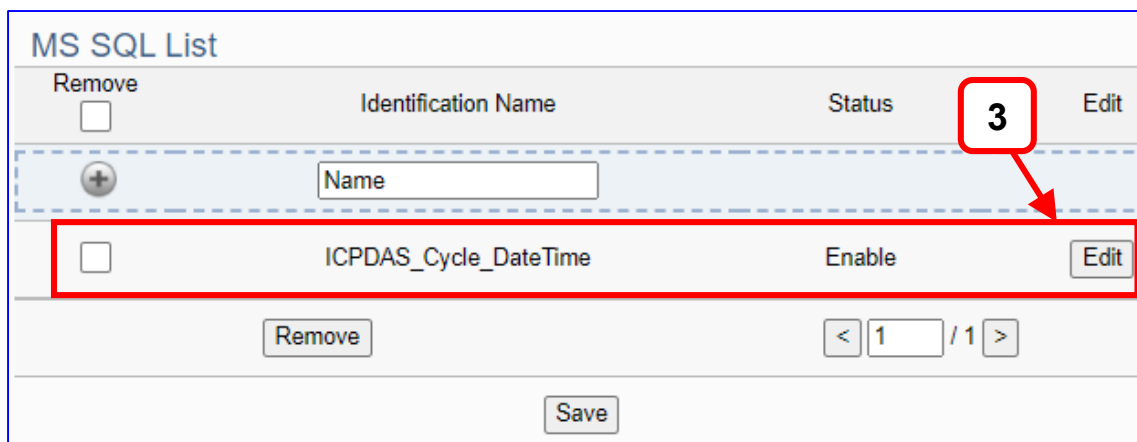
Click the next step, and enter the **Step 2 [MS SQL Connection Setting]** of the UI setting.

This page is for setting the connecting remote database.

We select the “Modbus TCP / Remote Database” at the beginning, so this step will auto enter the [**Advanced Setting > Data Logger > MS SQL**] Setting. The “Step Box” will prevent the user from selecting the wrong platform.



Add a database identification name (Ex: **ICPDAS\_Cycle\_DateTime**) as below, and then click [**Edit**] button to enter the “MS SQL Content Setting” page.



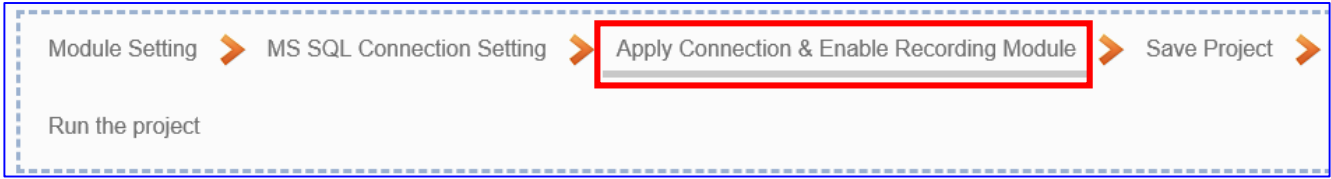
If set up a wrong module, user can click the box in the left side of the module number and click the [**Remove**] button to delete the module.

**[MS SQL Content Setting]** can set up the database relational setting.

MS SQL content settings	
Identification Name	ICPDAS_Cycle_DateTime
Database Name	ICPDAS
Table Name	Module_All_DateTime
Server Name	192.168.85.11\ICPDAS
Port	1433
Account	root
Password	....
Log Mode	Cycle
Interval Seconds	5
Date Time Format	[yyyy-MM-dd HH:mm:ss]
Enable	<input checked="" type="checkbox"/>
Test Connection	Connection
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

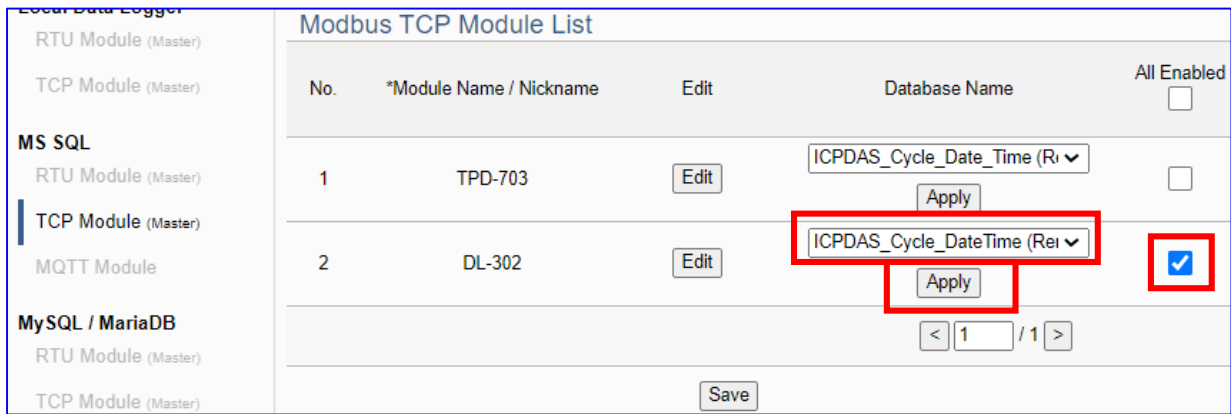
Advanced Setting > Data Logger > MS SQL – Content Setting	
Identification Name	User defined name to identify the database.
Database Name	The name of the remote database. If not exist, It will create one.
Table Name	The table name of the remote DB. If not exist, It will create one.
Server Name	The Server IP and name of the remote database.
Port	The port to link with database. Default: 1433 (for MS SQL)
Account	The login name of the remote database.
Password	The login password of the remote database.
Log Mode	<b>Cycle:</b> Record one log data at the interval time set below. <b>Data Change:</b> Only record when the data has changed.
Interval Seconds	Set up the interval time to save the I/O data to the remote database. Unit: Second.
Date Time Format	Select to separate the date and time into two [Columns] or combine the date and time in one [Column].
Enable	Enable the data logger to the remote database. Default: check.
Test Connection	Click to test the connection to the remote database. Result: Success or Failure.
OK / Cancel	Click “OK” to save the settings of this page. Click “Cancel” to exit the setting page without saving.

● **Step 3. Apply Connection & Enable Recording Module**



Click the next step, and enter the **Step 3 [Apply Connection & Enable Recording Module]** UI setting. This step is to enable the Modbus TCP module and connection. We select the “Modbus TCP /MS SQL” of “Data Log” at the beginning, so this step will auto enter the [**Logger Setting > MS SQL > TCP Module (Master)**] setting page. The “Step Box” will prevent the user from selecting the wrong platform.

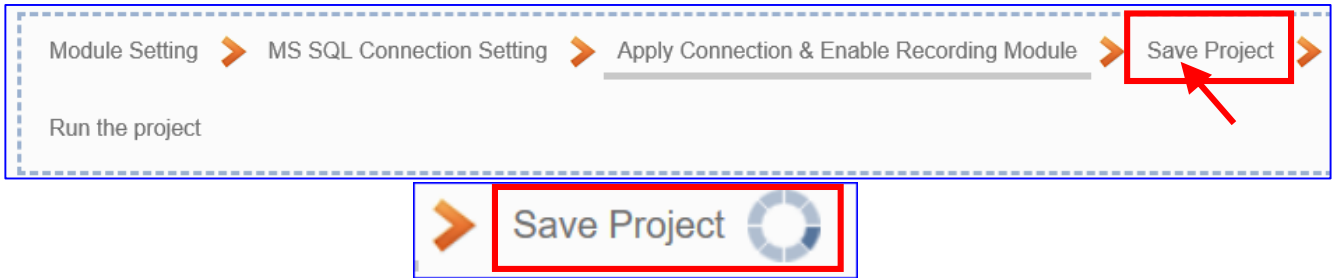
Here **select** and **apply** the Database name (Ex: **ICPDAS\_Cycle\_DateTime**), and **enable** the DL-302.



Logger Setting > MS SQL > TCP Module (Master) – Modbus TCP 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 data logger, click [Edit] of that module to enter the “Content Setting”. It is normal to set all channels as enabled, and the function will not affect the unconnected channels.
Database Name	Select and apply the recording remote database name.
All Enabled <input type="checkbox"/>	Check [All Enabled] box to enable all modules in list for data logger. Default: Uncheck. Check the “box” of each module can enable just that module for data logger.
<input type="button" value="1"/> / 1	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.

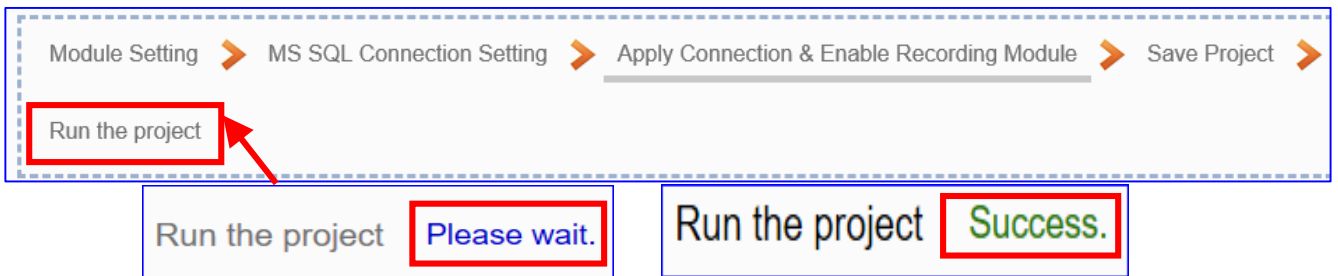
● **Step 4. Save Project**

The setting of this example is finished now. 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.



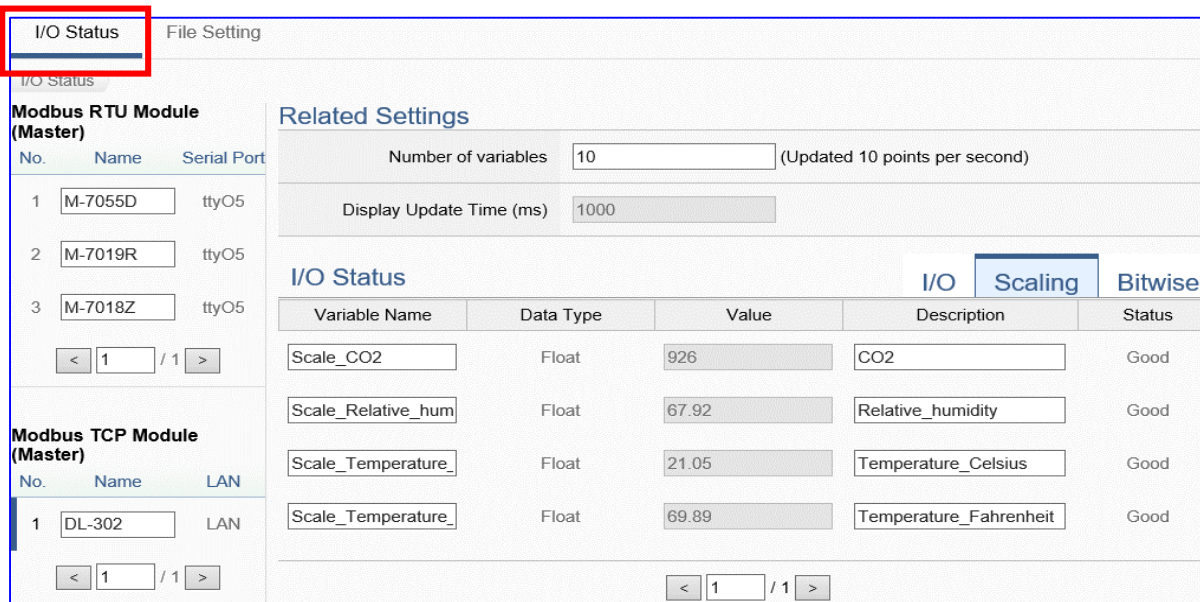
● **Step 5. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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 project communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.



● **MS SQL Remote Database Example Descriptions:**

Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

For database operation, please refer to **FAQ-001** of the **UA series FAQ list**:

**FAQ-001** How to save the UA collected data into SQL and then show trend chart in InduSoft?  
 (Take MS SQL 2017 Express as an example)

The connection screen view of the MS SQL Remote Database.

**1. MS SQL database screen view: Date/Time column separated (Reference)**

The screenshot displays the Microsoft SQL Server Management Studio interface. The central pane shows the results of a query executed on a remote database. The query is a SELECT TOP (1000) statement with columns [Date], [Time], [Name], [Attribute], [DbType], [Value], and [Status]. The results table contains 10 rows of data, showing timestamps from 2019/11/20 12:32:45 to 12:33:00 and various sensor names like 'MTCP\_No.1\_DL-302\_AI.Humidity' and 'MTCP\_No.1\_DL-302\_AI.CO2'. The status for all rows is 'GOOD'. The Properties window on the right shows connection parameters for '192.168.81.5\SQLEXPRESS', including connection name, state (Open), and login name (root).

Date	Time	Name	Attribute	DbType	Value	Status
2019/11/20	12:32:45	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20	12:32:45	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD
2019/11/20	12:32:50	MTCP_No.1_DL-302_AI.CO2	Read	Short	0	GOOD
2019/11/20	12:32:50	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20	12:32:50	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD
2019/11/20	12:32:55	MTCP_No.1_DL-302_AI.CO2	Read	Short	0	GOOD
2019/11/20	12:32:55	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20	12:32:55	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD
2019/11/20	12:33:00	MTCP_No.1_DL-302_AI.CO2	Read	Short	0	GOOD
2019/11/20	12:33:00	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20	12:33:00	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD

## 2. MS SQL database screen view: Date/Time column combined (Reference)

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The central pane shows the execution results of a query. The query text is as follows:

```

/***** Script for SelectTopRows command from SSMS *****/
SELECT TOP (1000) [Date]
, [Time]
, [Name]
, [Attribute]
, [DataType]
, [Value]
, [Status]
FROM [DatabaseName].[dbo].[TableName]
    
```

The results pane shows a table with the following columns: Date Time, Name, Attribute, DataType, Value, and Status. The data rows are as follows:

Date Time	Name	Attribute	DataType	Value	Status
2019/11/20 12:32:45	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20 12:32:45	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD
2019/11/20 12:32:50	MTCP_No.1_DL-302_AI.CO2	Read	Short	0	GOOD
2019/11/20 12:32:50	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20 12:32:50	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD
2019/11/20 12:32:55	MTCP_No.1_DL-302_AI.CO2	Read	Short	0	GOOD
2019/11/20 12:32:55	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20 12:32:55	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD
2019/11/20 12:33:00	MTCP_No.1_DL-302_AI.CO2	Read	Short	0	GOOD
2019/11/20 12:33:00	MTCP_No.1_DL-302_AI.Humidity	Read	Short	0	GOOD
2019/11/20 12:33:00	MTCP_No.1_DL-302_AI.Temperature	Read	Short	0	GOOD

The status bar at the bottom indicates: Query executed successfully. 192.168.81.5\SQLEXPRESS (14... | root (55) | DatabaseName | 00:00:00 | 1000 rows

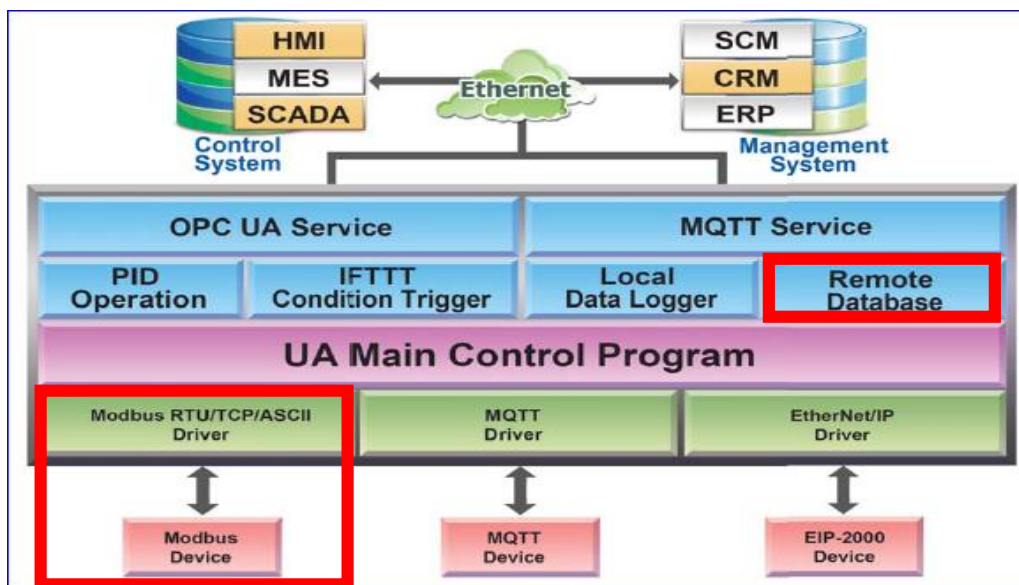


### 4.3.5 Function Wizard: Modbus / MySQL(MariaDB) (RTU Example, use M-7026)

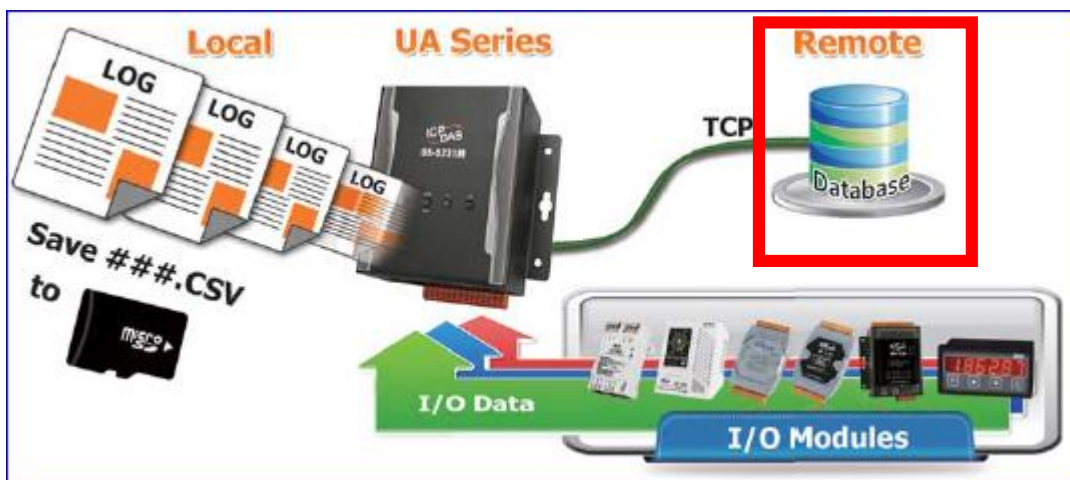
UA Data Logger supports to collect devices I/O status and then directly write into remote side MySQL/MariaDB Database for the Big Data analysis.

The connection steps for MySQL and MariaDB is the same, so here will introduce them together. The Modbus / MySQL and MariaDB Remote Database settings include Modbus RTU and TCP. Here will introduce Modbus RTU as the setting sample.

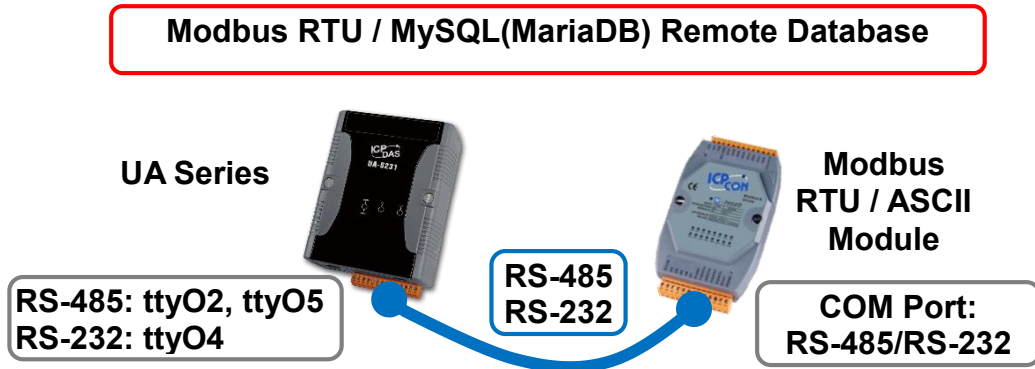
#### Modbus / Remote Database Function Diagram:



#### Application Solution:

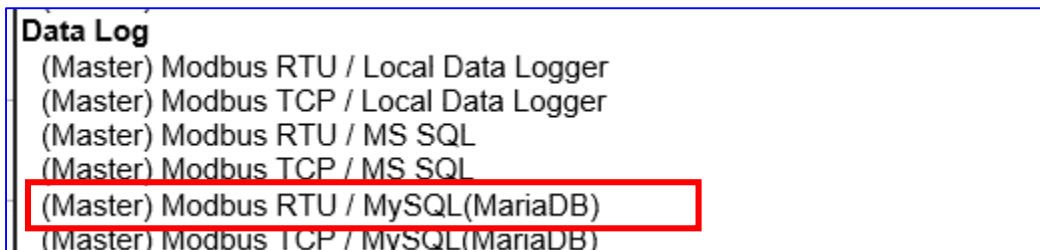


● **Modbus RTU / MySQL(MariaDB)**



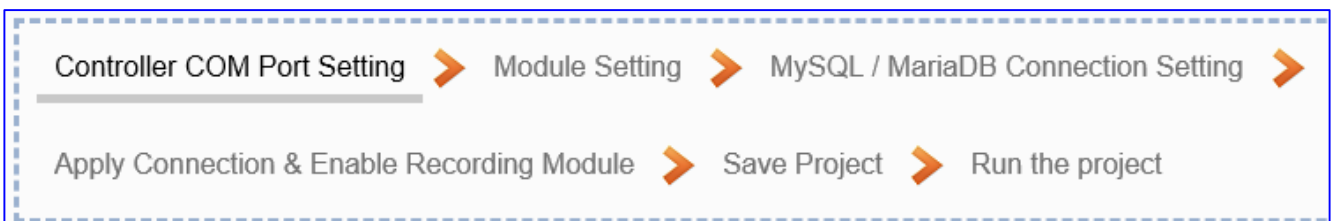
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus RTU module (via RS-485/232, as the picture), user can choose the item [**Modbus RTU / MySQL(MariaDB)**] of the “Data Log” in the Function Wizard.



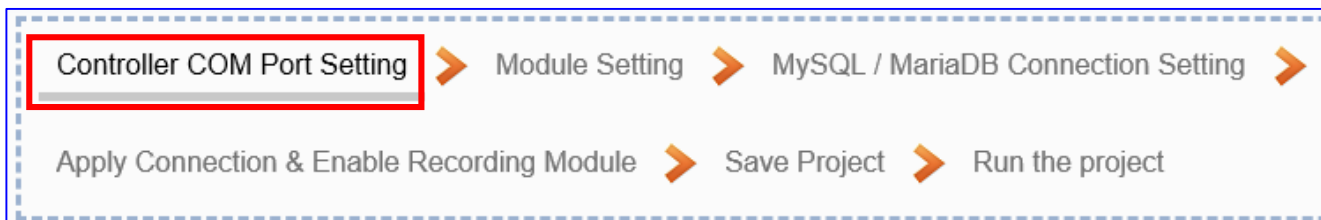
**[Step Box]:**

The Step Box of the [**Modbus RTU / MySQL(MariaDB)**] has 6 steps. 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.

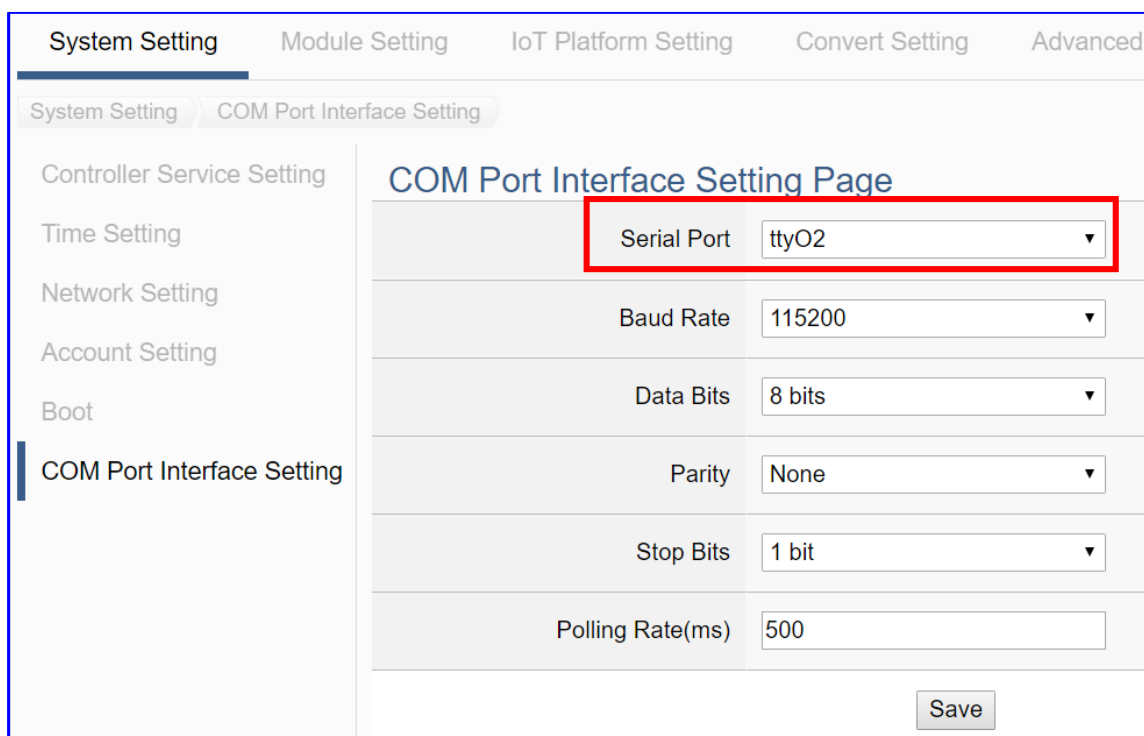


This example: UA-5231M-4GE via ttyO2 port to connect the device M-7026.

● **Step 1. Controller COM Port Setting**

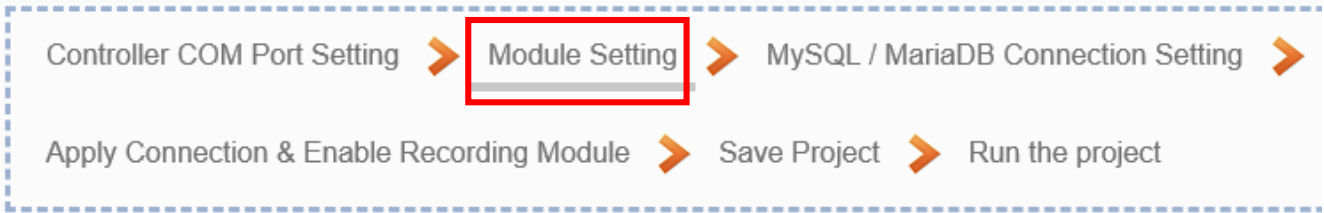


This page allows display and set the COM port interface of the controller for the RS-232/RS-485 serial communication. The user can find the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).



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**



It auto-enter the first step, **Step 2 [Module Setting]** of the UI setting.

This page is for setting the communication values with the connected modules. First check the 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.

The module (No.: 1, Name: M-7026) is 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 mapping table:

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. (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; width: fit-content;"> <p>01 Coil Status(0x)</p> <p>02 Input Status(1x)</p> <p>03 Holding Registers(4x)</p> <p>04 Input Registers(3x)</p> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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 mapping DO, DI, AO & AI.

**Address:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table				Address	Nickname	Scaling	Bitwise
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)		Input Registers(3x)	
Address	0	Address	32	Address	32	Address	0
Number	3	Number	3	Number	2	Number	6
Type	Bool	Type	Bool	Type	Short	Type	Short
<input type="button" value="Edit"/>		<input type="button" value="Edit"/>		<input type="button" value="Edit"/>		<input type="button" value="Edit"/>	
<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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
0	DO0	Bool						
1	DO1	Bool						
2	DO2	Bool						
<b>02 Input Status(1x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Description					
32	DI32	Bool						
33	DI33	Bool						
34	DI34	Bool						
<b>03 Holding Registers(4x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
32	AO32	Short	<input type="checkbox"/>					
33	AO33	Short	<input type="checkbox"/>					
<b>04 Input Registers(3x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	AI0	Short	<input type="checkbox"/>					
1	AI1	Short	<input type="checkbox"/>					

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.

**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the "**Advanced Setting**" button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check "**Enable**" box, The Scaling conversion function will be activated.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise
<b>03 Holding Registers(4x)</b>					
Table Display		<input type="button" value="Show"/>	<input type="button" value="Hide"/>		
Address	Reference	Output	Scaling	Enable	Description
32	<input type="text" value="AO32"/>	<input type="text" value="Scale_AO32"/>	<input type="button" value="Show Detail"/>	<input type="checkbox"/>	<input type="text"/>
33	<input type="text" value="AO33"/>	<input type="text" value="Scale_AO33"/>	<input type="button" value="Show Detail"/>	<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	Reference	Output	Scaling	Enable	Description
0	<input type="text" value="AI0"/>	<input type="text" value="Scale_AI0"/>	<input type="button" value="Show Detail"/>	<input type="checkbox"/>	<input type="text"/>

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.



**Bitwise:**

**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

The M-7055D has no AI/AO, so here uses other module's setting screen as an example.

Modbus Mapping Table	Address	Nickname	Scaling	Bitwise
----------------------	---------	----------	---------	---------

03 Holding Registers(4x)

Table Display			<input type="button" value="Show"/>	<input type="button" value="Hide"/>
Address	Reference	Bitwise		
32	AO32	<input type="button" value="Show Detail"/>		
	AO33	<input type="button" value="Hide Detail"/>		
33	<div style="display: flex; flex-direction: column; gap: 5px;"> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit0</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit2</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit4</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit6</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit8</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit10</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit12</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit14</span> <input style="width: 60px;" type="text"/></div> </div>	<div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit1</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit3</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit5</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit7</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit9</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit11</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit13</span> <input style="width: 60px;" type="text"/></div> <div style="display: flex; align-items: center;"><span style="background-color: #90EE90; border-radius: 5px; padding: 2px;">Bit15</span> <input style="width: 60px;" type="text"/></div>		

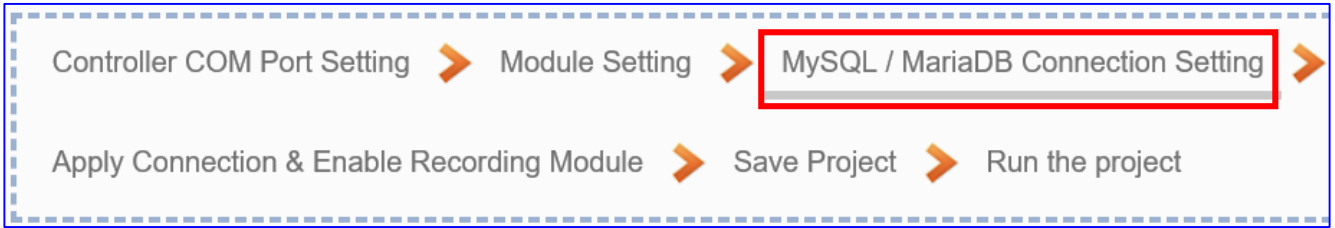
  

04 Input Registers(3x)

Table Display			<input type="button" value="Show"/>	<input type="button" value="Hide"/>
Address	Reference	Bitwise		

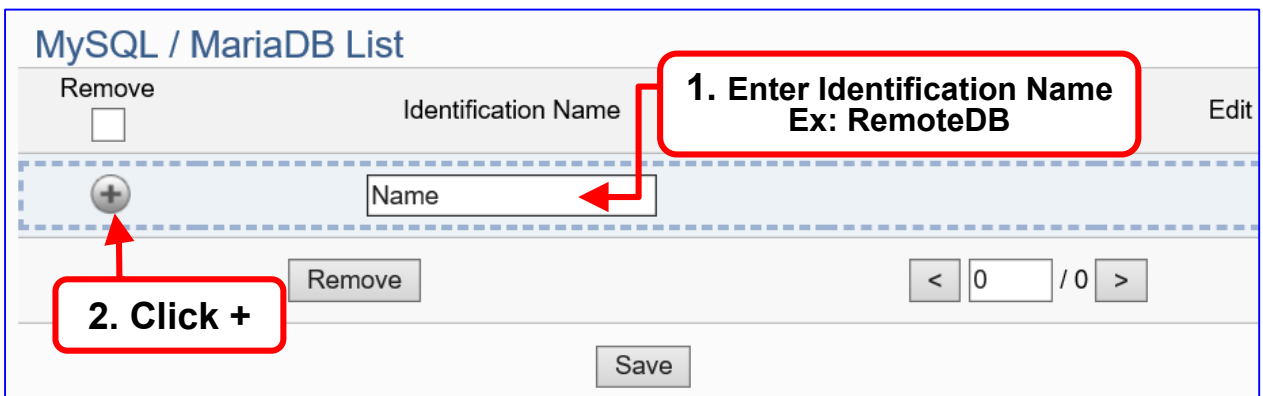
Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

● **Step 3. MySQL/MariaDB Connection Setting**

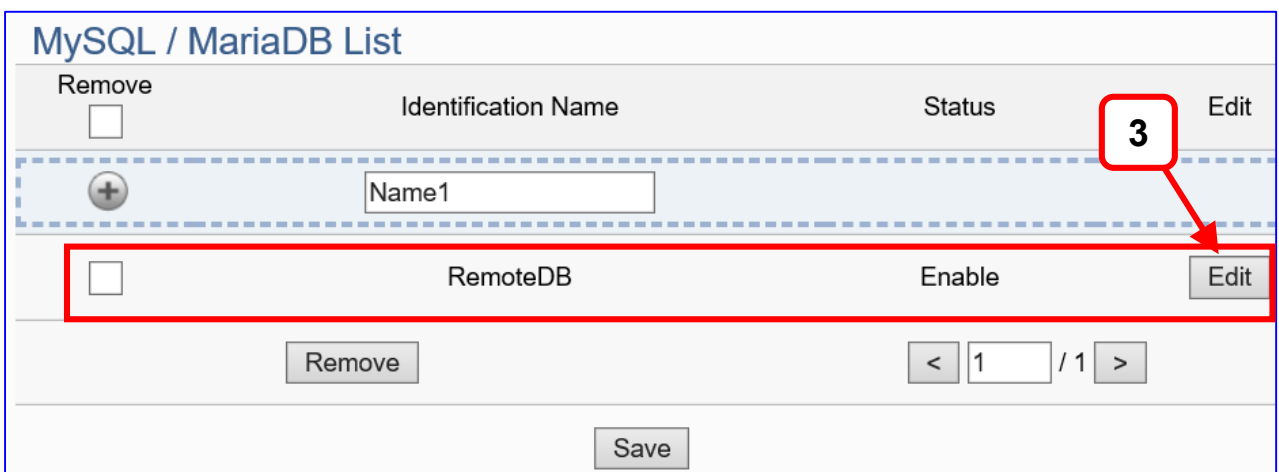


Click the next step, and enter the **Step 3 [MySQL/MariaDB Connection Setting]** of the UI setting. This page is for setting the connecting remote database.

We select the “Modbus TCP / Remote Database” at the beginning, so this step will auto enter the [**Advanced Setting > Data Logger > MySQL / MariaDB**] Setting. The “Step Box” will prevent the user from selecting the wrong platform.



Add a database identification name (Ex: RemoteDB) as below, and then click [**Edit**] button to enter the “MySQL / MariaDB 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.

[MySQL / MariaDB Content Setting] can set up the database relational setting.

MySQL / MariaDB Connection Settings	
Identification Name	ICPDAS_Cycle_DateTime
Database Name	ICPDAS
Table Name	Module_All_DateTime
IP	192.168.85.11
Port	3306
Account	chris
Password	....
Log Mode	Cycle
Interval Seconds	5
Date Time Format	[yyyy-MM-dd HH:mm:ss]
Enable	<input checked="" type="checkbox"/>
Test Connection	Connection
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

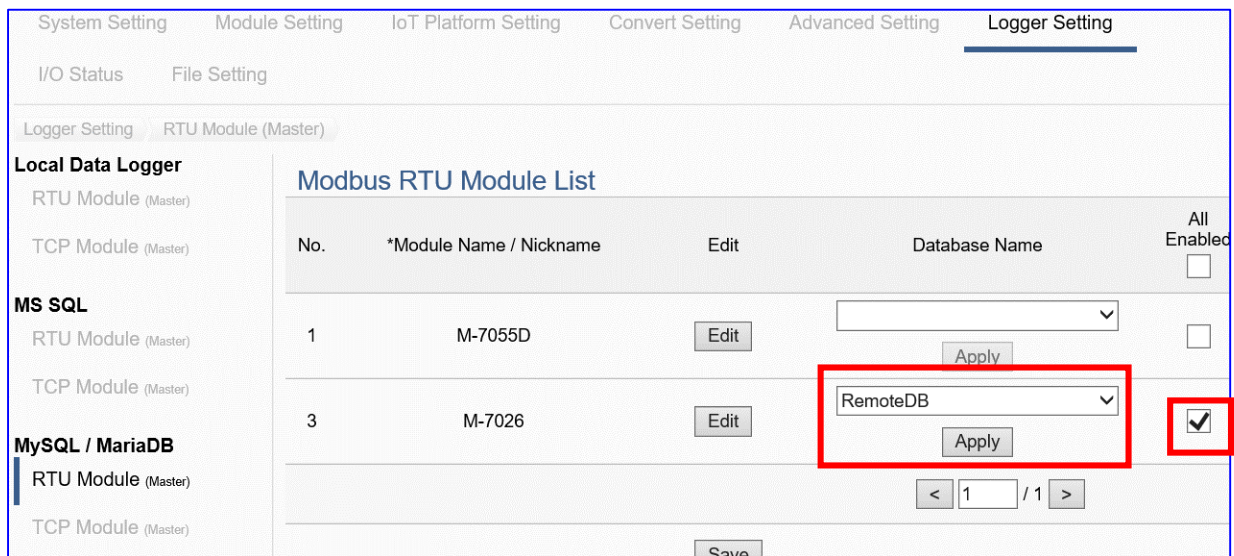
Advanced Setting > Data Logger > MySQL/MariaDB – Content Settings	
Identification Name	User defined name to identify the database.
Database Name	The name of the remote database. If it does not exist, it will add a new database with this name.
Table Name	The table name of the remote database. If it does not exist, it will add a new table with this name.
IP	The Server IP and name of the remote database.
Port	The port to connect with database. Default: 3306 (for MySQL/MariaDB)
Account	The login name of the remote database.
Password	The login password of the remote database.
Log Mode	<b>Cycle:</b> Record one log data at the interval time set below. <b>Data Change:</b> Only record when the data has changed.
Interval Seconds	Set up the interval time to save the I/O data to the remote database. Unit: Second.
Date Time Format	Select to separate the date and time into two [Columns] or combine the date and time in one [Column].
Enable	Check to enable the data logger to the remote database. Default: check.
Test Connection	Click to test the connection to the remote database. Result: Success or Failure.
OK / Cancel	Click “OK” to save the settings of this page. Click “Cancel” to exit the setting page without saving.

● **Step 4. Apply Connection & Enable Recording Module**



Click the next step, and enter the **Step 4 [Apply Connection & Enable Recording Module]** UI setting. This step is to enable the Modbus RTU module and connection. We select the “Modbus RTU /MySQL(MariaDB)” of “Data Log” at the beginning, so this step will auto enter the [**Logger Setting > MySQL/MariaDB > RTU Module (Master)**] setting page. The “Step Box” will prevent the user from selecting the wrong platform.

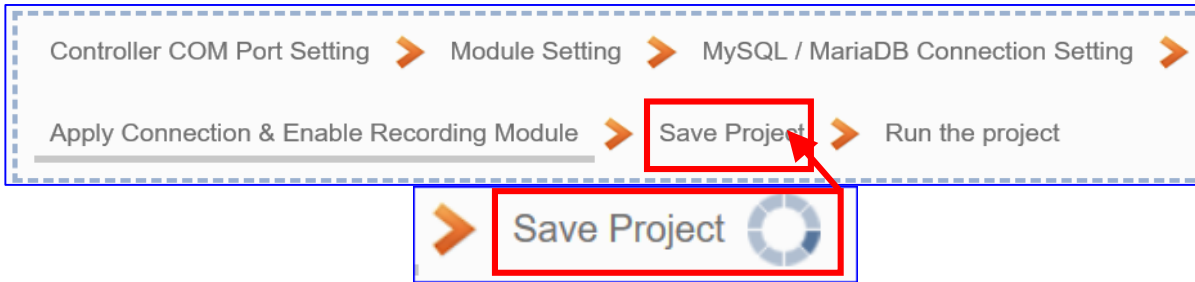
Here **select** and **apply** the Database name (Ex: RemoteDB), and **enable** the M-7026.



Logger Setting > MySQL/MariaDB > RTU Module (Master)	
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 data logger, click [Edit] of that module to enter the “Content Setting”. It is normal to set all channels as enabled, and the function will not affect the unconnected channels.
Database Name	Select and apply the recording remote database name.
All Enabled	Check [All Enabled] box to enable all modules in list for data logger. Default: Uncheck. Check the “box” of each module can enable just that module for data logger.
	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.

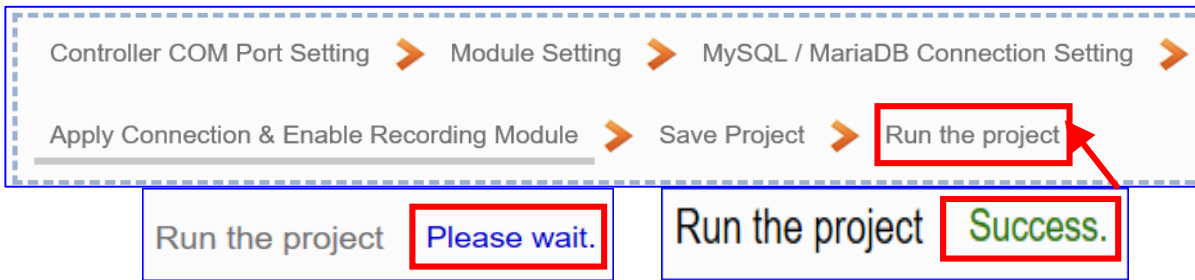
● **Step 5. Save Project**

The setting of this example is finished now. 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**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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 project communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.

**I/O Status** File Setting

I/O Status

**Modbus RTU Module (Master)**

No.	Name	Serial Port
3	M-7026	ttyO2
1	M-7055D	ttyO5

< 1 / 1 >

**Modbus TCP Module (Master)**

No.	Name	LAN
1	DL-302	LAN

< 1 / 1 >

**Related Settings**

Number of variables: 10 (Updated 10 points per second)

Display Update Time (ms): 1000

**I/O Status** I/O Scaling Bitwise

Variable Name	Data Type	Value	Description	Status
CO2	Short	824	room1	Good
Relative_humidity	Short	6798		Good
Temperature_Celsiu:	Short	2099		Good
Temperature_Fahrer	Short	6978		Good

● **MySQL/MariaDB Remote Database Example Descriptions:**

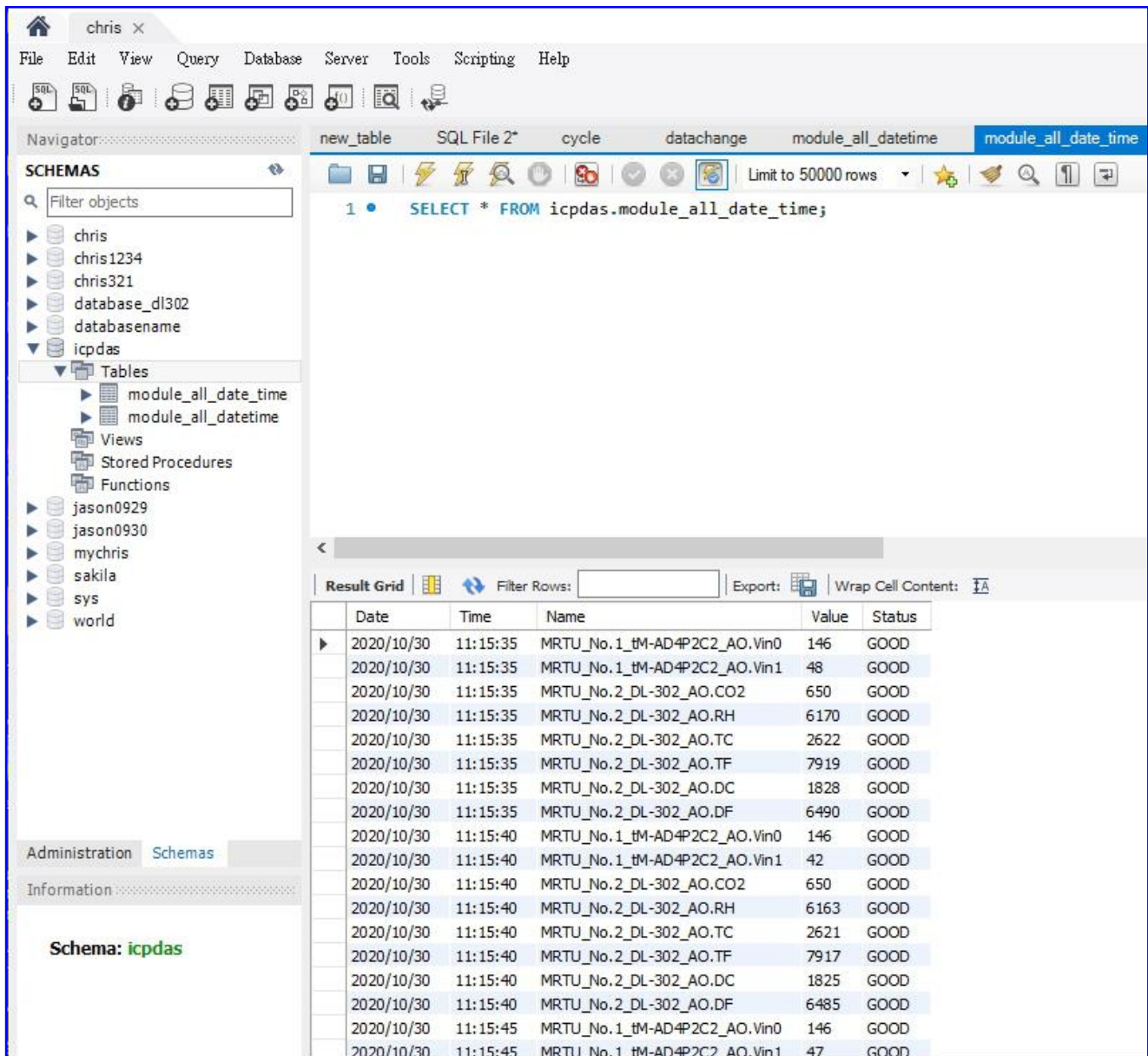
Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

For database operation, please refer to **FAQ-002 (MySQL)** of the **UA series FAQ list**:

[FAQ-002\\_How to save the UA collected data into SQL and then show trend chart in InduSoft?](#)  
(Take MySQL Installer 5.7.31 as an example)

The connection screen view of the **MySQL Remote Database**.

**1. MySQL database screen view: Date/Time column separated (reference)**



## 2. MySQL database screen view: Date/Time column combined (reference)

The screenshot shows a MySQL database management tool interface. The left sidebar displays a tree view of schemas, with 'icpdas' expanded to show tables including 'module\_all\_datetime'. The main window contains a query editor with the following SQL statement:

```
SELECT * FROM icpdas.module_all_datetime;
```

Below the query editor is a 'Result Grid' showing the following data:

DateTime	Name	Value	Status
2020-10-30 11:12:19	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:19	MRTU_No.1_tM-AD4P2C2_AO.Vin1	54	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.CO2	636	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.RH	6194	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.TC	2616	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.TF	7908	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.DC	1829	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.DF	6492	GOOD
2020-10-30 11:12:24	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:24	MRTU_No.1_tM-AD4P2C2_AO.Vin1	55	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.CO2	636	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.RH	6190	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.TC	2616	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.TF	7908	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.DC	1827	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.DF	6488	GOOD
2020-10-30 11:12:29	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:29	MRTU_No.1_tM-AD4P2C2_AO.Vin1	52	GOOD

The connection screen view of the **MariaDB Remote Database**.

**1. MariaDB database screen view: Date/Time column separated (reference)**

The screenshot shows the phpMyAdmin interface for a MariaDB 10 database. The current database is 'ICPDAS' and the selected table is 'Module\_All\_Date\_Time'. The SQL query displayed is 'SELECT \* FROM `Module\_All\_Date\_Time`'. The table view shows 25 rows of data. Each row contains a checkbox, edit, copy, and delete icons, followed by columns for Date, Time, Name, Value, and Status.

	Date	Time	Name	Value	Status
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.1_tM-AD4P2C2_AO.Vin1	59	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.CO2	637	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.DC	1822	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.DF	6479	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.RH	6099	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.TC	2635	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.TF	7943	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.1_tM-AD4P2C2_AO.Vin1	63	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.CO2	636	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.DC	1819	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.DF	6474	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.RH	6093	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.TC	2634	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.TF	7941	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.1_tM-AD4P2C2_AO.Vin1	64	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.CO2	636	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.DC	1820	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.DF	6476	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.RH	6092	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.TC	2635	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.TF	7943	GOOD



## 2. MariaDB database screen view: Date/Time column combined (reference)

The screenshot shows the phpMyAdmin interface for a MariaDB 10 database named 'ICPDAS'. The selected table is 'Module\_All\_DateTime'. The SQL query displayed is 'SELECT \* FROM `Module\_All\_DateTime`'. The table view shows 25 columns per page, with the first column being 'Date/Time'. The data rows show a list of records with columns: Date/Time, Name, Value, and Status. The status for all records is 'GOOD'.

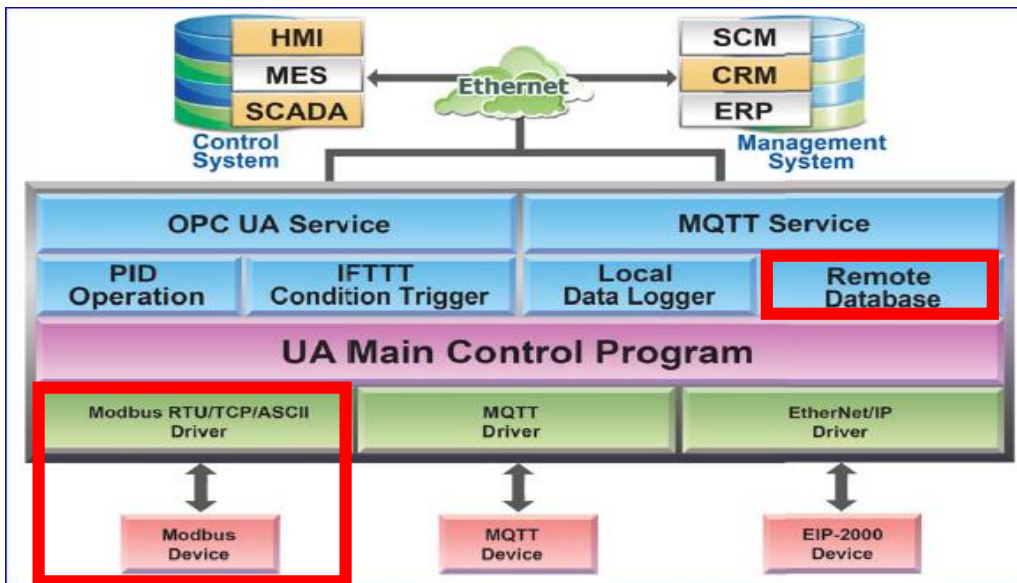
	Date/Time	Name	Value	Status
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.1_tM-AD4P2C2_AO.Vin1	61	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.DC	1812	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.DF	6461	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.RH	6036	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.TC	2642	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.TF	7955	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.1_tM-AD4P2C2_AO.Vin1	62	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.DC	1812	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.DF	6461	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.RH	6038	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.TC	2642	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.TF	7955	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.1_tM-AD4P2C2_AO.Vin1	59	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.DC	1811	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.DF	6459	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.RH	6038	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.TC	2641	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.TF	7953	GOOD

### 4.3.6 Function Wizard: Modbus / MySQL(MariaDB) (TCP Multi-Module Example)

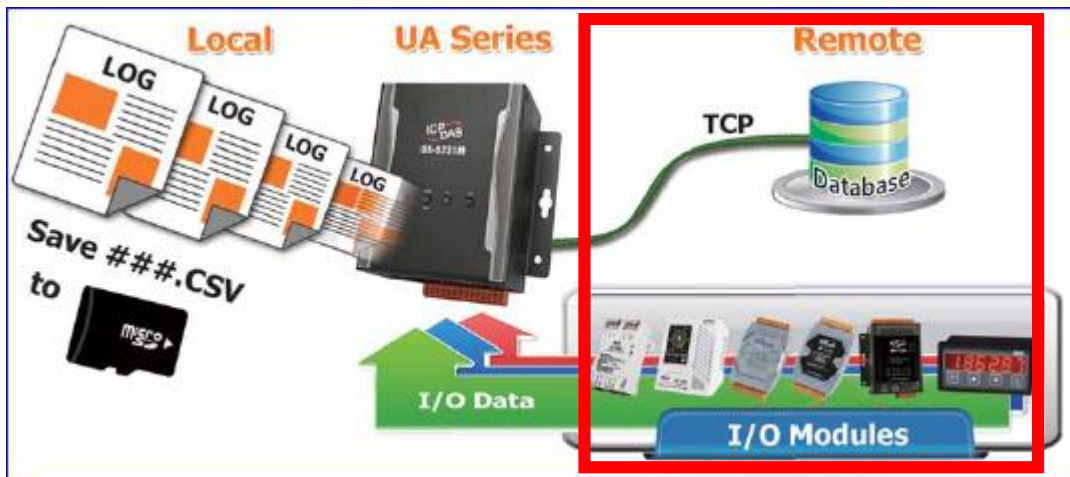
UA Data Logger supports to collect devices I/O status and then directly write into remote side MySQL/MariaDB Database for the Big Data analysis.

The connection steps for MySQL and MariaDB is the same, so here will introduce them together. The Modbus / MySQL and MariaDB Remote Database settings include Modbus RTU and TCP. Here will introduce Modbus TCP and multiple modules as the setting sample.

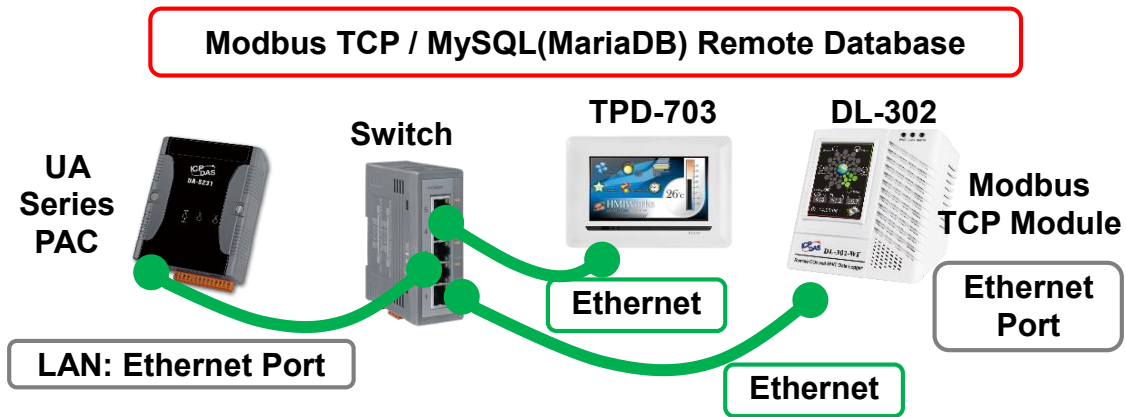
#### Modbus / Remote Database Function Diagram:



#### Application Solution:

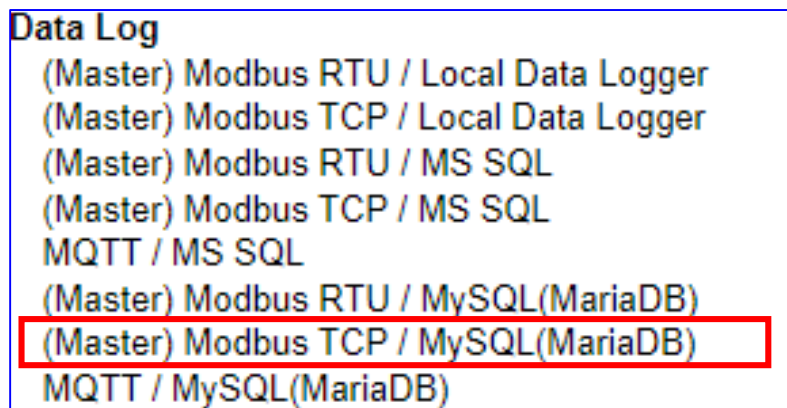


● **Modbus TCP / MySQL(MariaDB) :**



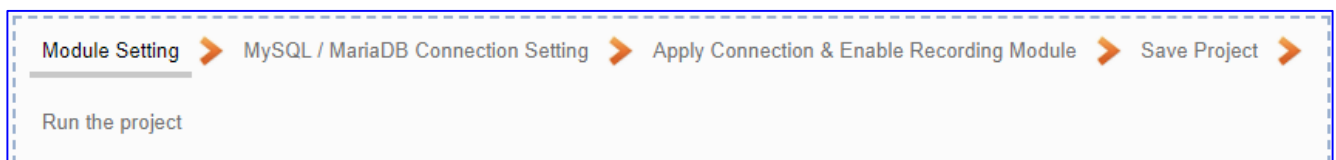
**Note:** The hardware/network connection methods please see the [Chapter 2](#).

When UA series controller connects the Modbus TCP modules (via Ethernet, as the picture), user can choose the item [**Modbus TCP / MySQL(MariaDB)**] of the “Data Log” in the Function Wizard.

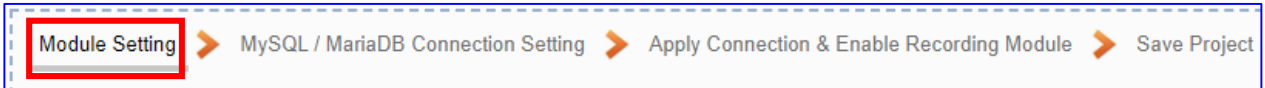


**[Step Box]:**

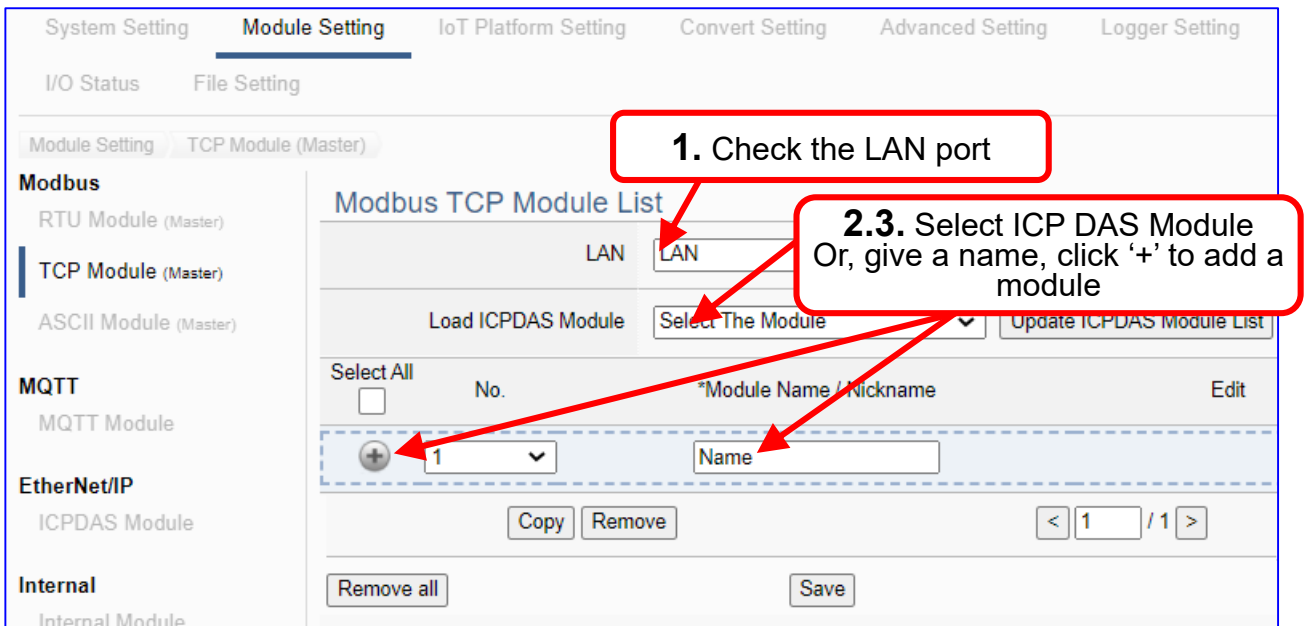
The Step Box of the [**Modbus TCP / MySQL(MariaDB)**] has the steps below. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step-by-step and then can complete the project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



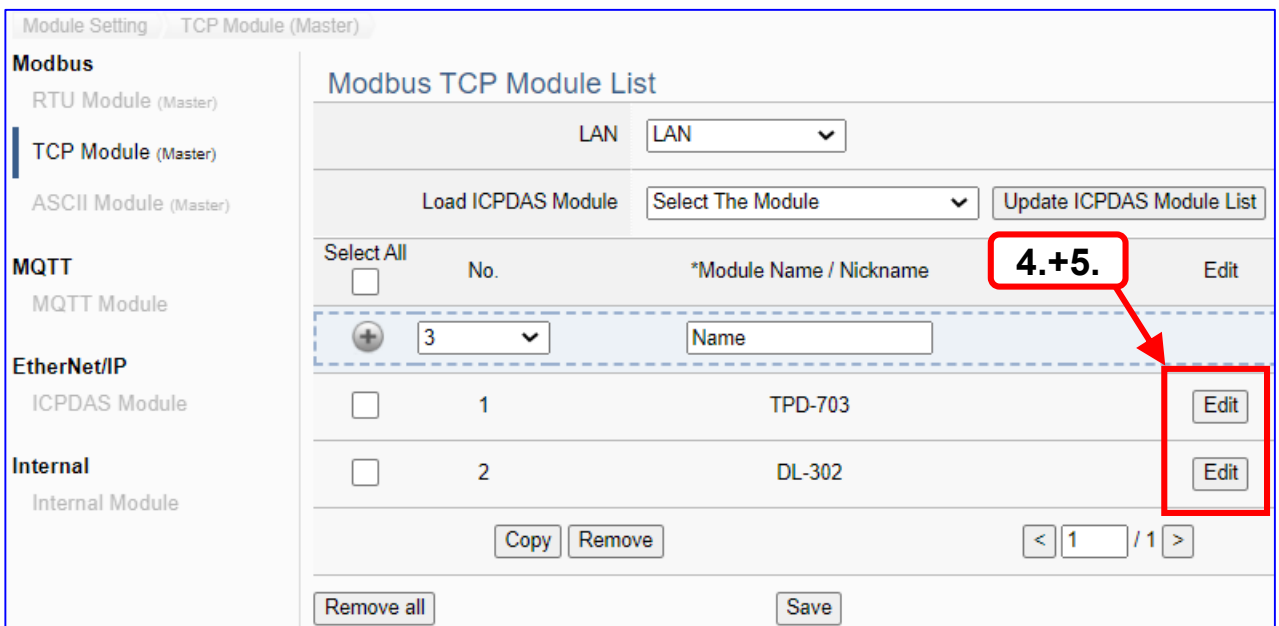
● **Step 1. Module Setting**



It auto-enter the first step, **Step 1 [Module Setting]** of the UI setting. This page is for setting the communication values with the connected modules. First check the port that connected with the module, and each module can give a name (Default name: Name). Click [ ⊕ ] button could add a new module. When connecting two modules, set up twice to the different modules in the same way. and then click each [Edit] button to configure the module content and the Modbus mapping table.



Add modules one by one, in this example, add **TPD-703** and **DL-302** one by one (**DL-302** can be directly selected from **ICP DAS Module List**, and the system will automatically set it). Please click [Edit] button one by one, to enter and set each Module content.



**TPD-703's [Module Content Setting]** page: set up the module and the Modbus mapping table:

Module Content Setting	
No.	1
Module Name	TPD-703
IP	192 168 85 20
Port	502
Slave ID	1
Timeout(ms)	500
Polling Rate(ms)	500
Modbus Mapping Table Setting	
Data Model	01 Coil Status(0x) ▼
Start Address	0
Data Number	1
Create Tables	Add

**Example: TPD-703**

**[IP] 192.168.85.20 (by user case)**

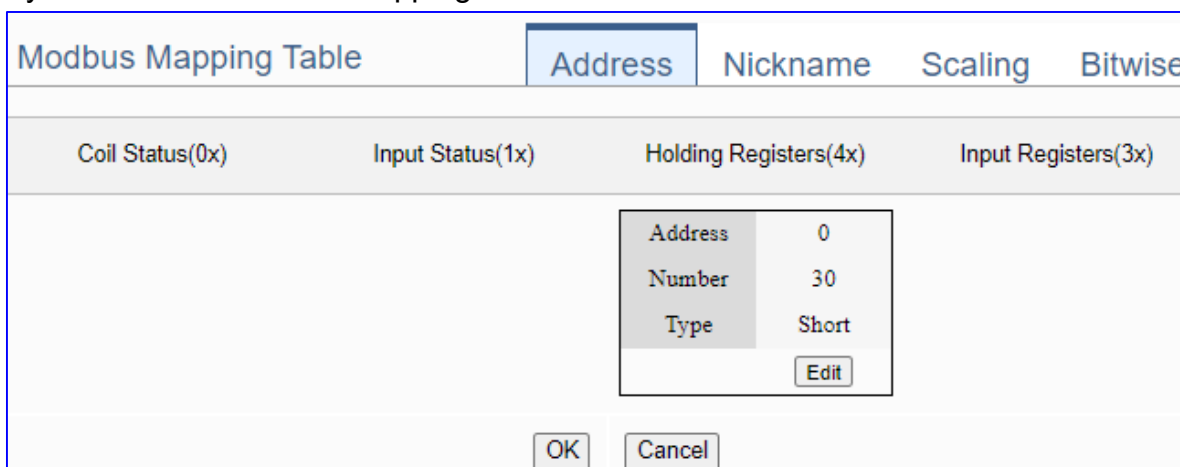
**[ Modbus Mapping Table Setting ]**  
**Data Model: 03 Holding Registers(4x)**  
**Start Address: 0**  
**Data Number: 30**  
**Type: 16-bit Short**  
**→ Click [ Add ]**

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. (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: 5px; width: fit-content; margin-top: 5px;"> 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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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 mapping DO, DI, AO & AI.

**Address:**

Display and edit the Modbus Mapping Table.



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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>02 Input Status(1x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>03 Holding Registers(4x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	lc_101_DO0	Short	<input type="checkbox"/>					
1	lc_101_DO1	Short	<input type="checkbox"/>					
2	lc_101_DO2	Short	<input type="checkbox"/>					
3	lc_101_DO3	Short	<input type="checkbox"/>					
4								
5	lc_101_DO5	Short	<input type="checkbox"/>					

User can define the name, e.g. the connected I/O module name for easy identification.

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.

**DL-302's [Module Content Setting]** page: set up the module and the Modbus mapping table:

Module Content Setting	
No.	<input type="text" value="2"/>
Module Name	<input type="text" value="DL-302"/>
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(ms)	<input type="text" value="500"/>
Polling Rate(ms)	<input type="text" value="500"/>
Modbus Mapping Table Setting	
Data Model	<input type="text" value="04 Input Registers(3x)"/>
Start Address	<input type="text" value="0"/>
Data Number	<input type="text" value="6"/>
Type	<input type="text" value="16-bit Short"/>
Create Tables	<input type="button" value="Add"/>

**Example: DL-302**

**[IP] 192.168.81.251 (by user case)**

**Because DL-302 is selected from the ICP DAS module list, the system will auto-setup the following items, the users do not need to add:**

**[ Modbus Mapping Table Setting ]**  
**Data Model: 04 Input Registers(3x)**  
**Start Address: 0**  
**Data Number: 6**  
**Type: 16-bit Short**

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. (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;"> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="background-color: #007bff; color: white;">01 Coil Status(0x)</td></tr> <tr><td>02 Input Status(1x)</td></tr> <tr><td>03 Holding Registers(4x)</td></tr> <tr><td>04 Input Registers(3x)</td></tr> </table> </div>	01 Coil Status(0x)	02 Input Status(1x)	03 Holding Registers(4x)	04 Input Registers(3x)
01 Coil Status(0x)					
02 Input Status(1x)					
03 Holding Registers(4x)					
04 Input Registers(3x)					
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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 mapping DO, DI, AO & AI.

**Address:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table												
Address		Nickname	Scaling	Bitwise								
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>6</td> </tr> <tr> <td>Type</td> <td>Short</td> </tr> <tr> <td colspan="2" style="text-align: right;"> <input type="button" value="Edit"/> </td> </tr> </table>		Address	0	Number	6	Type	Short	<input type="button" value="Edit"/>	
Address	0											
Number	6											
Type	Short											
<input type="button" value="Edit"/>												
<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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>02 Input Status(1x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type		Description				
<b>03 Holding Registers(4x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
<b>04 Input Registers(3x)</b>								
Table Display					Show		Hide	
Address	Variable name	Data Type	Swap	Description				
0	<input type="text" value="CO2"/>	Short	<input type="checkbox"/>	<input type="text"/>				
1	<input type="text" value="Relative_humidity"/>	Short	<input type="checkbox"/>	<input type="text"/>				
2	<input type="text" value="Temperature_Celsius"/>	Short	<input type="checkbox"/>	<input type="text"/>				
3	<input type="text" value="Temperature_Fahrenheit"/>	Short	<input type="checkbox"/>	<input type="text"/>				
4	<input type="text" value="Dew_point_temperature_"/>	Short	<input type="checkbox"/>	<input type="text"/>				

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.

**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

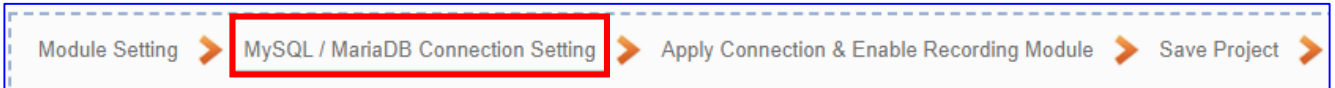
**Bitwise:**

**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

Modbus Mapping Table	Address	Nickname	Scaling	Bitwise
<b>03 Holding Registers(4x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference		Bitwise	
<b>04 Input Registers(3x)</b>				
Table Display <input type="button" value="Show"/> <input type="button" value="Hide"/>				
Address	Reference		Bitwise	
0	<input type="text" value="CO2"/> <input type="button" value="Bit0"/> <input type="text" value="aa"/> <input type="button" value="Bit2"/> <input type="text" value="bb"/> <input type="button" value="Bit4"/> <input type="text"/> <input type="button" value="Bit6"/> <input type="text"/> <input type="button" value="Bit8"/> <input type="text"/> <input type="button" value="Bit10"/> <input type="text"/> <input type="button" value="Bit12"/> <input type="text"/> <input type="button" value="Bit14"/> <input type="text"/>		<input type="button" value="Hide Detail"/> <input type="button" value="Bit1"/> <input type="text"/> <input type="button" value="Bit3"/> <input type="text"/> <input type="button" value="Bit5"/> <input type="text"/> <input type="button" value="Bit7"/> <input type="text"/> <input type="button" value="Bit9"/> <input type="text"/> <input type="button" value="Bit11"/> <input type="text"/> <input type="button" value="Bit13"/> <input type="text"/> <input type="button" value="Bit15"/> <input type="text"/>	

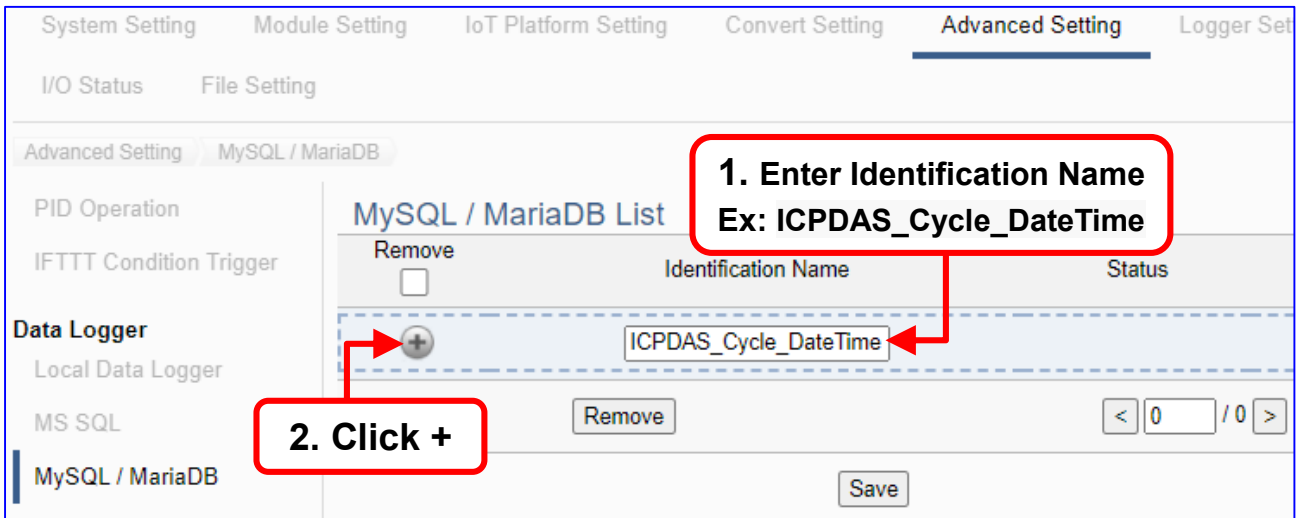
Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

● **Step 2. MySQL/MariaDB Connection Setting**

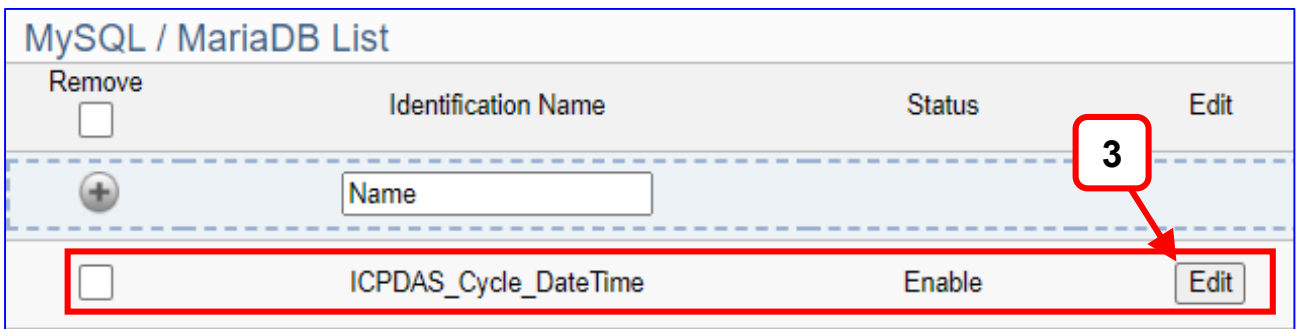


Click the next step, and enter the **Step 2 [MySQL/MariaDB Connection Setting]** of the UI setting. This page is for setting the connecting remote database.

We select the “Modbus TCP / MySQL/MariaDB” at the beginning, so this step will auto enter the **[Advanced Setting > Data Logger > MySQL/MariaDB]** Setting.



Add a database identification name (Ex: **ICPDAS\_Cycle\_DateTime**) as below, and then click **[Edit]** button to enter the “MySQL/MariaDB Connection Setting” page.



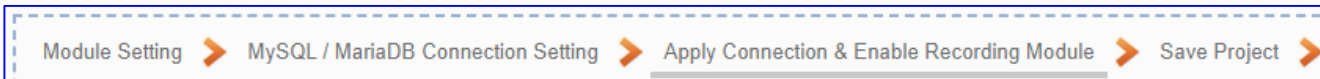
If set up a wrong module, user can click the box in the left side of the module number and click the **[Remove]** button to delete the module.

**[MySQL / MariaDB Connection Settings]** can set up the database relational setting.

MySQL / MariaDB Connection Settings	
Identification Name	<input type="text" value="ICPDAS_Cycle_DateTime"/>
Database Name	<input type="text" value="ICPDAS"/>
Table Name	<input type="text" value="Module_All_DateTime"/>
IP	<input type="text" value="192.168.85.11"/>
Port	<input type="text" value="3306"/>
Account	<input type="text" value="chris"/>
Password	<input type="password" value="...."/>
Log Mode	<input type="text" value="Cycle"/> ▼
Interval Seconds	<input type="text" value="5"/>
Date Time Format	<input type="text" value="[yyyy-MM-dd HH:mm:ss]"/> ▼
Enable	<input checked="" type="checkbox"/>
Test Connection	<input type="button" value="Connection"/>
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

Advanced Setting > Data Logger > MySQL/MariaDB – Content Settings	
Identification Name	User defined name to identify the database.
Database Name	The name of the remote database. If it does not exist, it will add a new database with this name.
Table Name	The table name of the remote database. If it does not exist, it will add a new table with this name.
IP	The Server IP and name of the remote database.
Port	The port to connect with database. Default: 3306 (for MySQL/MariaDB)
Account	The login name of the remote database.
Password	The login password of the remote database.
Log Mode	<b>Cycle:</b> Record one log data at the interval time set below. <b>Data Change:</b> Only record when the data has changed.
Interval Seconds	Set up the interval time to save the I/O data to the remote database. Unit: Second.
Date Time Format	Select to separate the date and time into two [Columns] or combine the date and time in one [Column].
Enable	Check to enable the data logger to the remote database. Default: check.
Test Connection	Click to test the connection to the remote database. Result: Success or Failure.
OK / Cancel	Click “OK” to save the settings of this page. Click “Cancel” to exit the setting page without saving.

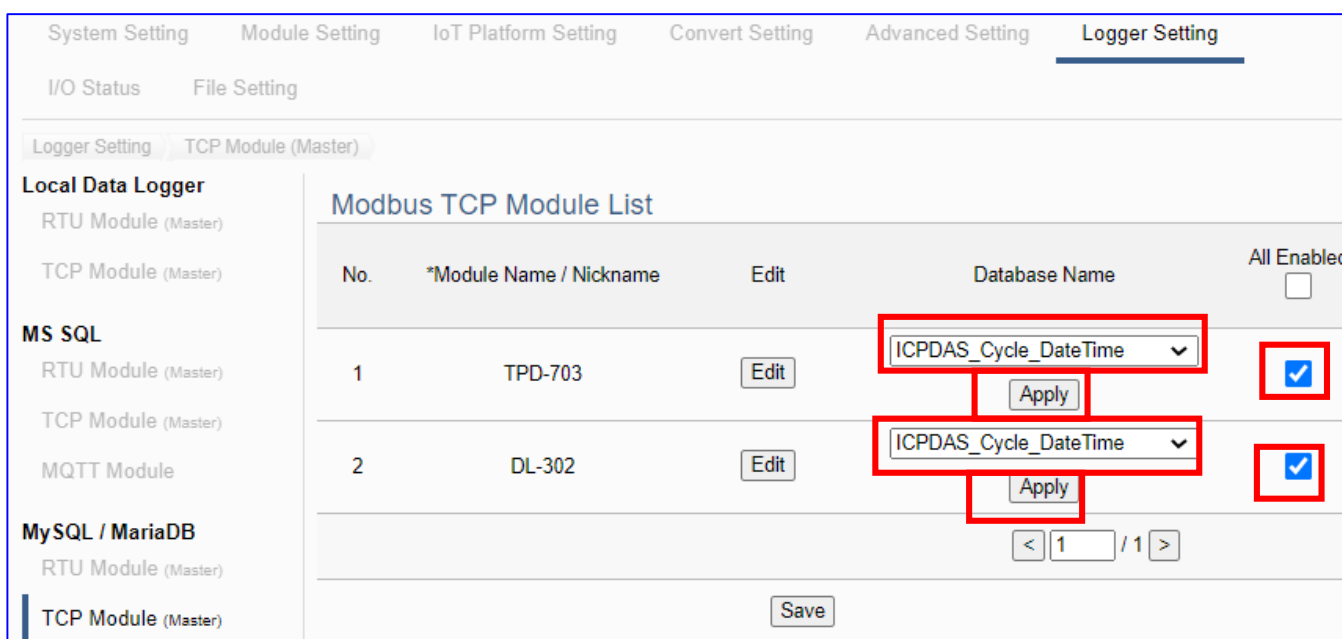
● **Step 3. Apply Connection & Enable Recording Module**



Click the next step, and enter the **Step 3 [Apply Connection & Enable Recording Module]** UI setting. This step is to enable the Modbus TCP module and connection.

We select the “Modbus TCP /MySQL/MariaDB” of “Data Log” at the beginning, so this step will auto enter the [**Logger Setting > MySQL/MariaDB > TCP Module (Master)**] setting page. The “Step Box” will prevent the user from selecting the wrong platform.

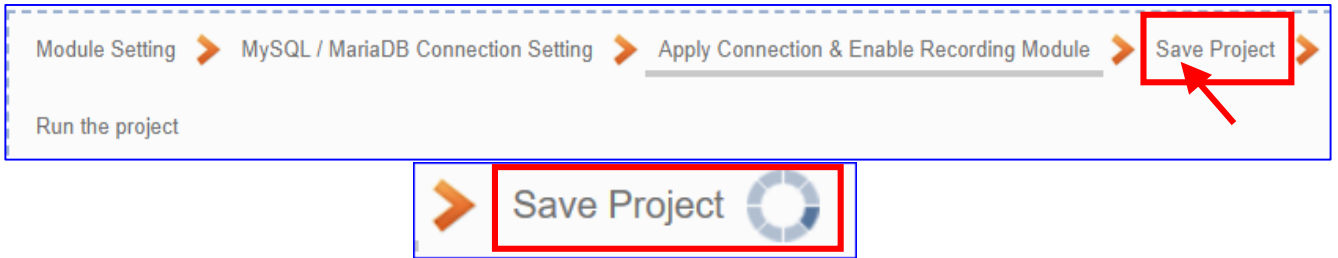
Here **select** and **apply** the Database name (Ex: **ICPDAS\_Cycle\_DateTime**), and **enable** both the TPD-703 and DL-302.



Logger Setting > MS SQL > TCP Module (Master) – Modbus TCP 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 data logger, click [Edit] of that module to enter the “Content Setting”. It is normal to set all channels as enabled, and the function will not affect the unconnected channels.
Database Name	Select and apply the recording remote database name.
All Enabled <input type="checkbox"/>	Check [All Enabled] box to enable all modules in list for data logger. Default: Uncheck. Check the “box” of each module can enable just that module for data logger.
<input type="button" value=" &lt; 1 / 1 &gt;"/>	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.

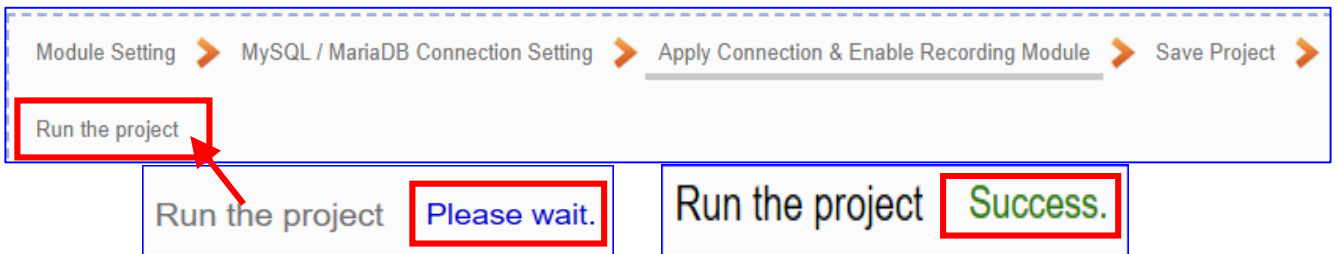
● **Step 4. Save Project**

The setting of this example is finished now. 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.



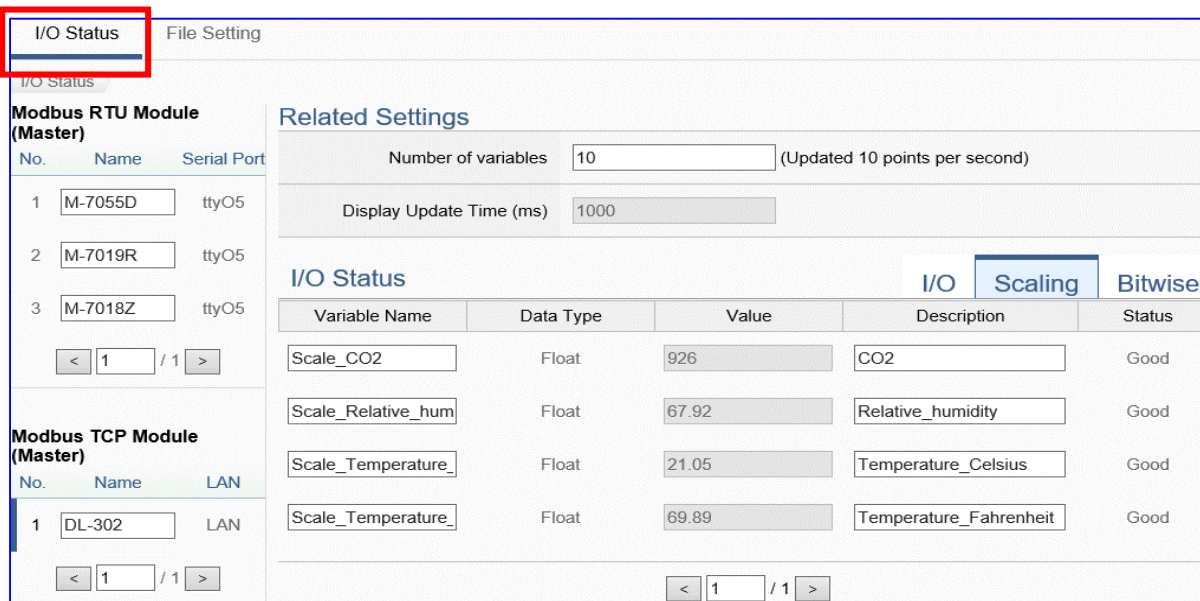
● **Step 5. Run the Project**

The project, after saving, needs to be executed. Click the next step [**Run the Project**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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 project communication. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.





● **MySQL/MariaDB Remote Database Example Descriptions:**

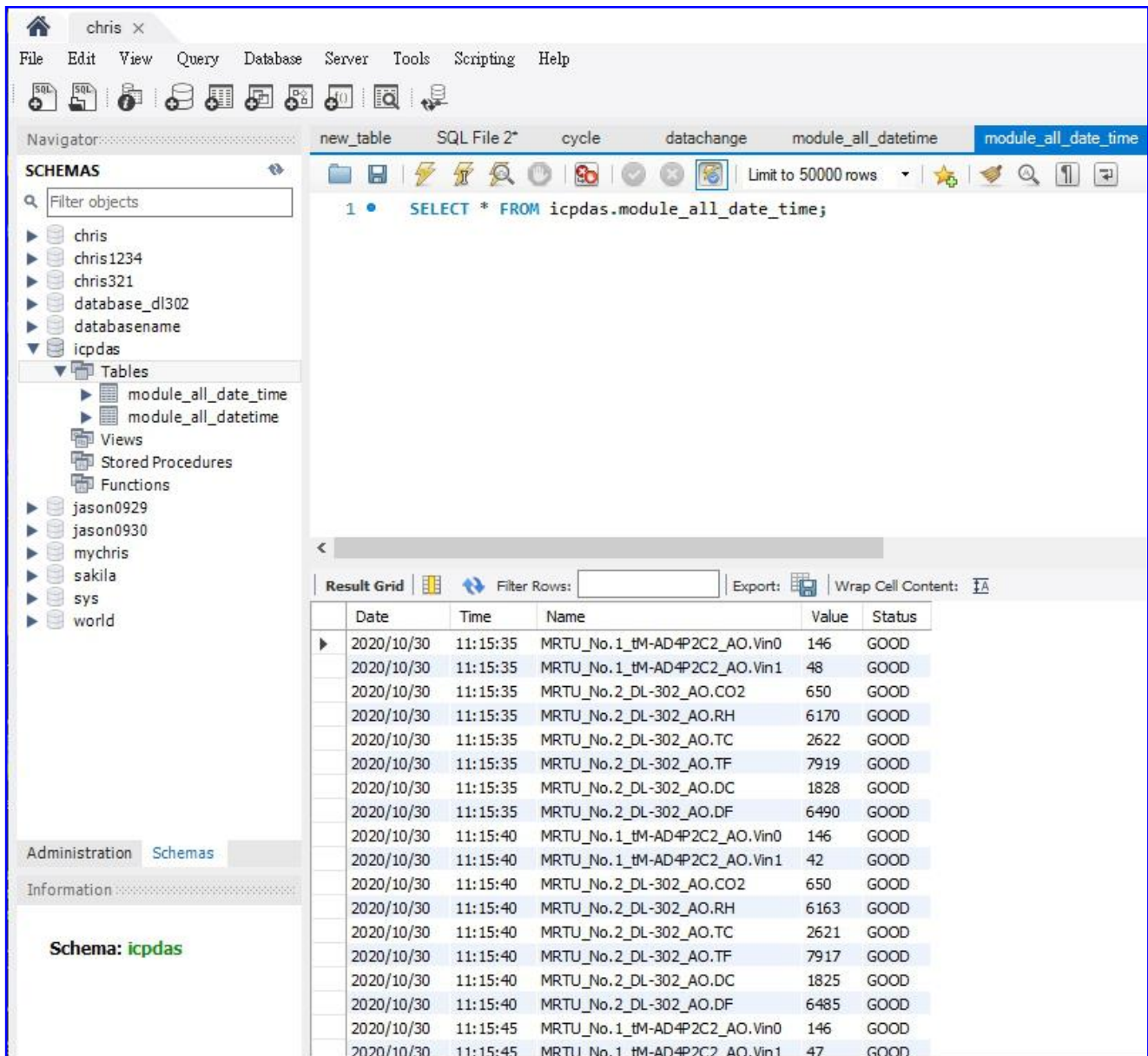
Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

For database operation, please refer to **FAQ-002 (MySQL)** of the **UA series FAQ list**:

[FAQ-002\\_How to save the UA collected data into SQL and then show trend chart in InduSoft?](#)  
(Take MySQL Installer 5.7.31 as an example)

The connection screen view of the **MySQL Remote Database**.

**1. MySQL database screen view: Date/Time column separated (reference)**



## 2. MySQL database screen view: Date/Time column combined (reference)

The screenshot shows a MySQL database management interface. The left sidebar displays a tree view of schemas, with 'icpdas' expanded to show tables like 'module\_all\_date\_time' and 'module\_all\_datetime'. The main window shows a SQL query: `SELECT * FROM icpdas.module_all_datetime;`. Below the query, a 'Result Grid' displays the following data:

DateTime	Name	Value	Status
2020-10-30 11:12:19	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:19	MRTU_No.1_tM-AD4P2C2_AO.Vin1	54	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.CO2	636	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.RH	6194	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.TC	2616	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.TF	7908	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.DC	1829	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.DF	6492	GOOD
2020-10-30 11:12:24	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:24	MRTU_No.1_tM-AD4P2C2_AO.Vin1	55	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.CO2	636	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.RH	6190	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.TC	2616	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.TF	7908	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.DC	1827	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.DF	6488	GOOD
2020-10-30 11:12:29	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:29	MRTU_No.1_tM-AD4P2C2_AO.Vin1	52	GOOD

The connection screen view of the **MariaDB Remote Database**.

**1. MariaDB database screen view: Date/Time column separated (reference)**

The screenshot shows the phpMyAdmin interface for a MariaDB 10 database. The current database is 'ICPDAS' and the selected table is 'Module\_All\_Date\_Time'. The SQL query displayed is 'SELECT \* FROM `Module\_All\_Date\_Time`'. The table contains 256 rows, with the first 24 rows visible. Each row has columns for Date, Time, Name, Value, and Status. The 'Date' and 'Time' columns are separated, as indicated by the section header.

	Date	Time	Name	Value	Status
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.1_tM-AD4P2C2_AO.Vin1	59	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.CO2	637	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.DC	1822	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.DF	6479	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.RH	6099	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.TC	2635	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.TF	7943	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.1_tM-AD4P2C2_AO.Vin1	63	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.CO2	636	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.DC	1819	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.DF	6474	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.RH	6093	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.TC	2634	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.TF	7941	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.1_tM-AD4P2C2_AO.Vin1	64	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.CO2	636	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.DC	1820	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.DF	6476	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.RH	6092	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.TC	2635	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.TF	7943	GOOD

## 2. MariaDB database screen view: Date/Time column combined (reference)

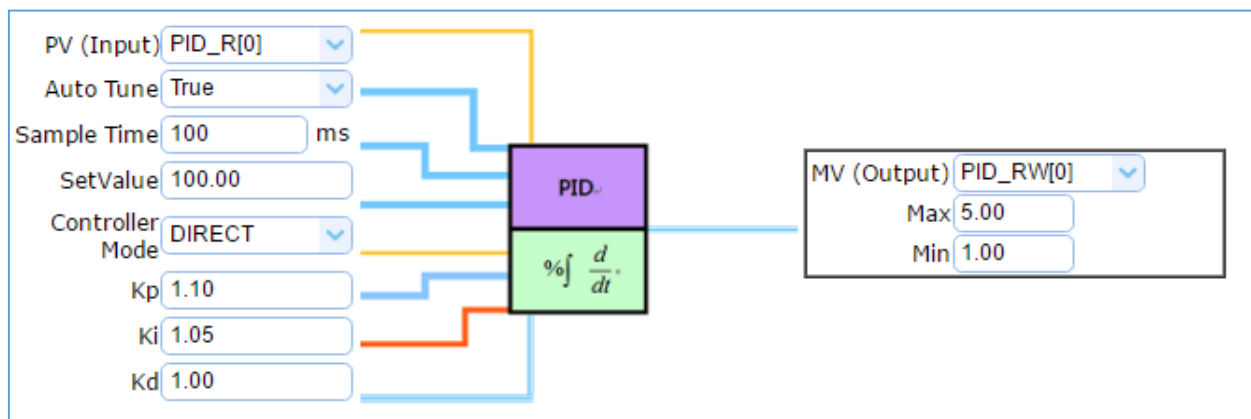
The screenshot shows the phpMyAdmin interface for a MariaDB 10 database named 'ICPDAS'. The selected table is 'Module\_All\_DateTime'. The SQL query displayed is 'SELECT \* FROM `Module\_All\_DateTime`'. The table view shows 25 columns per page, with the first column being 'Date/Time'. The data rows show a list of records with columns: Date/Time, Name, Value, and Status. The status for all records is 'GOOD'.

	Date/Time	Name	Value	Status
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.1_tM-AD4P2C2_AO.Vin1	61	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.DC	1812	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.DF	6461	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.RH	6036	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.TC	2642	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.TF	7955	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.1_tM-AD4P2C2_AO.Vin1	62	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.DC	1812	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.DF	6461	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.RH	6038	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.TC	2642	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.TF	7955	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.1_tM-AD4P2C2_AO.Vin1	59	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.DC	1811	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.DF	6459	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.RH	6038	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.TC	2641	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.TF	7953	GOOD

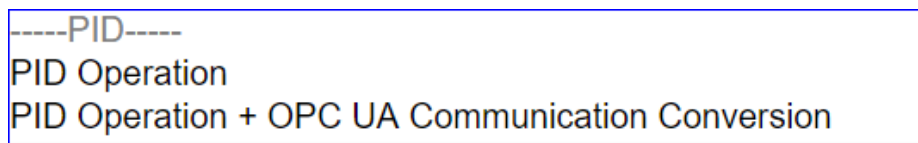
## 4.4 PID

PID (Proportional-Integral-Derivative) control is the most widely used in industrial control systems. A regulator that 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.

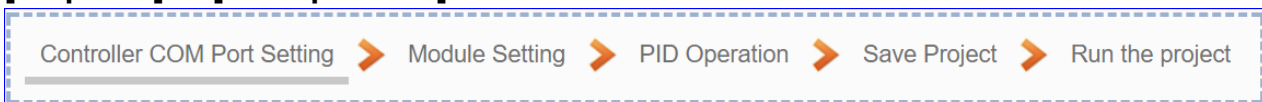
The PID controller is a common feedback loop component in industrial control applications. The controller compares the collected data with a reference value and then uses this difference to calculate a new input value whose purpose is to allow the system data to reach or remain at the reference value.



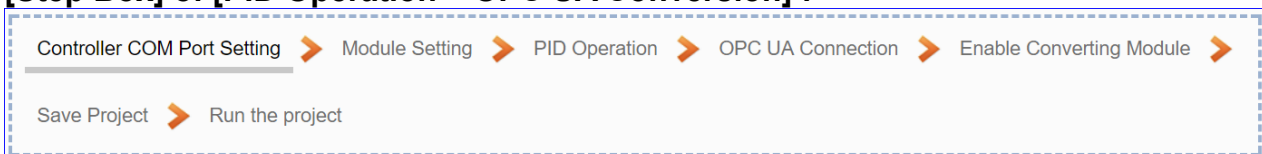
This section introduces the setting steps and the function parameters of the PID. There are 2 items about “PID” function in the “Function Wizard”. The 2<sup>nd</sup> item [PID Operation + OPC UA Communication Conversion] is combining the 1<sup>st</sup> item [PID Operation] and the [Section 4.1.1 Modbus / OPC UA Conversion](#). **The PID operation is for AI/AO data only, please select the AIO module when use the PID related Function Wizard.**



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



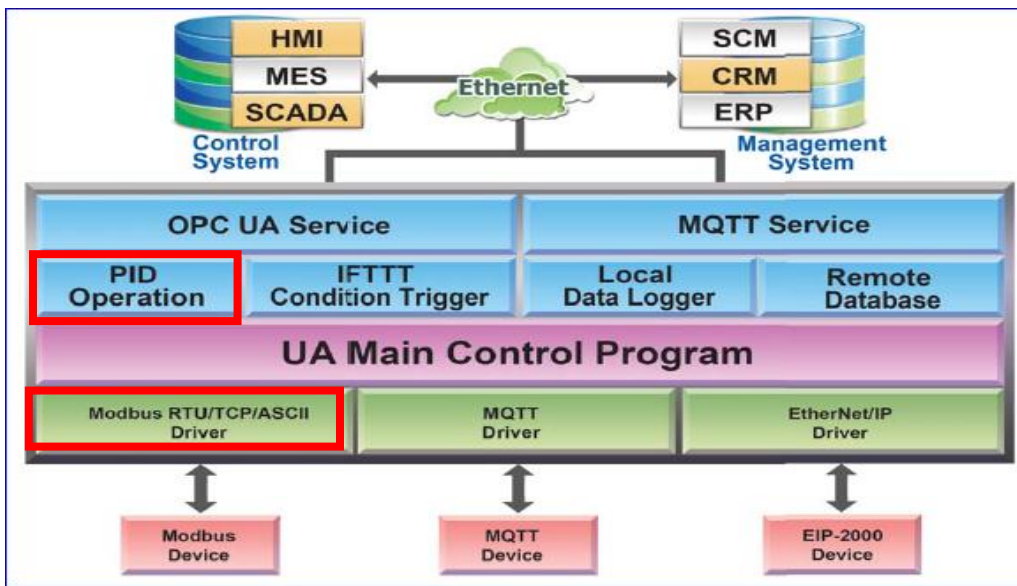
### [Step Box] of [PID Operation + OPC UA Conversion] :



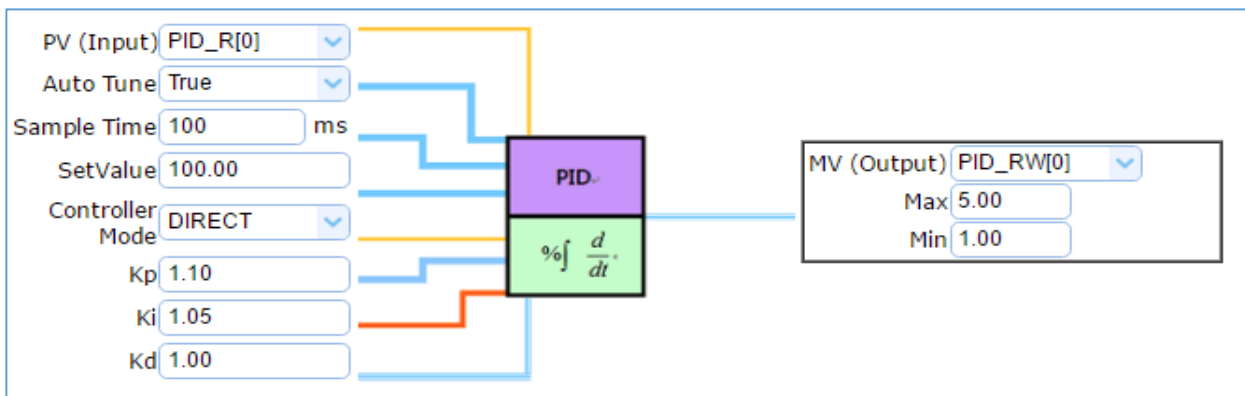
### 4.4.1 Function Wizard: PID Operation (Use AIO Module M-7026)

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]. **The PID operation is for AI/AO data only, please select the AIO module when use the PID related Function Wizard. This demo uses the M-7026(-G) module.**

#### Function Diagram for PID Operation:

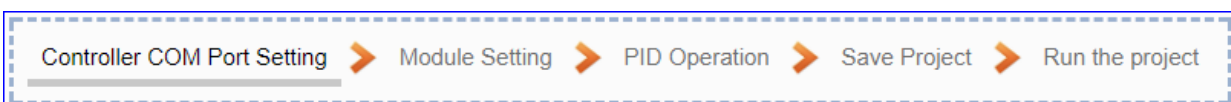


#### Application Solution Example:



#### [Step Box]:

The Step Box of the [PID Operation] is 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.



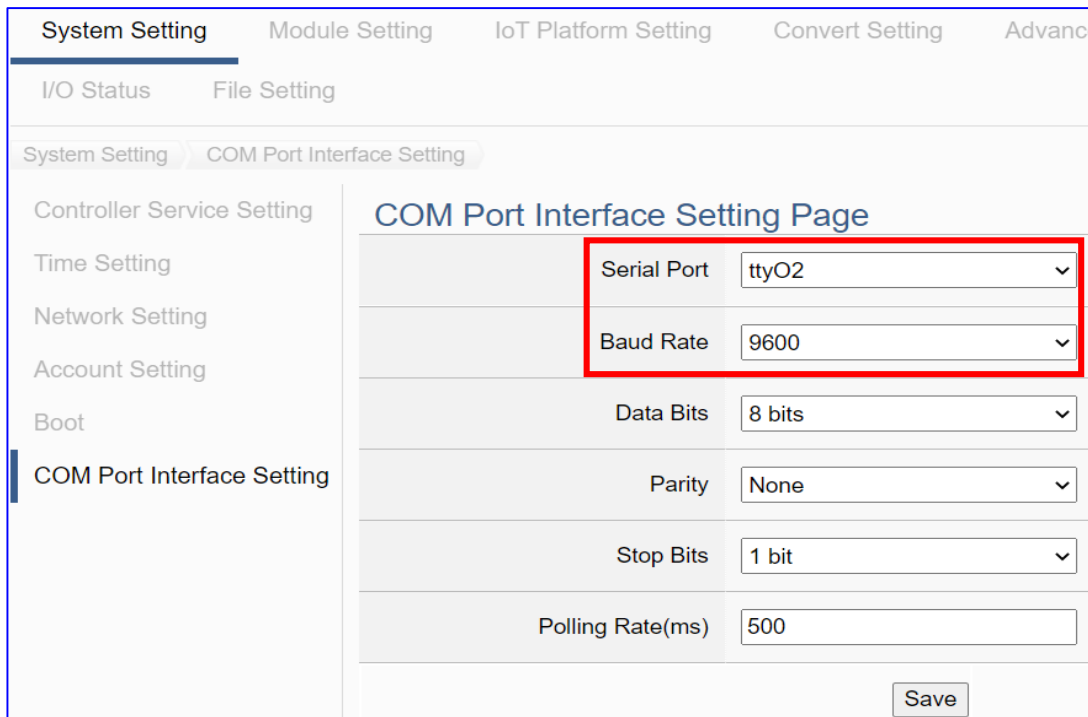
● **Step 1. Controller COM Port Setting**



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

**In this example: Use ttyO2 port to connect with ICP DAS module M-7026(-G), please choose ttyO2 Serial Port, 9600 Baud Rate.**



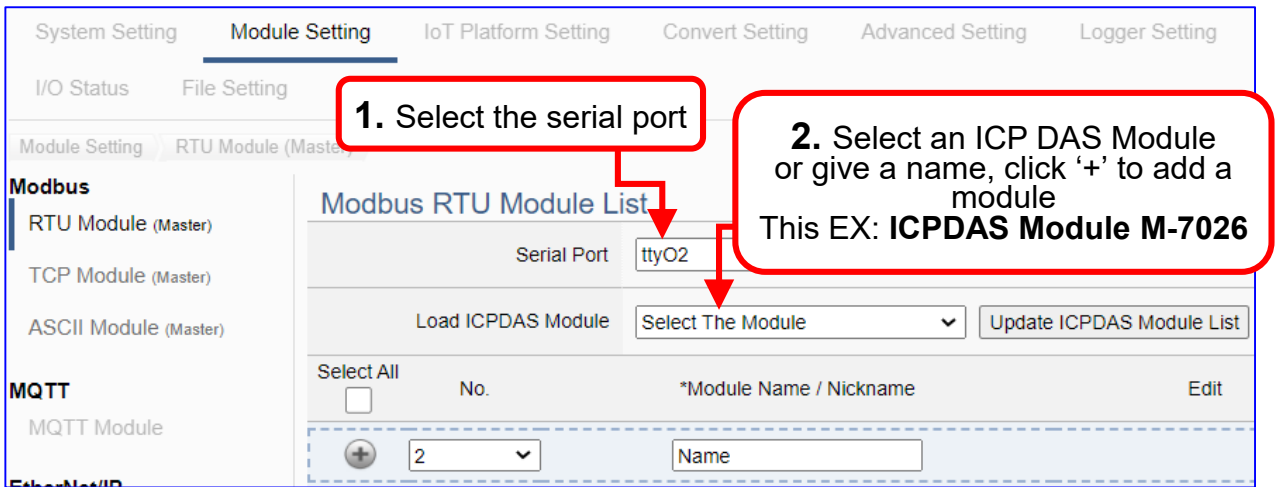
COM Port Interface Setting Page	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485 ; ttyO4: RS-232 ; ttyO5: RS-485
Baud Rate	Choose a baud rate to communicate with the module: 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. The UA controller and 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**

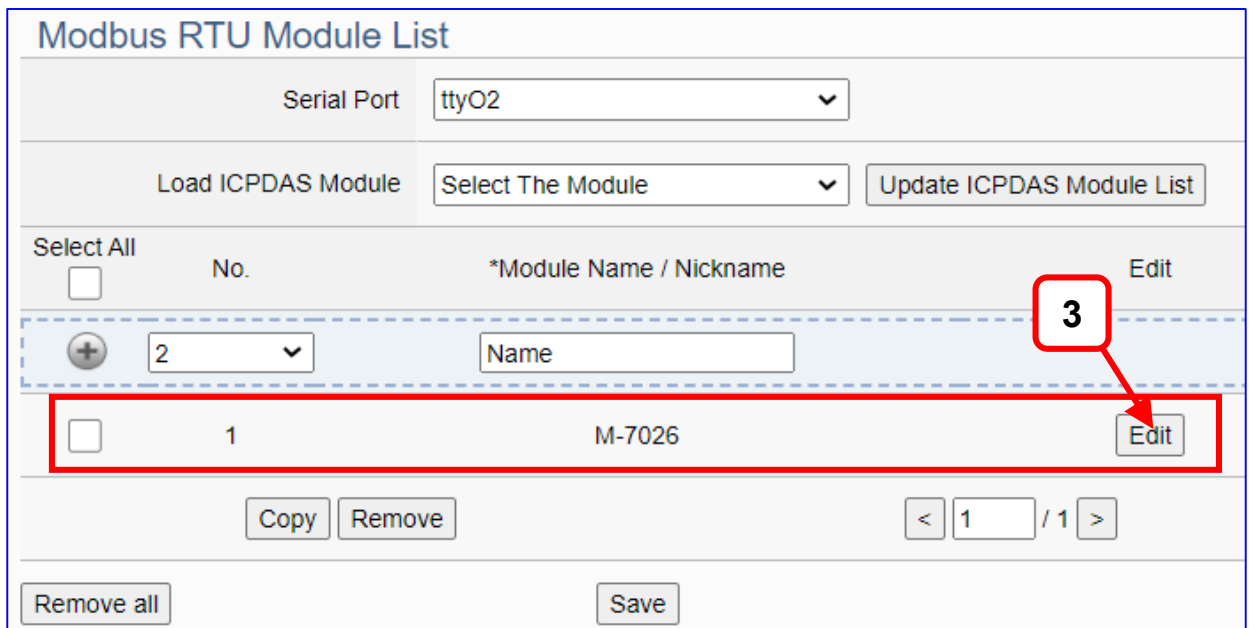


Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting. This page is for setting the communication values with the connected modules.

First, choose the serial port that connected with the module. If use ICP DAS module, select the model to auto load the module setting. If not, give a name (Default: Name), click [ + ] button to add a module. **This EX: select ttyO2, ICP DAS Module M-7026.**



Add a module as below, and then click [Edit] button to enter the “Module Content Setting” page. **Ex: M-7026(-G).**



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.	1
Module Name	M-7026-G
Slave ID	1
Timeout(ms)	500
Write Retry	<input type="checkbox"/> 1

### Modbus Mapping Table Setting

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

**This EX: M-7026(-G)**

**[Slave ID] 1** (set by user's case)

**[Modbus Mapping Table Setting]:**  
 Select ICP DAS module, system will auto set up the Modbus Mapping Table. If not, user needs to check the Modbus address or I/O number from the module user manual and set module by own self, for example:  
 Data Model: 04 Input Registers(3x)  
 Start Address: 0  
 Data Number: 6  
 Type: 16-bit Short  
 → Click [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. (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 gray; padding: 2px; margin-top: 5px;"> <span style="background-color: #0070c0; color: white; padding: 2px;">01 Coil Status(0x)</span>  <span style="color: #0070c0; padding: 2px;">02 Input Status(1x)</span>  <span style="color: #0070c0; padding: 2px;">03 Holding Registers(4x)</span>  <span style="color: #0070c0; padding: 2px;">04 Input Registers(3x)</span> </div>
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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 Modbus Mapping Table as below is in order of DO, DI, AO and AI.

**Address:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise		
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)		Input Registers(3x)	
Address	0	Address	32	Address	32	Address	0
Number	3	Number	3	Number	2	Number	6
Type	Bool	Type	Bool	Type	Short	Type	Short
<input type="button" value="Edit"/>		<input type="button" value="Edit"/>		<input type="button" value="Edit"/>		<input type="button" value="Edit"/>	
<input type="button" value="OK"/>				<input type="button" value="Cancel"/>			

If user selects ICP DAS module, the system will auto set up the Modbus Mapping Table. If not, user needs to check the module Modbus address or I/O number from the module user manual.

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table				
Address		Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>				
Table Display		Show	Hide	
Address	Variable name	Data Type	Description	
0	DO0	Bool		
1	DO1	Bool		
2	DO2	Bool		
<b>02 Input Status(1x)</b>				
Table Display		Show	Hide	
Address	Variable name	Data Type	Description	
32	DI32	Bool		
33	DI33	Bool		
34	DI34	Bool		
<b>03 Holding Registers(4x)</b>				
Table Display		Show	Hide	
Address	Variable name	Data Type	Swap	Description
32	AO32	Short	<input type="checkbox"/>	
33	AO33	Short	<input type="checkbox"/>	
<b>04 Input Registers(3x)</b>				
Table Display		Show	Hide	
Address	Variable name	Data Type	Swap	Description
0	AI0	Short	<input type="checkbox"/>	

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.

**Scaling:**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the "**Advanced Setting**" button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check "**Enable**" box, The Scaling conversion function will be activated.

**PID Operation needs set Scaling  
Click [Show Detail] Setting as:**

**[ 03 Holding Registers(4x) ]**  
**Reference AO32:**  
 Min. 0, Max. 10000  
**Output Scale\_AO32:**  
 Min. 0, Max. 10, Offset 0  
**Check Enable box**

**[ 04 Input Registers(3x) ]**  
**Reference AI0:**  
 Min. -10000, Max. 10000  
**Output Scale\_AI0:**  
 Min. -10, Max. 10, Offset 0  
**Check Enable box**

**→ Click [OK]**

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

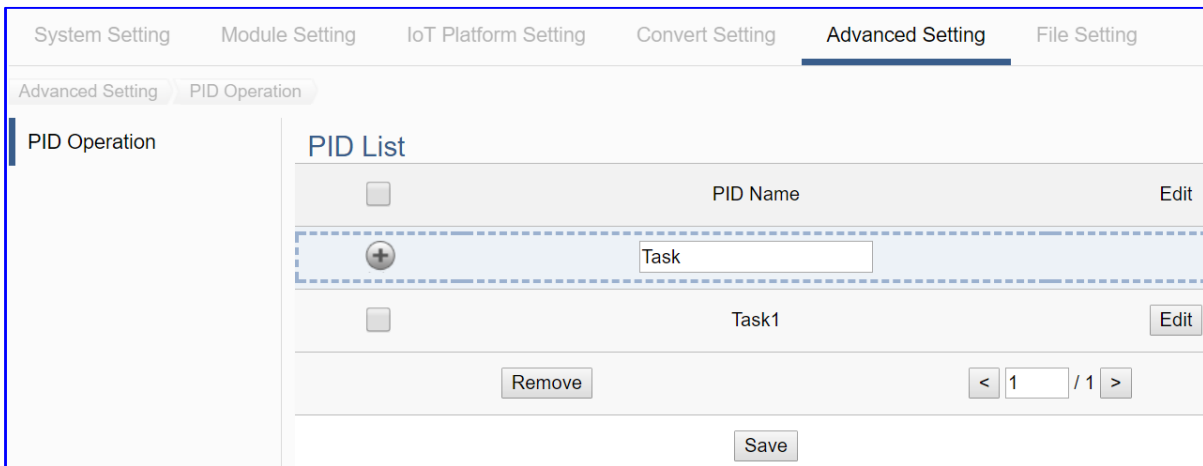
● **Step 3. PID Operation**



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

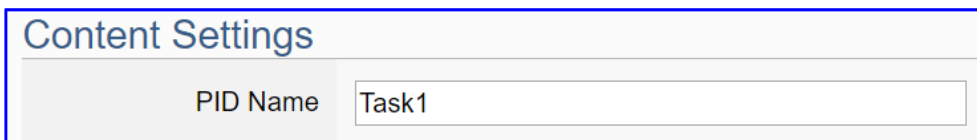
This page is for setting the Task and related parameters of the PID Operation, e.g. I/O module, I/O channels, variables, set point, control mode ....

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 to add a PID Task, and 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.

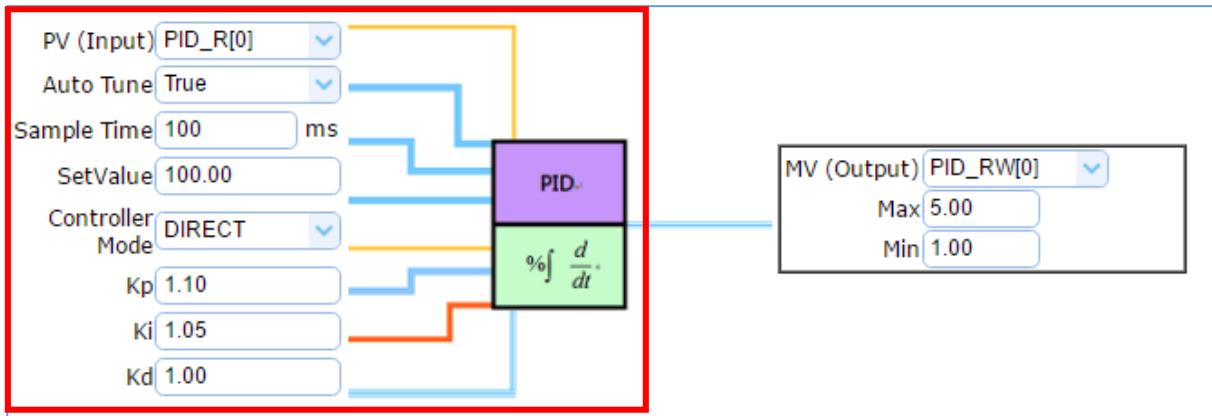
● Parameters Descriptions for Input Item of PID Operation

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. Un-Enable: 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.

● **Setting Example for Input Item of PID Operation**

Suppose the example formula is as shown in the figure below, its input items will be as shown in the left side of the figure:



According to the example formula, the Input Items are set as follows:

Input Item	
	Type : <b>Modbus RTU Scaling (Master)</b>
Module selection	No. : 1
	Name : M-7026-G
	Attribute : Read
Variable selection	Type : <b>32-bit Float</b>
	Name : <b>Scale_AI0</b>
Auto Tune	<input type="checkbox"/> Enabled
Sample Time(ms)	100
Setpoint	100
Controller Mode	DIRECT
Kp	1.1
Ki	1.05
Kd	1

● Parameters Descriptions for Output Item of PID Operation

**Output Item**

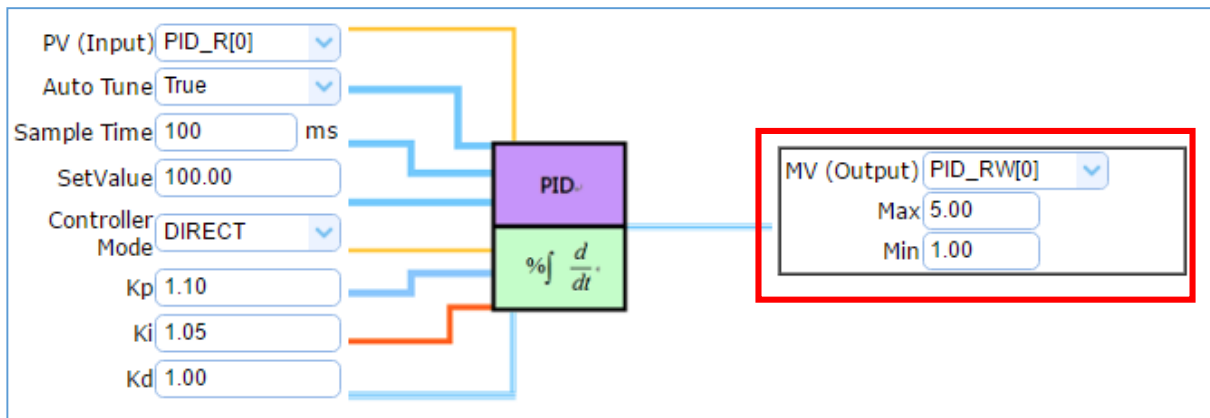
Module selection	Type : <input style="width: 150px;" type="text" value=""/>	Please select the module type.
	No. : <input style="width: 150px;" type="text" value=""/>	Please select the number. When no option is available, add a module.
	Name : <input style="width: 200px;" type="text" value=""/>	
Variable selection	Attribute <input style="width: 150px;" type="text" value=""/>	Please select item.
	Type : <input style="width: 150px;" type="text" value=""/>	Please select item.
	Name : <input style="width: 200px;" type="text" value=""/>	Please select name. When there is no option, add the variables in the module.
Max	<input style="width: 200px;" type="text" value="0"/>	
Min	<input style="width: 200px;" type="text" value="0"/>	

<b>Advanced Setting &gt; PID Operation &gt; Output Item</b>	
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.



● **Setting Example for Output Item of PID Operation**

Suppose the example formula is as shown in the figure below, its output items will be as shown in the right side of the figure:

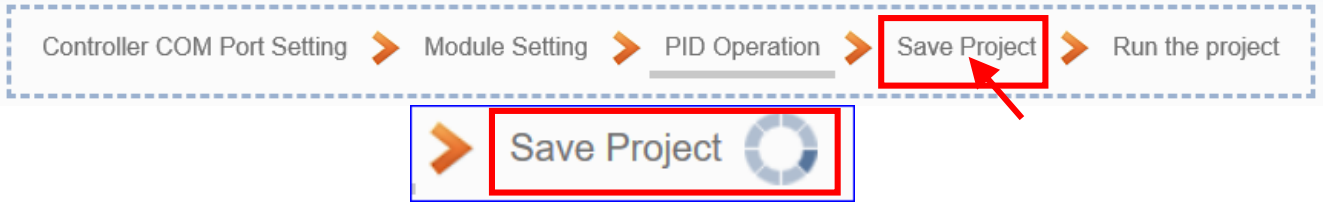


According to the example formula, the Output Items are set as follows:

Output Item	
Module selection	Type : <b>Modbus RTU Scaling (Master)</b>
	No. : 1
	Name : M-7026-G
Variable selection	Attribute : Write
	Type : <b>32-bit Float</b>
	Name : <b>Scale_AO32</b>
	Max : 5
	Min : 1
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

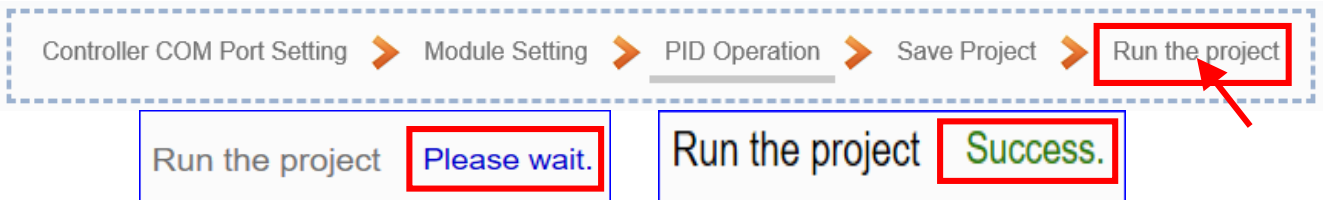
● **Step 4. Save Project**

The setting of this example is finished now. 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**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully. 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 PID function. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.

I/O Status

File Setting

**Modbus RTU Module (Master)**

No.	Name	Serial Port
1	M-7026-G	ttyO2

< 1 / 1 >

**Related Settings**

Number of variables:  (Updated 10 points per second)

Display Update Time (ms):

**Modbus TCP Module (Master)**

No.	Name	LAN
1	DL-302	LAN

< 1 / 1 >

**I/O Status**

Variable Name	Data Type	Value	Description
Scale_AI0	Float	<input type="text" value="0"/>	Input V
Scale_AO32	Float	<input type="text" value="0"/>	Output V

< 1 / 1 >

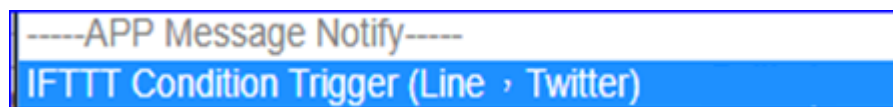
## 4.5 APP Message Notify

The "APP Message Notify" in the UA Function Wizard provides a condition trigger of IFTTT.

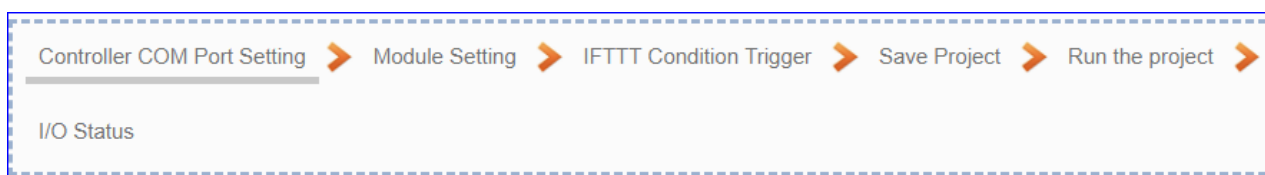
IFTTT (if this then that) is a cloud service platform that easy to get your apps and devices working together via creating chains of simple conditional statements (applets). An applet is triggered by changes that occur within other web services such as Line, Twitter, Gmail, Instagram, etc. For example, "if" Line (Service A) has a new message, "then" send an email to Gmail (Service B). With the IFTTT cloud platform and UA functions, the users can send messages to IFTTT-related cloud services such as Line, Twitter, etc. when the special events occur.



This section introduces the setting steps and the function parameters of the "APP Message Notify" and its item of "IFTTT Condition Trigger (Line, Face, Twitter)" function in the "Function Wizard".



**[Step Box] of [ IFTTT Condition Trigger (Line, Twitter) ] :**



### 4.5.1 Function Wizard: IFTTT Condition Trigger (Line, Twitter) (Use M-7055D)

The “IFTTT Condition Trigger (Line, Twitter)” combines the functions of the UA and IFTTT cloud platform. When the modules occur the special events that setting in the UA condition, it will trigger the IFTTT and send the message to the IFTTT-related cloud services (such as Line, Twitter, etc.)

The settings for sending the message to the APP with the "IFTTT Condition Trigger (Line, Twitter)" function includes two parts:

**1. IFTTT Cloud Platform Setting: (It must be set before setting up the UA project)**

In the IFTTT website, set up the “if” side service and event (**this**: use **webhooks** for the UA), the “then” side service and action (**that**: user can select the service, such as the Line, twitter, etc.). And then fill the “Event Name” and “Key” getting from the IFTTT website setting into the “Content Setting” of the UA We HMI. Please refer **FAQ-005**:

<https://www.icpdas.com/web/product/download/iiot/ua//faq/UA-en-FAQ-005.pdf>

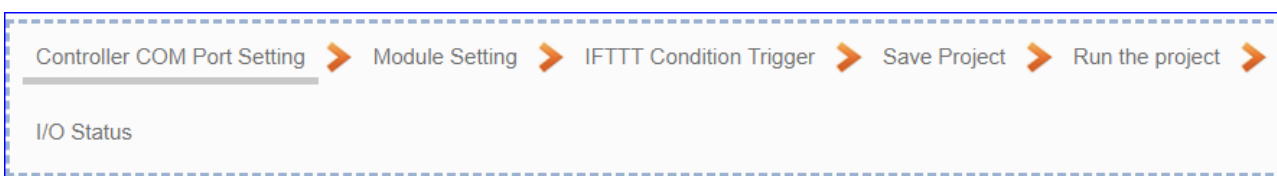


**2. UA Web Interface Setting: (Detail in 5.5.2 Advanced Setting > IFTTT Condition Trigger)**

In the UA Web HMI, set up the UA controller, modules, IFTTT trigger conditions, the condition variable table, and the IFTTT event connection. (Fill the IFTTT **Event Name** and authentication **Key** in the IFTTT step of the project into the "Content Setting" of the UA web interface)

**[Step Box]:**

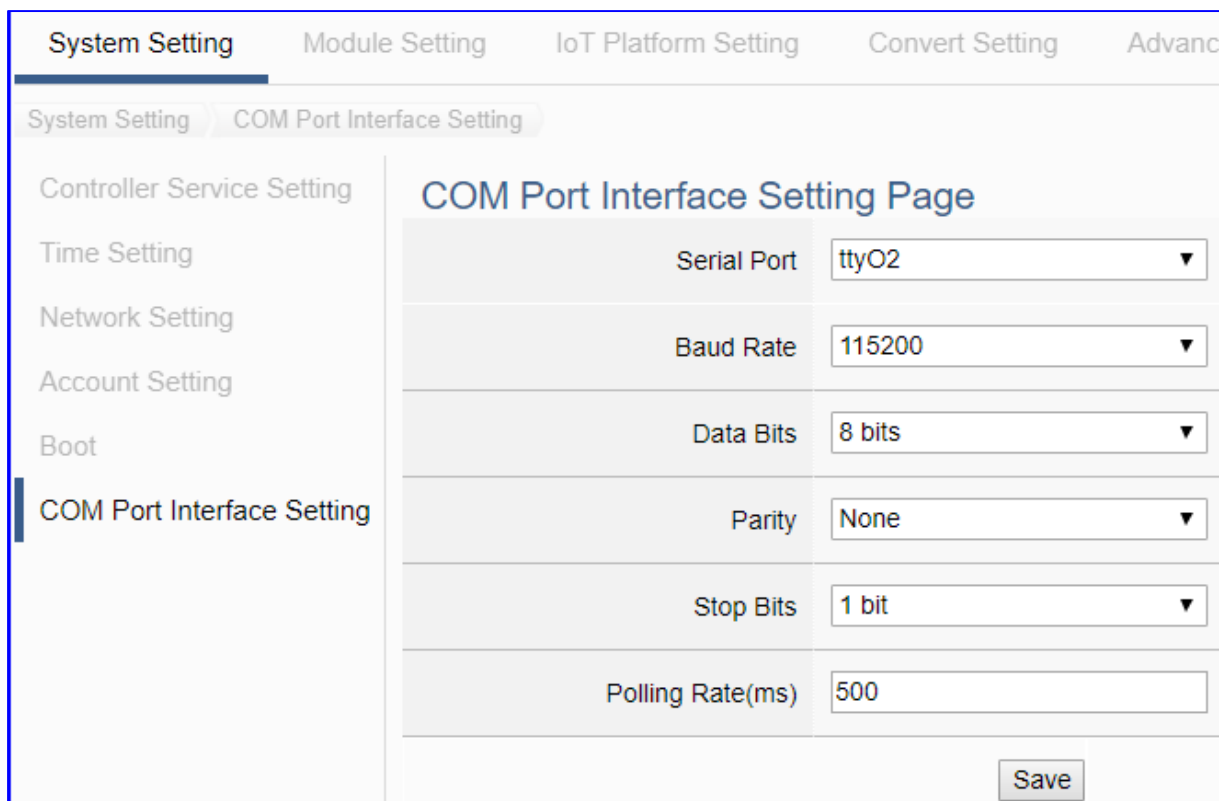
The Step Box of the [IFTTT Condition Trigger (Line, Twitter)] has 6 steps as below. When enabling the Step Box, it auto enters the first step setting page (The step with a bold underline means it is the current step.). The user just needs to follow the “Step Box” step-by-step and then can complete the project.



● **Step 1. Controller COM Port Setting**



This page allows display and set the COM port interface of the controller for the serial communication. The user can find the default communication values of our I/O modules from the module CD, manual or [I/O Module website](#).



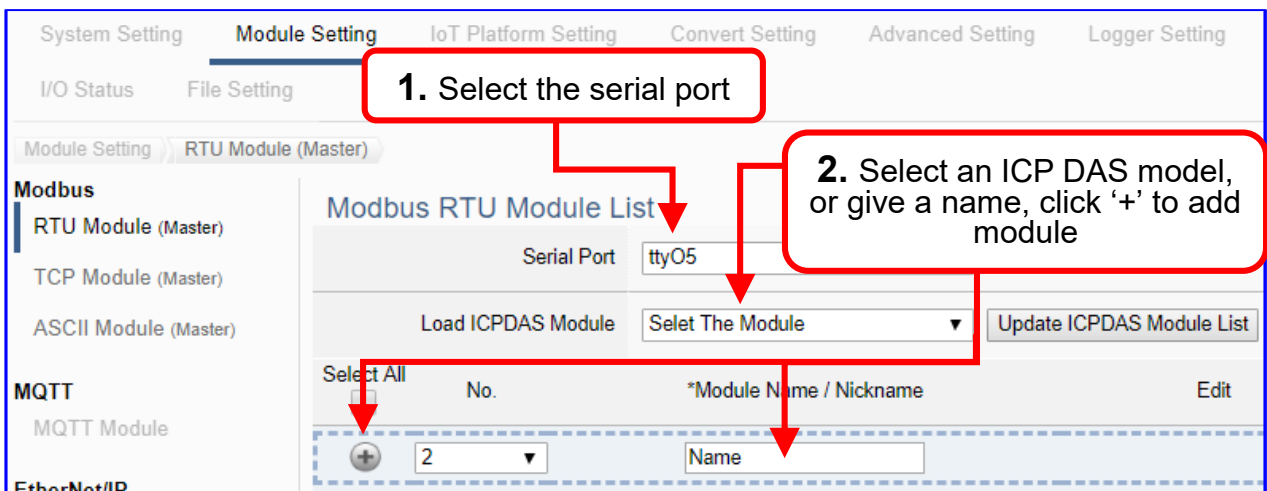
COM Port Interface Setting Page	
Serial Port	Choose the serial port of UA controller that links with the I/O module. ttyO2: RS-485 ; ttyO4: RS-232 ; ttyO5: RS-485
Baud Rate	Choose a baud rate to communicate with the module: 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. The UA controller and 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**

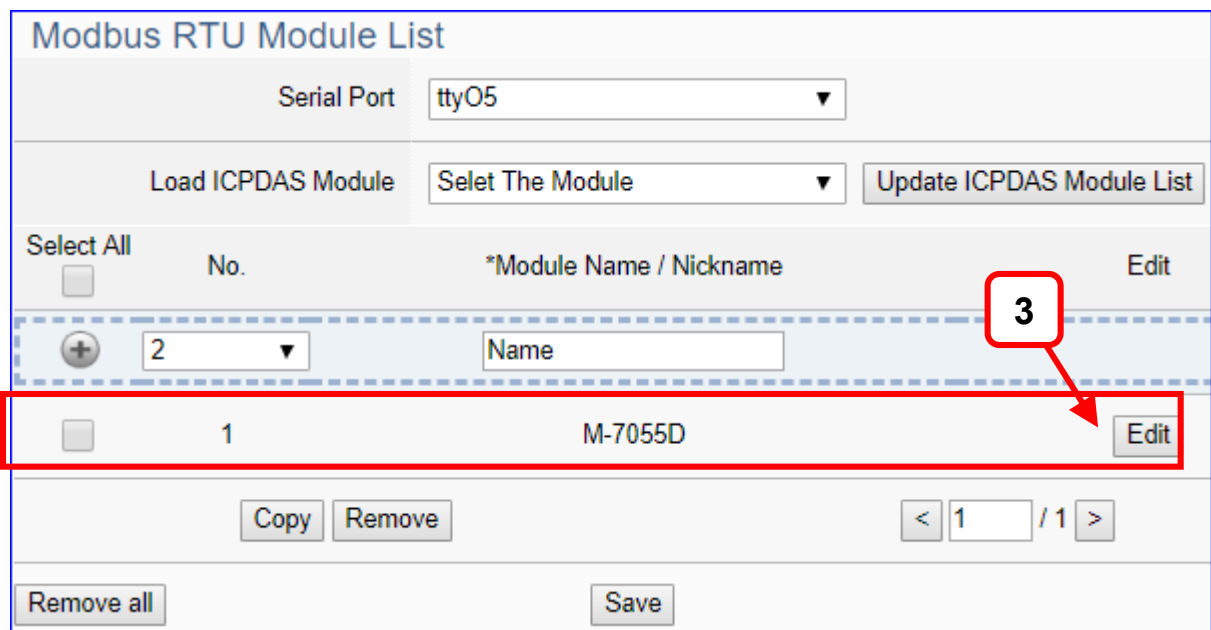


Click the next step, and enter the **Step 2 [Module Setting]** of the UI setting. This page is for setting the communication values with the connected modules.

First, choose the serial port that connected with the module. If use ICP DAS module, select the model to auto load the module setting. If not, give a name (Default: Name), click [ + ] button to add a module.



Add a module (Ex: No.: 1, Name: M-7055D) 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.	1
Module Name	M-7055D
Slave ID	1
Timeout(ms)	500
Modbus Mapping Table Setting	
Data Model	01 Coil Status(0x)
Start Address	0
Data Number	1
Create Tables	<input type="button" value="Add"/>

If use ICP DAS module, system can auto setup Modbus Mapping Table; if not, user needs to check Modbus address or I/O number from the module user manual.

**> Modbus Mapping Table Setting:**

Set up in the order of Data Model, Start Address and Data Number, then click "Add".

**Ex:** M-7055D has 8 Data Models of "01 Coil Status (0x)" (Mapping: DO), so select Model "01", Start Add. "0", Number "8", and click "Add".

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

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. (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)
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address:**

Display and edit the Modbus Mapping Table.

If user selects ICP DAS module, the system will auto set up the Modbus Mapping Table. If not, user needs to check the module Modbus address or I/O number from the module user manual.

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.



**Nickname:**

Setting the variable nickname and description.

Modbus Mapping Table				Address	Nickname	Scaling	Bitwise
<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="DO0"/>	Bool	<input type="text"/>				
1	<input type="text" value="DO1"/>	Bool	<input type="text"/>				
2	<input type="text" value="DO2"/>	Bool	<input type="text"/>				
3	<input type="text" value="DO3"/>	Bool	<input type="text"/>				
4	<input type="text" value="DO4"/>	Bool	<input type="text"/>				
5	<input type="text" value="DO5"/>	Bool	<input type="text"/>				
6	<input type="text" value="DO6"/>	Bool	<input type="text"/>				
7	<input type="text" value="DO7"/>	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="DI0"/>	Bool	<input type="text"/>				

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.

**Scaling:**

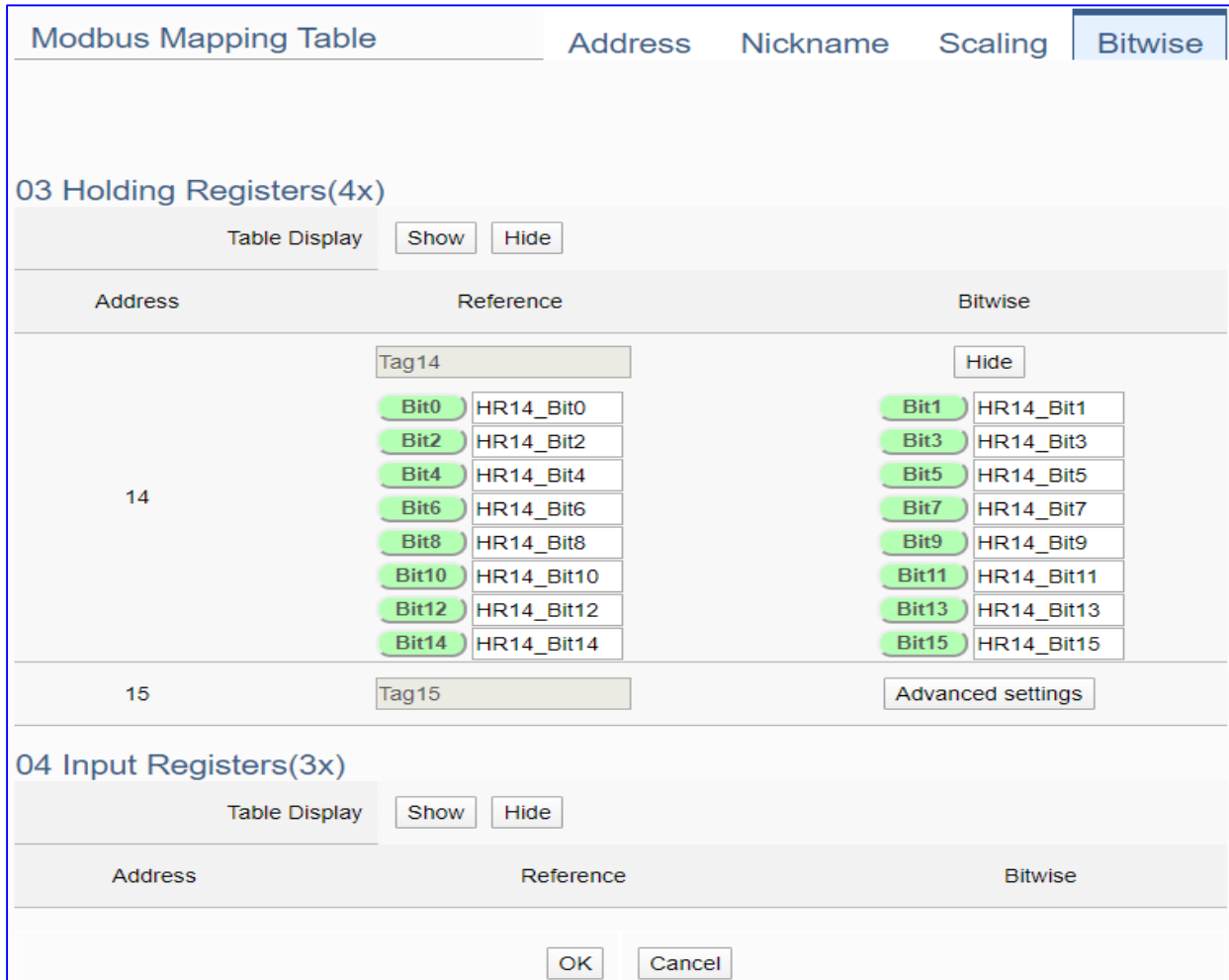
**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated. The M-7055D has no AI/AO, so here uses the screen of DL-302 for an example.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

**Bitwise:**

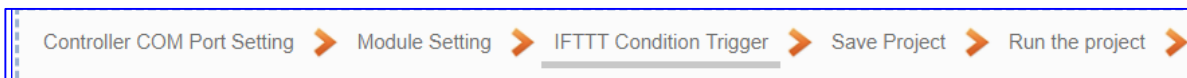
**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

The M-7055D has no AI/AO, so here uses other module’s setting screen as an example.



Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

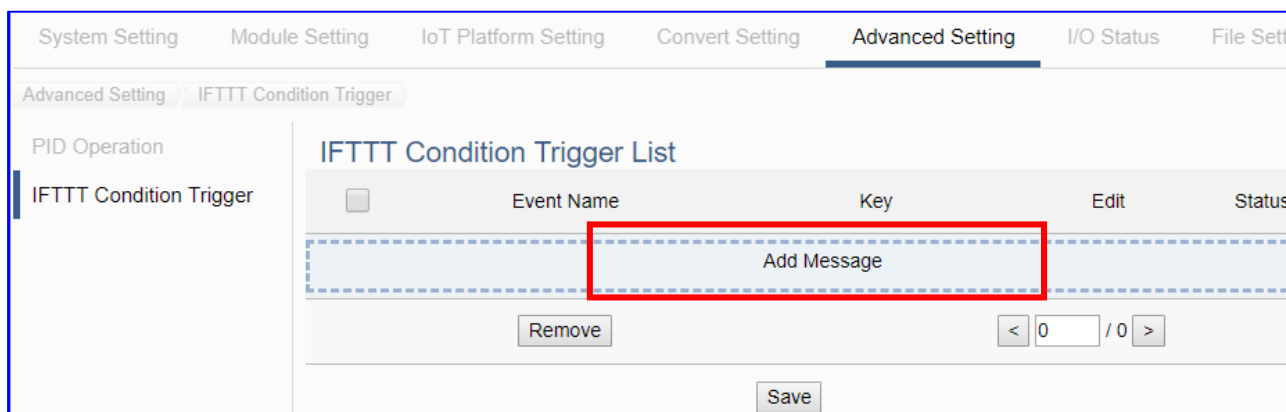
● **Step 3. IFTTT Condition Trigger**



Click the next step, and enter the **Step 3 [IFTTT Condition Trigger ]**.

This page is for the APP message related setting, e.g. IFTTT event name, key, trigger condition, I/O variables ....

We select the “**IFTTT Condition Trigger (Line, Twitter)**” at the beginning, so this step will auto enter the setting page [**Advanced Setting > IFTTT Condition Trigger**]. The “Step Box” will prevent the user from selecting the wrong platform.



Advanced Setting > IFTTT Condition Trigger > FTTT Condition Trigger List	
Add Message	Click to add a new IFTTT message. After setting, an IFTTT condition trigger list will show on the bottom includes left box, event name, key and status.
<input type="checkbox"/>	Check the box in the left of the list is to select and to delete the list. Check the box on the top will select all lists.
Event Name	Display the “Event Name” setting in the IFTTT website. ( <a href="#">FAQ-005</a> )
Key	Display the “Key” getting from the IFTTT website. ( <a href="#">FAQ-005</a> )
Edit	Click [Edit] can set the IFTTT condition trigger content.
Status	Display the enable status of the IFTTT condition trigger list.
Remove	Click the left box and [remove] can delete the IFTTT list.
<input type="text" value="1"/> / <input type="text" value="1"/>	The page number of the IFTTT 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 [Add Message] button to enter the IFTTT [Content Settings] page:

Content Setting	
Event Name	UA-5200 test
Key	fkCGvasDPR-xYe2ugpgQ7
Status	<input checked="" type="checkbox"/> Enabled

**Note: Case sensitive for Event Name and Key.**

**Note:** The “Event Name” and “Key” are set in the IFTTT website. If you are not familiar with IFTTT, please see the [FAQ-005](#) for the setting introductions.

Advanced Setting > IFTTT Condition Trigger > Content Setting	
Event Name	Input the “Event Name” setting in the IFTTT website. ( <a href="#">FAQ-005</a> )
Key	Input the “Key” getting from the IFTTT website. ( <a href="#">FAQ-005</a> )
Status	Check to enable the IFTTT condition trigger event.

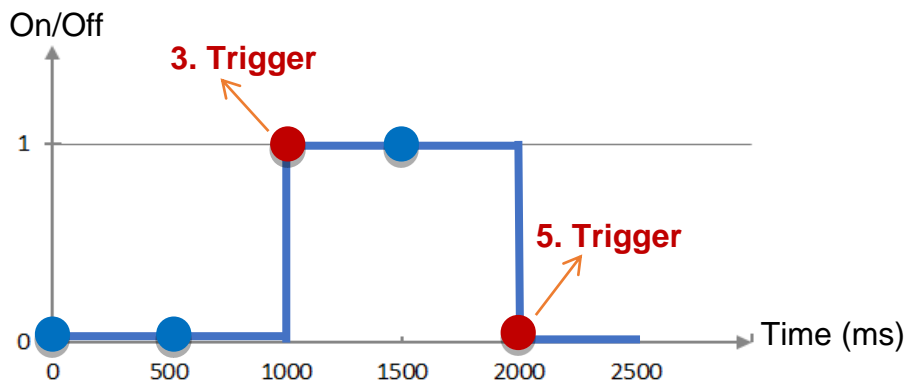
Condition Setting		
Module Variables	Operator	Value
↓ Module Type Modbus RTU (Master)		
↓ Module Name No.1 M-7		Type : User-Defined
↓ Variable Attribute Read	=	Dead Band : 1
↓ Variable Name Tag0 (Short)		
<input type="button" value="Add"/>		

Advanced Setting > IFTTT Condition Trigger > Condition Setting	
Module Variables	Select the module and variable for the condition trigger. Module Type: select the module type, Modbus RTU/TCP/ASCII... Module Name: select the module that set for condition trigger. Variable Attribute: select the variable attribute for condition trigger. Variable Name: select the variable name for condition trigger.
The following condition fields may different depending on the selected variable attribute. The condition trigger method will be described after this table.	
Operator	Select the operator for the trigger condition.
Value	Set up the value for the condition, include Type and Dead Band.
Status	Set up the status for the condition. Default: 0.
Add	Click to add a condition trigger list in the Condition Table..

## Condition Trigger Descriptions:

The condition trigger method will differ depending on the attribute of the selected variable and the trigger will be different. There are two operation styles: **DIO** and **AIO**.

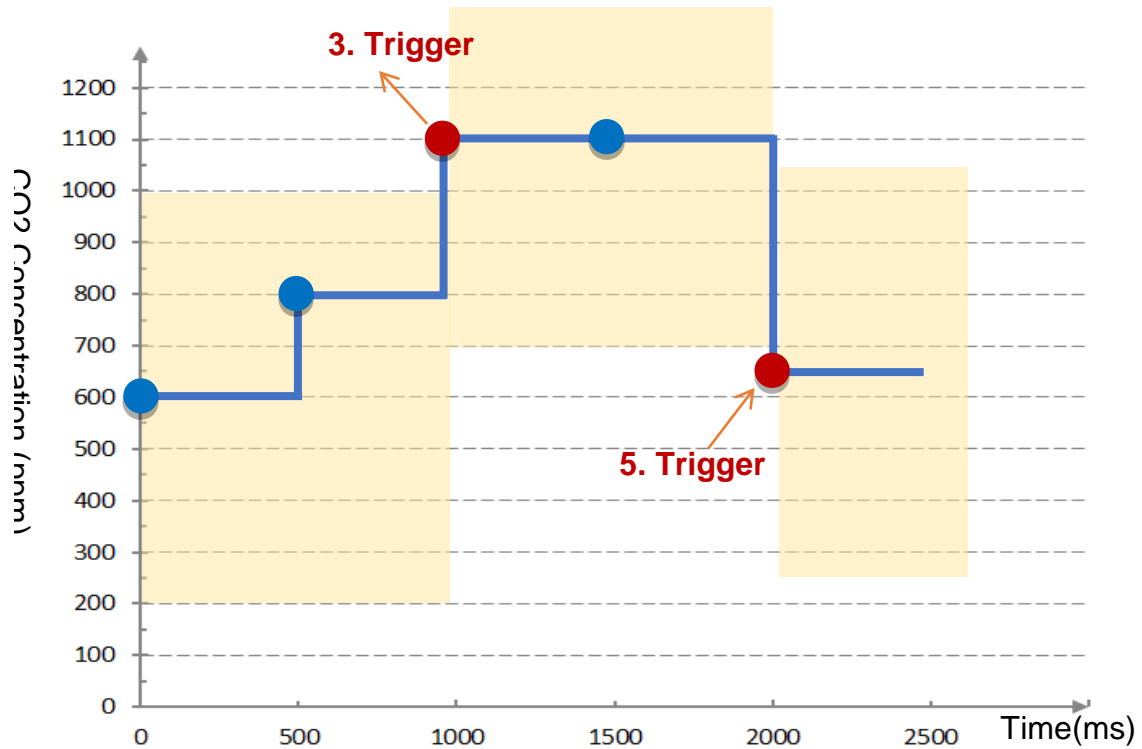
**(A)** If select **DIO variable**, then Condition is "Status Change". When detecting the status is changed, it will trigger the event and send the assigned message. (Below is a switch detecting example.)



**DIO Trigger:** (Detect per 500 ms)

1. Detect initial switch status "Off" (status = 0)
2. Detect "Off" (status = 0, status no change), no trigger
3. Detect "On" (status = 1, status changed), **trigger** a message notification
4. Detect "On" (status = 1, status no change), no trigger
5. Detect "Off" (status = 0, status changed), **trigger** a message notification

(B) If select **AIO variable**, then Condition is “Value” and can set the “Dead Band”. The condition will be triggered and send the message when the detected value exceeds the upper or lower Dead Band. (Below is a CO2 example. Detect per 500 ms)



**AIO Trigger:** (Detect per 500 ms. The yellow block means the Dead Band.)

1. Detect initial CO2 concentration 600 (ppm).  
Set Dead Band=400 (Initial Trigger Condition:  $\geq 1000$  or  $\leq 200$ )
2. Detect CO2 concentration 800. It is in the range of Dead Band.
3. Detect CO2 concentration 1100. It exceeds the upper value ( $\geq 1000$ ) of Dead Band, so **trigger** a message for danger notification.
4. Detect CO2 concentration 1100. It is in the new range of Dead Band.  
Dead Band=400 (New Trigger Condition:  $\geq 1500$  or  $\leq 700$ )
5. Detect CO2 concentration 650. It is below the lower value ( $\leq 700$ ) of Dead Band, so **trigger** a message for safety notification.

Please refer to the previous Condition Trigger Descriptions to set up your Condition. When complete, click the “Add” button. The setting will show in the Condition Table. Below Table is setting 2 conditions.

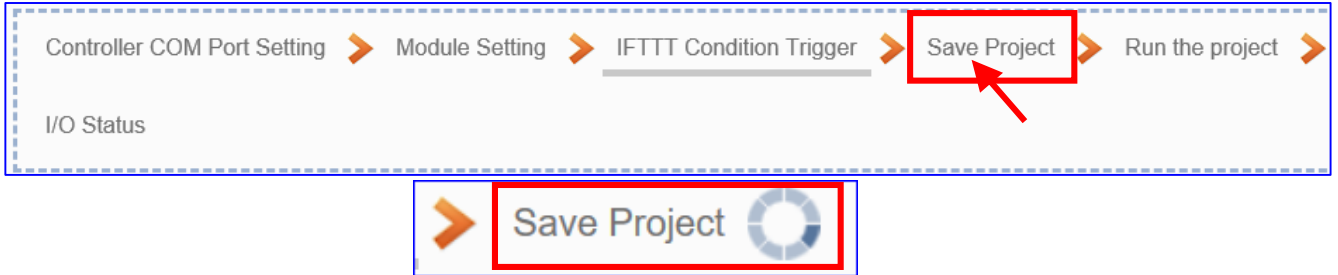
Advanced Setting > IFTTT Condition Trigger > Condition Table	
Module	Display the module type and name of the condition. (Not editable here)
Variable	Display the variable attribute and name of the condition. (Not editable here)
Condition	Display the trigger condition. (Not editable here)
Define Message	Default Message: module code_variable code. The user can define own message in the format of English character, number, general symbol...
Remove	Click the left box and [remove] can delete the IFTTT list.
OK	Click to save this page settings and back to the module list page.
Cancel	Click to exit without saving and back to the module list page.

When back to the IFTTT Condition Trigger List, the condition trigger message will show as below picture. If need more trigger conditions, click the “Add Message” again to combine the IFTTT APP message sending and the UA system. At last, click the Save button.



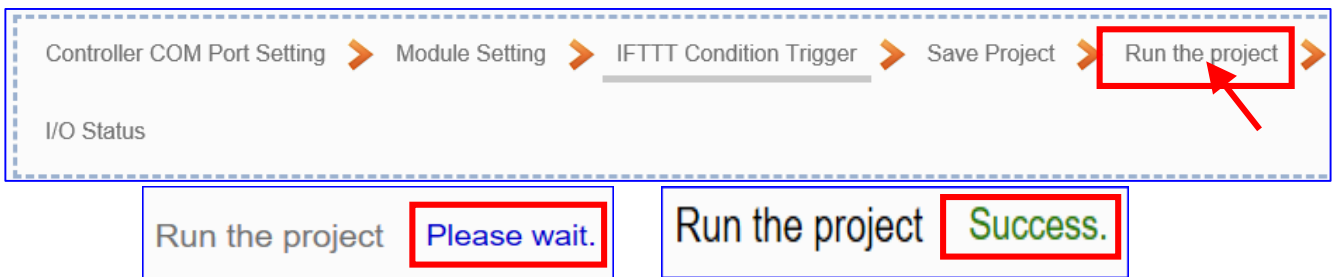
● **Step 4. Save Project**

The setting of this example is finished now. 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**]. This step can also via the [**System Setting > Controller Service Setting > Run Project**] to Stop and Run the project.



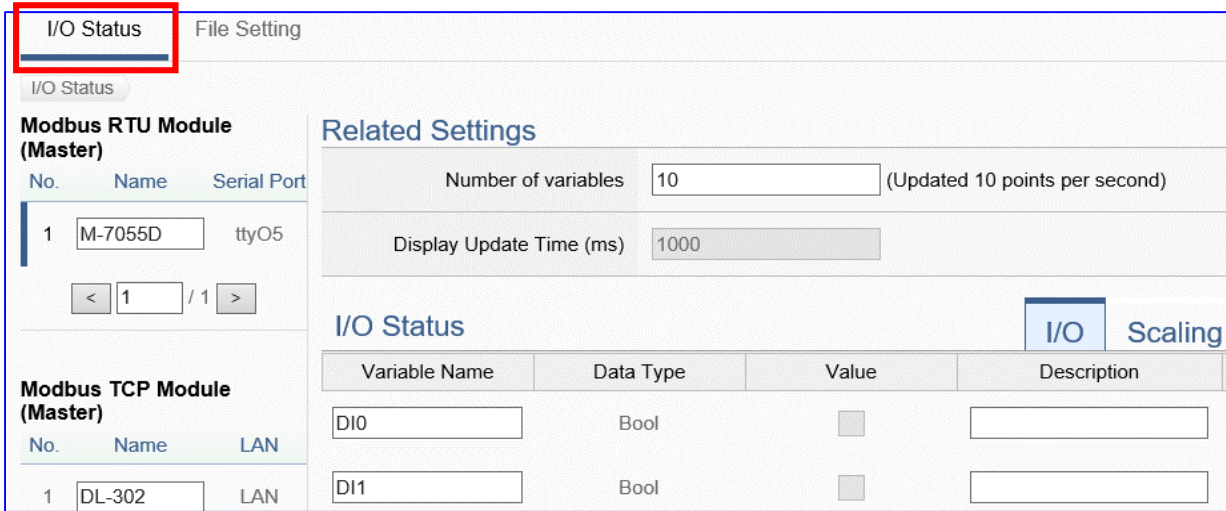
When the words “**Please wait**” disappears, the new words “**Success**” appears, that means the UA controller is running new project successfully.

● **Step 6. I/O Status**



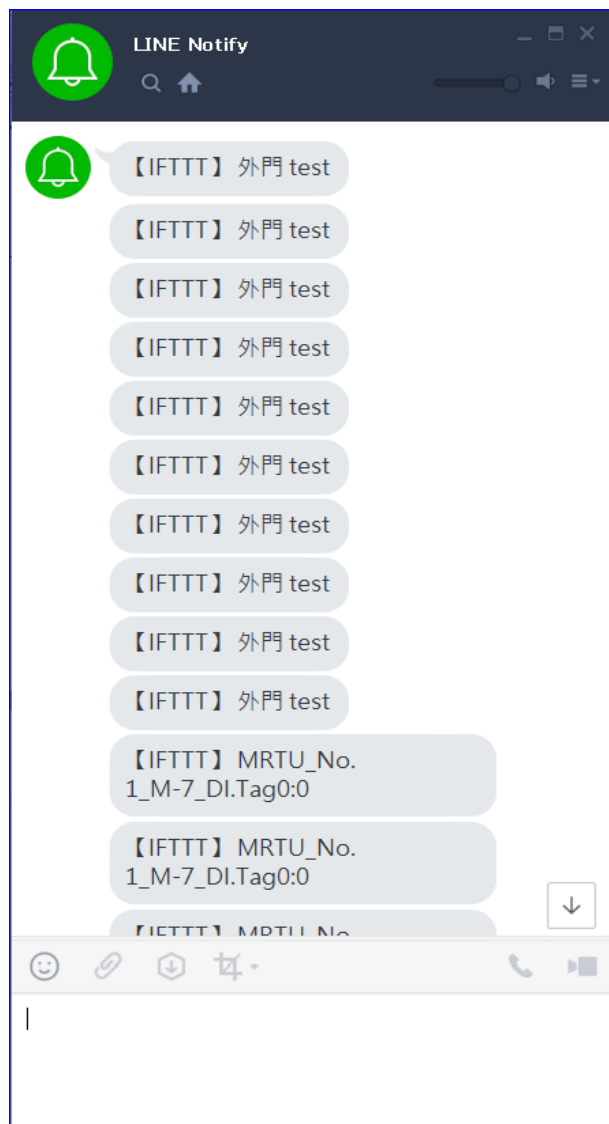
This step will go to the Main Menu [I/O Status]. The users can click the setup module to see its real time I/O status.

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 PID function. Users can see the I/O status from the menu [**I/O Status**]. For more about the Web UI settings, please refer to CH4 and CH5.

The project for APP message notifies via the IFTTT condition trigger (Line, Twitter) is now done.



# 5 Main Menu: Parameter Settings

## 5.1 Main Menu: System Setting

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

[System Setting] includes several sub-menu functions (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	
I/O Status File Setting	
System Setting	
Controller Service Setting	
Time Setting	
Network Setting	
Account Setting	
Boot	
COM Port Interface Setting	
Version Information	
Firmware Version	Version 1.3.2.0
Main Program	Version 1.1.60
Web Interface	Version : 6.5.21 Date : 2020/12/22
Install Information	2020/12/30-11:21:41_Factory_InstallSuccess
Flash Information NAND Flash	
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 for UA series controllers is to set up from the left to the right of the main menu functions. User can find the setting step and Web UI information in the following chapters.

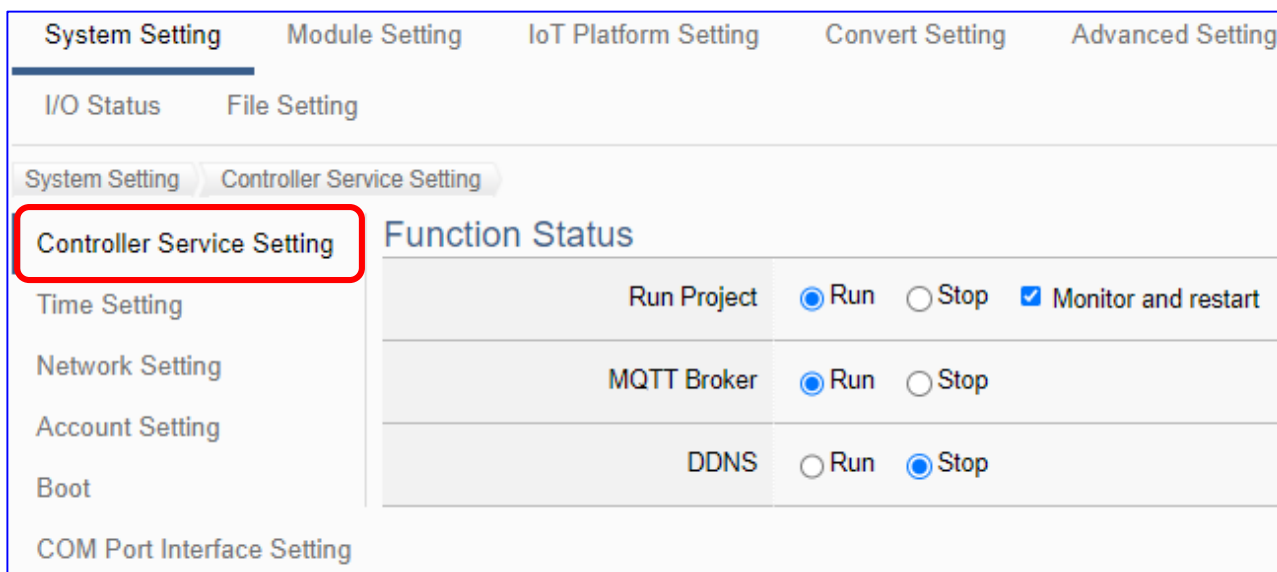
[CH2 Quick Start 1: Hardware/Network Connection](#)

[CH3 Quick Start 2: Web UI / Setting Steps](#)

[CH4 Function Wizard: Project Quick Setup & Example](#)

### 5.1.1 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.

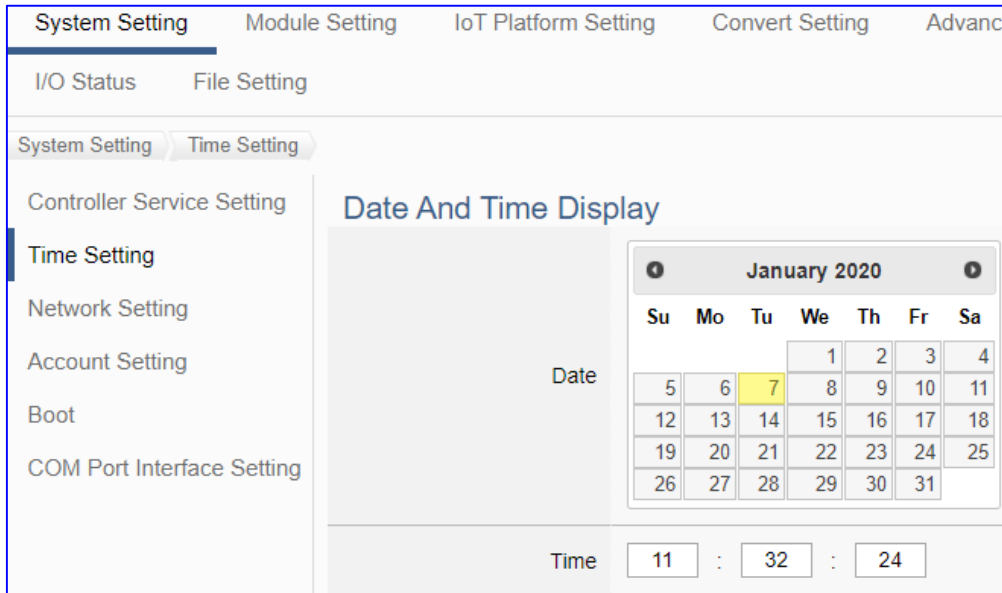


System Setting > Controller Service Setting > Functional status	
Run Project	Display the current status of project running in the UA series controller and provide “Run” and “Stop” button to switch the status. Default: Run. <b>Monitor and restart:</b> When check this function, if the project is stopped (including abnormal or manual stop), it will restart within 30 seconds.
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.

## 5.1.2 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.

### NTP Time Synchronization Setting

Functional Status	<input checked="" type="radio"/> NTP Time Server <input type="radio"/> Customize The Server
NTP Time Server	<input type="text" value="time1.google.com"/>
Time Zone	<input type="text" value="Taipei"/>

System Setting > Time Setting > NTP Time Synchronization Setting	
Functional Status / NTP Time Server	Set up one NTP Time Server from the google (4), windows 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.

### 5.1.3 Network Setting

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

**NOTE:**

- 3. UA-5200 series has 1 LAN (LAN1); UA-2200 series has 2 LANs (LAN1 / LAN2).
- 4. UA-2200 series supports static IP only, so it needs to specify an IP address for each LAN1 and LAN2.
- 5. UA-2200 series uses LAN1 to connect PC. LAN1 and LAN2 must set to different domains. Ex: Set LAN1 IP to “192.168.84.80”, and set LAN2 to a different domain, such as, “10.0.0.1”.

Network Setting (LAN1)				
Connection Mode	<input checked="" type="radio"/> Specify an IP address			
IP	192	.	168	.
	84	.	80	
Mask	255	.	255	.
	0	.	0	
Gateway	192	.	168	.
	1	.	1	
<input type="button" value="Save"/>				
Network Setting (LAN2)				
Connection Mode	<input checked="" type="radio"/> Specify an IP address			
IP	10	.	0	.
	0	.	0	.
	1			
Mask	255	.	255	.
	0	.	0	
Gateway	10	.	168	.
	1	.	1	
<input type="button" value="Save"/>				

<b>System Setting &gt; Network Setting &gt; Network Setting (LAN1)</b>	
Connection Mode	<b>Specify an IP address:</b> 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> .
IP	The LAN1 IP address of this UA. Factory Default: 192.168.255.1
Mask	The LAN1 mask address of this UA. Factory Default: 255.255.0.0
Gateway	The LAN1 gateway address of this UA. Factory Default: 192.168.1.1
Save	Click to save the settings of LAN1 item.
<b>System Setting &gt; Network Setting &gt; Network Setting (LAN2)</b>	
Connection Mode	<b>Specify an IP address:</b> 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> .
IP	The LAN2 IP address of this UA. Factory Default: 10.0.0.1
Mask	The LAN2 mask address of this UA. Factory Default: 255.255.0.0
Gateway	The LAN2 gateway address of this UA. Factory Default: 10.168.1.1
Save	Click to save the settings of LAN2 item.



### Hostname Setting

Hostname	<input style="width: 90%;" type="text" value="UA-5231-68C90BE4E5A5"/>
----------	---

<b>System Setting &gt; Network Setting &gt; Hostname Setting</b>	
Hostname	The host name of this UA. Default: system value. User can give a new name, but cannot be null.
Save	Click to save the settings of this item.

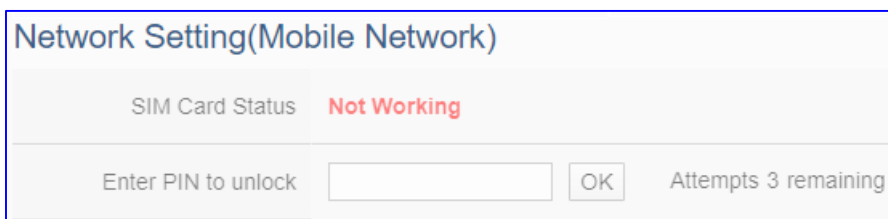
### Dynamic DNS Setting

Service Provider	<input style="border-bottom: 1px solid black;" 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, ChangelP.com, DynDNS, FreeDNS.
*Username	Set up the login user name. The star * means the field cannot be null. Default: undefined.
*Password	Set up the login password. The star * means the field cannot be null.
*Domain Name	Define the parked domain name of the DDNS. The star * means the field cannot be null. Default: undefined.
Save	Click to save the settings of this item.

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

● **\*\* Network Setting (Mobile Network):**



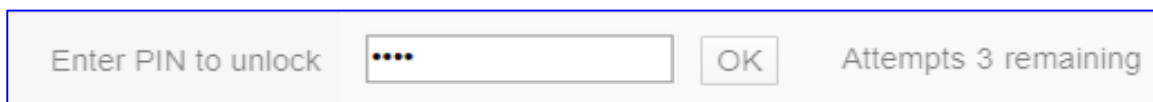
**Setting Steps & Notice:**

1. This “**Mobile Network**” setting items appears only on the **mobile model** of UA controller (-**4GE/4GC/3GWA**).
2. **Insert the SIM card** into the **SIM card socket** of UA, and then **power on or restart** the UA. Login the web UI again. When the “**SIM Card Status**” item appears “**Working**”, the network connection is successful. Users can check the **3G/4G LED Status** of UA Series to know the module status.



LED	LED Status	Module Status
3G	Green: ON	The modem is functioning normally, and the SIM card inside.
	OFF	The modem is not functioning, no SIM card inside, or the PIN is still locked.
4G	Green Light Flash: ON for 2 seconds, OFF for 1 sec, and flashing alternately.	The modem is functioning normally, and the SIM card inside.
	Green Light Flash: ON for 1 second, OFF for 2 secs, flashing alternately, or OFF .	The modem is not functioning, no SIM card inside, or the PIN is still locked.
	Green Light Flash: Quickly	Data is transmitting.


3. **Enter the PIN number** to unlock the SIM card:

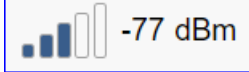


\* The SIM card of a telecom company is usually locked and protected by a PIN number. Please **unlock the PIN before setting the Mobile Network function**.

\* The PIN access will be locked out after **three fault attempts**. The user needs to **contact the telecom company of the SIM card to unlock it before proceeding**.

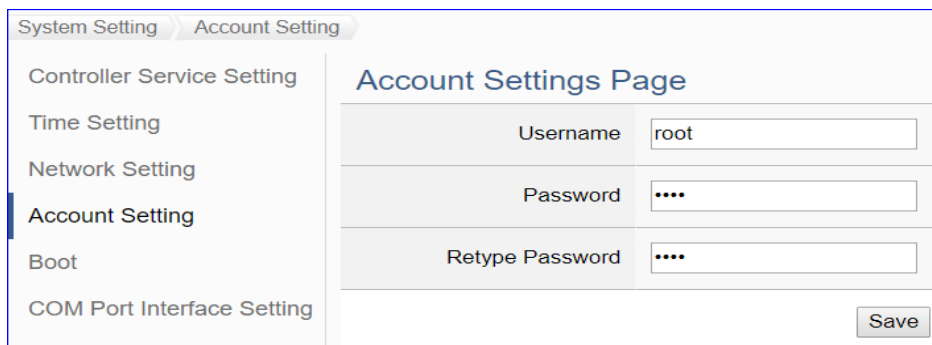
4. Set up the network settings (see next page), check the "Enable" items, and click the "Save and Connect" button. When the "Connection Status" becomes "Connected", the connection is successful.
5. When the mobile network is enabled, the mobile network will serve as the main route to the external network.

Signal	 -65 dBm
Connection Status	Connected
IP	10.238.52.120
Automatic Connection When Power On	<input checked="" type="checkbox"/> Enable
Dial-up Number	<input type="text" value="*99#"/>
APN	<input type="text" value="INTERNET"/>
Authentication	Username <input type="text" value="guest"/> Password <input type="password" value="....."/> Please refer to <a href="#">this document</a> to configure the setting.
Mobile Code	<input checked="" type="checkbox"/> Enable MCC <input type="text" value="466"/> MNC <input type="text" value="92"/> Please refer to <a href="#">this document</a> to configure the setting.
<input type="button" value="Disconnect"/> <input type="button" value="Save and Connection"/>	

System Setting > Network Setting > Network Setting(Mobile Network)	
Signal	Display the strength level of the Mobile Network Signal. Signal strength range: (The strongest signal) -51 ~ -113 (No signal)  -77 dBm
Connection Status	Display the connection status: Connected or Disconnected.
IP	Display the IP address that UA occupies through Mobile Network.
Automatic Connection When Power On	If check the “Enable” box, it will enable the UA controller to complete the Mobile Network connection automatically when power on UA controller.
Dial-up Number	Default: *99#. User can change it by the proprietary number provided by the Telecommunication Service Company.
APN	Please refer to “this document” provided on the UA Web HMI page to configure the setting.
Authentication	Please refer to “this document” provided on the UA Web HMI page to configure the setting.
Mobile Code	It is an optional setting. It depends on the service the Telecommunication Service Company provides
Disconnect	Click button can disconnect the Mobile Network connection.
Save and Connection	Click button to save the setting and start to connect the Mobile Network.

### 5.1.4 Account Setting

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



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

**After first login, change your password as soon as possible for your system safety.**

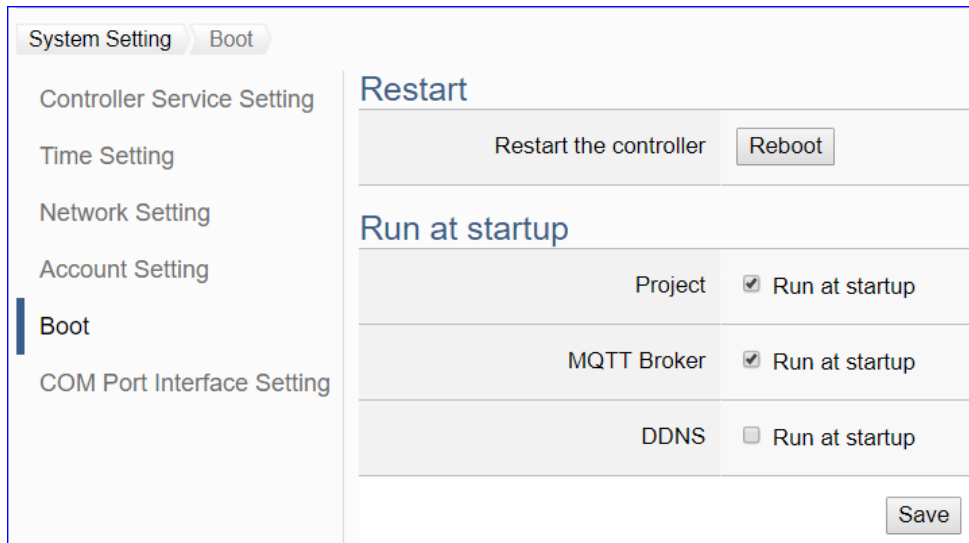
#### Steps to change the username and password:

1. Type the new username in the “Username” item.
2. Type the new password in the “Password” item.
3. Retype the new password in the “Retype Password” item.
4. Click the “Save”. Then re-login the UA Web UI with the new username and password.

Factory Default Settings of UA Series			
Network	IP (LAN1)	192.168.255.1	Assign UA a new IP address according to your case. For UA-2200 series, set the LAN1 for the connection.
	IP (LAN2)	10.0.0.1	
	Netmask	255.255.0.0	
	Gateway	LAN1: 192.168.1.1 LAN2: 10.168.1.1	
Web UI Account	Username	root	<b>After login, change your password as soon as possible.</b> (Section 5.1.4 for Web UI)
	Password	root	

### 5.1.5 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.



<b>System Setting &gt; Boot &gt; Restart</b>	
Restart the controller	Click "Reboot" can restart the UA controller at once.
<b>System Setting &gt; Boot &gt; Run at startup</b>	
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.

### 5.1.6 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.

## 5.2 Main Menu: 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 (RTU/TCP/ASCII module), remote MQTT and remote EtherNet/IP 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), ASCII Module (Master), MQTT and ICP DAS EIP Module. The Module Setting will support more modules in the future. This chapter will introduce the current function items and setting parameters.

Module Setting	
<b>Modbus</b>	
RTU Module (Master)	This setting is for connecting the remote Modbus RTU Slave module.
TCP Module (Master)	This setting is for connecting the remote Modbus TCP Slave module.
ASCII Module (Master)	This setting is for connecting the remote Modbus ASCII Slave module.
<b>MQTT</b>	
MQTT Module	This setting is for connecting the remote MQTT module.
<b>EtherNet/IP</b>	
ICPDAS Module	This setting is for connecting the remote ICPDAS EIP module.
<b>Internal</b>	
Internal Model	The internal module can create virtual internal variables for reading and writing or as an intermediary to provide data exchange of communication protocols.

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

[CH2 Quick Start 1: Hardware/Network Connection](#)

[CH3 Quick Start 2: Web UI / Setting Steps](#)

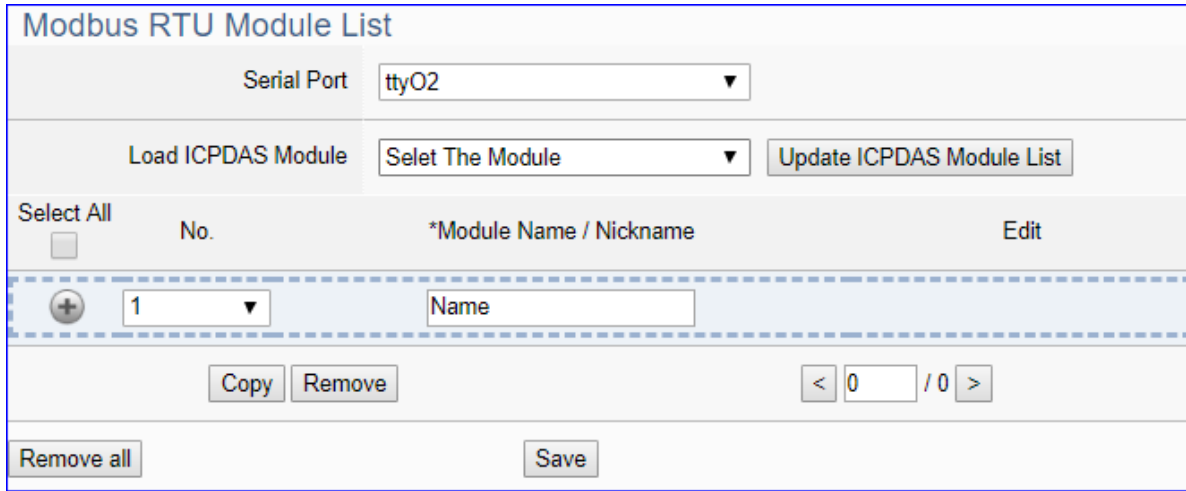
[CH4 Function Wizard: Project Quick Setup & Example](#)

Recommend to use ICP DAS module, system will auto set up the Modbus Mapping Table. The user can check the module Modbus address or I/O number from the user manual.

Website: <https://www.icpdas.com/en/product/p02.php?root=537>

### 5.2.1 Modbus RTU (Master)

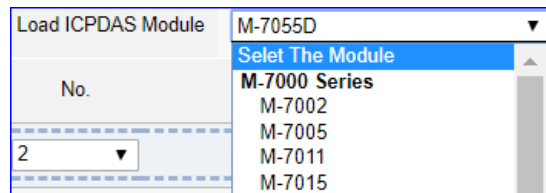
This setting is for UA Controller connecting the remote Modbus RTU Slave module.



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). Click [ + ] button could add a new module. If using ICP DAS module, user just need to select the model number, system will auto add and setup the module. Click [Edit] button can configure the module content and the Modbus mapping table.

**Setting Steps:**

1. Select the module Serial port
2. a) Select an ICP DAS model (as the pic), system will auto load/setup the module.  
 b) Give the module name or nickname, e.g. M-7055D. Click [ + ] to add a new module
3. Click the button [Edit] to enter the Module Content Setting page  
 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]:



Module Setting > Modbus - RTU Module (Master) > Modbus RTU 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.	
Load ICPDAS Module	Select the ICP DAS module number, system will auto add and setup the module. Click "Update ICPDAS Module List" can update new list.	
	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 nickname. User can give a new name. (The star * means this field cannot 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.	
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] button to enter the "Module Content Setting" page.

Enter the **Module Content Setting** page.

Module Content Setting	
No.	<input type="text" value="3"/>
Module Name	<input type="text" value="M-7055D"/>
Slave ID	<input type="text" value="2"/>
Timeout(ms)	<input type="text" value="500"/>
Write Retry	<input type="checkbox"/> <input type="text" value="1"/>
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"/>

If select ICP DAS module, system will auto set up the Modbus Mapping Table, or user needs to check the module Modbus address or I/O number from the module user manual.

> **Modbus Mapping Table Setting:**  
Set module in the order of Data Model, Start Address and Data Number, then click "Add".

**Ex:** M-7055D has 8 Data Models of "01 Coil Status (0x)" (Mapping: DO), so select Model "01", Start Add. "0", Number "8", and click "Add".

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

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. (Range: 1 ~ 247)
Timeout	Set the timeout value for the module. Default: 500 ms
Write Retry	Check to retry writing again when there is no response after the set time is up, and it can be set up to retry 3 times
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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address Setting:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)	
Coil Status(0x)		Input Status(1x)		Holding Registers(4x)	
Input Registers(3x)		Input Registers(3x)		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"/>			

If user selects ICP DAS module, the system will auto set up the Modbus Mapping Table. If not, user needs to check the module Modbus address or I/O number from the module user manual.

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

### Nickname Setting:

Setting the variable nickname and description.

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 type="text"/>
1	<input type="text" value="Tag1"/>	Bool	<input type="text"/>

02 Input Status(1x)

Table Display

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

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.

● **\*\* Scaling**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

● **\*\* Bitwise**

**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

Modbus Mapping Table
Address
Nickname
Scaling
Bitwise

**03 Holding Registers(4x)**

Table Display
Show
Hide

Address	Reference	Bitwise
14	<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">Tag14</div> <div style="display: flex; flex-wrap: wrap;"> <div style="width: 50%;"> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit0</span> HR14_Bit0</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit2</span> HR14_Bit2</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit4</span> HR14_Bit4</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit6</span> HR14_Bit6</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit8</span> HR14_Bit8</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit10</span> HR14_Bit10</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit12</span> HR14_Bit12</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit14</span> HR14_Bit14</div> </div> <div style="width: 50%;"> <div style="margin-bottom: 5px; text-align: right;"><span style="border: 1px solid #ccc; padding: 2px;">Hide</span></div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit1</span> HR14_Bit1</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit3</span> HR14_Bit3</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit5</span> HR14_Bit5</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit7</span> HR14_Bit7</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit9</span> HR14_Bit9</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit11</span> HR14_Bit11</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit13</span> HR14_Bit13</div> <div style="margin-bottom: 2px;"><span style="background-color: #90EE90; border: 1px solid #ccc; padding: 2px;">Bit15</span> HR14_Bit15</div> </div> </div>	
15	<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">Tag15</div>	<span style="border: 1px solid #ccc; padding: 2px;">Advanced settings</span>

**04 Input Registers(3x)**

Table Display
Show
Hide

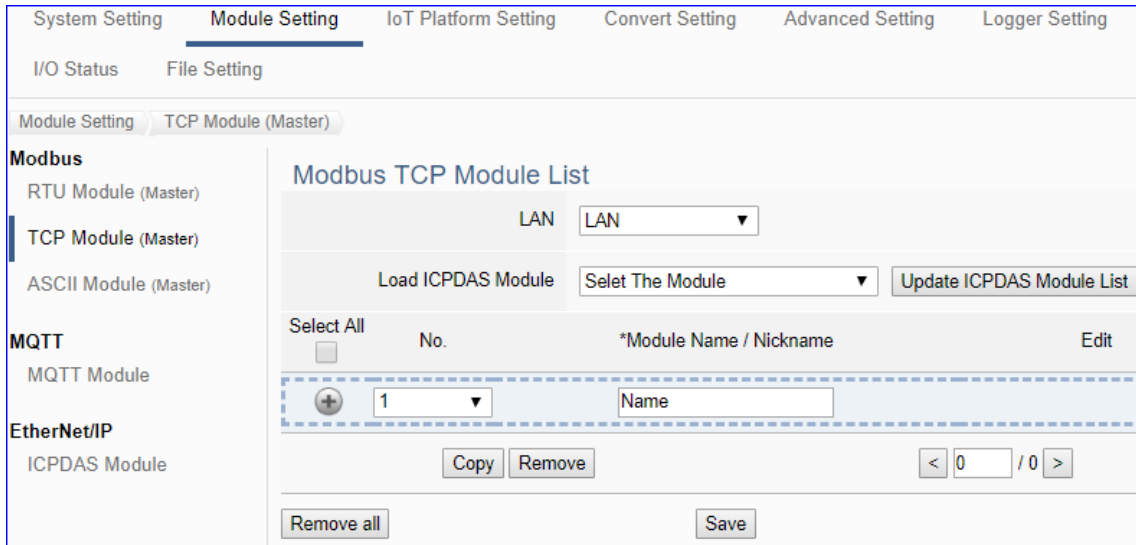
Address	Reference	Bitwise

OK
Cancel

<b>Modbus Mapping Table – Bitwise</b>	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

### 5.2.2 Modbus TCP (Master)

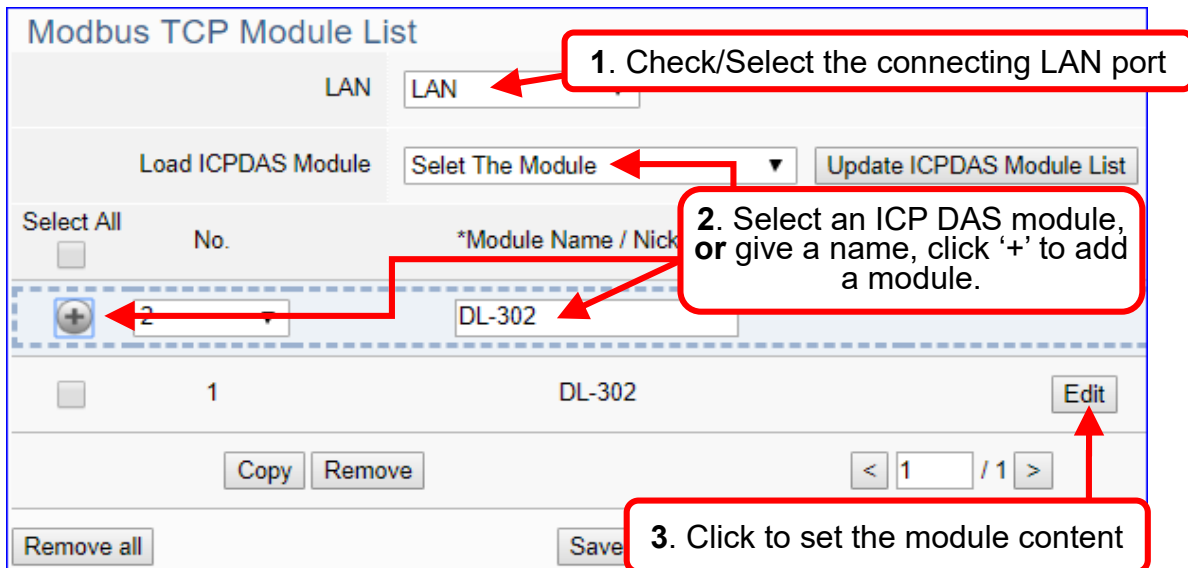
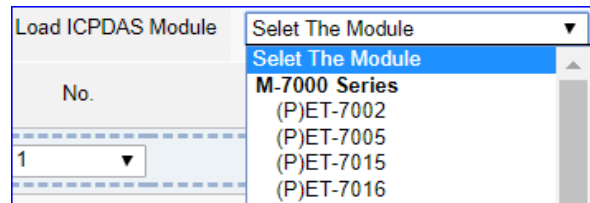
This setting is for UA Controller connecting the remote Modbus TCP Slave module.



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). Click [ + ] button could add a new module. If using ICP DAS module, user just need to select the model number, system will auto add and setup the module. Click [Edit] button to configure the module content and the Modbus mapping table.

**Setting Steps:**

1. Check/Select the module LAN port
  2. a) Select an ICP DAS model (as the pic), system will auto load/setup the module.  
 b) Give the module name or nickname, e.g. DL-302. Click [ + ] to add a new module
  3. Click the button [Edit] to enter the Module Content Setting page
- 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 cannot 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.	1
Module Name	Name
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

If select ICP DAS module, system will auto set up the Modbus Mapping Table, or user needs to check the module Modbus address or I/O number from the module user manual.

> **Modbus Mapping Table Setting:**

Set module in the order of Data Model, Start Address and Data Number, then click "Add".

**Ex:** DL-302 has 6 Data Models of "04 Input Registers (3x)" (Mapping: AI), so select Model "04", start 0, number 6, type Short, and click "Add".

Input Registers(3x)	
Address	0
Number	6
Type	Short
Edit	

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. (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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address Setting:**

Display and edit the Modbus Mapping Table.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise								
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>6</td> </tr> <tr> <td>Type</td> <td>Short</td> </tr> <tr> <td colspan="2" style="text-align: right;"> <input type="button" value="Edit"/> </td> </tr> </table>		Address	0	Number	6	Type	Short	<input type="button" value="Edit"/>	
Address	0												
Number	6												
Type	Short												
<input type="button" value="Edit"/>													
		<input type="button" value="OK"/> <input type="button" value="Cancel"/>											

If user selects ICP DAS module, the system will auto set up the Modbus Mapping Table. If not, user needs to check the module Modbus address or I/O number from the module user manual.

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Nickname Setting:**

Setting the variable nickname and description.

Modbus Mapping Table					Address	Nickname	Scaling	Bitwise
<b>01 Coil Status(0x)</b>								
Table Display					<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Variable name	Data Type	Description					
<b>02 Input Status(1x)</b>								
Table Display					<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Variable name	Data Type	Description					
<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				
<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="CO2"/>	Short	<input type="checkbox"/>	<input type="text" value="room1"/>				
1	<input type="text" value="Relative_humidity"/>	Short	<input type="checkbox"/>	<input type="text"/>				
2	<input type="text" value="Temperature_Celsius"/>	Short	<input type="checkbox"/>	<input type="text"/>				

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.

● **\*\* Scaling**

**Scaling is only available in the AI/AO settings of Modbus RTU/TCP.** When the variable value needs to be scaled or converted before output, click the **"Advanced Setting"** button of the variable on the **Scaling** page, input the **Min./Max./Offset** of the Reference/Output items, add a description, and check **"Enable"** box, The Scaling conversion function will be activated.

Modbus Mapping Table – Scaling	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Scaling do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The I/O variable of the Modbus address.
Output	The scaling variable for scaling output. User can define the variable name.
Scaling	Click [Show Detail] to set up the Scaling parameters, and click [Hide Detail] to hide the parameters. Fill in the Min/Max range values of the source in the Reference column. Fill in the Min/Max range values after scaling in the Output column. If needs offset, fill the offset value in the Offset item. Remember check “Enable” box.
Enable	Check the box of the variable can enable just that variable for scaling.
Description	Write a note for this variable.
OK	Click to save this page settings and back to the module list page.

● **\*\* Bitwise**

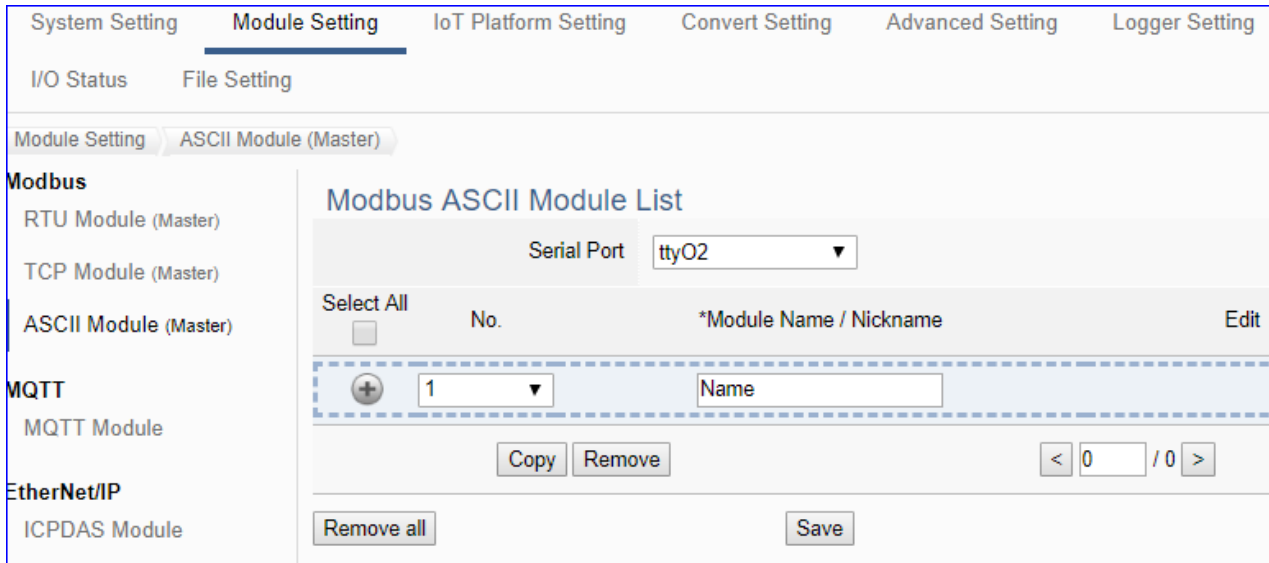
**Bitwise is only available in the AI/AO settings of Modbus RTU/TCP.** When the data needed to take out the value of the specified bit, fill in the variable name in the specified Bit# of the required address, and the value of the bit can be output to the filled variable.

Modbus Mapping Table		Address	Nickname	Scaling	Bitwise
<b>03 Holding Registers(4x)</b>					
Table Display		<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Reference	Bitwise			
<b>04 Input Registers(3x)</b>					
Table Display		<input type="button" value="Show"/> <input type="button" value="Hide"/>			
Address	Reference	Bitwise			
0	<input type="text" value="CO2"/>	<input type="button" value="Hide Detail"/>			
	<input type="button" value="Bit0"/>	<input type="text" value="aa"/>	<input type="button" value="Bit1"/>	<input type="text"/>	
	<input type="button" value="Bit2"/>	<input type="text" value="bb"/>	<input type="button" value="Bit3"/>	<input type="text"/>	
	<input type="button" value="Bit4"/>	<input type="text"/>	<input type="button" value="Bit5"/>	<input type="text"/>	
	<input type="button" value="Bit6"/>	<input type="text"/>	<input type="button" value="Bit7"/>	<input type="text"/>	
	<input type="button" value="Bit8"/>	<input type="text"/>	<input type="button" value="Bit9"/>	<input type="text"/>	
	<input type="button" value="Bit10"/>	<input type="text"/>	<input type="button" value="Bit11"/>	<input type="text"/>	
	<input type="button" value="Bit12"/>	<input type="text"/>	<input type="button" value="Bit13"/>	<input type="text"/>	
	<input type="button" value="Bit14"/>	<input type="text"/>	<input type="button" value="Bit15"/>	<input type="text"/>	

Modbus Mapping Table – Bitwise	
Modbus Mapping Table	Holding Registers(4x): Mapping to AO Modbus address Input Registers(3x): Mapping to AI Modbus address <b>Bitwise do not support 01 Coil Status(0x):DO &amp; 02 Input Status(1x):DI</b> <b>Bitwise do not supports 32-bit Float &amp; 64-bit Double data types.</b>
Table Display	Click [Show] to display all fields, click [Hide] to hide some fields.
Address	Modbus address. System auto arrange.
Reference	The Bit# variables of the Modbus address.
Bitwise	Set up the variables for Bitwise. Click [Advanced Settings] to set up the Bitwise parameters, and click [Hide] to hide the parameters. Fill in the variable names to the Bit# that wanted to do the Bitwise. The value in the fixed bit number will be assigned into the variable.
OK	Click to save this page settings and back to the module list page.

### 5.2.3 Modbus ASCII (Master)

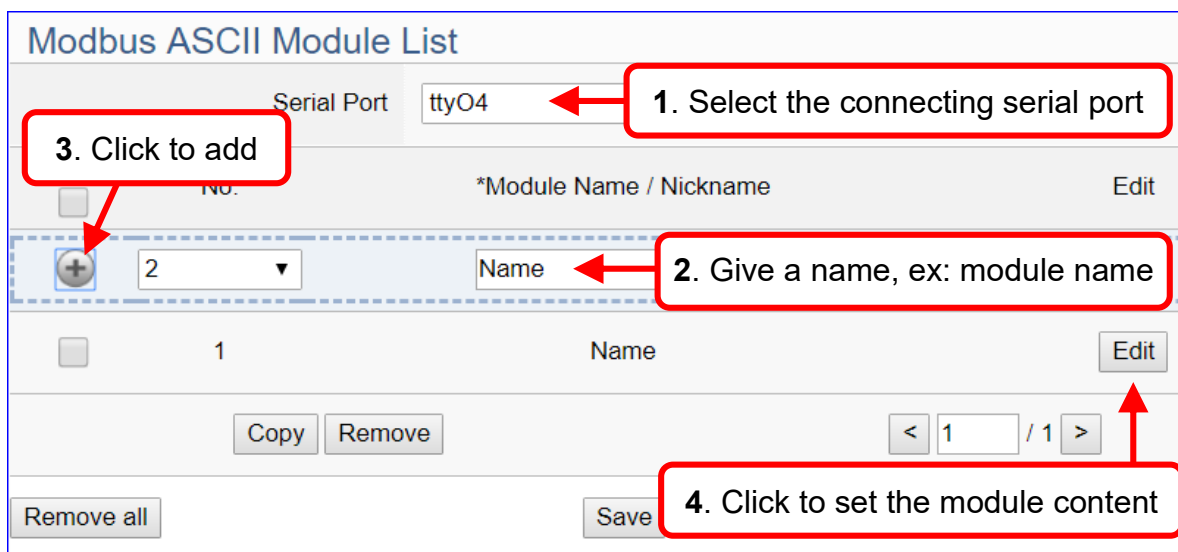
This setting is for UA Controller connecting the remote Modbus ASCII Slave module.



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. module number. Default: Name
3. Click the button [ + ] to add a new module
4. Click the button [Edit] to enter the Module Content Setting page  
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

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 cannot 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: 0;">Copy to : <input style="width: 100px;" type="text" value="ttyO5"/></p> <p style="margin: 0;">Quantity : <input style="width: 100px;" type="text"/></p> <p style="text-align: center; margin: 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.
	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.	1
Module Name	Name
Slave ID	1
Timeout	500

### Modbus Mapping Table Setting

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

**> Modbus Mapping Table Setting:**  
 Set module in the order of Data Model, Start Address and Data Number, then click “Add”.

**Ex:** Module has 6 Data Models of “04 Input Registers (3x)” (Mapping: AI), so select Model “04”, start 0, number 6, type Short, and click “Add”.

Input Registers(3x)

Address	0
Number	6
Type	Short
<input type="button" value="Edit"/>	

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. (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;"> <table style="width: 100%; border-collapse: collapse;"> <tr style="background-color: #007bff; color: white;"><td style="padding: 2px;">01 Coil Status(0x)</td></tr> <tr><td style="padding: 2px;">02 Input Status(1x)</td></tr> <tr><td style="padding: 2px;">03 Holding Registers(4x)</td></tr> <tr style="background-color: #007bff; color: white;"><td style="padding: 2px;">04 Input Registers(3x)</td></tr> </table> </div>	01 Coil Status(0x)	02 Input Status(1x)	03 Holding Registers(4x)	04 Input Registers(3x)
01 Coil Status(0x)					
02 Input Status(1x)					
03 Holding Registers(4x)					
04 Input Registers(3x)					
Start Address	The start address of the Modbus command. <b>Note:</b> the Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

**Address Setting:**

Display and edit the Modbus Mapping Table.

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 Start Address of UA is bass on 0, even if some modules are bass on 1, here it needs to follow UA to set bass on 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.

### Nickname Setting:

Setting the variable nickname and description.

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)**

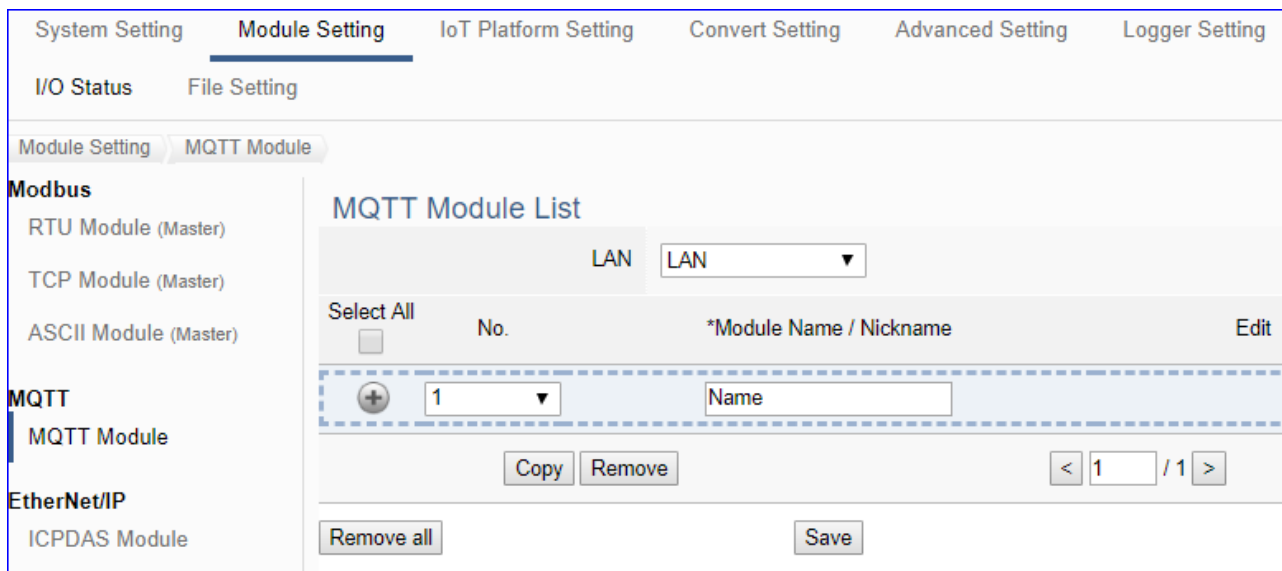
Table Display

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.

### 5.2.4 MQTT Module

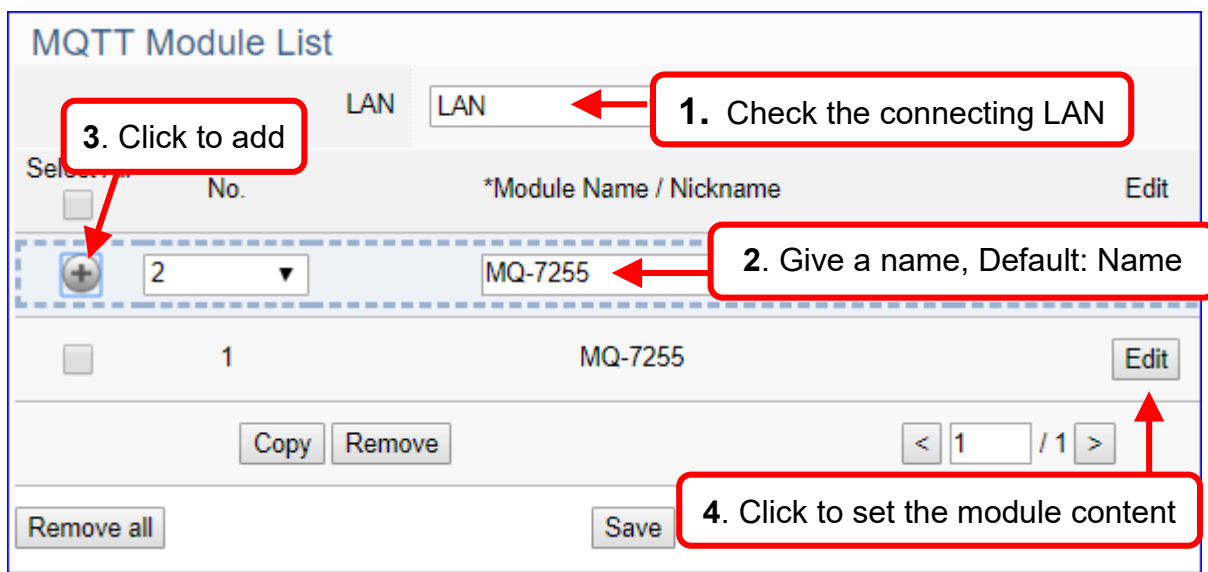
This setting is for UA Controller connecting the remote MQTT module.



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 MQTT variable table.

#### Setting Steps:

1. Select the module connecting Ethernet LAN port
2. Give the module name or nickname, e.g. MQ-7255, DL-302. Default: Name
3. Click the button [ + ] to add a new module
4. Click the button [Edit] to enter the Module Content Setting page  
Set up the Modbus Mapping Table for the UA controller and module I/O channels



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

Module Setting > MQTT - MQTT Module > MQTT Module List		
LAN	Choose the LAN port of UA controller that links with the MQTT module.	
	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 cannot 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 [MQTT Client Setting] page.

[MQTT Client Setting] page: to set up the module and the variable table

MQTT Client Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="MQ-7255"/>
MQTT Connection	<input checked="" type="radio"/> Broker (Local)
MQTT Variable Setting	
Attribute	<input type="text" value="Read"/>
Data Type	<input type="text" value="Bool"/>
Data Number	<input type="text" value="1"/>
Create Tables	<input type="button" value="Add"/>
Details	<input type="button" value="Show"/> <input type="button" value="Hide"/>

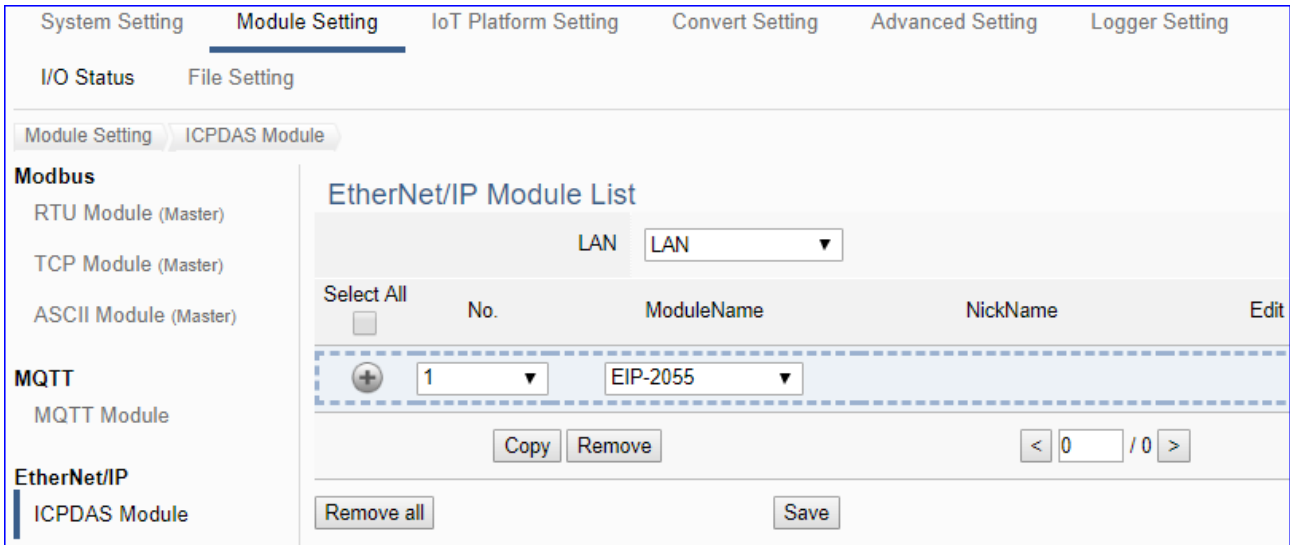
MQTT Client Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
MQTT Connection	The using Broker (Local).
MQTT Variable Setting	
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the MQTT variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String.
Data Number	The number for the I/O variables of the module. Default: 1.
Create Tables	Click [Add] button, it will add a variable list in the MQTT Variable Table.
Details Show / Hide	Click [Show] to display all fields, click [Hide] to hide some fields. The hide fields: Subscribe QoS, Publish QoS, Retain.

**[MQTT Variable Table] :**

MQTT Variable Table	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Show / Hide	The hide fields: Subscribe QoS, Publish QoS and Retain.
Remove Table / Remove	Check the box in the left of the variable is to select that variable list, and click the “remove” on the box can delete that variable list. Click the “Remove” of the “Remove Table” will delete all lists.
Name	The name of the MQTT variable. Default: Tag#
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String
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.
Description	For users set up the description for the variables.
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.
OK / Cancer	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.

### 5.2.5 EtherNet/IP ICP DAS Module

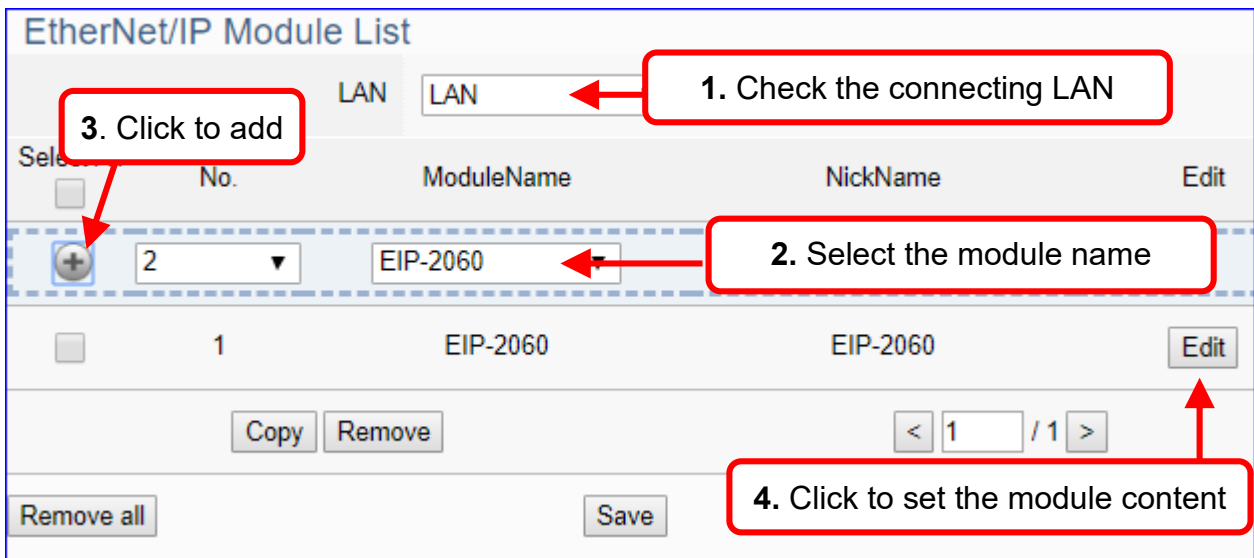
This setting is for UA Controller connecting the remote ICP DAS EIP module.



This page is for setting the communication values with the connected EIP modules. First, choose the Ethernet LAN port that connected with the module, and select the name of EIP module. Click [ + ] button could add a new module, and then click [Edit] button to configure the module content and the MQTT variable table.

**Setting Steps:**

1. Select the module connecting Ethernet LAN port
2. Select the module name of EIP-2000
3. Click the button [ + ] to add a new module
4. Click the button [Edit] to enter the Module Content Setting page  
Set up the module IP and module I/O channels



The function items and setting parameters of the [EtherNet/IP Module List]:

Module Setting > EtherNet/IP – ICPDAS Module > EtherNet/IP Module List		
LAN	Choose the LAN port of UA controller that links with the EIP module. UA-5200 has 1 LAN port; UA-2200 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)	
ModuleName	Select the connecting EIP-2000 module name.	
NickName	User can define a nickname for the EIP-2000 module.	
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.



[Module Content Setting] page:

Module Content Setting	
No.	<input type="text" value="1"/>
Module Name	<input type="text" value="EIP-2060"/>
NickName	<input type="text" value="EIP-2060"/>
IP	<input type="text" value="0"/> . <input type="text" value="0"/> . <input type="text" value="0"/> . <input type="text" value="0"/>
ChannelNumber	<input type="text" value="12-ch(6DI+6DO) ▼"/>

Module Setting > EtherNet/IP – ICPDAS Module > Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	The selected EIP module number.
NickName	User can define a nickname for the module. Default: Module name.
IP	Enter the IP address of the module. Default: 0.0.0.0
ChannelNumber	System auto setup the I/O channel numbers and the I/O table. Some module provides 2 or more channel mode needed user to select one.

DI/DO/AI/AO Channel Table: System auto setup the table according to the module name.

Digital Input				
Channel	Name	Attributes	Data Type	Description
0	<input type="text" value="DI0"/>	<input type="text" value="Read"/> ▼	Bool	<input type="text"/>

Digital Output				
Channel	Name	Attributes	Data Type	Description
0	<input type="text" value="DO0"/>	<input type="text" value="Read / Write"/> ▼	Bool	<input type="text"/>

Analogy Input				
Channel	Name	Attributes	Data Type	Description
0	<input type="text" value="AI0"/>	<input type="text" value="Read"/> ▼	Float	<input type="text"/>

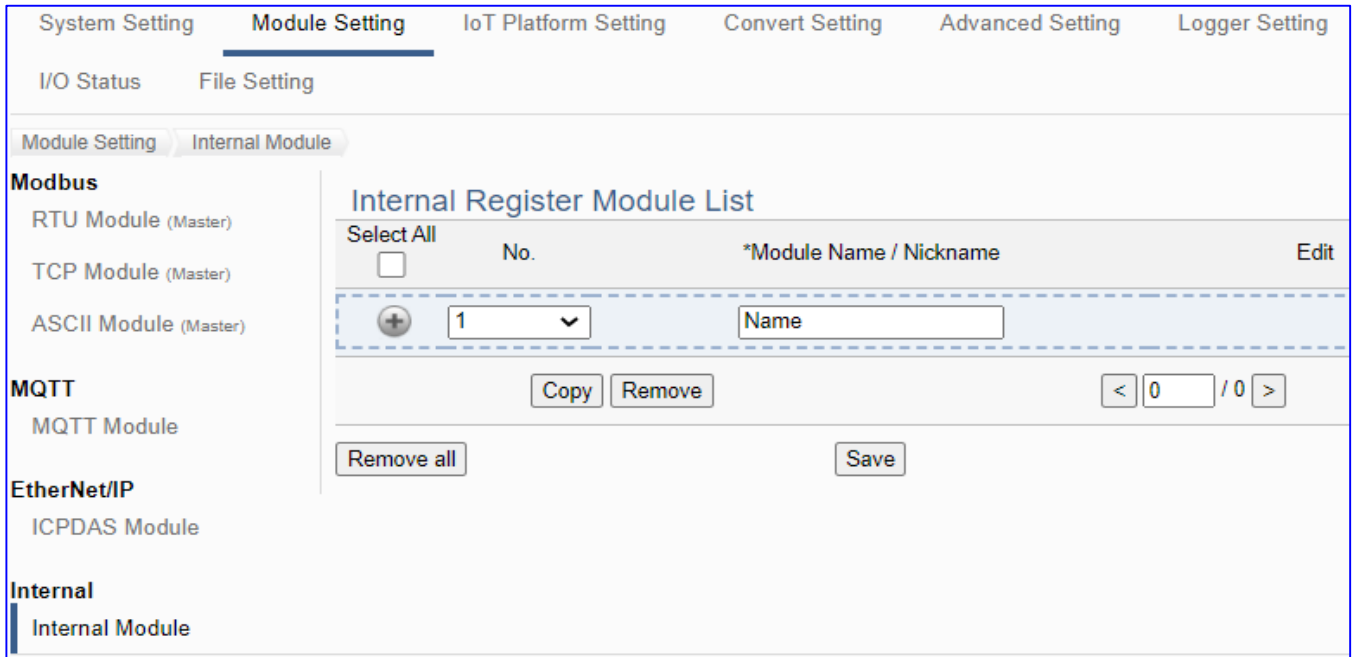
Analogy Output				
Channel	Name	Attributes	Data Type	Description
<input type="button" value="OK"/> <input type="button" value="Cancel"/>				

Module Setting > EtherNet/IP – ICPDAS Module > Di/Do/AI/AO Channel Table	
Channel	Channel number set by system. (Not editable)
Name	Channel name. User can define a new channel name.
Attribute	Display data attribute of the channel. (Not editable) Include: Read, Read / Write...
Data Type	Display data type of the channel. Include: Bool, Short, Float, ... according to the module.
Description	User can set up the description for the channel.
OK / Cancer	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.

## 5.2.6 Internal Module

The function can create internal modules and variables for virtual reading and writing, or as an intermediary to provide data exchange of communication protocols.

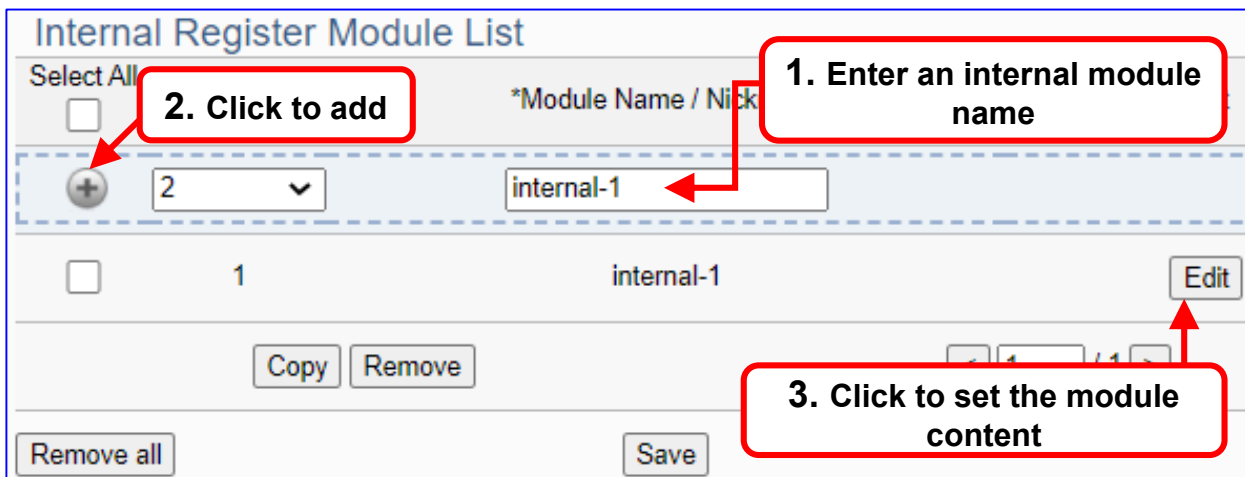
- \* Internal Variable: max. 8 internal modules; max. 100 internal variables (tags) per module.
- \* Protocol Communication Conversion: provide Internal to OPC UA (Server)



This page is for setting the internal module list. First, enter the name for the module. Click [ + ] button to add a new module, and then click [Edit] button to configure the module variables.

### Setting Steps:

1. Enter a name for the internal module
2. Click the button [ + ] to add the module
3. Click the button [Edit] to enter the Module Content Setting page.  
Set up the module variables



The function items and setting parameters of the [Internal Register Module List]:

Module Setting > Internal – Internal Module > Internal Register Module List	
Select All <input type="checkbox"/>	Check the box “Select All” will select all modules in the list. Check the box in the left of the module is to select that module list, can delete or copy the module.
	Click to add a list of module.
No.	The module number in the module list (System arrange, not editable)
*Module Name / Nickname	Module name or nickname. User can give a new name. (The star * means this field cannot 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 [**Internal Module Content Setting**] page.

[Internal Module Content Setting] page: to set up the module and the variable table

**Internal Module Content Setting**

No.

Module name

---

**Internal Variable Setting**

Attribute

Data Type

Data Number

Create Tables

**> Internal Variable Table:**  
 Sequentially select the Attribute, Data Type, and Data Number of the variable, and then click [Add].

**EX:** Internal-1 has 2 "Bool" type variables, then select Type "Bool", Number "2", then click [Add]. If user needs other variables, please add more variables by your case.

Internal 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.
Internal Variable Setting	
Attribute	Display data attribute of the variable. (Not editable) Include: Read/Write...
Data Type	Display data type of the MQTT variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String.
Data Number	The number for the I/O variables of the module. Default: 1.
Create Tables	Click [Add] button, it will add a variable list in the Variable Table.

The Internal Variable Table is as below.

**Internal Variable Table**

Remove Tables

<input type="button" value="Remove"/>	No.	Name	Attribute	Data Type	Description
<input type="checkbox"/>	1	<input type="text" value="Tag0"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>
<input type="checkbox"/>	2	<input type="text" value="Tag1"/>	<input type="text" value="Read / Write"/>	Bool	<input type="text"/>
<input type="checkbox"/>	3	<input type="text" value="Tag2"/>	<input type="text" value="Read / Write"/>	String	<input type="text"/>
<input type="checkbox"/>	4	<input type="text" value="Tag3"/>	<input type="text" value="Read / Write"/>	String	<input type="text"/>

**[Internal Variable Table] :**

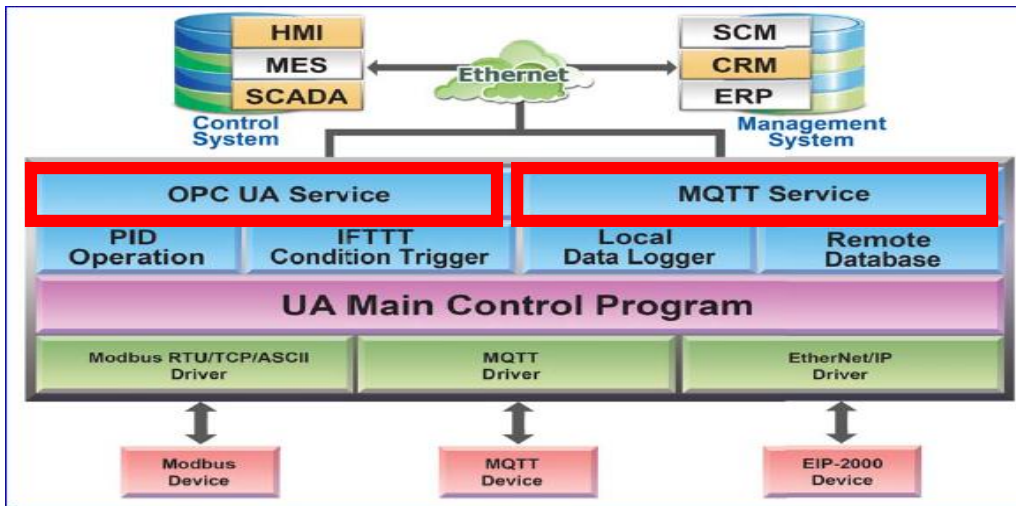
Internal Variable Table					
Remove Tables		<input type="button" value="Remove"/>			
<input type="button" value="Remove"/>	No.	Name	Attribute	Data Type	Description
<input type="checkbox"/>	1	<input type="text" value="Tag0"/>	Read / Write ▾	Bool	<input type="text"/>
<input type="checkbox"/>	2	<input type="text" value="Tag1"/>	Read / Write ▾	Bool	<input type="text"/>
<input type="checkbox"/>	3	<input type="text" value="Tag2"/>	Read / Write ▾	String	<input type="text"/>
<input type="checkbox"/>	4	<input type="text" value="Tag3"/>	Read / Write ▾	String	<input type="text"/>
		<input type="button" value="OK"/> <input type="button" value="Cancel"/>			

Internal Variable Table	
Remove Table / Remove	Check the box in the left of the variable is to select that variable list, and click the “remove” on the box can delete that variable list. Click the “Remove” of the “Remove Table” will delete all lists.
No.	The module number in the module list (Not editable here)
Name	The name of the variable. Default: Tag#
Attribute	Display data attribute of the variable. (Not editable) Include: Read/Write...
Data Type	Display data type of the variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String (Not editable)
Description	For users set up the description for the variables.
OK / Cancer	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.

### 5.3 Main Menu: IoT Platform Setting

UA series can connect to Amazon AWS, IBM Bluemix, Microsoft Azure or other IoT Cloud platforms. The Azure connection is more complicated and will set up in another setting item.

**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    Logger Setting

I/O Status    File Setting

IoT Platform Setting

**MQTT Connection**

- Local Broker
- Remote Broker
- MQTT Group Connection
- Microsoft Azure Platform

**OPC UA Connection**

- 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.

### 5.3.1 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 configuring MQTT settings. At the top, there are tabs for 'System Setting', 'Module Setting', 'IoT Platform Setting' (which is selected), and 'Convert Setting'. Under 'IoT Platform Setting', there is a sub-tab for 'Local Broker'. On the left, a sidebar lists 'MQTT Connection' options: 'Local Broker' (selected), 'Remote Broker', 'MQTT Group Connection', and 'Microsoft Azure Platform'. The main area is titled 'Local Broker Setting' and contains two rows of settings: 'Port' with a text input field containing '1883', and 'Anonymous Login' with a checked checkbox and the text '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.



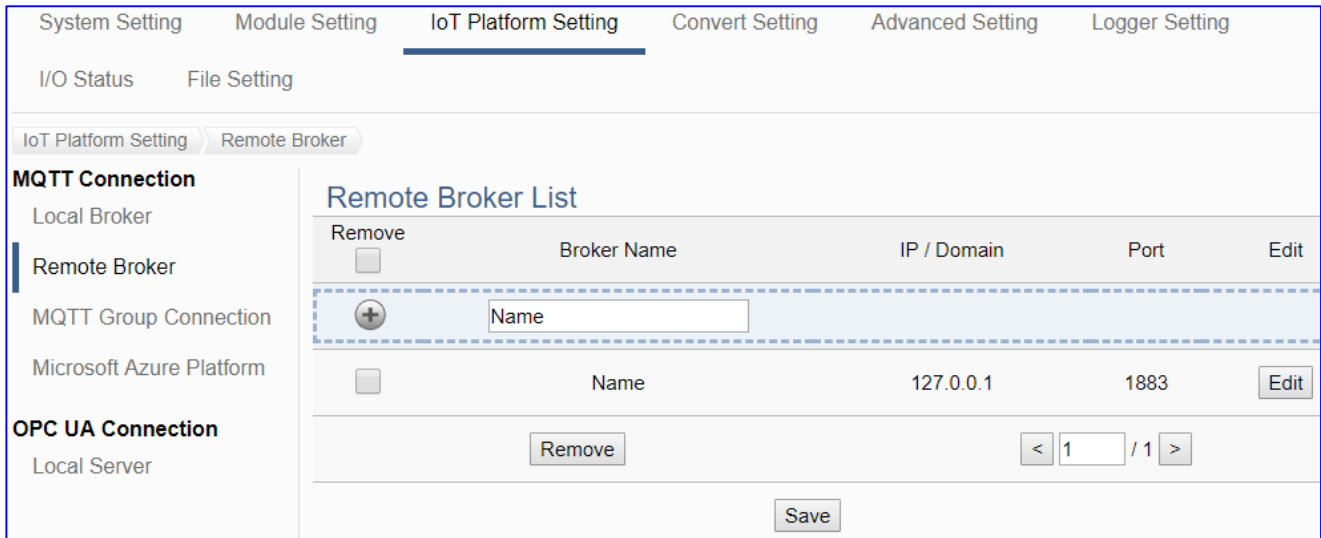
### 5.3.2 MQTT Remote Broker

UA series can connect to Amazon AWS, IBM Bluemix, Microsoft Azure or other IoT Cloud platforms. The Azure connection is more complicated and will set up in another setting item.



UA series controller built-in MQTT Broker, but when users want to use the external MQTT Broker, UA system also provides the settings to connect and publish/subscript 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.



**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 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 7](#).

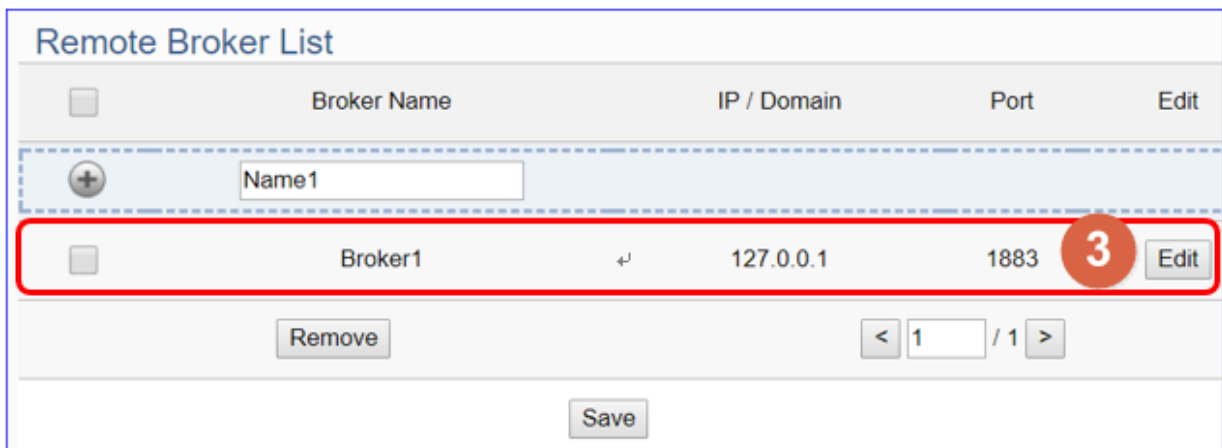
This section will introduce the function items and setting parameters.



**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	<input type="text" value="Broker1"/>
IP / Domain	<input type="text" value="127.0.0.1"/>
Port	<input type="text" value="1883"/>
Keep Alive Time(second)	<input type="text" value="60"/>
SSL/TLS	<input checked="" type="checkbox"/> Enabled
Trusted Certificate	<input type="text"/>
Certificate	<input type="text"/>
Private Key	<input type="text"/>
Anonymous Login	<input checked="" type="checkbox"/> Enabled
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

MQTT Connection > Remote Broker List > Broker Content Settings	
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)	Set the time in second that pass away without communication between the UA controller and Cloud platform. Default: 60 second.
SSL/TLS	Check to enable SSL/TLS security communication. Default: Uncheck. Sub-item: Trusted Certificate/Certificate/Private Key. Before enabling, upload the needed file from [File Setting] function menu.
Trusted Certificate	Select the trusted root CA file name uploaded to the controller via the File Setting function to verify the broker side certificate.
Certificate	Select the name of the certificate file uploaded to the controller via the File Setting function as the client side certification. When the "Certificate" field is not empty, the "Private Key" field cannot be empty.
Private Key	Select the name of the Private Key file uploaded to the controller via the File Setting function as the client side Private Key. When the "Private Key" field is not empty, the "Certificate" field cannot be empty.
Anonymous Login	Check to allow anonymous login. Default: Check Enabled.
OK / Cancel	Click: save the setting and exit this page. Cancel: exit without saving.

### 5.3.3 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.

The screenshot displays the 'MQTT Group Connection' configuration page. The left sidebar includes 'MQTT Connection' (with sub-items: Local Broker, Remote Broker, MQTT Group Connection, Microsoft Azure Platform) and 'OPC UA Connection' (with sub-item: Local Server). The main content area is titled 'MQTT Connection Group Name List' and features a table with the following structure:

Remove	Group Name	Edit
<input type="checkbox"/>		
<input type="checkbox"/>	Name	
<input type="checkbox"/>	Default	Edit

Below the table, there is a 'Remove' button, a pagination control showing '< 1 / 1 >', and a 'Save' button.

#### 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 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 7](#).

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. (Un-editable)
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 style="width: 90%;" 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; text-align: right; vertical-align: bottom; margin-right: 5px;" type="text" value="2"/> ▼
Subscribe Topic	<input style="width: 90%;" 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; text-align: right; vertical-align: bottom; margin-right: 5px;" 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; text-align: right; vertical-align: bottom; margin-right: 5px;" 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.

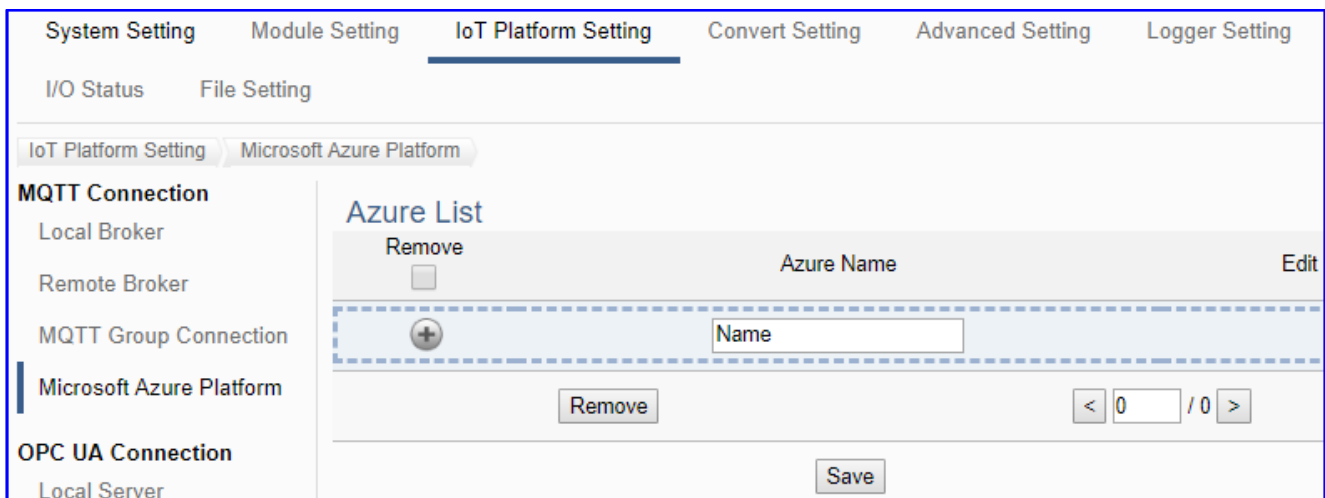


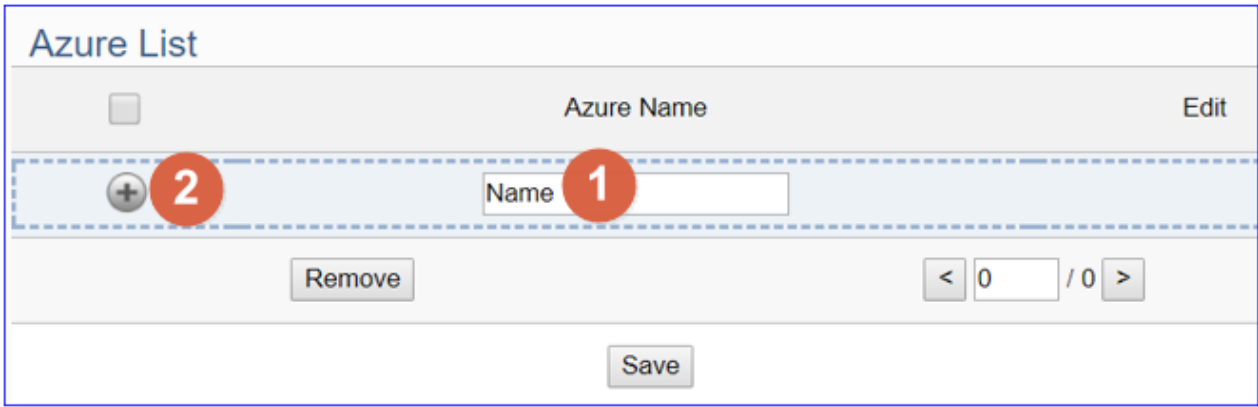
### 5.3.4 MQTT Connection - Microsoft Azure Platform

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.



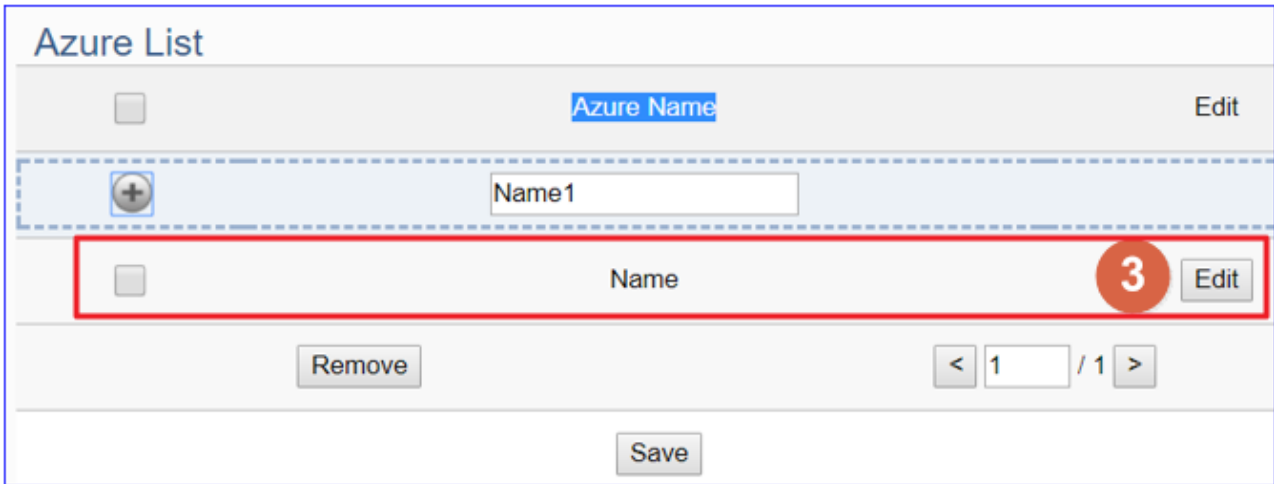
This page will introduce the settings for UA series controller using MQTT service to connect to the Microsoft Azure Platform. It includes new, remove and set up the Azure list and the function parameters





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 Azure:



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: 60px;" type="text" value="HostName=;DeviceId=;SharedAccessSignature="/>
Trusted Certificate	<input type="text" value=""/>
Keep Alive Time(second)	<input type="text" value="60"/>
Scan Rate(ms)	<input type="text" value="1000"/>
Dead Band	<input type="text" value="0"/>

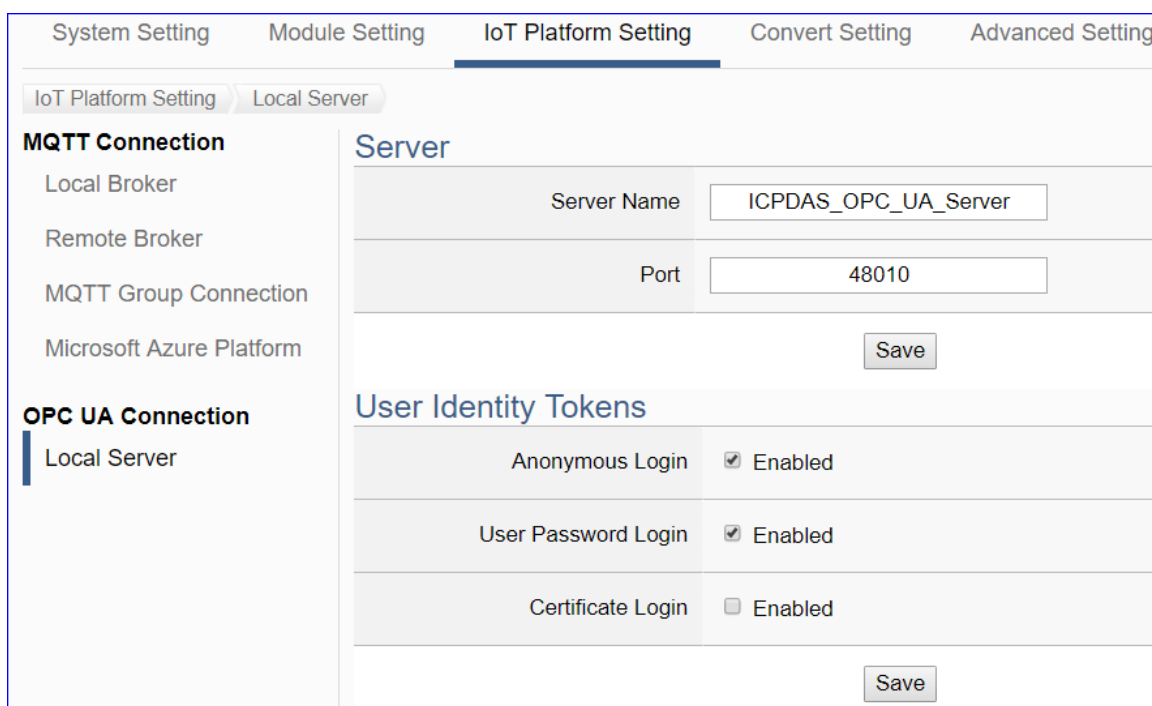
IoT Platform Setting > MQTT Connection > Microsoft Azure Platform > Azure Content Settings	
Azure Name	Azure name. User can define the name. Default: Name.
SAS Token	Input the SAS Token which you previously registered for the UA controller from Microsoft Azure. For the procedure to generate a SAS Token, please refer to the “Documentation > Azure IoT Hub > IoT Hub MQTT support” section on the Microsoft Azure Web Site for detailed information.
Trusted Certificate	Select the Trusted Certificate file that you previously got for the UA controller from Microsoft Azure, and uploaded in the menu [MQTT Certificate] of [File Setting] function. Sub-filename: <b>.crt</b>
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
OK / Cancel	OK: save and exit this page. Cancel: exit without saving.

### 5.3.5 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 setting is for the related service function of OPC UA Server built in UA series controller. The Security policy is **default to enable** the function of **None**, **Basic128Rsa15 (Sign)**, **Basic128Rsa15 (Sign and Encrypt)**, **Basic256 (Sign)**, and **Basic256 (Sign and Encrypt)**.

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 values: 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.

## 5.4 Main Menu: Convert Setting

**Convert Setting** is the 4<sup>th</sup> item of the Main Menu for the communication conversion.

There are 3 converting types: OPC UA, MQTT & MQTT JSON. Each type has some convert settings items for conversion with the Modbus RTU/TCP/ASCII (Master), MQTT, EtherNet/IP protocols and the OPC UA with Internal converting for the virtual communication. The function descriptions are on the home page of the Main Menu. This chapter will introduce these function items and setting parameters.

System Setting	Module Setting	IoT Platform Setting	<b>Convert Setting</b>	Advanced Setting	Logger Setting
I/O Status	File Setting				

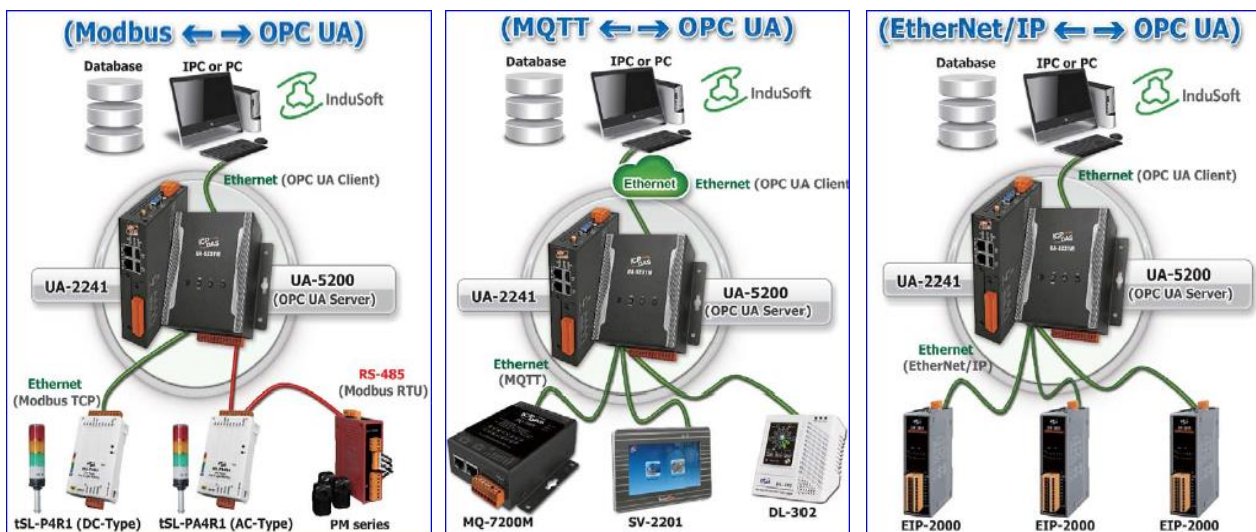
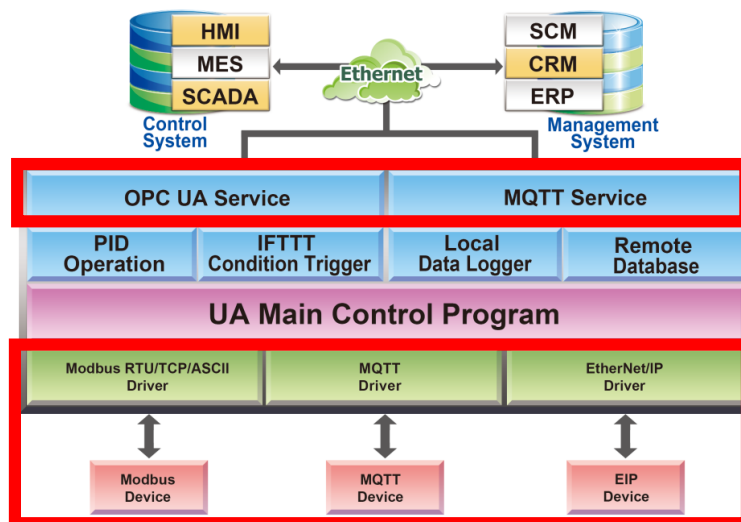
  

<p>Convert Setting</p> <p><b>OPC UA</b></p> <p>Modbus RTU (Master)</p> <p>Modbus TCP (Master)</p> <p>Modbus ASCII (Master)</p> <p>MQTT</p> <p>EtherNet/IP</p> <p>Internal</p> <p><b>MQTT</b></p> <p>Modbus RTU (Master)</p> <p>Modbus TCP (Master)</p> <p>Modbus ASCII (Master)</p> <p>EtherNet/IP</p> <p><b>MQTT JSON</b></p> <p>Modbus RTU (Master)</p> <p>Modbus TCP (Master)</p> <p>Modbus ASCII (Master)</p>	<h3>Convert Setting</h3> <p><b>OPC UA</b></p> <table border="1"> <tr> <td>Modbus RTU (Master)</td> <td>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.</td> </tr> <tr> <td>Modbus TCP (Master)</td> <td>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.</td> </tr> <tr> <td>Modbus ASCII (Master)</td> <td>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.</td> </tr> <tr> <td>MQTT</td> <td>Provides OPC UA and MQTT communication protocol conversion. With this function, the OPC UA Server can read and write the MQTT device that connected to the controller.</td> </tr> <tr> <td>EtherNet/IP</td> <td>Provides OPC UA and ICPDAS EIP communication protocol conversion. With this function, the OPC UA Server can read and write the MQTT device that connected to the controller.</td> </tr> </table> <p><b>MQTT</b></p> <table border="1"> <tr> <td>Modbus RTU (Master)</td> <td>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.</td> </tr> </table>	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.	MQTT	Provides OPC UA and MQTT communication protocol conversion. With this function, the OPC UA Server can read and write the MQTT device that connected to the controller.	EtherNet/IP	Provides OPC UA and ICPDAS EIP communication protocol conversion. With this function, the OPC UA Server can read and write the MQTT device that connected to the controller.	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 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.												
MQTT	Provides OPC UA and MQTT communication protocol conversion. With this function, the OPC UA Server can read and write the MQTT device that connected to the controller.												
EtherNet/IP	Provides OPC UA and ICPDAS EIP communication protocol conversion. With this function, the OPC UA Server can read and write the MQTT device that connected to the controller.												
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.												

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

<p><b>OPC UA</b></p>	<p>Use <b>OPC UA Service</b> to convert with <b>Modbus RTU/ASCII</b> protocol. <b>(5.4.1)</b>                  Use <b>OPC UA Service</b> to convert with <b>Modbus TCP</b> protocol. <b>(5.4.2)</b>                  Use <b>OPC UA Service</b> to convert with <b>MQTT</b> protocol. <b>(5.4.3)</b>                  Use <b>OPC UA Service</b> to convert with <b>EtherNet/IP</b> protocol. <b>(5.4.4)</b>                  Use <b>OPC UA Service</b> to convert with <b>Internal</b> communication. <b>(5.4.5)</b></p>
<p><b>MQTT</b></p>	<p>Use <b>MQTT Service</b> to convert with <b>Modbus RTU/ASCII</b> protocol. <b>(5.4.6)</b>                  Use <b>MQTT Service</b> to convert with <b>Modbus TCP</b> protocol. <b>(5.4.7)</b>                  Use <b>MQTT Service</b> to convert with <b>EtherNet/IP</b> protocol. <b>(5.4.8)</b></p>
<p><b>MQTT JSON</b></p>	<p>Use <b>MQTT Service</b> in group of JSON format to convert with <b>Modbus RTU/ASCII</b> protocol. <b>(5.4.9)</b>                  Use <b>MQTT Service</b> in group of JSON format to convert with <b>Modbus TCP</b> protocol. <b>(5.4.10)</b></p>

**UA Series Function Diagram:**

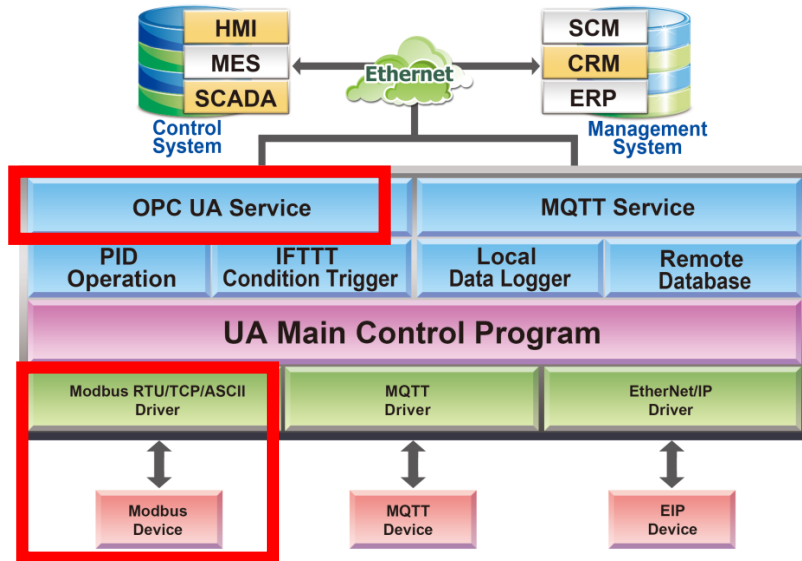


### 5.4.1 OPC UA and Modbus RTU/ASCII Conversion

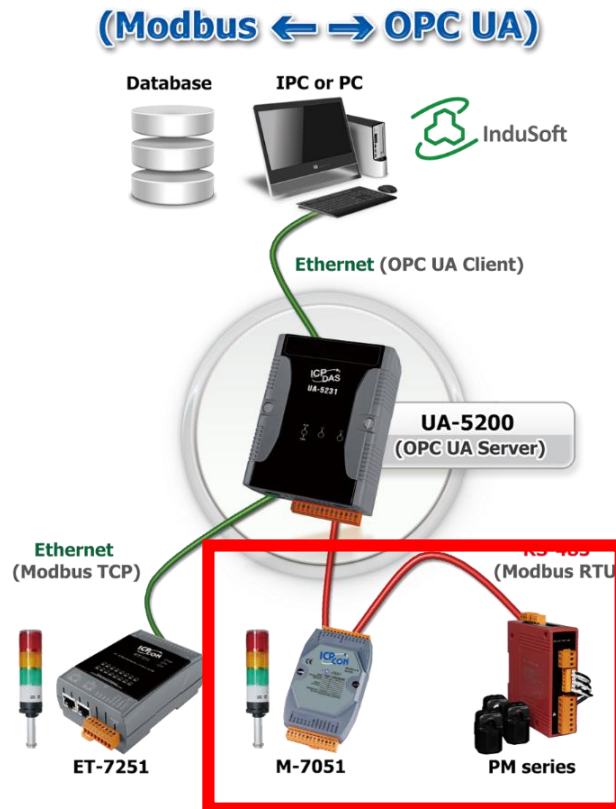
This page provides OPC UA and Modbus RTU/ASCII (Master) communication protocol conversion. With this function, the OPC UA Server can read and write the Modbus RTU / ASCII device that connected to the controller.

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

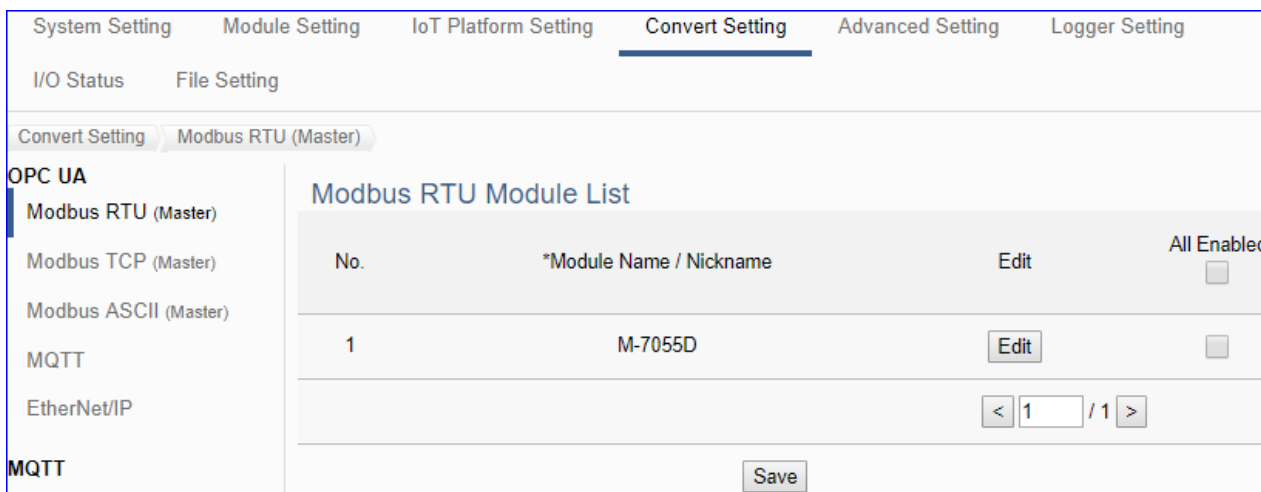
#### Function Diagram:



#### 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 5.2](#) 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 <input type="checkbox"/> Enable <input type="checkbox"/>	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.
<input type="button" value="&lt;"/> <input type="text" value="1"/> / 1 <input type="button" value="&gt;"/>	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.

This function is to enable the communication conversion module, please  check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to enter the “Module Content Setting” page.



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

**Module Content Setting**

No.

Module Name

**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"/>

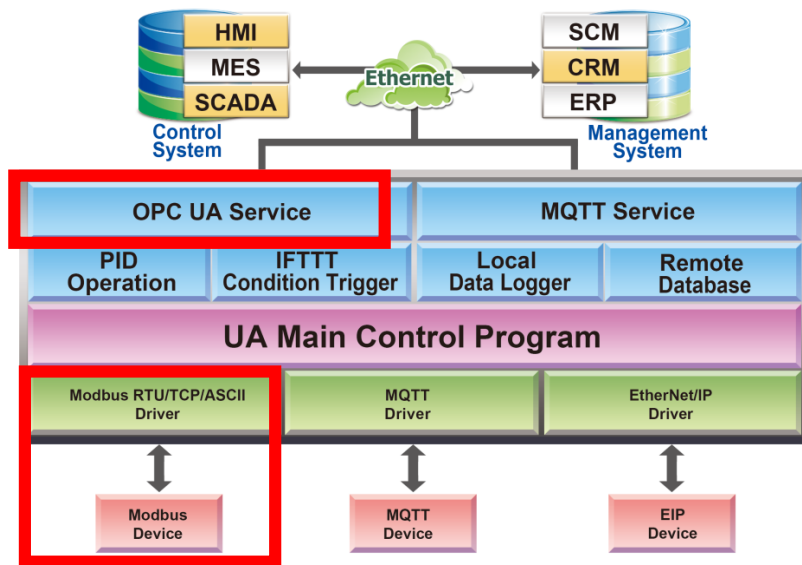
<b>Convert Setting &gt; OPC UA &gt; Modbus RTU (Master) – Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Convert Setting &gt; OPC UA &gt; Modbus RTU (Master) – Variable Table</b>	
Name	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. Remember to click [Save] to save the Convert Setting.

### 5.4.2 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.

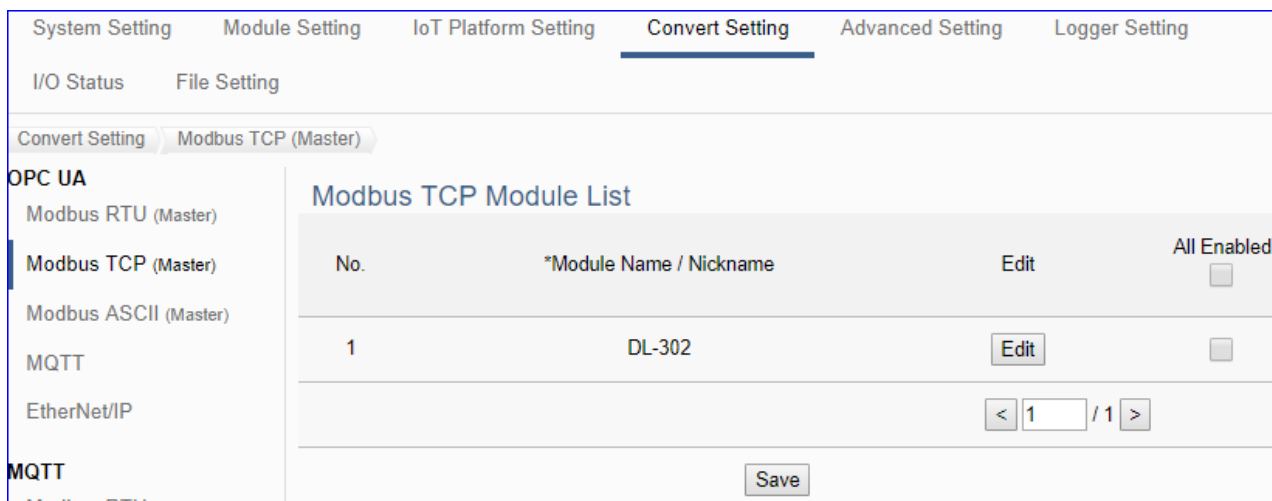
**Function Diagram:**



**Application Solution:**



When entering the menu [Convert Setting] and the sub-menu [OPC UA] > Modbus TCP (Master), the Modbus TCP modules preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 5.2](#) for the Module Setting.)



Convert Setting > OPC UA > 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 <input type="checkbox"/> <input type="checkbox"/> Enable	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 “Module Content Setting” page to set up and enable the I/O.
<input type="button" value=" &lt; 1 / 1 &gt;"/>	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.

This function is to enable the communication conversion module, please  check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to 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"/>	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"/>	

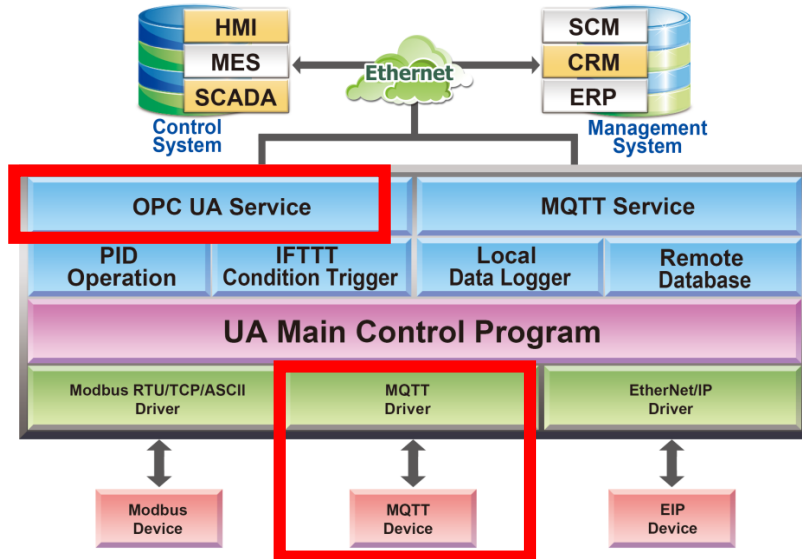
Convert Setting > OPC UA > Modbus TCP (Master) – Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
Convert Setting > OPC UA > Modbus TCP (Master) – Variable Table	
Name	Display the variable name 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.

### 5.4.3 OPC UA and MQTT Conversion

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

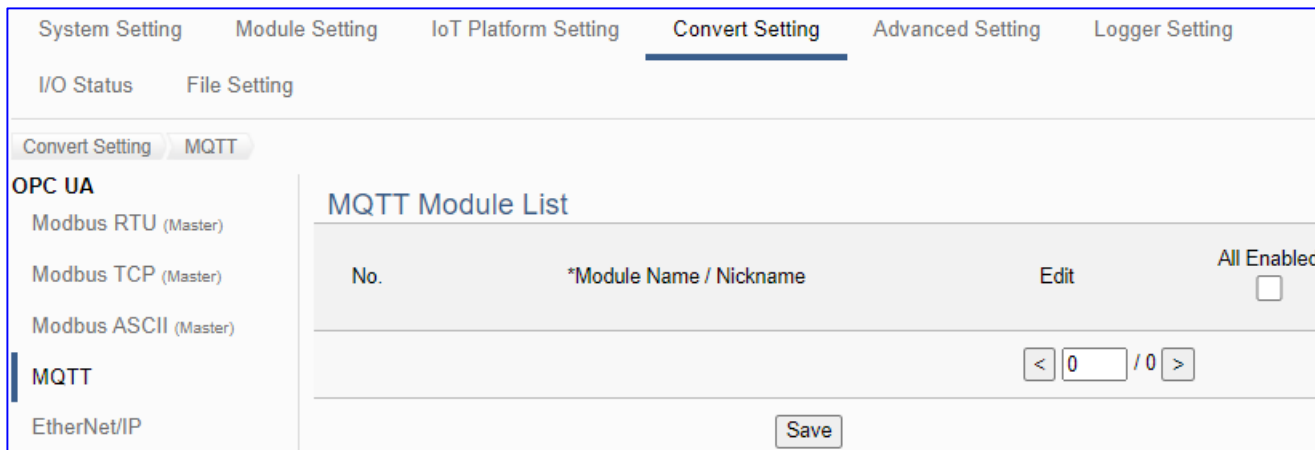
#### Function Diagram:



#### Application Solution:



When entering the menu [Convert Setting] and the sub-menu [OPC UA] > MQTT, the MQTT modules preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 5.2](#) for the Module Setting.)



Convert Setting > OPC UA > MQTT - MQTT 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 <input type="checkbox"/> <input type="checkbox"/> Enable	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 enable the I/O.
<input style="display: inline-block; width: 20px; height: 20px; border: 1px solid #ccc;" type="button" value=" &lt; 1 "/> / <input style="display: inline-block; width: 20px; height: 20px; border: 1px solid #ccc;" type="button" value=" 1 "/> >	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.

This function is to enable the communication conversion module, please check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to enter the “Module Content Setting” page.

[Module Content Setting] page:

**Module Content Setting**

No.

---

Module Name

---

**Variable Table**

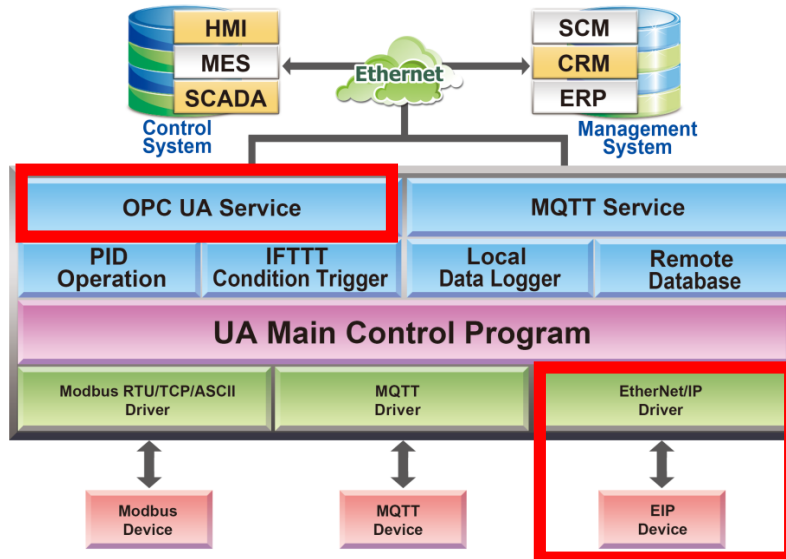
Name	Attribute	Data Type	Enabled
Temperature	<input type="text" value="Read"/>	Float	<input checked="" type="checkbox"/>
Humidity	<input type="text" value="Read"/>	Float	<input checked="" type="checkbox"/>
CO2	<input type="text" value="Read"/>	Short	<input checked="" type="checkbox"/>

<b>Convert Setting &gt; OPC UA &gt; MQTT - MQTT Module List &gt; Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
<b>Convert Setting &gt; OPC UA &gt; MQTT - MQTT Module List &gt; Variable Table</b>	
No.	The module name in the module list (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the MQTT variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String.
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 / Cancer	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.

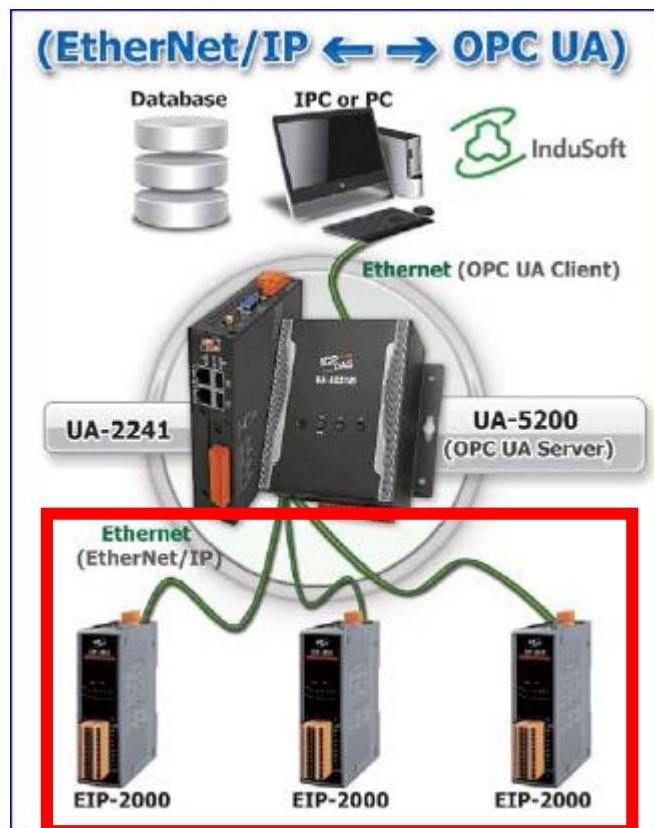
### 5.4.4 OPC UA and EtherNet/IP Conversion

This page provides OPC UA and EtherNet/IP communication protocol conversion. With this function, the OPC UA Server can read and write the EtherNet/IP EIP-2000 device that connected to the controller.

**Function Diagram:**

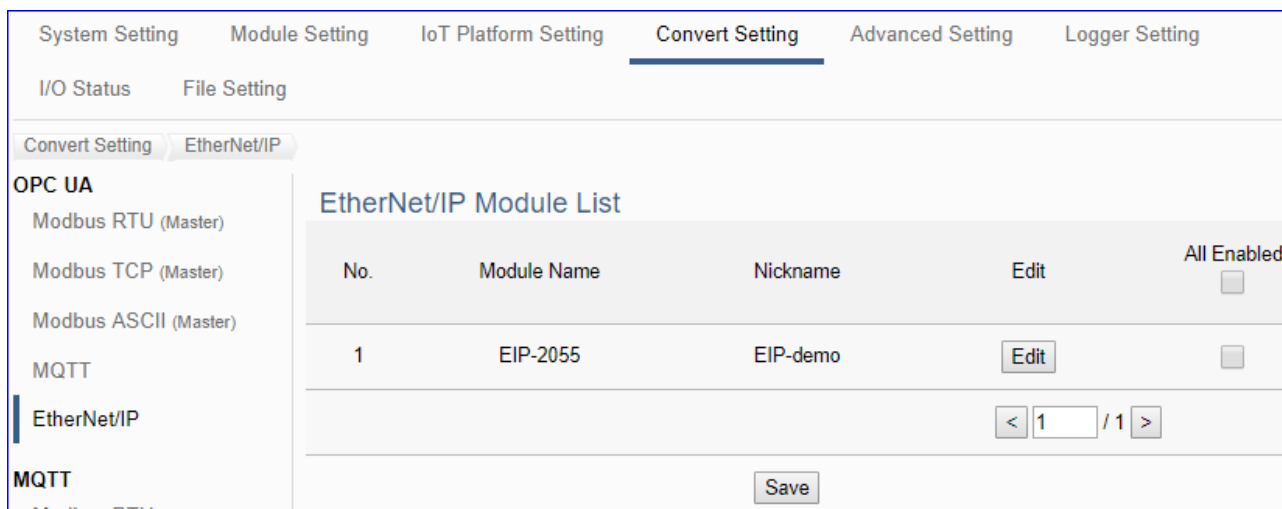


**Application Solution:**





When entering the menu [Convert Setting] and the sub-menu [OPC UA] > EtherNet/IP, the EIP-2000 modules preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 5.2](#) for the Module Setting.)



Convert Setting > OPC UA > EtherNet/IP Module List	
No.	The module number in the module list (Not editable here)
Module Name	The module name selected in the module list (Not editable here)
Nickname	The user defined name for the module (Not editable here)
All Enabled <input type="checkbox"/> Enable <input type="checkbox"/>	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 “Module Content Setting” page to set up and enable the I/O.
<input type="button" value="1"/> / 1	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.

This function is to enable the communication conversion module, please  check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to 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="EIP-2055"/>		
NickName	<input type="text" value="EIP-demo"/>		
Variable Table			
Name	Attribute	Data Type	Enabled <input type="checkbox"/>
DI0	<input type="text" value="Read"/>	Bool	<input type="checkbox"/>
DI1	<input type="text" value="Read"/>	Bool	<input type="checkbox"/>
DO6	<input type="text" value="Read / Write"/>	Bool	<input type="checkbox"/>
DO7	<input type="text" value="Read / Write"/>	Bool	<input type="checkbox"/>
		<input type="button" value="OK"/>	<input type="button" value="Cancel"/>

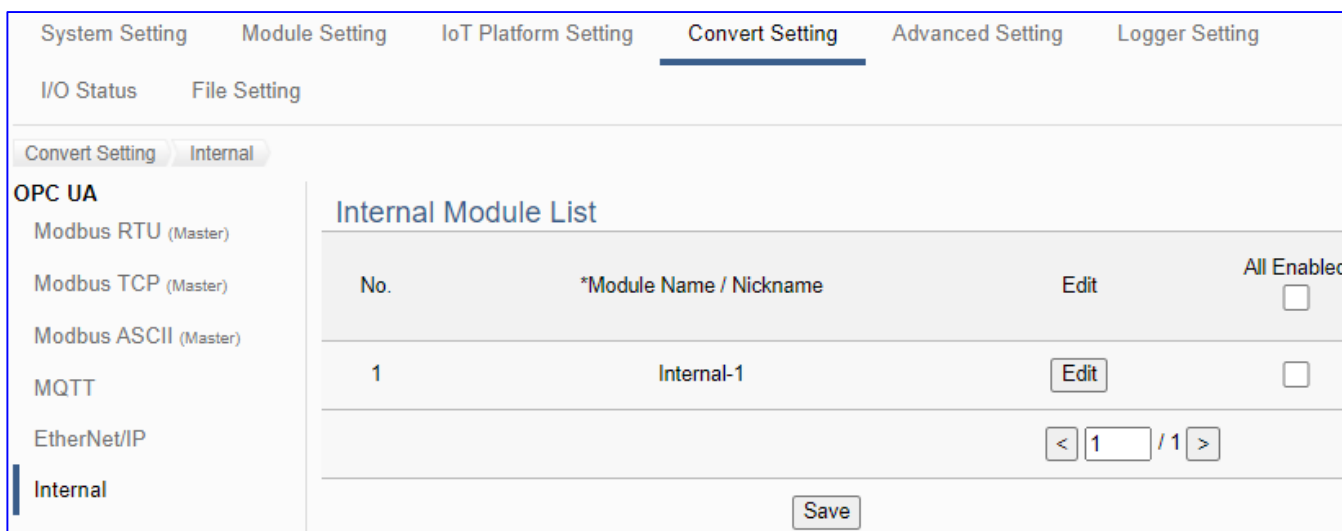
Convert Setting > OPC UA > EtherNet/IP – Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
Convert Setting > OPC UA > EtherNet/IP – Variable Table	
Name	Display the variable name that set in the 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. Remember to click [Save] to save the Convert Setting.

### 5.4.5 OPC UA 與 Internal 轉換設定

The function can create internal modules and variables for virtual reading and writing, or as an intermediary to provide data exchange of communication protocols, e.g. for 2 or more OPC UA clients.

When entering the menu [Convert Setting] and the sub-menu [OPC UA] > Internal, the internal modules preset in the [Module Setting] will show up in the Module List. (Refer to Chapter 5.2 for the Module Setting.)



Convert Setting > OPC UA > Internal - Internal 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	Click to enter the “Internal Client Setting” page to enable the I/O.
All Enabled <input type="checkbox"/> <input type="checkbox"/> Enable	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.
<input type="button" value="&lt;"/> <input type="text" value="1"/> / 1 <input type="button" value="&gt;"/>	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.

This function is to enable the communication conversion module, please  check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to enter the “Module Content Setting” page.

[Module Content Setting] page:

Module Content Setting			
No.	<input type="text" value="1"/>		
Module Name	<input type="text" value="Internal-1"/>		
Variable Table			
Variable Name	Attribute	Data Type	Enabled <input type="checkbox"/>
Tag2	<input type="text" value="Read / Write"/>	String	<input type="checkbox"/>
Tag3	<input type="text" value="Read / Write"/>	String	<input type="checkbox"/>
Tag0	<input type="text" value="Read / Write"/>	Bool	<input type="checkbox"/>
Tag1	<input type="text" value="Read / Write"/>	Bool	<input type="checkbox"/>
<input type="button" value="OK"/>		<input type="button" value="Cancel"/>	

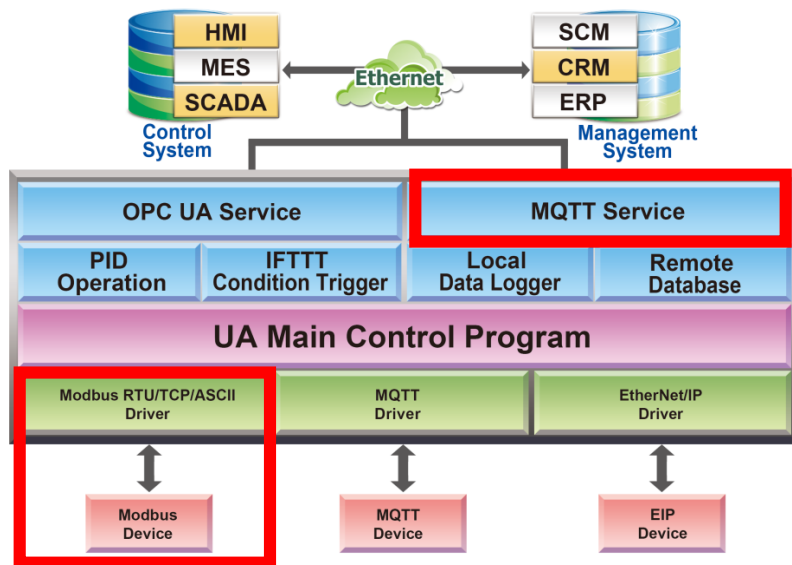
Convert Setting > OPC UA > Internal - Internal Module List > Module Content Setting	
No.	The module number in the module list (Not editable here)
Module Name	Give a name, e.g. model number or name. Default: Name.
Convert Setting > OPC UA > Internal - Internal Module List > Variable Table	
Variable Name	The Variable name in the module list (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: Read, Read/Write...
Data Type	Display data type of the Internal variable. Include: Bool, Short, Unsigned Short, Long, Unsigned Long, Float, Double, String. (Not editable here)
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 / Cancer	Click [OK] to save and exit the page settings. Click [Cancer] to exit without saving.

### 5.4.6 MQTT and Modbus RTU/ASCII Conversion

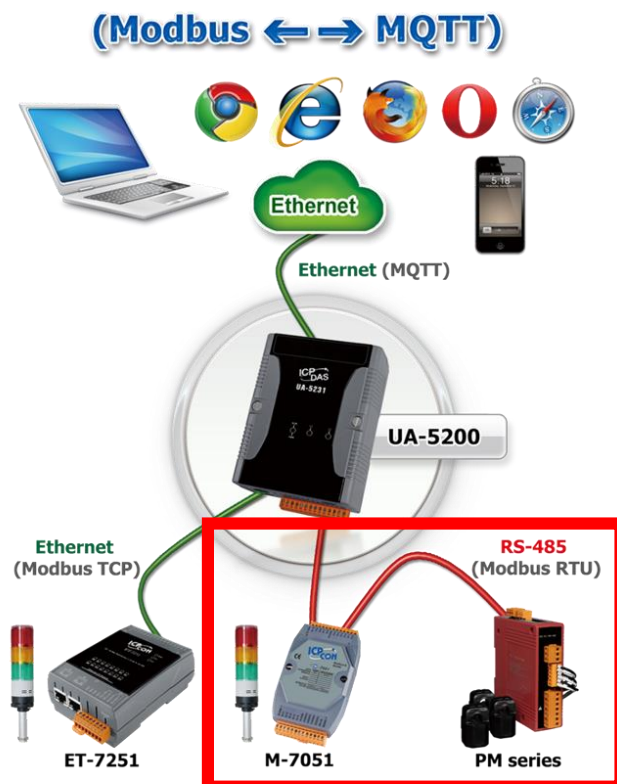
This page provides MQTT and Modbus RTU/ASCII (Master) communication protocol conversion. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus device that connected 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 7](#).

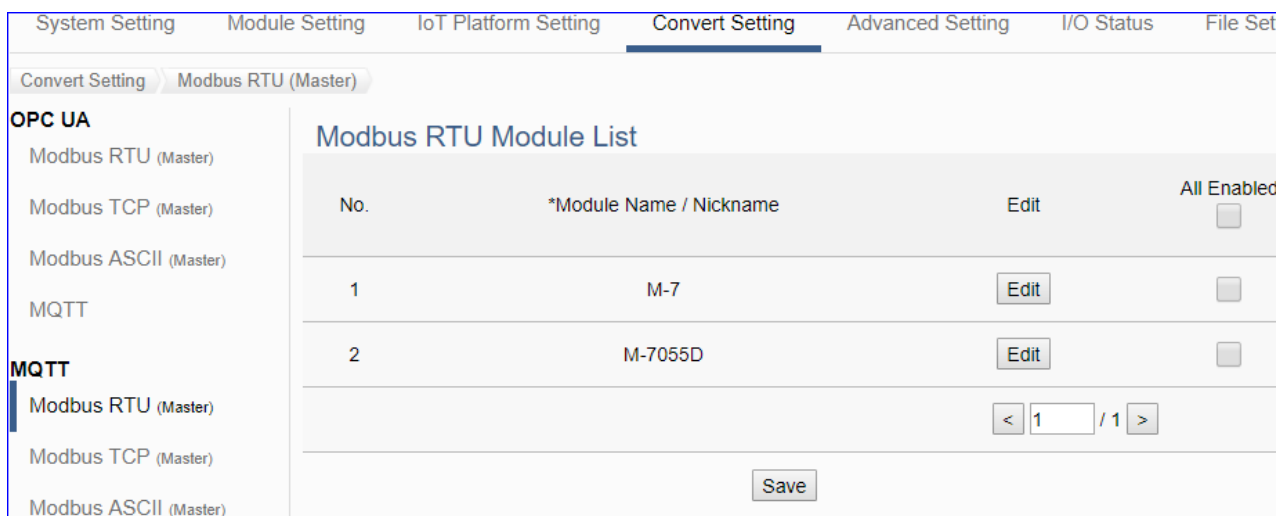
#### Function Diagram:



#### 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 5.2](#) 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 <input type="checkbox"/> Enable <input type="checkbox"/>	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 enable I/O or set up the Topic, QoS, Publish, Subscribe ...
<input type="button" value="&lt;"/> <input type="text" value="1"/> / 1 <input type="button" value="&gt;"/>	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.

This function is to enable the communication conversion module, please  check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to 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 (Un-editable)
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 <input type="checkbox"/>	Enabled <input type="checkbox"/>
Tag0	Read	Short		2	/MRTU_No.1_M-7/Input_Registers/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>
Tag0	Read / Write	Short	/MRTU_No.1_M-7/Holding_Registers/Tag0/Subscribe	2	/MRTU_No.1_M-7/Holding_Registers/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>
Tag0	Read	Bool		2	/MRTU_No.1_M-7/Input_Status/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>
Tag0	Read / Write	Bool	/MRTU_No.1_M-7/Coil_Status/Tag0/Subscribe	2	/MRTU_No.1_M-7/Coil_Status/Tag0/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>
Tag1	Read / Write	Bool	/MRTU_No.1_M-7/Coil_Status/Tag1/Subscribe	2	/MRTU_No.1_M-7/Coil_Status/Tag1/Publish	2	<input type="checkbox"/>	<input type="checkbox"/>

**Convert Setting > MQTT > Modbus RTU (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.

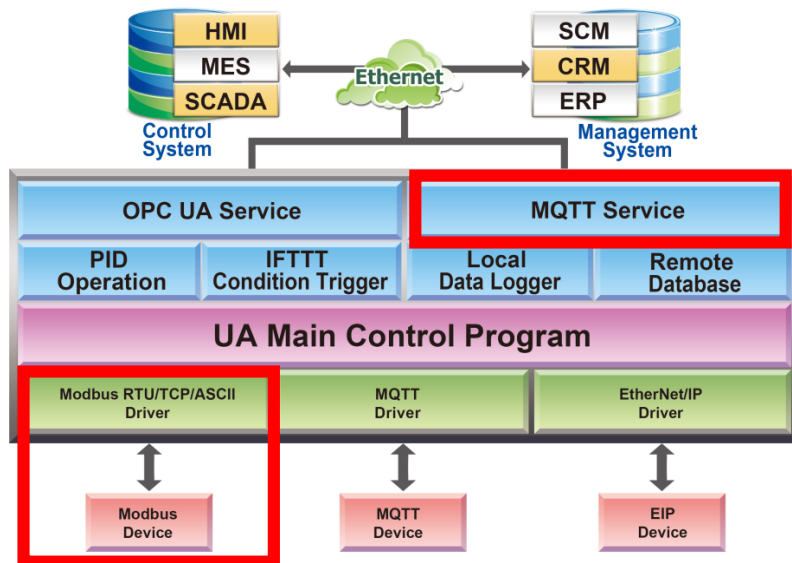


### 5.4.7 MQTT and Modbus TCP Conversion

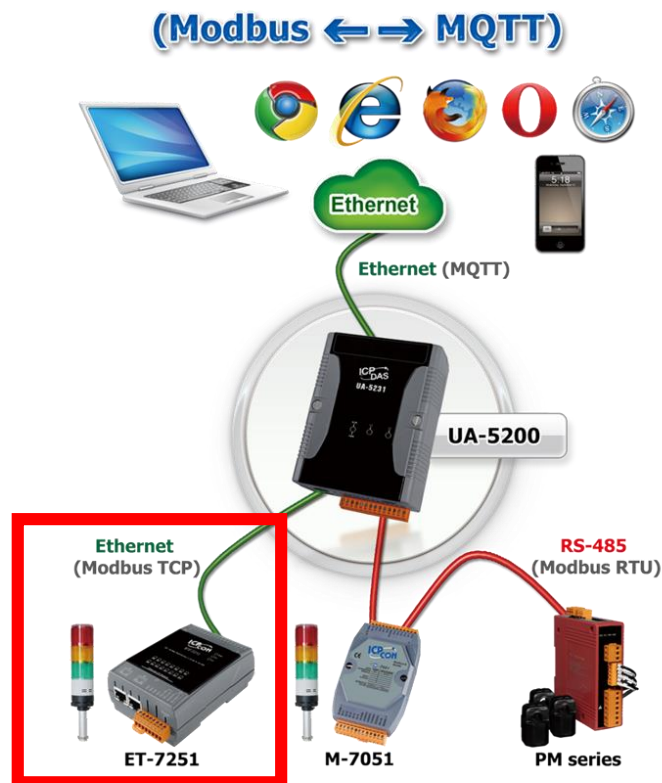
This page provides MQTT and Modbus TCP (Master) communication protocol conversion. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and so to read and write the single channel of the Modbus device that connected to the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

For the certificate about the communication security, please refer to [Chapter 7](#).

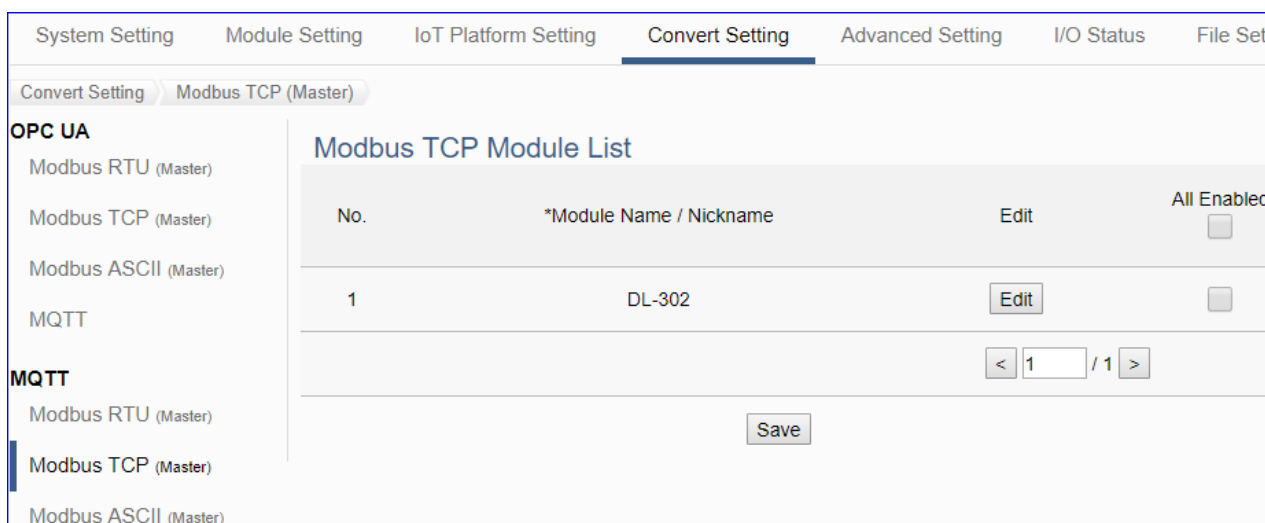
#### Function Diagram:



#### 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 5.2 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 <input type="checkbox"/> Enable <input type="checkbox"/>	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 enable I/O or set up the Topic, QoS, Publish, Subscribe ...
<input type="button" value="1"/> / 1	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.

This function is to enable the communication conversion module, please  check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to 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 (Un-editable)
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"/>

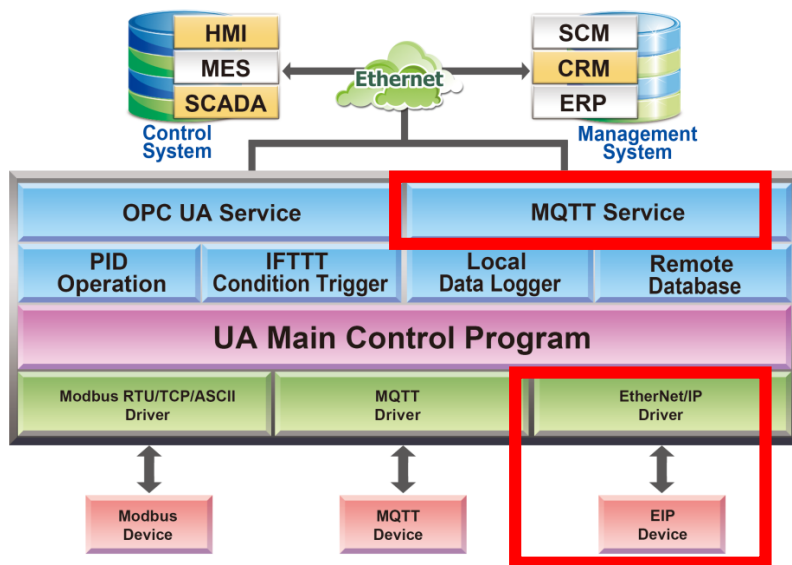
<b>Convert Setting &gt; MQTT &gt; Modbus TCP (Master) – Publish &amp; Subscribe</b>	
Details	Click [Show] to display all fields, click [Hide] to hide some fields.
Name	The variable name of the mapping address. (Not editable here)
Attribute	Display data attribute of the variable. (Not editable) Include: 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.

### 5.4.8 MQTT and EtherNet/IP Conversion

This page provides MQTT and EtherNet/IP communication protocol conversion. With this 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 ICP DAS EIP-2000 device that connected to the controller.

For the certificate about the communication security, please refer to [Chapter 7](#).

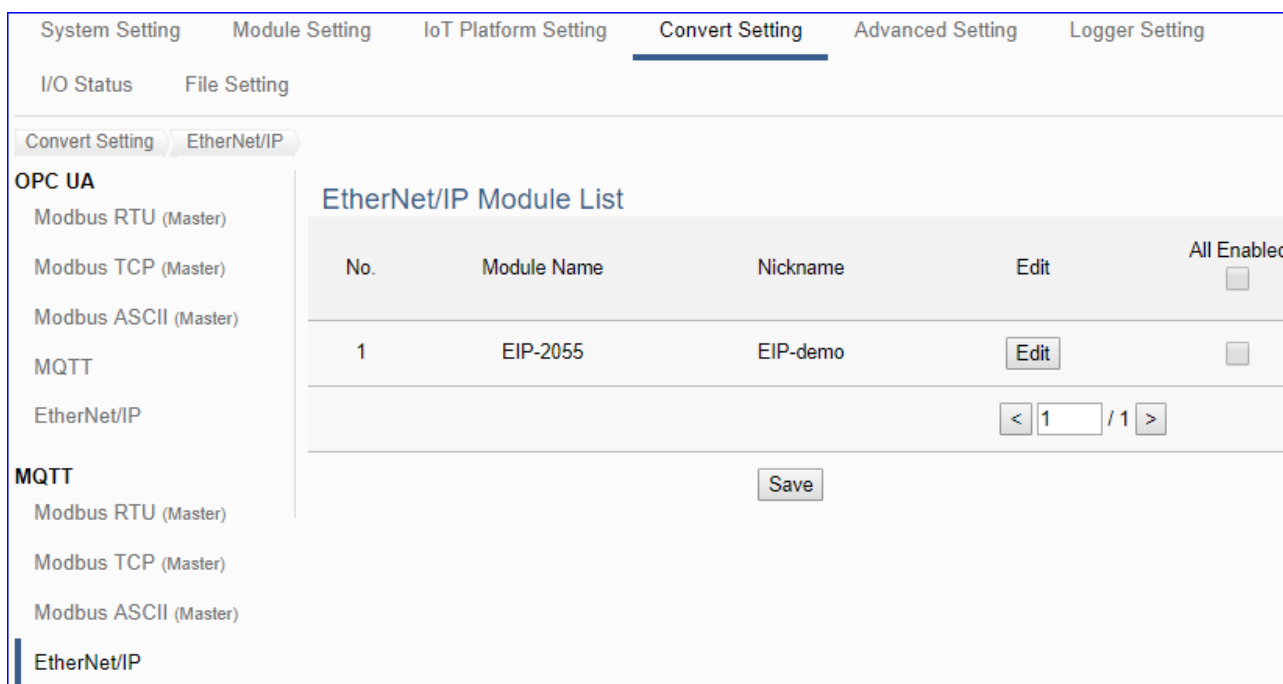
#### Function Diagram:



#### Application Solution:



When entering the menu [Convert Setting] and the sub-menu [MQTT] > EtherNet/IP, the ICP DAS EtherNet/IP modules EIP Series preset in the [Module Setting] will show up in the Module List. (Refer to [Chapter 5.2](#) for the Module Setting.)



Convert Setting > MQTT > EtherNet/IP Module List	
No.	The module number in the module list (Not editable here)
Module Name	The EIP series model selected in the module list (Not editable here)
Nickname	The user defined name for the module (Not editable here)
All Enabled <input type="checkbox"/> Enable <input type="checkbox"/>	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 enable I/O or set up the Topic, QoS, Publish, Subscribe ...
<input type="button" value=" &lt; 1 / 1 &gt;"/>	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.

**This function is to enable the communication conversion module, please  check the box of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to 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 > EtherNet/IP – MQTT Client Setting	
No.	The module number in the module list (Un-editable)
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"/>

Convert Setting > MQTT > EtherNet/IP – 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.

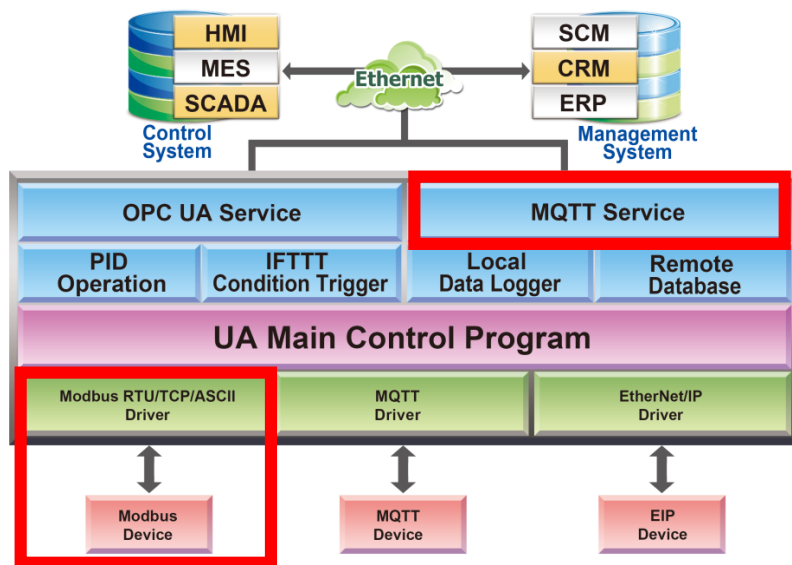


### 5.4.9 MQTT JSON and Modbus RTU/ASCII Conversion

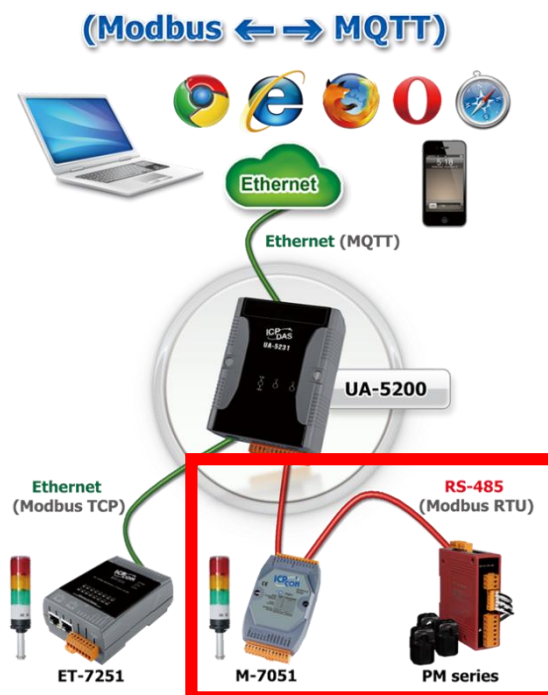
This page provides MQTT JSON and Modbus RTU/ASCII (Master) communication protocol conversion. With the MQTT Service function, users can set the MQTT client to publish the message to the specified broker or subscribe the topic, and combine several messages that converted in JSON format into a group to 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 7](#).

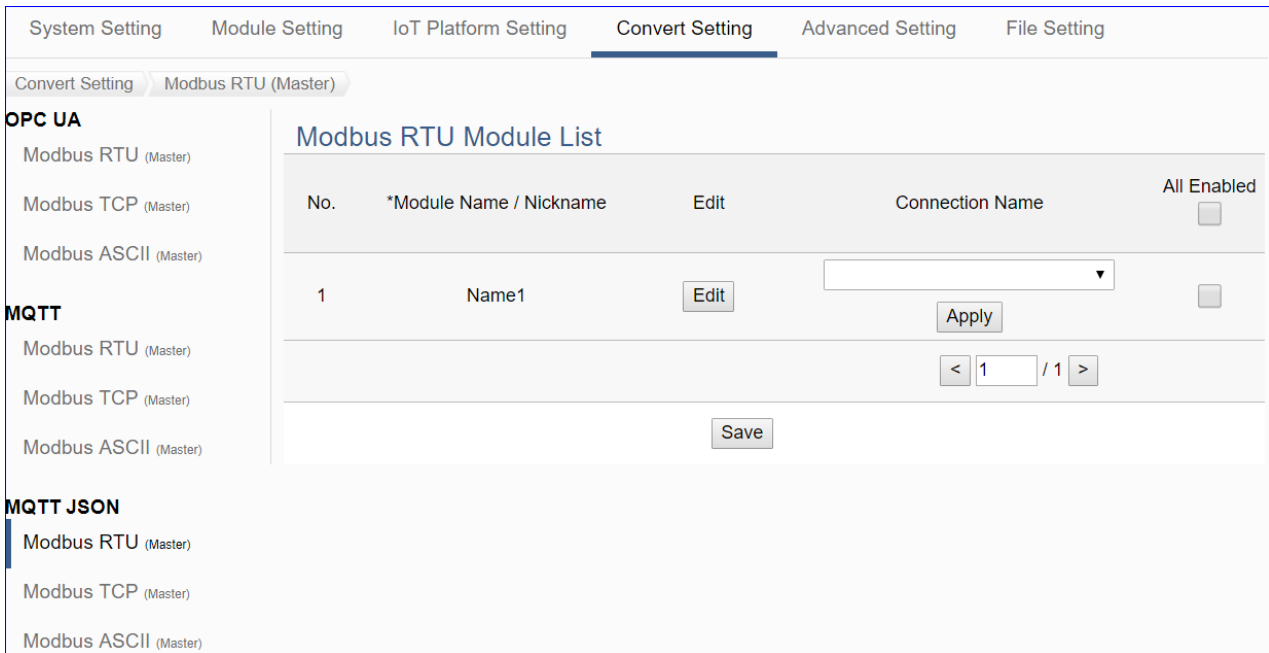
**Function Diagram:**



**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 5.2](#) 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 <input type="checkbox"/> <input type="checkbox"/> Enable	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 enable I/O or check the “Module Content Setting” and “Variable Tale” page.
<input type="button" value="&lt;"/> <input type="text" value="1"/> / <input type="text" value="1"/> <input type="button" value="&gt;"/>	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.

This function is to enable the communication conversion module, first select the [Connection Name] and click [Apply] button, and then check the box  of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to enter the “Module Content Setting” page.

### Module Content Setting

No.	<input style="width: 60%;" type="text" value="1"/>
Module Name	<input style="width: 60%;" 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: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read ▾	Float	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read / Write ▾	Short	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read ▾	Bool	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag1"/>	<input style="width: 80%;" type="text" value="Tag1"/>	Read ▾	Bool	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read / Write ▾	Bool	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag1"/>	<input style="width: 80%;" type="text" value="Tag1"/>	Read / Write ▾	Bool	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>

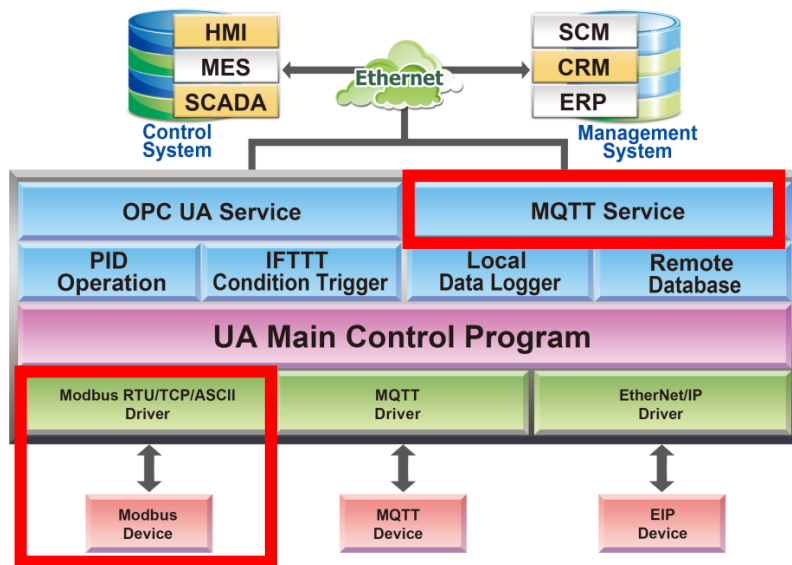
<b>Convert Setting &gt; MQTT JSON &gt; Modbus RTU (Master) – Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Convert Setting &gt; MQTT JSON &gt; Modbus RTU (Master) – Variable Table</b>	
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 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.

### 5.4.10 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 7](#).

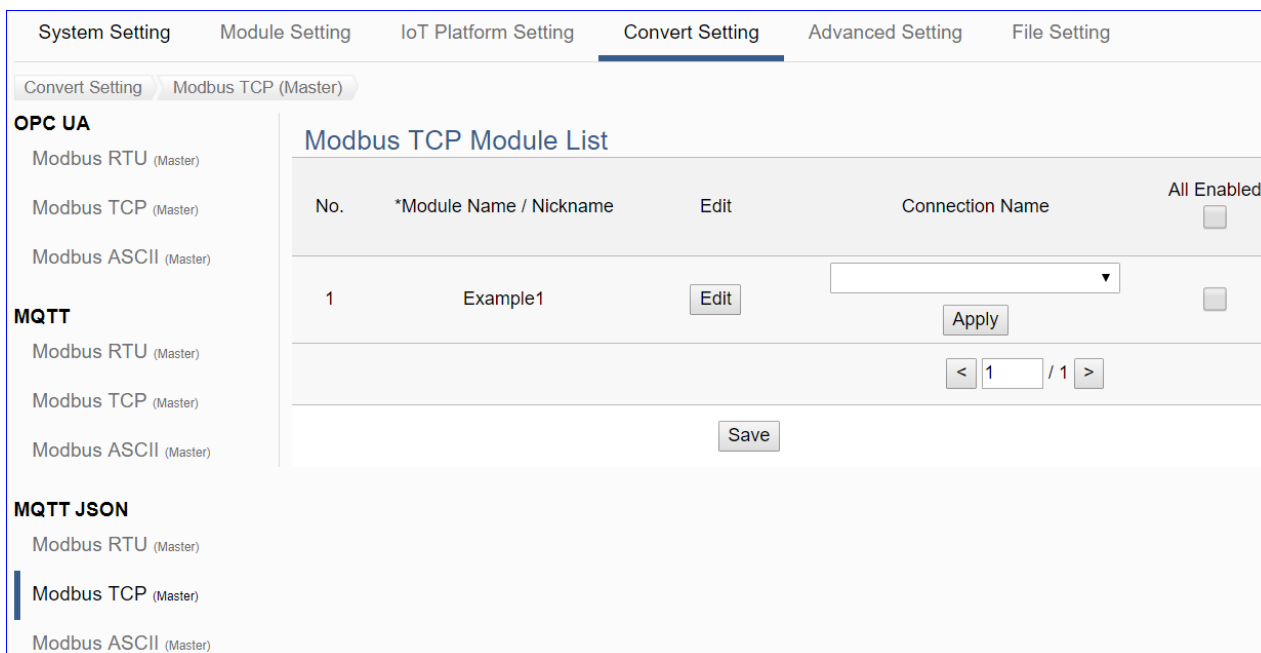
#### Function Diagram:



#### 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 5.2 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 <input type="checkbox"/> Enable <input type="checkbox"/>	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.
<input type="button" value="&lt;"/> <input type="text" value="1"/> / <input type="text" value="1"/> <input type="button" value="&gt;"/>	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.

This function is to enable the communication conversion module, first select the [Connection Name] and click [Apply] button, and then check the box  of the converting module. If user wants to enable some I/O of the module, please click [Edit] button to enter the “Module Content Setting” page.

### Module Content Setting

No.	<input style="width: 60%;" type="text" value="1"/>
Module Name	<input style="width: 60%;" 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: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read ▾	Float	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read / Write ▾	Short	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read ▾	Bool	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag1"/>	<input style="width: 80%;" type="text" value="Tag1"/>	Read ▾	Bool	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag0"/>	<input style="width: 80%;" type="text" value="Tag0"/>	Read / Write ▾	Bool	<input style="width: 80%;" type="text"/>	<input type="checkbox"/>
<input style="width: 80%;" type="text" value="Tag1"/>	<input style="width: 80%;" type="text" value="Tag1"/>	Read / Write ▾	Bool	<input style="width: 80%;" 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 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.

## 5.5 Main Menu: 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 or cloud service function. The description is on the page of the Main Menu. It will support more functions in the future.

The items in the advanced setting functions are “PID Operation” and “IFTTT Condition Trigger” and “Data Logger” that includes “Local Data Logger”, “MS SQL” and “MySQL / MariaDB”. This chapter will introduce the function items and setting parameters.

Advanced Setting	
PID Operation	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.
IFTTT Condition Trigger	With the IFTTT cloud platform, the users can send messages to IFTTT-related cloud services such as Line, Facebook, Twitter, etc. when the special events occur.
Data Logger	
Local Data Logger	Set local data log.
MS SQL	Set the MS SQL data log.
MySQL / MariaDB	Set the MySQL / MariaDB data log.

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

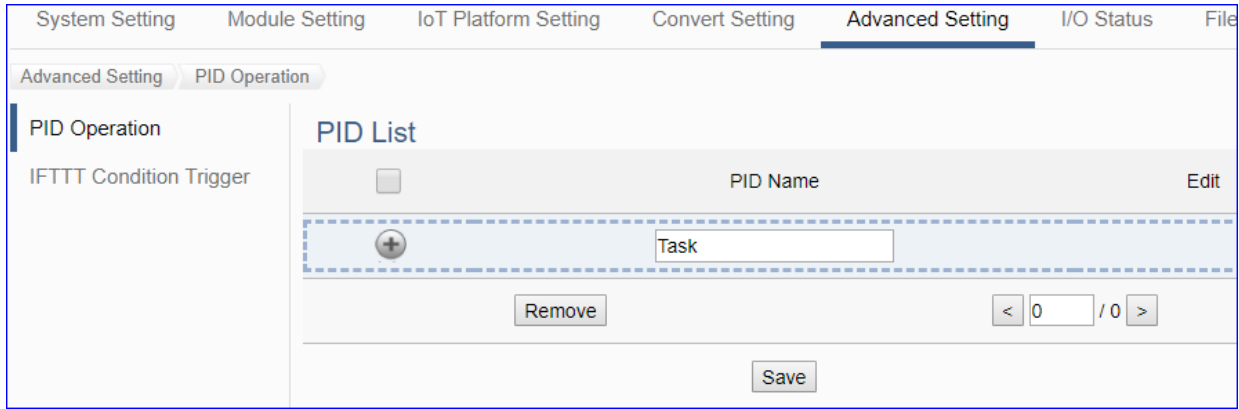
[CH2 Quick Start 1: Hardware/Network Connection](#)

[CH3 Quick Start 2: Web UI / Setting Steps](#)

[CH4 Function Wizard: Project Quick Setup & Example](#)

### 5.5.1 PID Operation

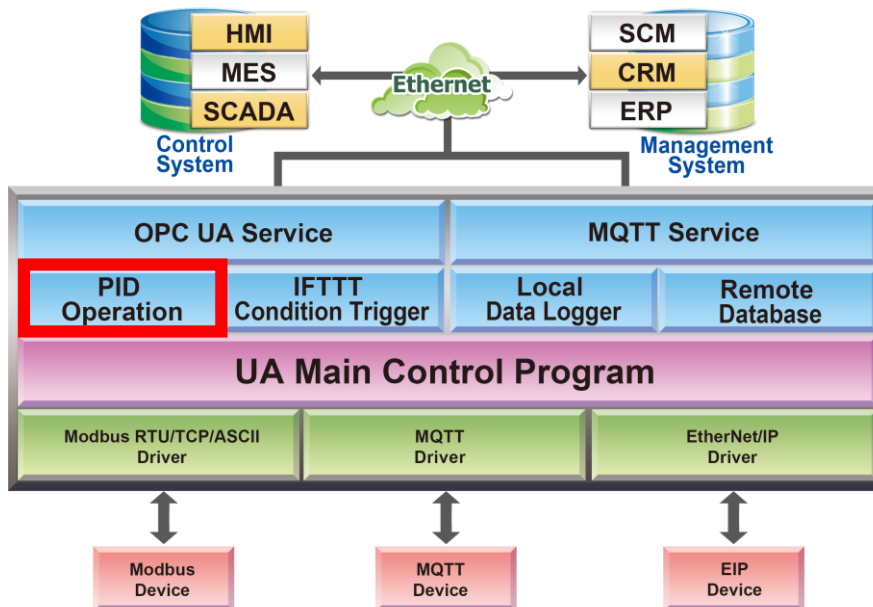
This page is about the virtual device function to allow users to simulate various devices with the real I/O by using the tuning function of PID operation.



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.

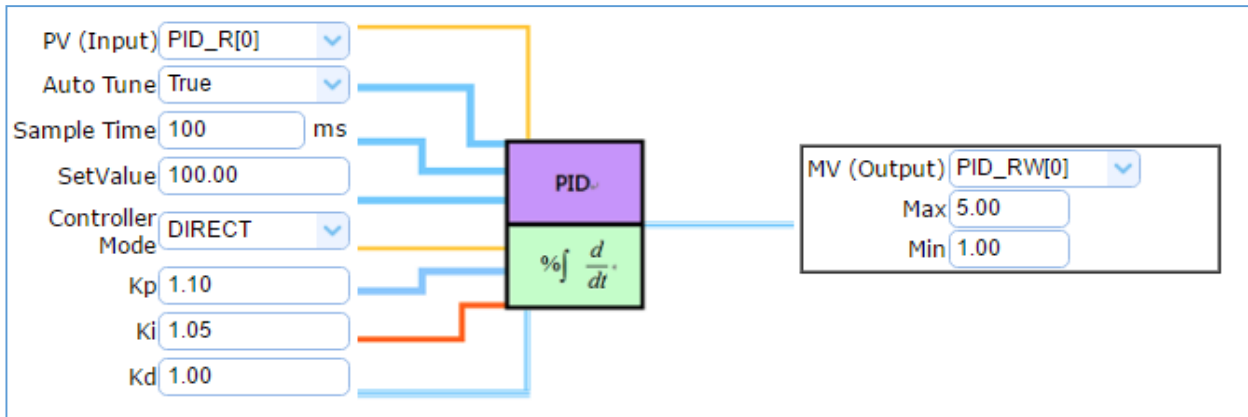
The PID controller is a common feedback loop component in industrial control applications. The controller compares the collected data with a reference value and then uses this difference to calculate a new input value whose purpose is to allow the system data to reach or remain at the reference value.

#### Function Diagram:





**PID Operation Solution Example:**



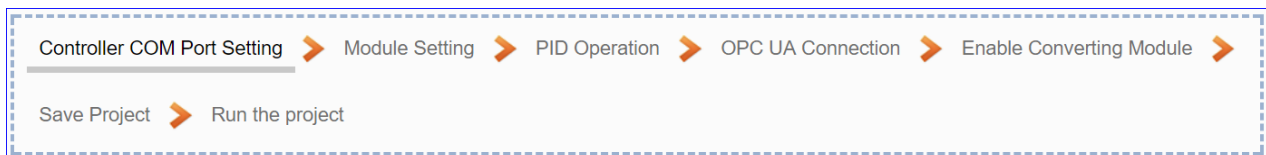
In the PID Operation function, UA controller collects the module’s data to operate via the feedback loop component of PID control. The controller compares the collected data with a reference value and then uses this difference to calculate a new input value whose purpose is to allow the system data to 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 4.4 “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 type="checkbox"/>	Task1	<input type="button" value="Edit"/>
<input type="button" value="Remove"/>		<input type="button" value="&lt;"/> <input style="width: 30px; text-align: center;" type="text" value="1"/> <input type="button" value="/"/> <input style="width: 30px; text-align: center;" type="text" value="1"/> <input type="button" value="&gt;"/>
<input type="button" value="Save"/>		

**Advanced Setting > PID Operation > PID List**

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: 30px; text-align: center;" type="text" value="1"/> <input type="button" value="/"/> <input style="width: 30px; text-align: center;" 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] botton to enter the [Content Settings] page:

**Content Settings**

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

**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"/> <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"/>

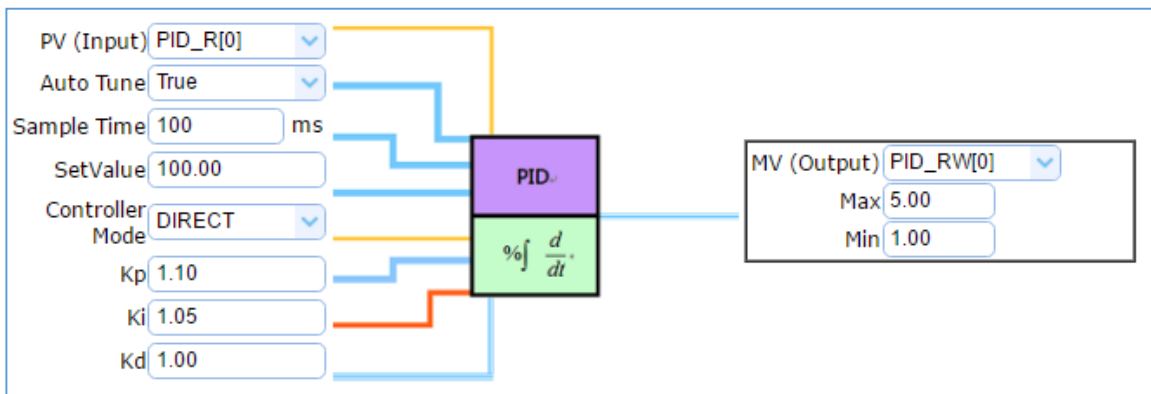
Advanced Setting > PID Operation > Input Item	
Module selection	Choose a predefined module for input data of the PID. Select the type, number and name of the input module. If no option is available, add a new module.
Variable selection	Choose a predefined float variable as the input parameter for PID operation. Select the attribute, type and name of the float variable.
Auto Tune	Enable: Auto-tuning PID parameters for your system. Default: check. Un-Enable: 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	

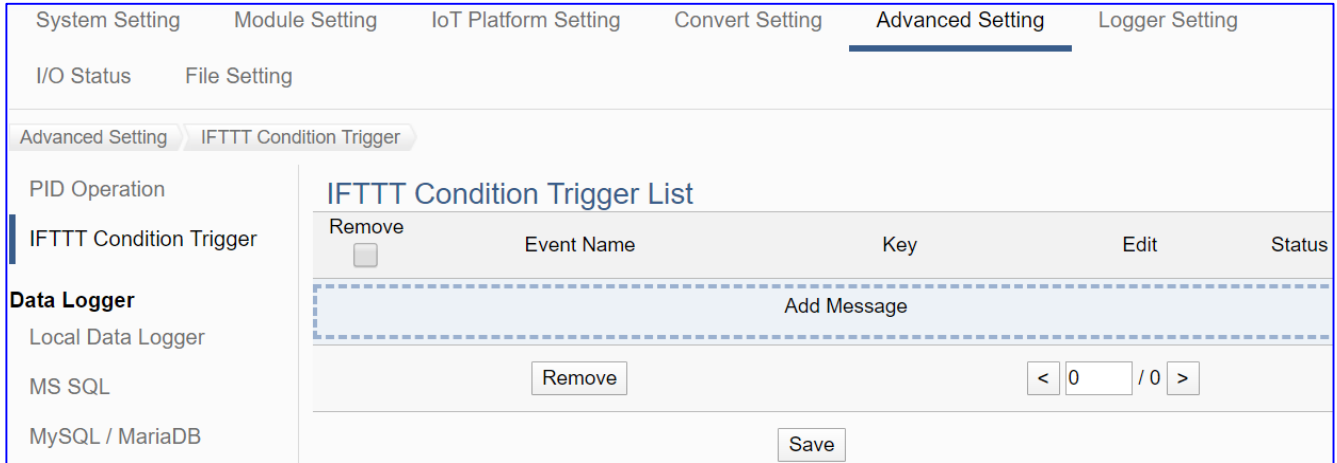
Advanced Setting > PID Operation > Output Item	
Module selection	Choose a predefined module for output data of the PID. Select the type, number and name of the input module. If no option is available, add a new module.
Variable selection	Choose a predefined float variable as the output parameter for 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.

**PID Operation Solution Example:**



### 5.5.2 IFTTT Condition Trigger

This page is about use the IFTTT cloud platform function. Combine with the IFTTT Condition Trigger function, when the special events occur, the users can send messages to IFTTT-related cloud services (such as Line, Twitter, etc.).



IFTTT (if this then that) is a cloud service platform that easy to get your apps and devices working together via creating chains of simple conditional statements (applets). An applet is triggered by changes that occur within other web services such as Line, Twitter, Gmail, Instagram, etc. For example, “if” Line (Service A) has a new message, “then” send an email to Gmail (Service B).

UA using the IFTTT cloud platform functions, the users can send messages to cloud services such as Line, Twitter, etc. when the special events occur.



The settings for sending the message to the APP with the "IFTTT Condition Trigger (Line, Twitter)" function includes two parts:

1. **IFTTT Cloud Platform Setting:** (Refer to [FAQ-005](#).)

In the IFTTT website, set up the “if” side service and event (**this**: use **webhooks** for the UA), the “then” side service and action (**that**: user can select the service, such as the Line, twitter, etc.). And then fill the “**Event Name**” and “**Key**” getting from the IFTTT website setting into the “**Content Setting**” of the UA We HMI. (Detail in the [FAQ-005](#).)



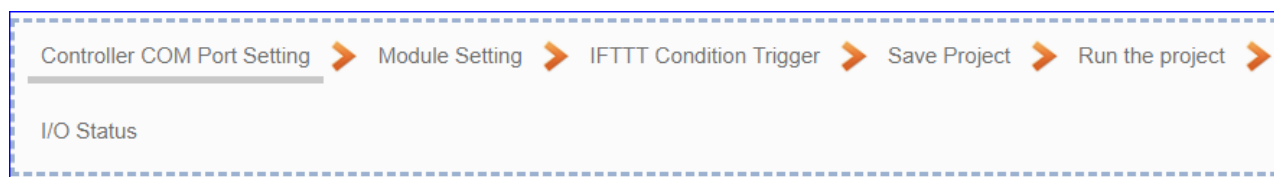
2. **UA Web Interface Setting:** ([Sec.5.5.2 Advanced Setting > IFTTT Condition Trigger](#))

In the UA Web HMI, set up the UA controller, modules, IFTTT trigger conditions, the condition variable table, and the IFTTT event connection. (Fill the **Event Name and Key** from IFTTT website into the “**Content Setting**” of the UA Web UI.)

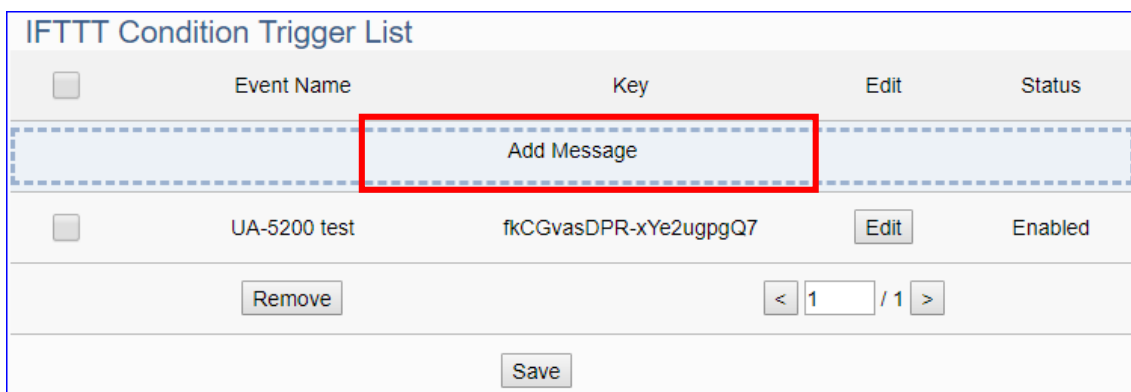
This section introduces the setting part on the UA Web Interface, including the IFTTT trigger condition, variable table and the event message. About the setting on the IFTTT Cloud Platform, user can set up on the IFTTT website and get the “**Event Name**” and “**Key**” for the configuration here. If you are not familiar about the IFTTT, please refer to the [FAQ-005](#).

For the whole steps to send the message to an APP from setting the UA controller, module, I/O variables to the IFTTT Condition Trigger, the users can refer to the [Section 4.5](#) and the step box below.

**[Step Box : IFTTT Condition Trigger (Line, Twitter)]:**



This section will introduce the setting of the IFTTT condition trigger list, variable table and the event message.



**Advanced Setting > IFTTT Condition Trigger > FTTT Condition Trigger List**

Add Message	Click to add a new IFTTT message. After setting, an IFTTT condition trigger list will show on the bottom, includes left box, event name, key and status.
<input type="checkbox"/>	Check the box in the left of the list is to select and to delete the list. Check the box on the top will select all lists.
Event Name	Display the “Event Name” setting in the IFTTT website. (FAQ-005)
Key	Display the “Key” getting from the IFTTT website. (FAQ-005)
Edit	Click [Edit] can set the IFTTT condition trigger content.
Status	Display the enable status of the IFTTT condition trigger list.
Remove	Click the left box and [remove] can delete the IFTTT list.
<input type="text" value="1"/>	The page number of the IFTTT 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 **[Add Message]** button to enter the IFTTT [Content Settings] page:

**Content Setting**

Event Name	<input style="width: 90%;" type="text" value="UA-5200 test"/>
Key	<input style="width: 90%;" type="text" value="fkCGvasDPR-xYe2ugpgQ7"/>
Status	<input checked="" type="checkbox"/> Enabled

**Note:** The “Event Name” and “Key” are set in the IFTTT website. If you are not familiar with IFTTT, please see the [FAQ-005](#) for the setting introductions.

**Advanced Setting > IFTTT Condition Trigger > Content Setting**

Event Name	Input the “Event Name” setting in the IFTTT website. (FAQ-005)
Key	Input the “Key” getting from the IFTTT website. (FAQ-005)
Status	Check to enable the IFTTT condition trigger event.

Condition Setting		Module Variables	Operator	Value
↓ Module Type	Modbus RTU (Master) ▼			
↓ Module Name	No.1 M-7 ▼			Type : User-Defined ▼
↓ Variable Attribute	Read ▼	= ▼		Dead Band : 1
↓ Variable Name	Tag0 (Short) ▼			
Add				

The condition setting field may different depending on the selected variable attribute.

Condition Setting		Module Variables	Status
↓ Module Type	Modbus RTU (Master) ▼		
↓ Module Name	No.2 M-7055D ▼		Status Change ▼
↓ Variable Attribute	Read ▼		
↓ Variable Name	Tag0 (Bool) ▼		

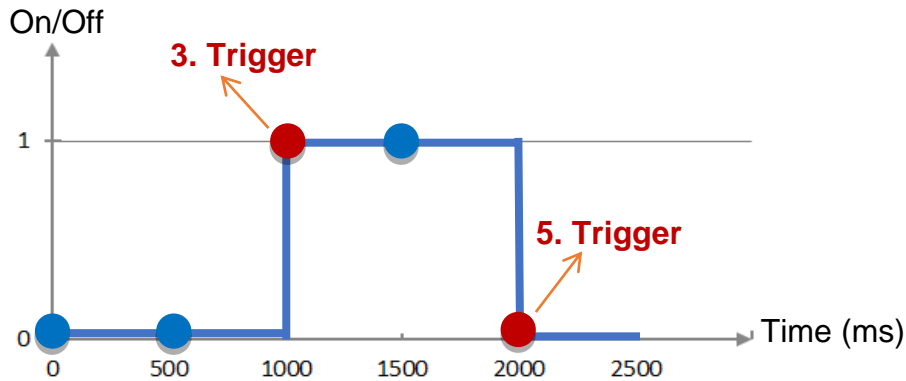
Advanced Setting > IFTTT Condition Trigger > Condition Setting	
Module Variables	Select the module and variable for the condition trigger. Module Type: select the module type, Modbus RTU/TCP/ASCII... Module Name: select the module that set for condition trigger. Variable Attribute: select the variable attribute for condition trigger. Variable Name: select the variable name for condition trigger.
The following condition fields may different depending on the selected variable attribute. The condition trigger method will be described after this table.	
Operator	Select the operator for the trigger condition.
Value	Set up the value for the condition, include Type and Dead Band.
Status	Set up the status for the condition. Default: 0.
Add	Click to add a condition trigger list in the Condition Table..



## Condition Trigger Descriptions:

The condition trigger method will differ depending on the attribute of the selected variable and the trigger will be different. There are two operation styles: **DIO** and **AIO**.

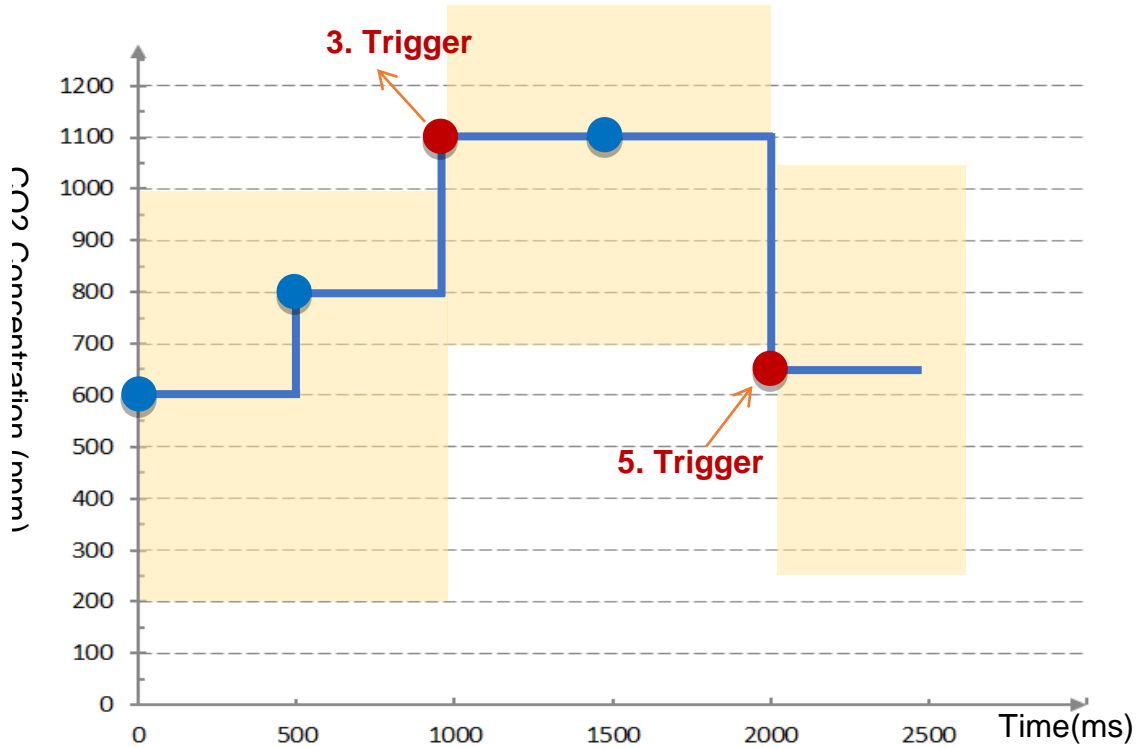
**(A)** If select **DIO variable**, then Condition is "Status Change". When detecting the status is changed, it will trigger the event and send the assigned message. (Below is a switch detecting example.)



**DIO Trigger:** (Detect per 500 ms)

1. Detect initial switch status "Off" (status = 0)
2. Detect "Off" (status = 0, status no change), no trigger
3. Detect "On" (status = 1, status changed), trigger a message notification
4. Detect "On" (status = 1, status no change), no trigger
5. Detect "Off" (status = 0, status changed), trigger a message notification

(B) If select **AIO variable**, then Condition is “Value” and can set the “Dead Band”. The condition will be triggered and send the message when the detected value exceeds the upper or lower Dead Band. (Below is a CO2 example. Detect per 500 ms)



**AIO Trigger:** (Detect per 500 ms. The yellow block means the Dead Band.)

1. Detect initial CO2 concentration 600 (ppm).  
Set Dead Band=400 (Initial Trigger Condition:  $\geq 1000$  or  $\leq 200$ )
2. Detect CO2 concentration 800. It is in the range of Dead Band.
3. Detect CO2 concentration 1100. It exceeds the upper value ( $\geq 1000$ ) of Dead Band, so trigger a message for danger notification.
4. Detect CO2 concentration 1100. It is in the new range of Dead Band.  
Dead Band=400 (New Trigger Condition:  $\geq 1500$  or  $\leq 700$ )
5. Detect CO2 concentration 650. It is below the lower value ( $\leq 700$ ) of Dead Band, so trigger a message for safety notification.

Please refer to the previous Condition Trigger Descriptions to set up your Condition. When complete, click the “Add” button. The setting will show in the Condition Table. Below Table is setting 2 conditions.

Advanced Setting > IFTTT Condition Trigger > Condition Table	
Module	Display the module type and name of the condition. (Not editable here)
Variable	Display the variable attribute and name of the condition. (Not editable here)
Condition	Display the trigger condition. (Not editable here)
Define Message	Default Message: module code_variable code. The user can define own message in the format of English character, number, general symbol...
Remove	Click the left box and [remove] can delete the IFTTT list.
OK	Click to save this page settings and back to the module list page.
Cancel	Click to exit without saving and back to the module list page.

When back to the IFTTT Condition Trigger List, the condition trigger message will show as below picture. If need more trigger conditions, click the “Add Message” again to combine the IFTTT APP message sending and the UA system. At last, click the Save button.

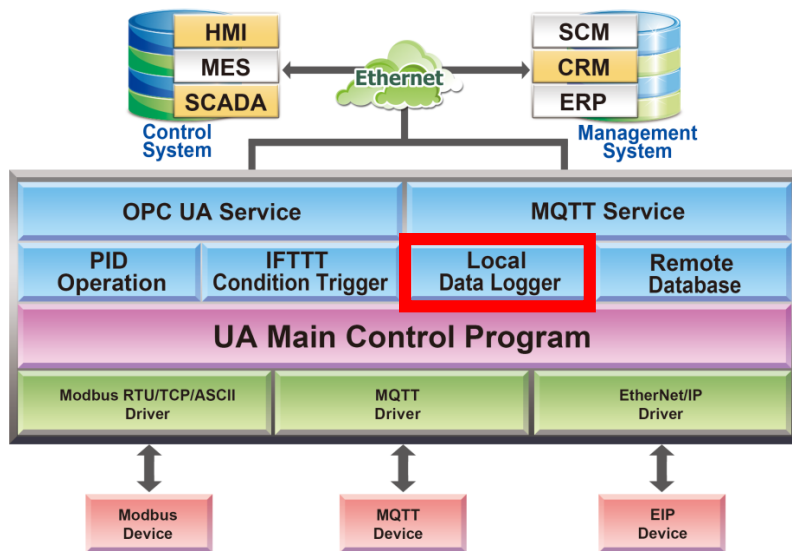
### 5.5.3 Data Logger: Local Data Logger

UA supports Data Logger function to save I/O data into Local CSV log files of the microSD card in UA, or import the I/O data into remote database directly.

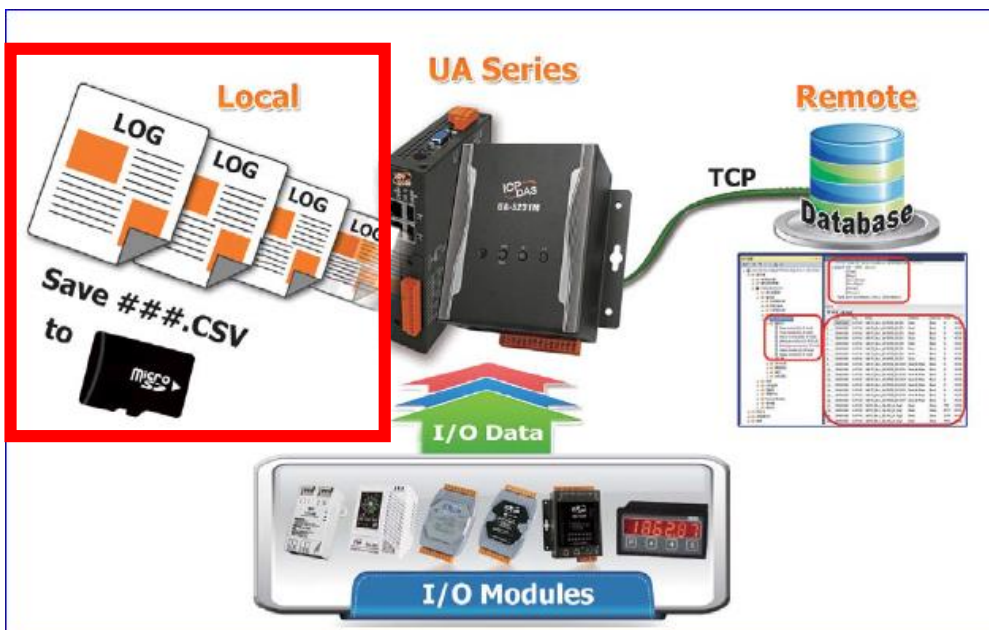
This function is for setting the local data logger and the microSD card. For the setting about the logger and module, please refer to [Chapter 5.6](#).

When the local data logger function is running, the system will allocate memory as a read/write file cache temporarily. When the function runs for a long time, it will increase the memory resource utilization. This situation is normal and does not affect the long-term stability of UA operation.

#### Function Diagram:



#### Application:



Enter the main menu [Advanced Setting] > [Data Logger] > [Local Data Logger]:

System Setting		Module Setting		IoT Platform Setting		Convert Setting		<b>Advanced Setting</b>	
I/O Status		File Setting							
Advanced Setting		Local Data Logger							
PID Operation		<b>Local Data Logger</b>							
IFTTT Condition Trigger		Folder Name		<input type="text" value="Datalog"/>					
<b>Data Logger</b>		File Length		1 hour ▼					
Local Data Logger		Log Interval		1 minute ▼					
MS SQL		Max SD Card Usage Rate(%)		<input type="text" value="90"/>					
MySQL / MariaDB		SD Card Currently Usage Rate		7%					
		SD Card		<input checked="" type="radio"/> Mount <input type="radio"/> Unmount					
		<input type="button" value="Save"/>							

Advanced Setting > Data Logger > Local Data Logger	
Folder Name	The folder name in microSD card of UA, user definable. The I/O data will save into the file “log.csv” under this folder.
File Length	Unit: hour. User can select per 1, 2, 3, ... 8, 12, or 24 hours to divide the log.csv into the file “log-Y-M-D-H-M-S.csv” under the folder “Y-M”. (e.g. 2018-12)
Log Interval	The interval to save I/O data per seconds, minutes or hours.
Max SD Card Usage Rate (%)	Set up the maximum usage rate (Unit: %) of UA microSD card. If the data current rate meet the max rate, the oldest data will be removed first.
SD Card Currently Usage Rate	Display the current usage rate of UA microSD card (show %).
SD Card	Mount: Click to mount microSD card and begin to record data. Unmount: Click to unmount microSD card and stop record data.
Save	Click to save the settings of this item.

● **CVS local data log file: fields and example**

1. The Log record will be stored to the microSD card in the UA PAC, and the default name is the **folder "Datalog"** which can be customized by the user.
2. I/O data records will be stored in the **file "log.csv"** under this folder.
3. The log data file is divided every 1, 2, 3... 8, 12 or 24 hours according to user settings, and saved under the **folder "YYYY-MM"**.
4. Save to the **file "log-YYYY-MM-DD-HH-MM-SS.csv"**. Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

**[ EX ] Folder Name: [Datalog]** ①  
**Save I/O data per 10 sec to the file [log.csv].** ②  
**Divide file per 1 hr to the folder of [Y-M-folder]** ③  
**into the file [log-Y-M-D-H-M-S.csv]** ④

The screenshot shows the 'Local Data Logger' configuration page. The 'Folder Name' is set to 'Datalog', 'File Length' is '1 hour', and 'Log Interval' is '10 seconds'. Callouts 1-4 point to the folder name, the log.csv file, the monthly folder structure, and the specific timestamped CSV file names respectively.

	A	B	C	D
1	# Log file created/rotated Thursday	29 Oct 20 05:50:10 GMT		
2	Timestamp	Name	Value	Status
3	2020-10-29-13-50-10	MRTU_No.1_tM-AD4P2C2_AO.Vin0	11979	Good
4	2020-10-29-13-50-10	MRTU_No.1_tM-AD4P2C2_AO.Vin1	5495	Good
5	2020-10-29-13-50-10	MRTU_No.2_DL-302_AO.CO2	736	Good
6	2020-10-29-13-50-10	MRTU_No.2_DL-302_AO.RH	6371	Good
7	2020-10-29-13-50-10	MRTU_No.2_DL-302_AO.TC	2694	Good
8	2020-10-29-13-50-10	MRTU_No.2_DL-302_AO.TF	8049	Good
9	2020-10-29-13-50-10	MRTU_No.2_DL-302_AO.DC	1947	Good
10	2020-10-29-13-50-10	MRTU_No.2_DL-302_AO.DF	6704	Good
11	2020-10-29-13-50-20	MRTU_No.1_tM-AD4P2C2_AO.Vin0	11979	Good
12	2020-10-29-13-50-20	MRTU_No.1_tM-AD4P2C2_AO.Vin1	5771	Good
13	2020-10-29-13-50-20	MRTU_No.2_DL-302_AO.CO2	734	Good
14	2020-10-29-13-50-20	MRTU_No.2_DL-302_AO.RH	6370	Good
15	2020-10-29-13-50-20	MRTU_No.2_DL-302_AO.TC	2694	Good
16	2020-10-29-13-50-20	MRTU_No.2_DL-302_AO.TF	8049	Good
17	2020-10-29-13-50-20	MRTU_No.2_DL-302_AO.DC	1947	Good
18	2020-10-29-13-50-20	MRTU_No.2_DL-302_AO.DF	6704	Good
19	2020-10-29-13-50-30	MRTU_No.1_tM-AD4P2C2_AO.Vin0	11979	Good

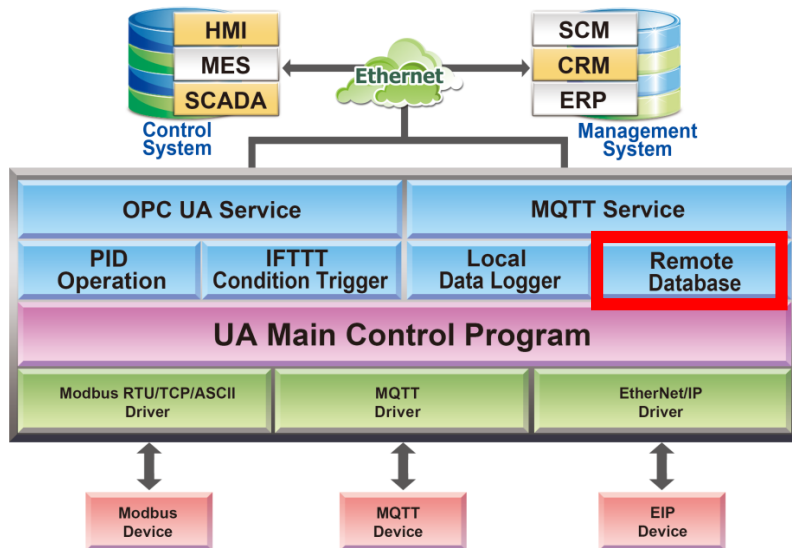
log-2020-10-29-13-51-20

### 5.5.4 Data Logger: MS SQL

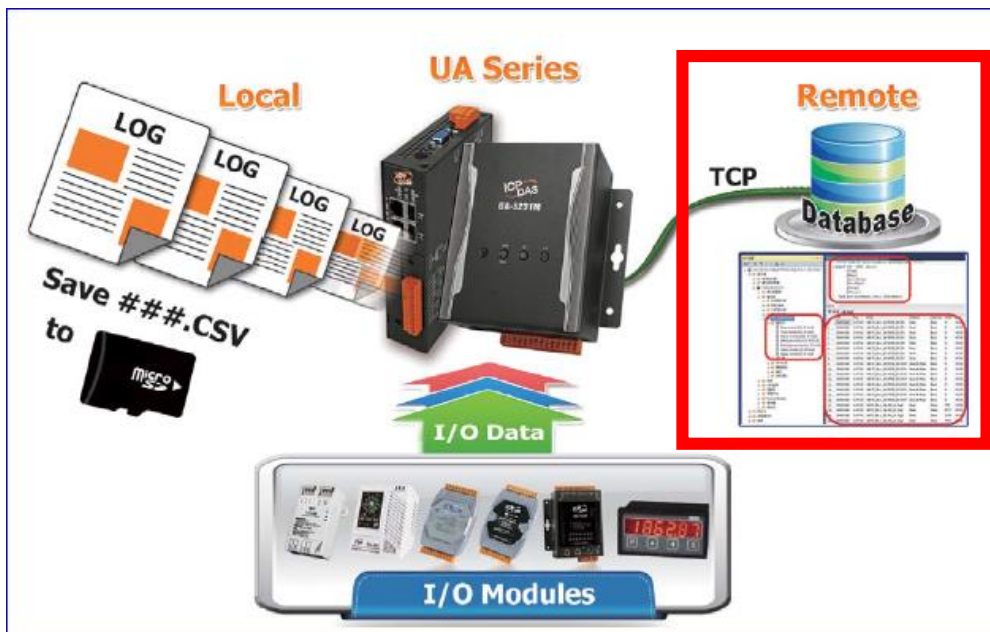
UA supports Data Logger function to save I/O data into Local CSV log files of the microSD card in UA, or write the I/O data directly into the remote database, e.g. MS SQL, MySQL or MariaDB.

This function is for setting the remote database connection. For the setting about the logger and module, please refer to [Chapter 5.6](#).

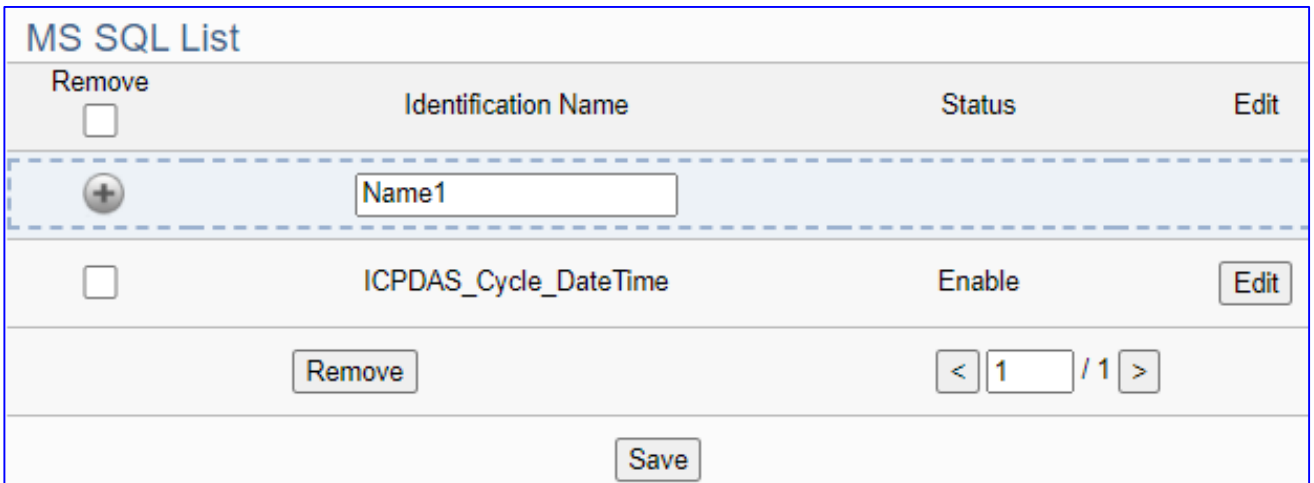
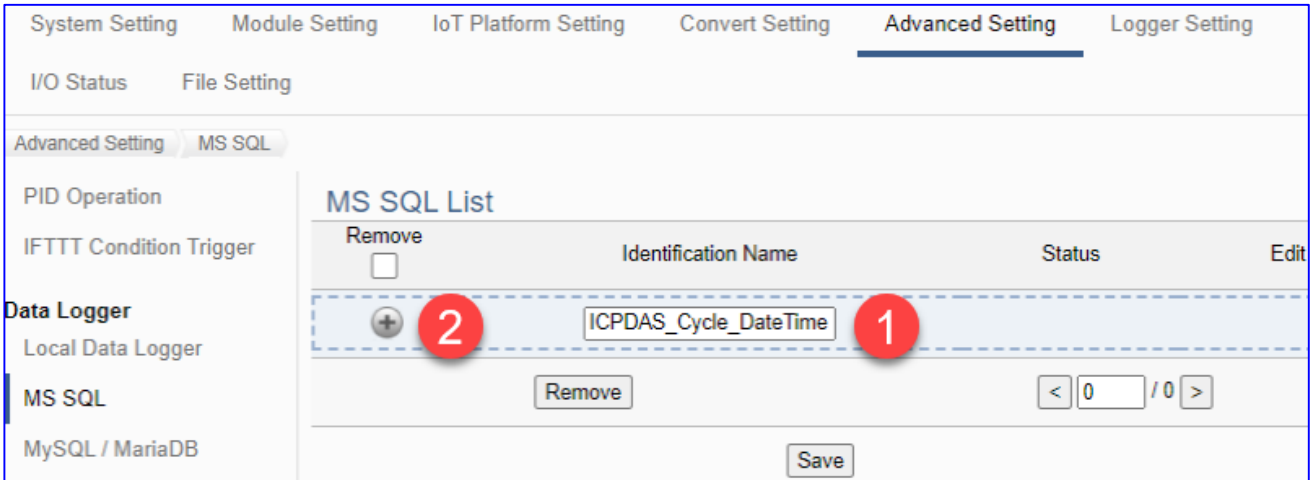
#### Function Diagram:



#### Application:



Enter the main menu [Advanced Setting] > [Data Logger] > [MS SQL] screen, enter a Name (e.g. "ICPDAS\_Cycle\_DateTime"), and click the plus sign to add a MS SQL remote database list as below.



Advanced Setting > Data Logger > MS SQL List	
<input type="checkbox"/> Remove	Check Remove box to remove all database connection in list. Check the box of each database and click the "Remove" button can remove just that database connection.
Identification Name	User defined name to identify the remote database. Default: Name.
Status	Display the status (Enable/Disable) of the database connection. Default: Enable.
<input type="button" value="+"/> +	Click to add a new remote database connection.
Edit	Click to enter the "Content Setting" page of the remote database.
<input type="button" value="&lt; 1 / 1 &gt;"/>	The page number of the database 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" to enter the "MS SQL connect settings" page.



MS SQL content settings	
Identification Name	<input type="text" value="ICPDAS_Cycle_DateTime"/>
Database Name	<input type="text" value="ICPDAS"/>
Table Name	<input type="text" value="Module_All_DateTime"/>
Server Name	<input type="text" value="192.168.85.11\ICPDAS"/>
Port	<input type="text" value="1433"/>
Account	<input type="text" value="root"/>
Password	<input type="password" value="...."/>
Log Mode	<input type="text" value="Cycle"/> ▼
Interval Seconds	<input type="text" value="5"/>
Date Time Format	<input type="text" value="[yyyy-MM-dd HH:mm:ss]"/> ▼
Enable	<input checked="" type="checkbox"/>
Test Connection	<input type="button" value="Connection"/>
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

Advanced Setting > Data Logger > MS SQL – Content Setting	
Identification Name	User defined name to identify the database.
Database Name	The name of the remote database. If not exist, It will create one.
Table Name	The table name of the remote DB. If not exist, It will create one.
Server Name	The Server IP and name of the remote database.
Port	The port to link with database. Default: 1433 (for MS SQL)
Account	The login name of the remote database.
Password	The login password of the remote database.
Log Mode	<b>Cycle:</b> Record one log data at the interval time set below. <b>Data Change:</b> Only record when the data has changed.
Interval Seconds	Set up the interval time to save the I/O data to the remote database. Unit: Second.
Date Time Format	Select to separate the date and time into two [Columns] or combine the date and time in one [Column].
Enable	Enable the data logger to the remote database. Default: check.
Test Connection	Click to test the connection to the remote database. Result: Success or Failure.
OK / Cancel	Click “OK” to save the settings of this page. Click “Cancel” to exit the setting page without saving.

● **MS SQL Remote Database Example Descriptions:**

Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

For database operation, please refer to **FAQ-001** of the **UA series FAQ list**:

**FAQ-001** How to save the UA collected data into SQL and then show trend chart in InduSoft?  
 (Take MS SQL 2017 Express as an example)

The connection screen view of the MS SQL Remote Database.

**1. MS SQL database screen view: Date/Time column separated**

The screenshot displays the SQL Server interface. On the left is the Object Explorer showing the server structure. The main window shows a query window with the following SQL code:

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Date]
, [Time]
, [Name]
, [Attribute]
, [DataType]
, [Value]
, [Status]
FROM [DatabaseName].[dbo].[TableName]
    
```

Below the query window, the Results pane shows the following data:

	Date	Time	Name	Attribute	DataType	Value	Status
1	2020-10-29	17:55:54	MRTU_No.1_IN-AD4P2C2_AO.VinU	Read & Write	Short	11979	GOOD
2	2020-10-29	17:55:54	MRTU_No.1_IN-AD4P2C2_AO.Vin1	Read & Write	Short	6155	GOOD
3	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.CO2	Read & Write	Short	694	GOOD
4	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.RH	Read & Write	Short	6351	GOOD
5	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.TC	Read & Write	Short	2650	GOOD
6	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.TF	Read & Write	Short	7970	GOOD
7	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.DC	Read & Write	Short	1901	GOOD
8	2020-10-29	17:55:54	MRTU_No.2_DL-302_AO.DF	Read & Write	Short	6621	GOOD
9	2020-10-29	17:55:59	MRTU_No.1_IN-AD4P2C2_AO.Vin0	Read & Write	Short	11980	GOOD
10	2020-10-29	17:55:59	MRTU_No.1_IN-AD4P2C2_AO.Vin1	Read & Write	Short	6002	GOOD
11	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.CO2	Read & Write	Short	693	GOOD
12	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.RH	Read & Write	Short	6353	GOOD
13	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.TC	Read & Write	Short	2650	GOOD
14	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.TF	Read & Write	Short	7970	GOOD
15	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.DC	Read & Write	Short	1901	GOOD
16	2020-10-29	17:55:59	MRTU_No.2_DL-302_AO.DF	Read & Write	Short	6621	GOOD

## 2. MS SQL database screen view: Date/Time column combined

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. On the left, the Object Explorer shows the server structure for '192.168.81.5\SQLEXPRESS (SQL S...)', including Databases, System Databases, Database Snapshots, DatabaseName, Database Diagrams, Tables, Views, External Resources, Synonyms, Programmability, Service Broker, Storage, Security, Server Objects, Replication, and PolyBase.

The main window shows a SQL query in the 'SQLQuery2.sql' window:

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [DateTime]
, [Name]
, [Attribute]
, [DataType]
, [Value]
, [Status]
FROM [ICPDAS].[dbo].[Module_All_DateTime]
    
```

Below the query editor, the 'Results' pane shows the output of the query as a table with 16 rows and 6 columns: Date/Time, Name, Attribute, Data Type, Value, and Status. The data is as follows:

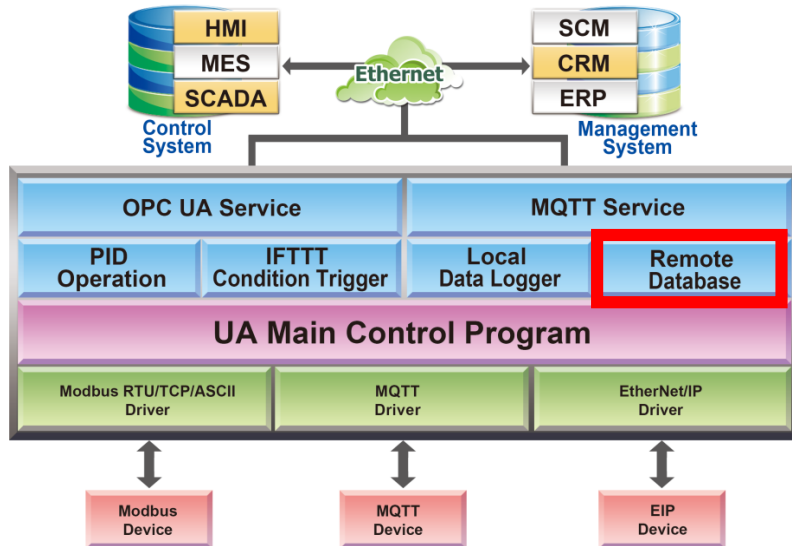
	Date/Time	Name	Attribute	Data Type	Value	Status
1	2020-10-29 10:12:21	MRIU_No.1_0M-AD4F2C2_AO_Vin0	Read & Write	Short	11979	GOOD
2	2020-10-29 10:12:21	MRIU_No.1_0M-AD4F2C2_AO_Vin1	Read & Write	Short	5042	GOOD
3	2020-10-29 10:12:21	MRIU_No.2_DL-302_AO.CO0	Read & Write	Short	662	GOOD
4	2020-10-29 10:12:21	MRIU_No.2_DL-302_AO.RH	Read & Write	Short	6627	GOOD
5	2020-10-29 10:12:21	MRIU_No.2_DL-302_AO.TC	Read & Write	Short	2659	GOOD
6	2020-10-29 10:12:21	MRIU_No.2_DL-302_AO.TF	Read & Write	Short	7996	GOOD
7	2020-10-29 10:12:21	MRIU_No.2_DL-302_AO.DC	Read & Write	Short	1977	GOOD
8	2020-10-29 10:12:21	MRIU_No.2_DL-302_AO.DF	Read & Write	Short	6750	GOOD
9	2020-10-29 10:12:26	MRIU_No.1_0M-AD4F2C2_AO_Vin0	Read & Write	Short	11979	GOOD
10	2020-10-29 10:12:26	MRIU_No.1_0M-AD4F2C2_AO_Vin1	Read & Write	Short	5169	GOOD
11	2020-10-29 10:12:26	MRIU_No.2_DL-302_AO.CO0	Read & Write	Short	662	GOOD
12	2020-10-29 10:12:26	MRIU_No.2_DL-302_AO.RH	Read & Write	Short	6627	GOOD
13	2020-10-29 10:12:26	MRIU_No.2_DL-302_AO.TC	Read & Write	Short	2650	GOOD
14	2020-10-29 10:12:26	MRIU_No.2_DL-302_AO.TF	Read & Write	Short	7994	GOOD
15	2020-10-29 10:12:26	MRIU_No.2_DL-302_AO.DC	Read & Write	Short	1977	GOOD
16	2020-10-29 10:12:26	MRIU_No.2_DL-302_AO.DF	Read & Write	Short	6750	GOOD

### 5.5.5 Data Logger: MySQL / MariaDB

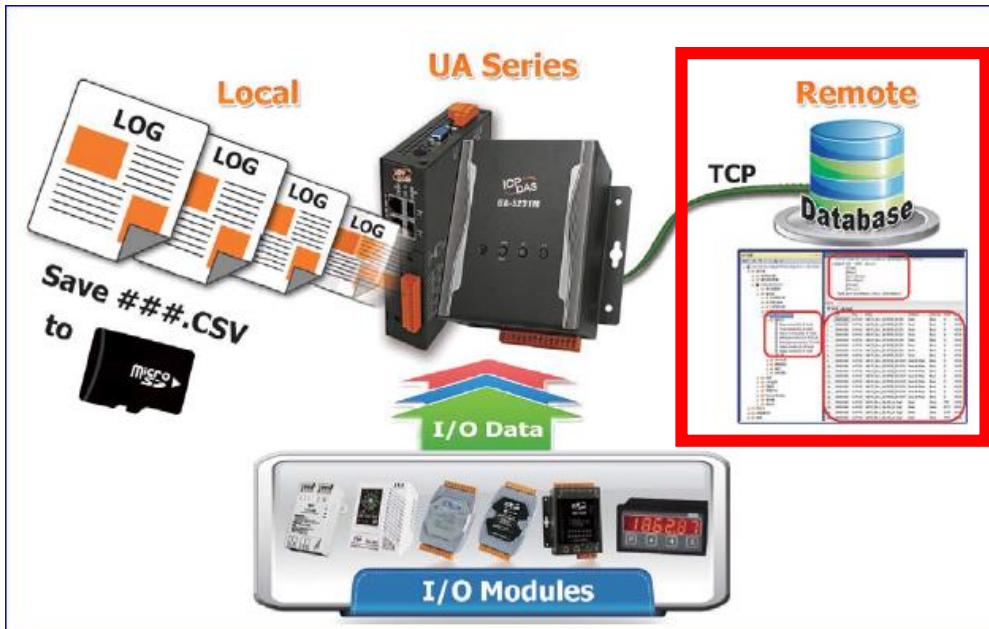
UA supports Data Logger function to save I/O data into Local CSV log files of the microSD card in UA, or write the I/O data directly into the remote database, e.g. MS SQL, MySQL or MariaDB.

This function is for setting the remote database connection. For the setting about the logger and module, please refer to [Chapter 5.6](#).

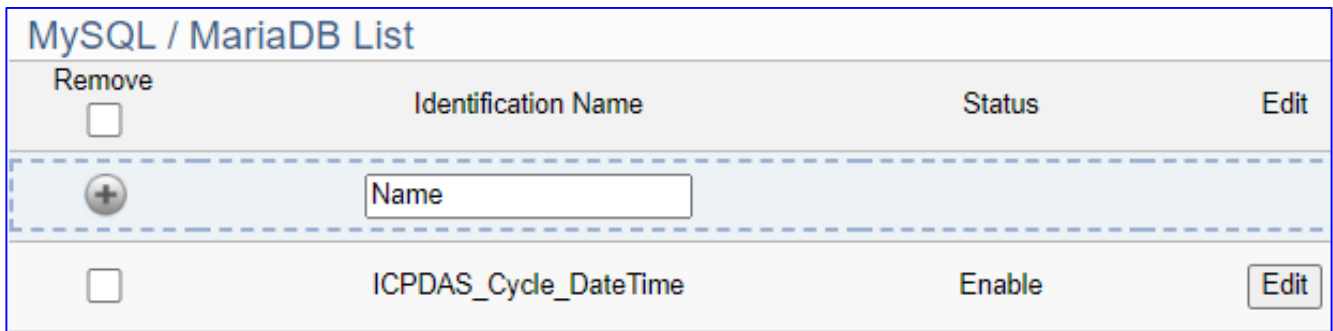
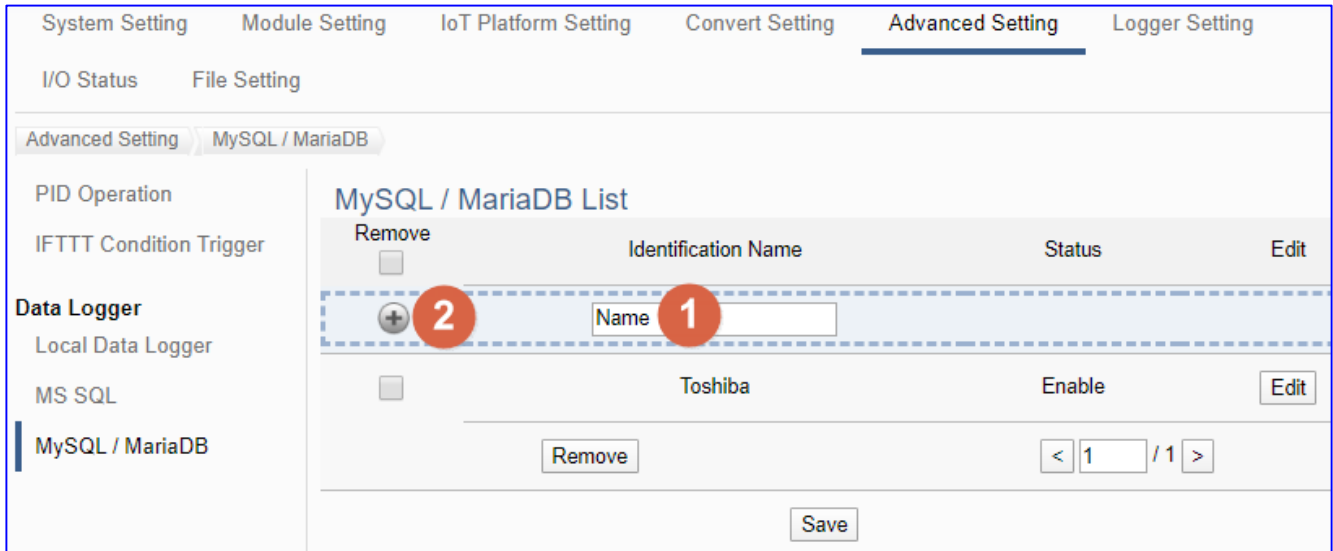
#### Function Diagram:



#### Application:



Enter the main menu [Advanced Setting] > [Data Logger] > [MySQL/MariaDB], enter a name (e.g. "ICPDAS\_Cycle\_DateTime"), and click the plus sign to add a MySQL or DariaDB remote database list.



Advanced Setting > Data Logger > MySQL/MariaDB – MySQL/MariaDB List	
<input type="checkbox"/> Remove	Check Remove box to remove all database connection in list. Check the box of each database and click the "Remove" button can remove just that database connection.
Identification Name	User defined name to identify the remote database. Default: Name.
Status	Display the status (Enable/Disable) of the database connection. Default: Enable.
	Click to add a new remote database connection.
Edit	Click to enter the "Content Setting" page of the remote database.
	The page number of the database 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" to enter the "Remote database connect settings" page.

MySQL / MariaDB Connection Settings	
Identification Name	ICPDAS_Cycle_DateTime
Database Name	ICPDAS
Table Name	Module_All_DateTime
IP	192.168.85.11
Port	3306
Account	chris
Password	....
Log Mode	Cycle
Interval Seconds	5
Date Time Format	[yyyy-MM-dd HH:mm:ss]
Enable	<input checked="" type="checkbox"/>
Test Connection	Connection
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

Advanced Setting > Data Logger > MySQL/MariaDB – Content Settings	
Identification Name	User defined name to identify the database.
Database Name	The name of the remote database. If it does not exist, it will add a new database with this name.
Table Name	The table name of the remote database. If it does not exist, it will add a new table with this name.
IP	The Server IP and name of the remote database.
Port	The port to connect with database. Default: 3306 (for MySQL/MariaDB)
Account	The login name of the remote database.
Password	The login password of the remote database.
Log Mode	<b>Cycle</b> : Record one log data at the interval time set below. <b>Data Change</b> : Only record when the data has changed.
Interval Seconds	Set up the interval time to save the I/O data to the remote database. Unit: Second.
Date Time Format	Select to separate the date and time into two [Columns] or combine the date and time in one [Column].
Enable	Check to enable the data logger to the remote database. Default: check.
Test Connection	Click to test the connection to the remote database. Result: Success or Failure.
OK / Cancel	Click “OK” to save the settings of this page. Click “Cancel” to exit the setting page without saving.

● **MySQL/MariaDB Remote Database Example Descriptions:**

Each tag data and status are recorded in each separate row, **the row is added down for each interval**, and the tag data is recorded in time sequence.

For database operation, please refer to **FAQ-002 (MySQL)** of the **UA series FAQ list**:

[FAQ-002\\_How to save the UA collected data into SQL and then show trend chart in InduSoft?](#)  
(Take MySQL Installer 5.7.31 as an example)

The connection screen view of the **MySQL Remote Database**.

**1. MySQL database screen view: Date/Time column separated (reference)**

The screenshot shows a MySQL database interface with a query result grid. The query executed is `SELECT * FROM icpdas.module_all_date_time;`. The result grid displays the following data:

Date	Time	Name	Value	Status
2020/10/30	11:15:35	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020/10/30	11:15:35	MRTU_No.1_tM-AD4P2C2_AO.Vin1	48	GOOD
2020/10/30	11:15:35	MRTU_No.2_DL-302_AO.CO2	650	GOOD
2020/10/30	11:15:35	MRTU_No.2_DL-302_AO.RH	6170	GOOD
2020/10/30	11:15:35	MRTU_No.2_DL-302_AO.TC	2622	GOOD
2020/10/30	11:15:35	MRTU_No.2_DL-302_AO.TF	7919	GOOD
2020/10/30	11:15:35	MRTU_No.2_DL-302_AO.DC	1828	GOOD
2020/10/30	11:15:35	MRTU_No.2_DL-302_AO.DF	6490	GOOD
2020/10/30	11:15:40	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020/10/30	11:15:40	MRTU_No.1_tM-AD4P2C2_AO.Vin1	42	GOOD
2020/10/30	11:15:40	MRTU_No.2_DL-302_AO.CO2	650	GOOD
2020/10/30	11:15:40	MRTU_No.2_DL-302_AO.RH	6163	GOOD
2020/10/30	11:15:40	MRTU_No.2_DL-302_AO.TC	2621	GOOD
2020/10/30	11:15:40	MRTU_No.2_DL-302_AO.TF	7917	GOOD
2020/10/30	11:15:40	MRTU_No.2_DL-302_AO.DC	1825	GOOD
2020/10/30	11:15:40	MRTU_No.2_DL-302_AO.DF	6485	GOOD
2020/10/30	11:15:45	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020/10/30	11:15:45	MRTU_No.1_tM-AD4P2C2_AO.Vin1	47	GOOD

## 2. MySQL database screen view: Date/Time column combined (reference)

The screenshot shows a MySQL database management tool interface. The left sidebar displays a tree view of schemas, with 'icpdas' expanded to show tables 'module\_all\_date\_time' and 'module\_all\_datetime'. The main window shows a query: `SELECT * FROM icpdas.module_all_datetime;` and a result grid with the following data:

DateTime	Name	Value	Status
2020-10-30 11:12:19	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:19	MRTU_No.1_tM-AD4P2C2_AO.Vin1	54	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.CO2	636	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.RH	6194	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.TC	2616	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.TF	7908	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.DC	1829	GOOD
2020-10-30 11:12:19	MRTU_No.2_DL-302_AO.DF	6492	GOOD
2020-10-30 11:12:24	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:24	MRTU_No.1_tM-AD4P2C2_AO.Vin1	55	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.CO2	636	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.RH	6190	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.TC	2616	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.TF	7908	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.DC	1827	GOOD
2020-10-30 11:12:24	MRTU_No.2_DL-302_AO.DF	6488	GOOD
2020-10-30 11:12:29	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
2020-10-30 11:12:29	MRTU_No.1_tM-AD4P2C2_AO.Vin1	52	GOOD



The connection screen view of the **MariaDB Remote Database**.

**1. MariaDB database screen view: Date/Time column separated (reference)**

The screenshot shows the phpMyAdmin interface for a MariaDB 10 database. The current database is 'ICPDAS' and the selected table is 'Module\_All\_Date\_Time'. The SQL query displayed is 'SELECT \* FROM `Module\_All\_Date\_Time`'. The table contains 256 rows, with the first 24 rows visible. Each row has columns for Date, Time, Name, Value, and Status. The 'Date' and 'Time' columns are separated, as indicated by the section header.

	Date	Time	Name	Value	Status
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.1_tM-AD4P2C2_AO.Vin1	59	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.CO2	637	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.DC	1822	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.DF	6479	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.RH	6099	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.TC	2635	GOOD
<input type="checkbox"/>	2020/10/30	12:14:44	MRTU_No.2_DL-302_AO.TF	7943	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.1_tM-AD4P2C2_AO.Vin1	63	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.CO2	636	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.DC	1819	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.DF	6474	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.RH	6093	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.TC	2634	GOOD
<input type="checkbox"/>	2020/10/30	12:14:49	MRTU_No.2_DL-302_AO.TF	7941	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.1_tM-AD4P2C2_AO.Vin1	64	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.CO2	636	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.DC	1820	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.DF	6476	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.RH	6092	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.TC	2635	GOOD
<input type="checkbox"/>	2020/10/30	12:14:54	MRTU_No.2_DL-302_AO.TF	7943	GOOD

## 2. MariaDB database screen view: Date/Time column combined (reference)

The screenshot shows the phpMyAdmin interface for a MariaDB 10 database named 'ICPDAS'. The selected table is 'Module\_All\_DateTime'. The SQL query displayed is 'SELECT \* FROM `Module\_All\_DateTime`'. The table view shows 25 columns per page, with the first column being 'Date/Time'. The data rows show a list of records with columns: Date/Time, Name, Value, and Status. The status for all records is 'GOOD'.

	Date/Time	Name	Value	Status
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.1_tM-AD4P2C2_AO.Vin1	61	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.DC	1812	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.DF	6461	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.RH	6036	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.TC	2642	GOOD
<input type="checkbox"/>	2020-10-30 12:22:17	MRTU_No.2_DL-302_AO.TF	7955	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.1_tM-AD4P2C2_AO.Vin1	62	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.DC	1812	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.DF	6461	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.RH	6038	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.TC	2642	GOOD
<input type="checkbox"/>	2020-10-30 12:22:22	MRTU_No.2_DL-302_AO.TF	7955	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.1_tM-AD4P2C2_AO.Vin0	146	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.1_tM-AD4P2C2_AO.Vin1	59	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.CO2	640	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.DC	1811	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.DF	6459	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.RH	6038	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.TC	2641	GOOD
<input type="checkbox"/>	2020-10-30 12:22:27	MRTU_No.2_DL-302_AO.TF	7953	GOOD

## 5.6 Main Menu: Logger Setting

**Logger Setting** is the 6<sup>th</sup> item of the Main Menu, mainly to provide the data logger and the connecting modules related settings.

“Logger Setting” provides “Local Data Logger”, “MS SQL” and “MySQL/MariaDB”, and all have RTU/TCP module two setting items. The Local Data Logger provides users to record data such as RTU/TCP module (Master) channel data into Local CSV log files of the microSD card in UA. The “MS SQL” or “MySQL / MariaDB” Remote Database provides users to record data between RTU/TCP module (Master) channel data directly into remote database, such as SQL DB.

Logger Setting	
<b>Local Data Logger</b>	
RTU Module (Master)	Provide users to record data such as RTU module (Master) channel and internal register.
TCP Module (Master)	Provide users to record data such as TCP module (Master) channel and internal register.
<b>MS SQL</b>	
RTU Module (Master)	Provide users to record data between RTU module (Master) channel and MS SQL.
TCP Module (Master)	Provide users to record data between TCP module (Master) channel and MS SQL.
MQTT Module	Provide users to record data between MQTT module channel and MS SQL.
<b>MySQL / MariaDB</b>	
RTU Module (Master)	Provide users to record data between RTU module (Master) channel and MySQL / MariaDB.
TCP Module (Master)	Provide users to record data between TCP module (Master) channel and MySQL / MariaDB.
MQTT Module	Provide users to record data between MQTT channel and MySQL / MariaDB.

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

[CH2 Quick Start 1: Hardware/Network Connection](#)

[CH3 Quick Start 2: Web UI / Setting Steps](#)

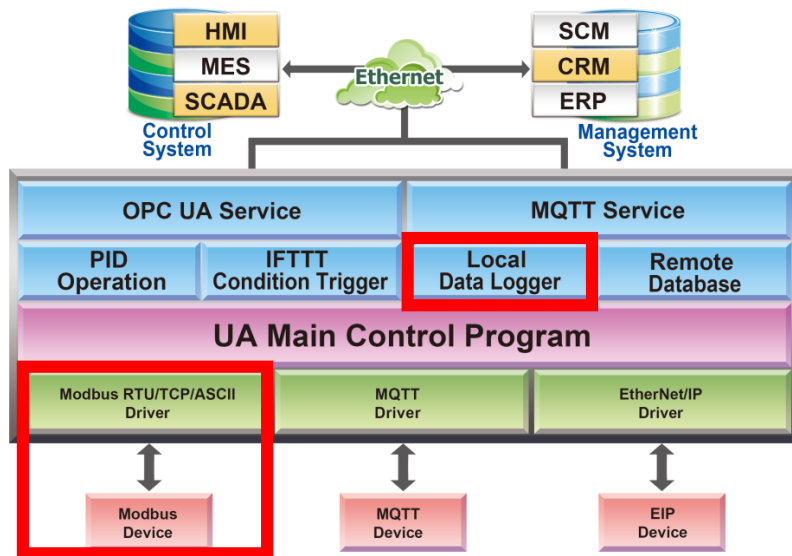
[CH4 Function Wizard: Project Quick Setup & Example](#)

### 5.6.1 Local Data Logger: RTU / TCP Module (Master)

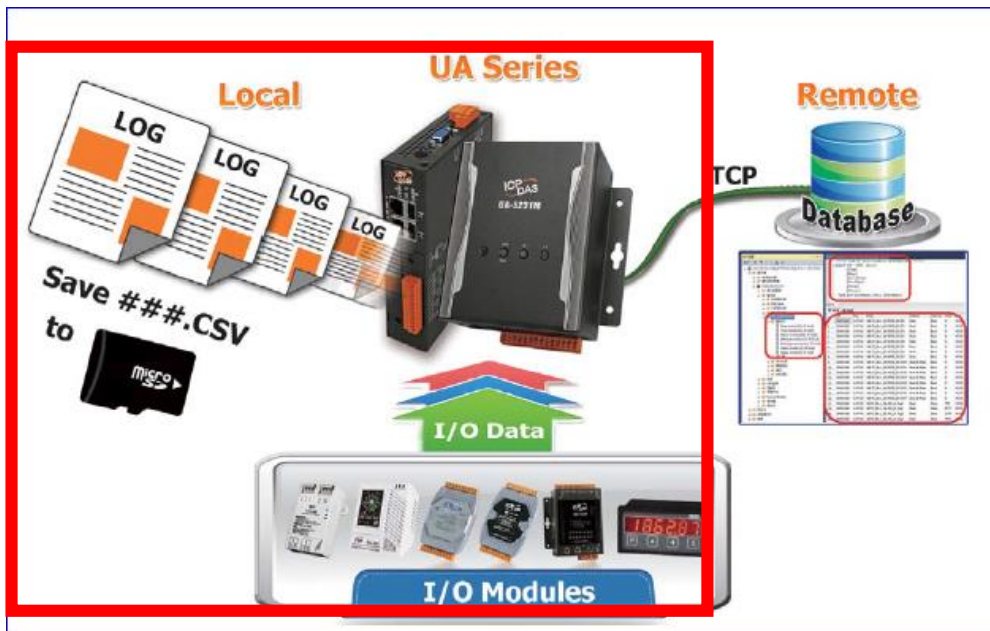
“Local Data Logger” of “Logger Setting” provides users to record I/O log data of the connecting RTU/TCP module (Master) into the local microSD card.

This function is for setting the local data logger and the RTU/TCP modules, using “RTU” module setting as an example. For the setting about the logger and microSD card, please refer to [Chapter 5.5.3](#).

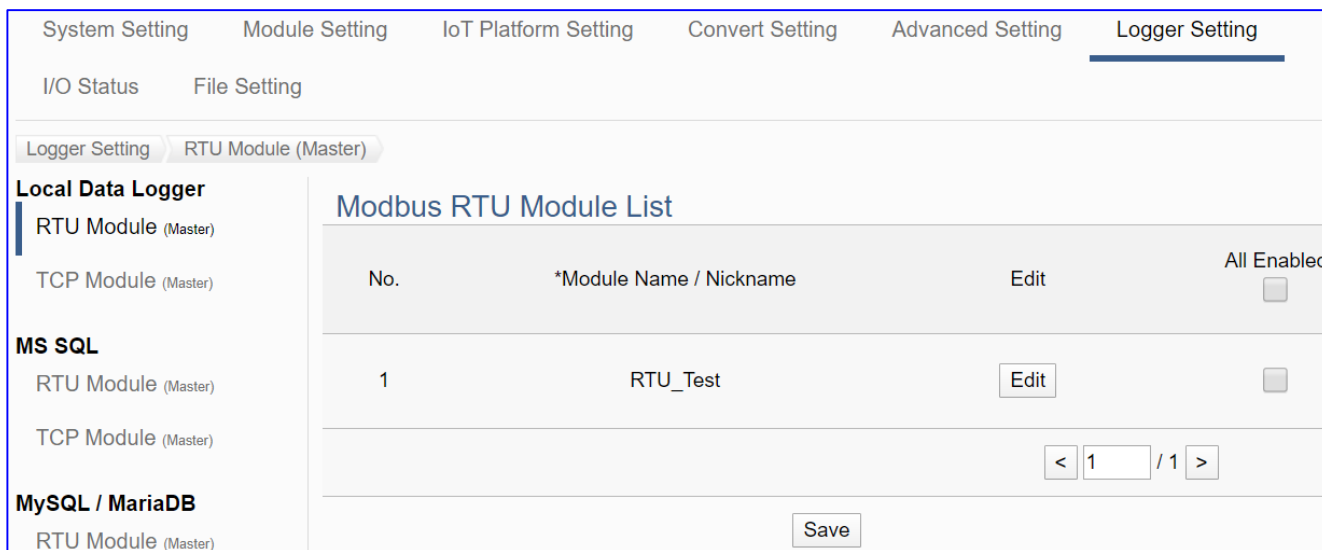
#### Function Diagram:



#### Application:



Enter the main menu [Logger Setting] > [Local Data Logger] > [RTU Module (Master)].  
 This setting page is to enable the module(s) or I/O channels for data logger.



Logger Setting > Local Data Logger - RTU Module (Master) – Modbus 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 data logger, 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.
All Enabled <input type="checkbox"/> <input type="checkbox"/> Enable	Check [All Enabled] box to enable all modules in list for data logger. Default: Uncheck. Check the box of each module can enable just that module for data logger.
<input type="button" value="&lt;"/> <input type="text" value="1"/> / 1 <input type="button" value="&gt;"/>	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.

This function is to enable the module for data logger, please check  the box of the module. If user wants to enable some I/O of the module, please click [Edit] button to enter the “Module Content Setting” page.

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

### Module Content Setting

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

### Variable Table

Variable Name	Attribute	Data Type	Enabled
<input style="width: 90%;" type="text" value="Tag14"/>	<input style="width: 90%;" type="text" value="Read / Write"/>	Unsigned Short	<input checked="" type="checkbox"/>
<input style="width: 90%;" type="text" value="Tag15"/>	<input style="width: 90%;" type="text" value="Read / Write"/>	Unsigned Short	<input checked="" type="checkbox"/>

<b>Logger Setting &gt; Local Data Logger &gt; RTU Module (Master) – 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>Logger Setting &gt; OPC UA &gt; Modbus RTU (Master) – Variable Table</b>	
Variable 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 data logger. Default: Uncheck.
OK	Click to save this page settings and back to the module list page.

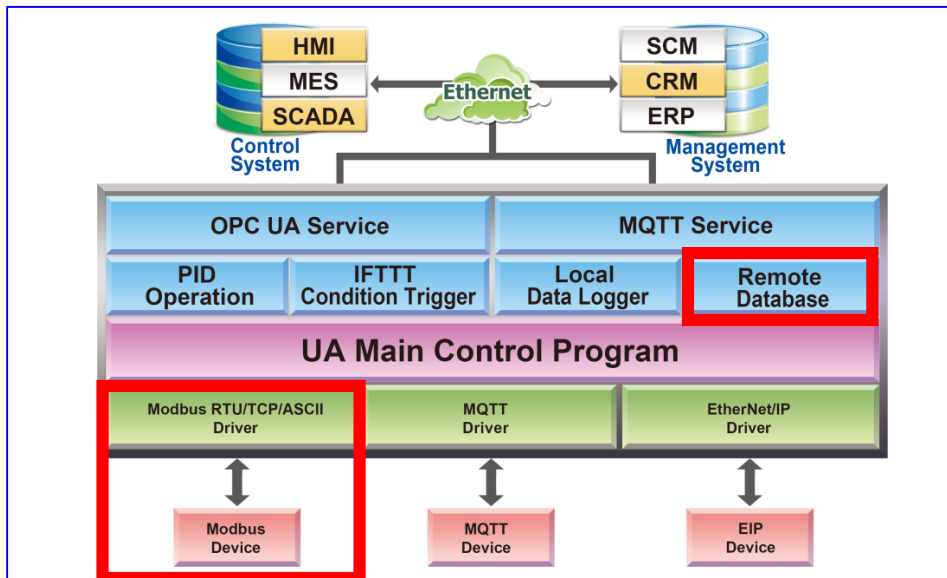
When complete the setting, click [OK] to save this page settings and back to the module list page. Remember to click [Save] to save the Convert Setting.

### 5.6.2 MS SQL: RTU / TCP Module (Master)

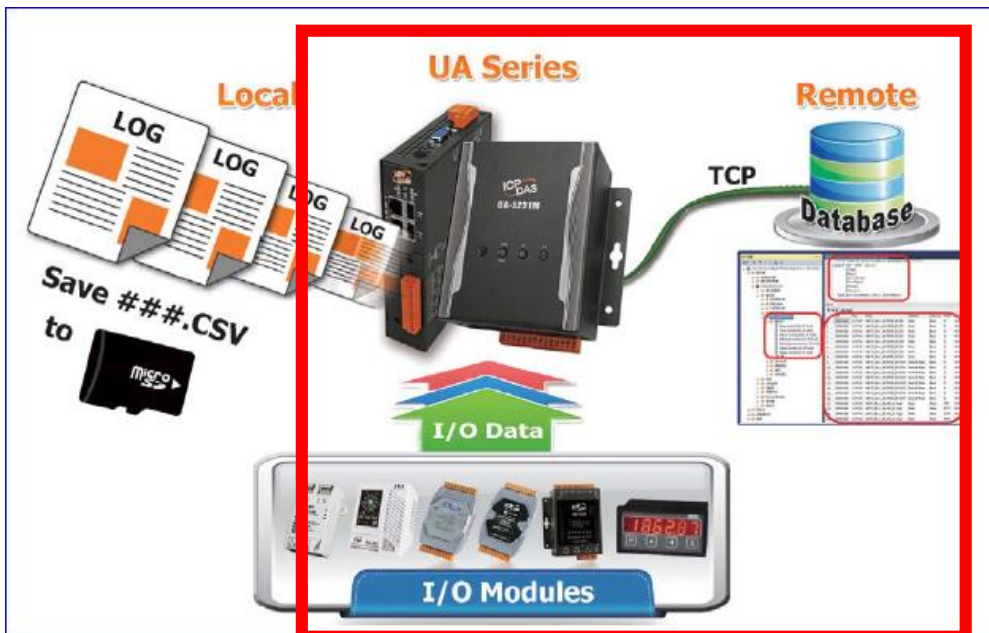
“MS SQL” of “Logger Setting” provides users to record I/O log data of the connecting RTU/TCP module (Master) into the MS SQL remote database.

This function is for setting the remote data logger and the RTU/TCP modules, using “**TCP module**” setting as an example. For the setting about the MS SQL data logger, please refer to [Chapter 5.5.4](#).

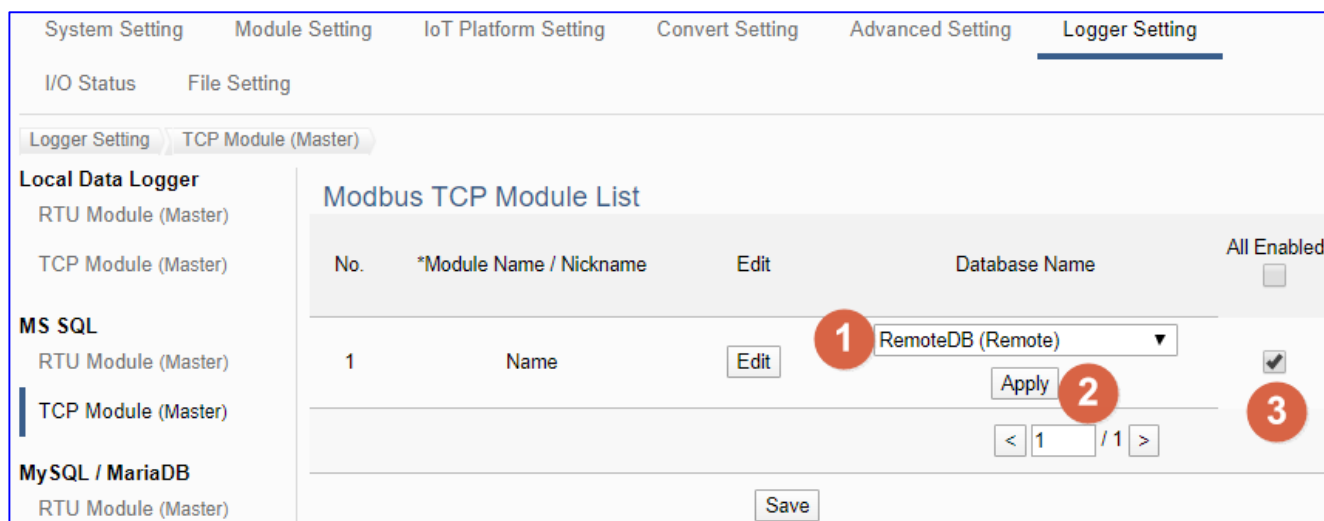
#### Function Diagram:



#### Application:



Enter the main menu [Logger Setting] > [MS SQL] > [TCP Module (Master)], e.g. the remote database Name as below.



Logger Setting > MS SQL - TCP Module (Master) > Modbus TCP 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 data logger, 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.
Database Name Apply	Select the database name set in the “MS SQL” of the “Advanced Setting”. Click “Apply” (Text color will change from black to gray).
All Enabled <input type="checkbox"/>	Check [All Enabled] box to enable all modules in list for data logger. Default: Uncheck.
<input type="checkbox"/> Enabled	Check the box of each module can enable just that module for data logger.
<input type="button" value=" &lt; 1 / 1 &gt;"/>	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.

First select [**Database name**] and click [**Apply**] button, and check the **enable box** of the module. If user wants to enable some I/O of the module, please click [Edit] button to 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="Name"/>

### Variable Table

Details

Variable Name	Attribute	Data Type	Database Name	Enabled
<input style="width: 90%;" type="text" value="Tag30"/>	<input style="width: 90%;" type="text" value="Read"/>	Short	<input style="width: 90%;" type="text" value="RemoteDB (Remote)"/>	<input checked="" type="checkbox"/>
<input style="width: 90%;" type="text" value="eagle"/>	<input style="width: 90%;" type="text" value="Read / Write"/>	Short	<input style="width: 90%;" type="text" value="RemoteDB (Remote)"/>	<input checked="" type="checkbox"/>

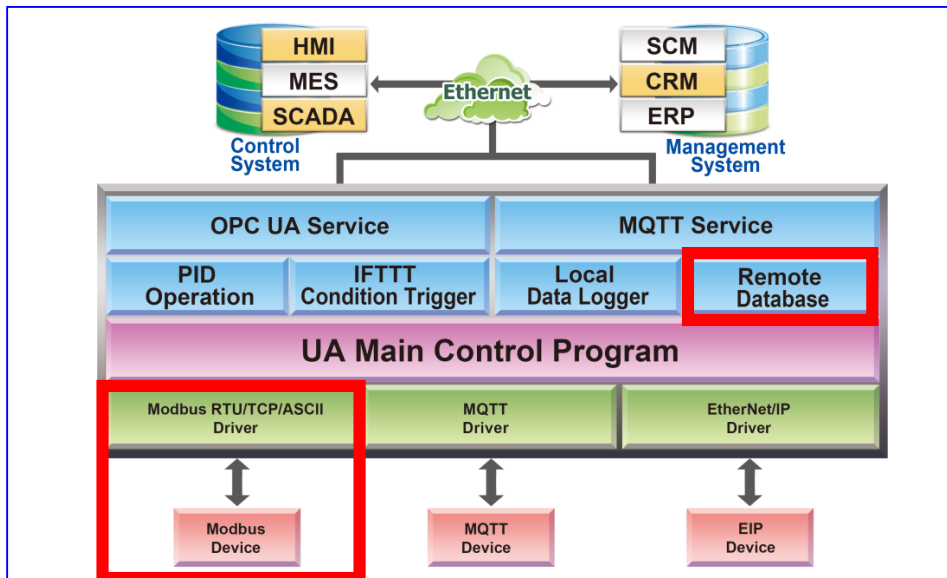
<b>Logger Setting &gt; MS SQL &gt; TCP Module (Master) – Module Content Setting</b>	
No.	The module number in the module list (Not editable here)
Module Name	The module name set in the module list (Not editable here)
<b>Logger Setting &gt; MS SQL &gt; TCP Module (Master) – Variable Table</b>	
Variable 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...
Database Name	Display the database name select in previous setting page. (Not editable here)
Enabled <input type="checkbox"/>	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 / Cancel	Click "OK" to save this page settings and back to the module list page. Click "Cancel" to leave this page without save.

### 5.6.3 MySQL / MariaDB: RTU / TCP Module (Master)

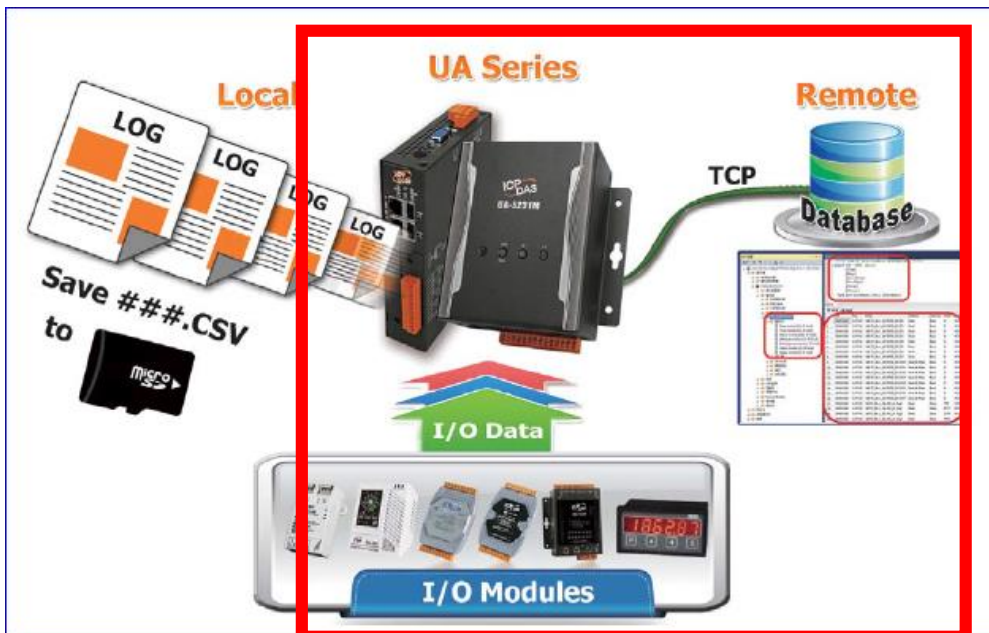
“MySQL / MariaDB” of “Logger Setting” provides users to record I/O log data of the connecting RTU/TCP module (Master) into the MySQL / MariaDB remote database.

This function is for setting the remote data logger and the RTU/TCP modules, using “**RTU module**” setting as an example. For the setting about the MySQL / MariaDB data logger, please refer to [Chapter 5.5.5](#).

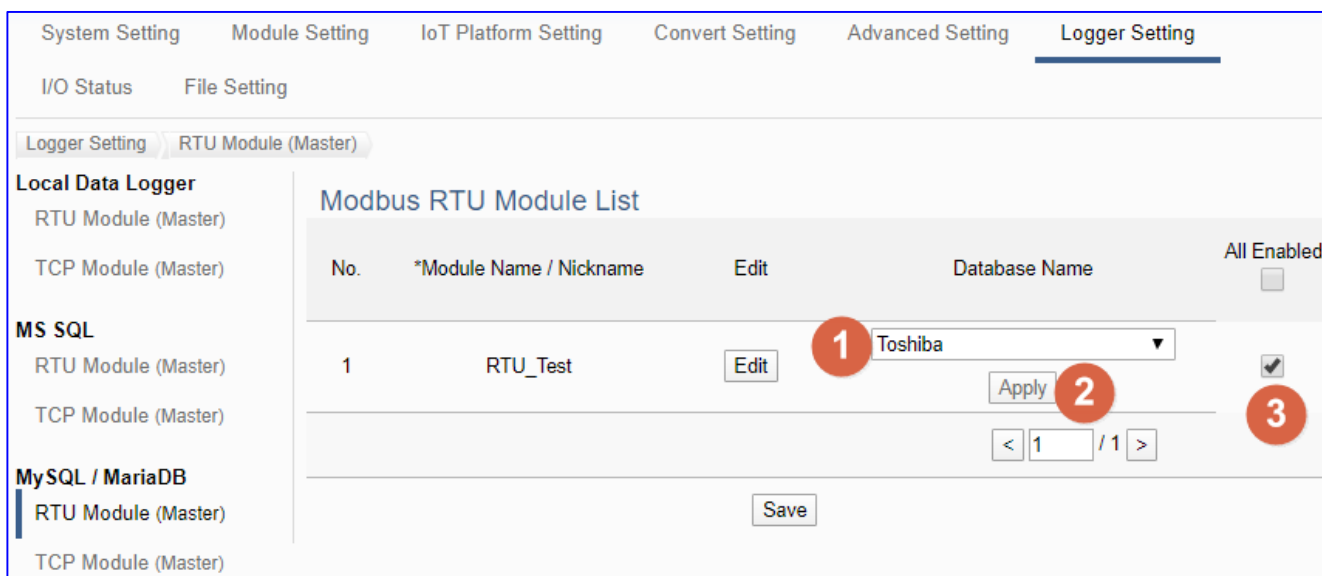
#### Function Diagram:



#### Application:



Enter the main menu [Logger Setting] > [MySQL / MariaDB] > [RTU Module (Master)], e.g. the remote database Name “Toshiba”, as below.



Logger Setting > MySQL/MariaDB - RTU Module (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 data logger, 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.
Database Name Apply	Select the database name set in the “MS SQL” of the “Advanced Setting”. Click “Apply” (Text color will change from black to gray).
All Enabled <input type="checkbox"/>  <input type="checkbox"/> Enabled	Check [All Enabled] box to enable all modules in list for data logger. Default: Uncheck. Check the box of each module can enable just that module for data logger.
<input type="button" value=" &lt; 1 / 1 &gt;"/>	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.

First select [**Database name**] and click [**Apply**] button, and check the **enable box** of the module. If user wants to enable some I/O of the module, please click [Edit] button to enter the “Module Content Setting” page.

**Module Content Setting**

No.

Module Name

---

**Variable Table**

Details

Variable Name	Attribute	Data Type	Database Name	Enabled
<input type="text" value="Tag14"/>	<input type="text" value="Read / Write"/>	Unsigned Short	<input type="text" value="Toshiba"/>	<input checked="" type="checkbox"/>
<input type="text" value="Tag15"/>	<input type="text" value="Read / Write"/>	Unsigned Short	<input type="text" value="Toshiba"/>	<input checked="" type="checkbox"/>

<b>Logger Setting &gt; MySQL/MariaDB - RTU Module (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>Logger Setting &gt; MySQL/MariaDB - RTU Module (Master) – Variable Table</b>	
Variable 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...
Database Name	Display the database name select in previous setting page. (Not editable here)
Enabled <input type="checkbox"/>	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 / Cancel	Click "OK" to save this page settings and back to the module list page. Click "Cancel" to leave this page without save.

## 5.7 Main Menu: I/O Status

**I/O Status** is the 6<sup>th</sup> item of the Main Menu, mainly to display the realtime I/O status of all the modules.

I/O Status page offers an easy way to view monitoring page that allows you to view important controller information in real time. The I/O Status page includes the following information.

- i. System default I/O Status page: It displays the all I/O channel information based on the sorting of all I/O Modules.
- ii. Related settings and the user-defined I/O Status page: It displays the I/O channel status based on the user-defined arrangement.

The user can click the module name on the left site, and the right will show all the real time I/O status of the selected module.

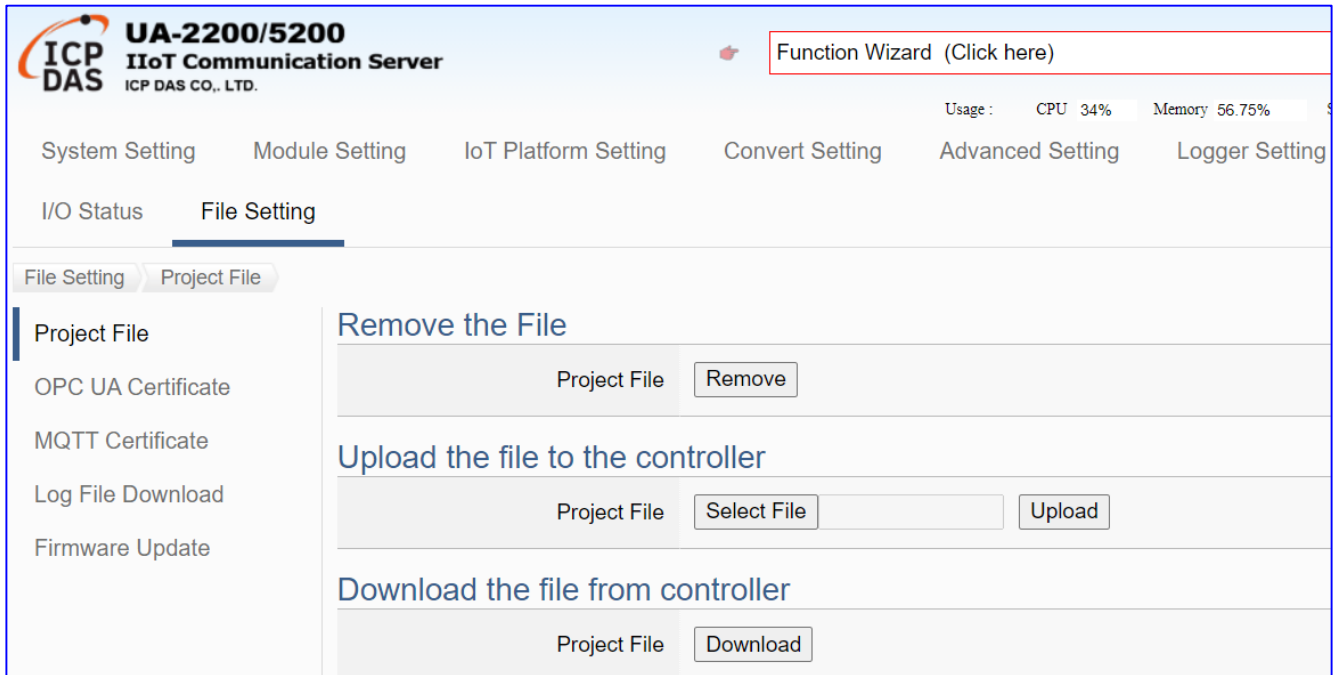
The screenshot displays the I/O Status page for the UA-2200/5200 IICoT Communication Server. The interface includes a top navigation bar with 'Function Wizard (Click here)' and several menu items: System Setting, Module Setting, IoT Platform Setting, Convert Setting, Advanced Setting, and Logger Setting. Below this, there are sub-menus for 'I/O Status' and 'File Setting'. The left sidebar lists module types: Modbus RTU Module (Master), Modbus TCP Module (Master), Modbus ASCII Module (Master), MQTT Module, and EtherNet/IP Module. The 'Modbus TCP Module (Master)' is currently selected, showing a table with one entry: No. 1, Name DL-302, LAN. The main content area is divided into 'Related Settings' and 'I/O Status'. 'Related Settings' includes 'Number of variables' (10) and 'Display Update Time (ms)' (1000). The 'I/O Status' section has tabs for 'I/O', 'Scaling', and 'Bitwise', with 'Scaling' selected. It contains a table with the following data:

Variable Name	Data Type	Value	Description	Status
Scale_CO2	Float	920	CO2	Good
Scale_Relative_humi	Float	66.29	Relative_humidity	Good
Scale_Temperature_	Float	21.64	Temperature_Celsius	Good
Scale_Temperature_	Float	70.95	Temperature_Fahrenheit	Good

## 5.8 Main Menu: File Setting

**File Setting** is the last 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 4 sub-menu functions. This chapter will introduce the function items and setting parameters.



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

[CH2 Quick Start 1: Hardware/Network Connection](#)

[CH3 Quick Start 2: Web UI / Setting Steps](#)

[CH4 Function Wizard: Project Quick Setup & Example](#)

## 5.8.1 Project File

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

The screenshot shows a web interface with a top navigation bar containing 'I/O Status' and 'File Setting'. Below this, there are two sub-tabs: 'File Setting' and 'Project File'. A left sidebar lists several options: 'Project File', 'OPC UA Certificate', 'MQTT Certificate', 'Log File Download', and 'Firmware Update'. The main content area is divided into three sections:

- Remove the File:** A section with a 'Project File' label and a 'Remove' button.
- Upload the file to the controller:** A section with a 'Project File' label, a 'Select File' button, a text input field, and an 'Upload' button.
- Download the file from controller:** A section with a 'Project File' label and a 'Download' button.

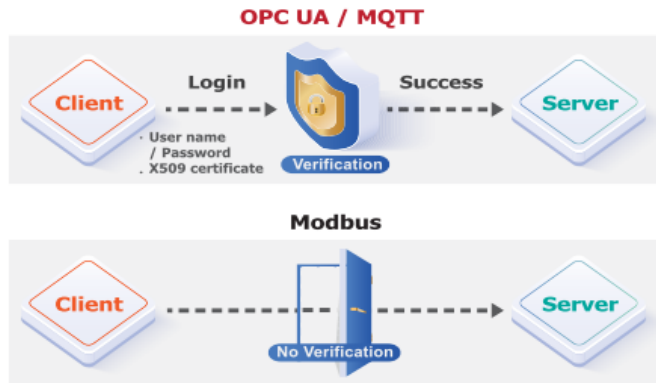
File Setting > Project File > Remove the File	
Project File	Click [Remove] to delete all project settings current in the UA series controller.
File Setting > Project File > Upload the file to the controller	
Project File	<p><b>Select File:</b> click to select the project that want to upload to the UA controller. (Extension name of the project file: “.tar”)</p> <p><b>Upload:</b> upload the project file into the UA controller. If select a wrong file (extension name is not “.tar”), the system will show an error message.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <div style="display: flex; align-items: center;"> <span style="margin-right: 10px;">Project File</span> <div style="border: 1px solid #ccc; padding: 2px;">Select File</div> <div style="border: 1px solid #ccc; padding: 2px; margin-left: 5px;">20191211.csv</div> <div style="margin-left: 10px; color: red; font-size: small;">File name is incorrect, default name: Project.tar.</div> <div style="margin-left: 20px; border: 1px solid #ccc; padding: 2px;">Upload</div> </div> </div> <p>Select a right format file with extension name of “.tar”, as below.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <div style="display: flex; align-items: center;"> <span style="margin-right: 10px;">Project File</span> <div style="border: 1px solid #ccc; padding: 2px;">Select File</div> <div style="border: 1px solid #ccc; padding: 2px; margin-left: 5px;">Project_192.168.255.</div> <div style="margin-left: 20px; border: 1px solid #ccc; padding: 2px;">Upload</div> </div> </div>
File Setting > Project File > Download the file to the local computer	
Project File	<b>Download:</b> Download the project with all Web UI settings to the current computer. (Extension name of the project file: “.tar”)

## 5.8.2 OPC UA Certificate

UA controller supports OPC UA Server security connection, including identity authentication, data encryption, data signature. Server and Client authenticate each other through x.509 certificate. Compared with the traditional method, it has the following functional advantages.

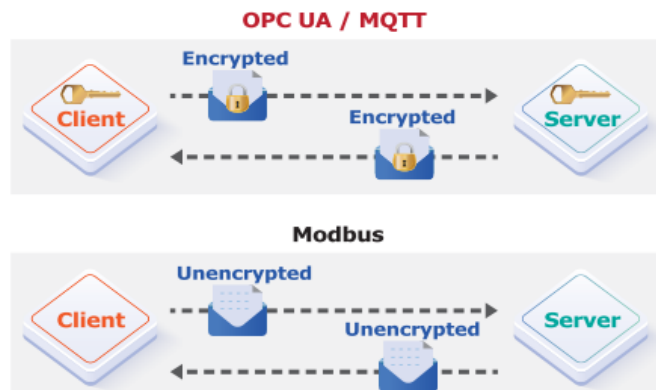
### Support Identity Authentication

Identity Authentication			
ICP DAS UA Solution	OPC UA	ID/Password, Anonymous, Certificate	Yes
	MQTT	ID/Password, Anonymous, Certificate	✓
Traditional	Modbus	None	



### Support Data Encryption

Data Encryption			
ICP DAS UA Solution	OPC UA	SSL/TLS Encryption	Yes
	MQTT	SSL/TLS Encryption	✓
Traditional	Modbus	None	



This function is the **certificate** file management function of **OPC UA Server**. There are 3 setting items about OPC UA Certificate: Remove, Upload the file to the controller, download the file from the controller. If your project need not the certificate, you can skip this step.

In the [OPC UA Certificate] step, users can add mutual credentials on both side's devices to strengthen security encryption.

- ① First, obtain the **OPC UA Client** trust certificate file of the device from the connected party, save it to the PC. In this step, select this file and upload it to the UA controller. (If there was an old certificate file in UA, remove it first.)
- ② The device of the other side needs the UA certificate also. In this step, download the **OPC UA Server** certificate file (**Certificate\_IPAddress\_.tar**) to the other party, so that they can decompress the file (**icpdasuaserver.der**) and upload to their device.





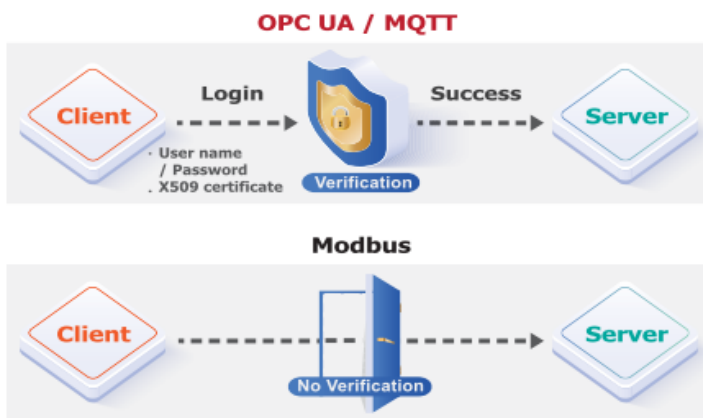
File Setting > OPC UA Certificate > Remove the File	
Trusted Certificate	<b>Remove:</b> Click to delete the OPC UA client Trusted Certificate current in the UA controller.
OPC UA Server Certificate	<b>Remove:</b> Click to delete the OPC UA Server Certificate current in the UA controller.
File Setting > OPC UA Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the OPC UA Trusted Certificate file in PC to upload to the UA controller.</p> <p><b>Upload:</b> upload the Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>DER</b>. Extension name must be "<b>der / cer / crt</b>".</li> </ul> <p></p> <ul style="list-style-type: none"> <li>If select a wrong file, the system will show an error message.</li> </ul> <p></p>
File Setting > OPC UA Certificate > Download the file from controller	
OPC UA Server Certificate	<p><b>Download:</b> Download the OPC UA Server Certificate file to the current using computer.</p> <ul style="list-style-type: none"> <li>File format: <b>DER</b>. File name: <b>Certificate_IP-address_.tar</b></li> </ul> <p>e.g. </p> <p>Before using, decompress to <b>icpdasuaserver.der</b>, as below.</p> <p></p>

### 5.8.3 MQTT Certificate

UA controller supports MQTT Client secure encrypted certificate file management. There are three types of files: Trusted Certificate, Certificate, and Private Key. Compared with the traditional method, it has the following functional advantages.

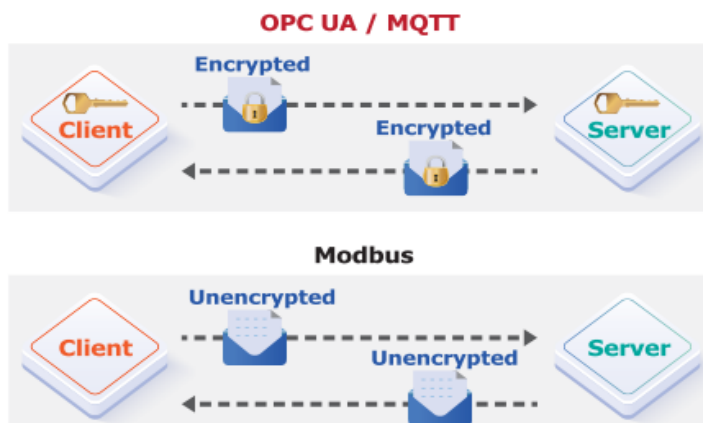
#### Support Identity Authentication

Identity Authentication			
ICP DAS UA Solution	OPC UA	ID/Password, Anonymous, Certificate	Yes ✓
	MQTT	ID/Password, Anonymous, Certificate	
Traditional	Modbus	None	



#### Support Data Encryption

Data Encryption			
ICP DAS UA Solution	OPC UA	SSL/TLS Encryption	Yes ✓
	MQTT	SSL/TLS Encryption	
Traditional	Modbus	None	



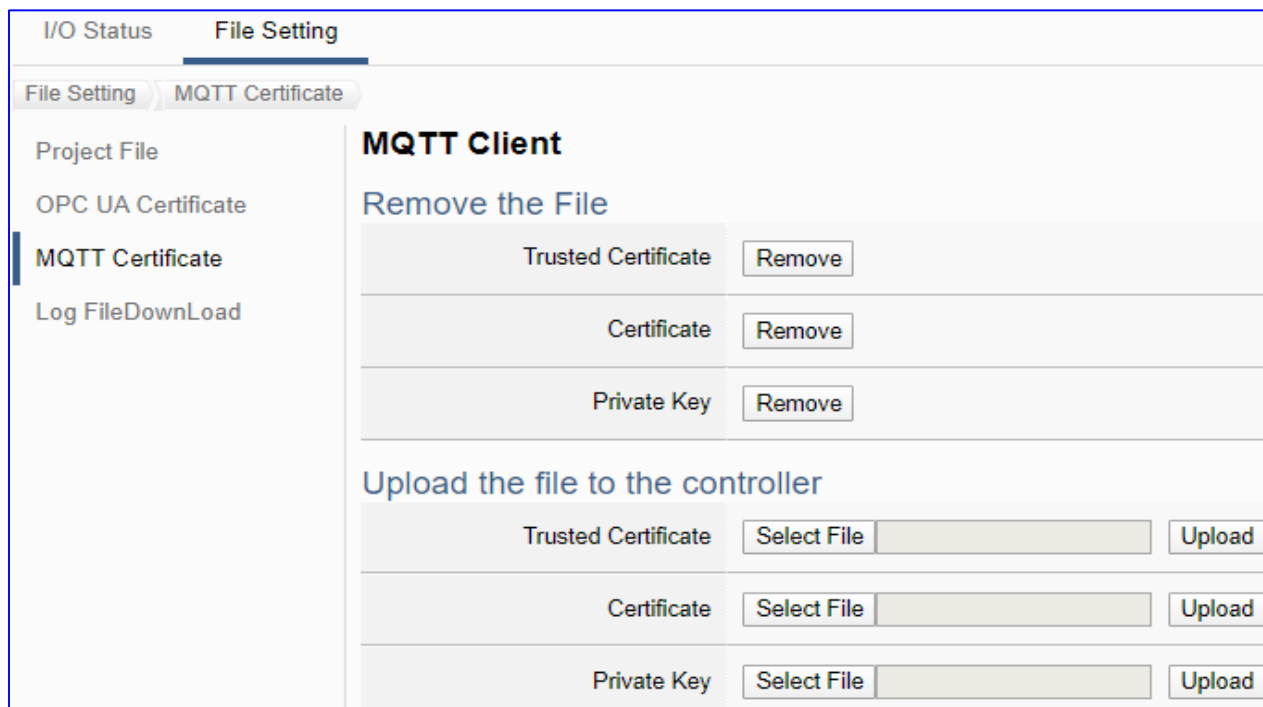
The [MQTT Certificate] is for setting up security communications to upload and remove the MQTT Certificate.

The users upload the file to the UA controller according to the type of obtained certificate. There are 3 types of MQTT Certificate:

- Trusted Certificate
- Certificate
- Private Key

If you want to perform **Broker authentication**, you need to upload the **Trusted Certificate**. If you want to perform the **Broker/Client two-way authentication**, you need to upload the **Credential and Private Key additionally**. The user can skip this step if the user project does not use certificate transmission security.

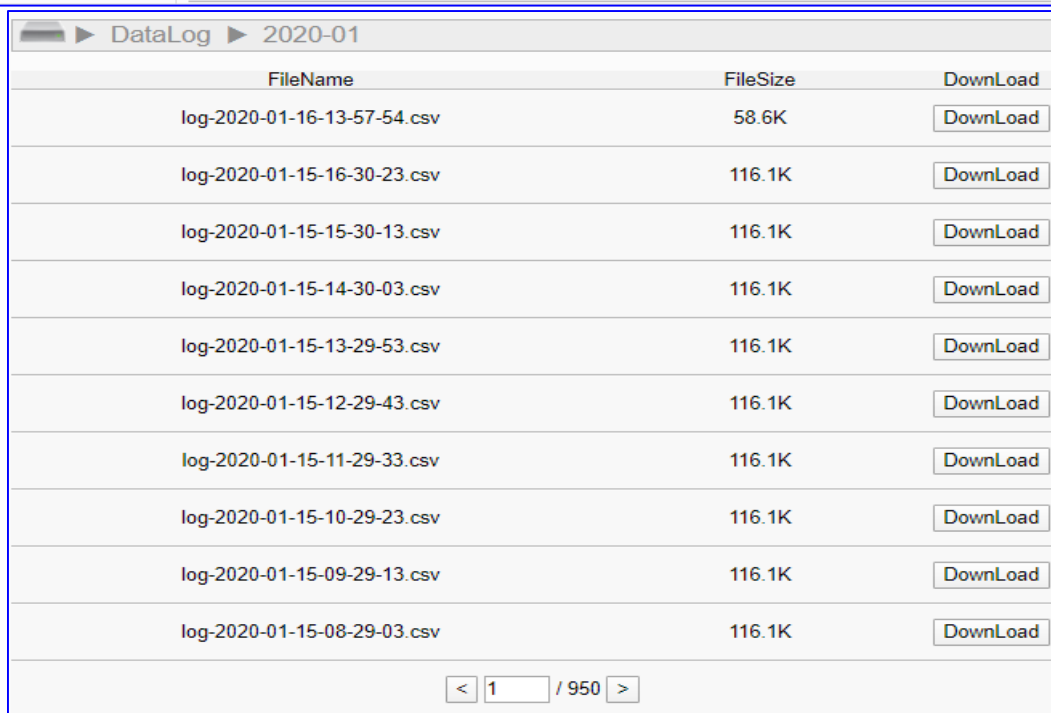
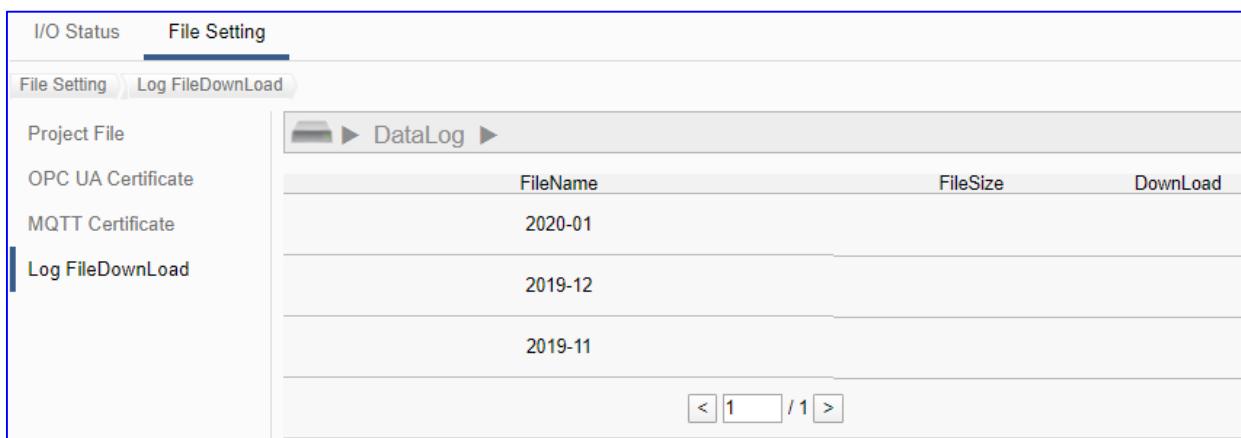
This page provides 2 setting items: Remove the file, and Upload the file to the controller for the MQTT Trusted Certificate, Certificate and Private Key.




File Setting > MQTT Certificate > Remove the File	
Trusted Certificate	<b>Remove:</b> delete the MQTT Trusted Certificate current in the UA series controller.
Certificate	<b>Remove:</b> delete the MQTT Certificate current in the UA series controller.
Private Key	<b>Remove:</b> delete the MQTT Private Key current in the UA series controller.
File Setting > MQTT Certificate > Upload the file to the controller	
Trusted Certificate	<p><b>Select File:</b> select the MQTT Trusted Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Trusted Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be "<b>pem / cer / crt</b>".</li> <li>If select a wrong file, the system will show an error message.</li> </ul> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <span>Trusted Certificate</span> <span>Select File</span> <input type="text" value="Certificate_192.168.255.10"/> <span style="color: red;">Certificate type is wrong.</span> <span>Upload</span> </div>
Certificate	<p><b>Select File:</b> select the MQTT Certificate file of the device.</p> <p><b>Upload:</b> upload the MQTT Certificate file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be "<b>pem / cer / crt</b>".</li> <li>If select a wrong file, the system will show an error message.</li> </ul>
Private Key	<p><b>Select File:</b> select the MQTT Private Key of the device.</p> <p><b>Upload:</b> upload the MQTT Private Key file to the UA controller.</p> <ul style="list-style-type: none"> <li>File format must be <b>PEM</b>. Extension name must be "<b>.key</b>".</li> <li>If select a wrong file, the system will show an error message.</li> </ul>

### 5.8.4 Log File Download

This function page is to view and download the Local Data Logger files of the microSD card in the UA series controller.



File Setting > Log File Download > DataLog	
File Name	At first, display the year-month folder (e.g. 2020-01), click the year-month folder to list all log files in this month, with the file name "log-YYYY-MM-DD-HH-mm-ss.csv".
File Size	The size of the log file. Unit: KB
Download	Click to download the Log file to the specific folder of the PC.
	The page number of the Log file list: Current page / Total pages. Click < or > to go to the previous or next page.

### 5.8.5 Firmware Update

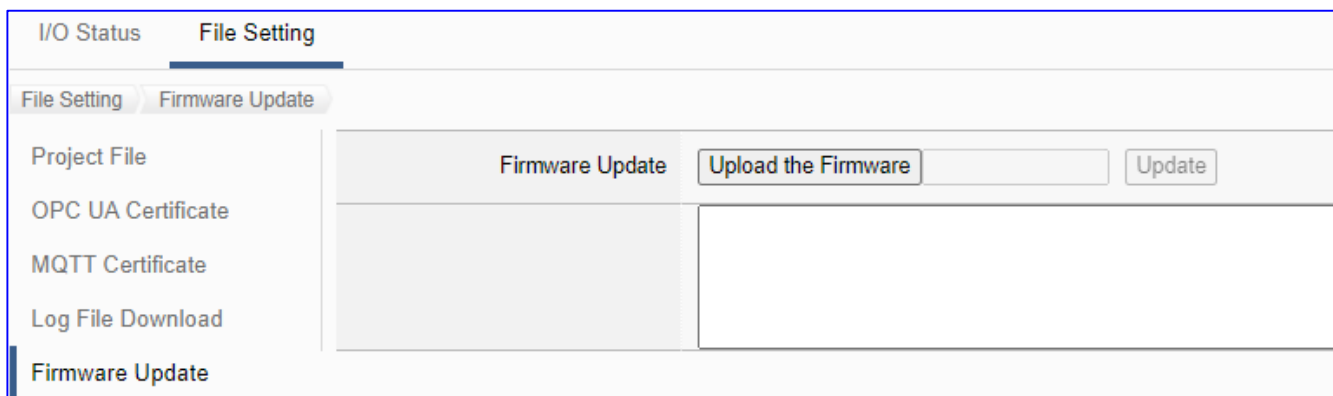
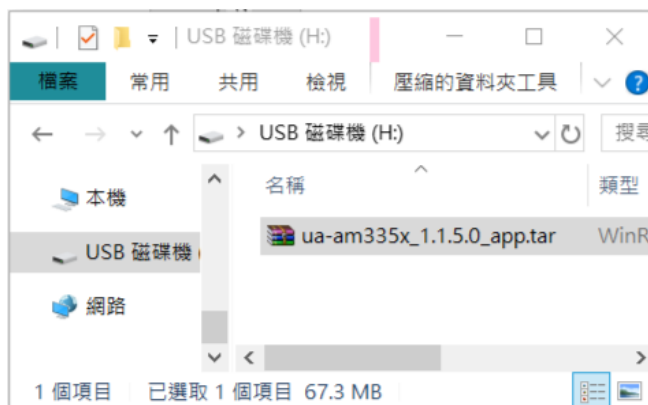
This function is about the update operation of the firmware file, mainly uploading the firmware file to update the version of the UA controller.

Before updating, please download the latest Firmware software file from ICP DAS UA series download center, save it to your computer, and then can upload the file to the UA controller.

**UA series Download Center):**

<https://www.icpdas.com/en/download/index.php?nation=US&kind1=&model=&kw=ua->

**Note: unzip the zip file to “.tar”, DO NOT decompress the “.tar” file again.**



File Setting > Firmware Update	
Upload the Firmware	Go to the UA series download center of the ICP DAS website to download the latest Firmware software file, save it to your computer. <b>Note: unzip the zip file to “.tar”, DO NOT decompress the “.tar” file again.</b> Click “Upload the Firmware” button and select the firmware file.
Update / Information Box	Select the Firmware file and click the “Update” button, it will update the Firmware version automatically. In the update operation, the information box below will display the updating status, and a final message will show if the updating succeeded or failed.

## 6. Factory Setting Recovering and Firmware

### Updating

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

#### 6.1 Recovering to Factory Setting (Rotary Switch: 8)

Turn the Rotary Switch of UA series to “8” can recover to the factory setting. Before that, first to connect the UA controller via a network cable to a PC or a Switch.

The steps:

1. After connecting the network, power off the UA hardware, and turn the Rotary Switch to “8”.
2. Reboot the UA and wait a long buzzer sound that means of doing the recovering.
3. Wait about 3 minutes until **2 long** buzzer sounds, and then turn the Rotary Switch to “0”.



**Note:**

If the buzzer makes **4 short** beeps, it means the network is not connected properly. Please check the network cable again.

4. Reboot the UA again, and the system will recover to the factory settings.

Factory Default Settings of UA Series			
Network	IP	LAN1: 192.168.255.1 LAN2: 10.0.0.1	Assign UA a new IP address according to your case. For UA-2200 series, set the LAN1 for the connection.
	Netmask	255.255.0.0	
	Gateway	LAN1: 192.168.1.1 LAN2: 10.168.1.1	
Web UI Account	Username	root	<b>After login, change your password as soon as possible.</b> (Section 5.1.4 for Web UI)
	Password	root	

## 6.2 Updating Firmware A – via Web UI of UA

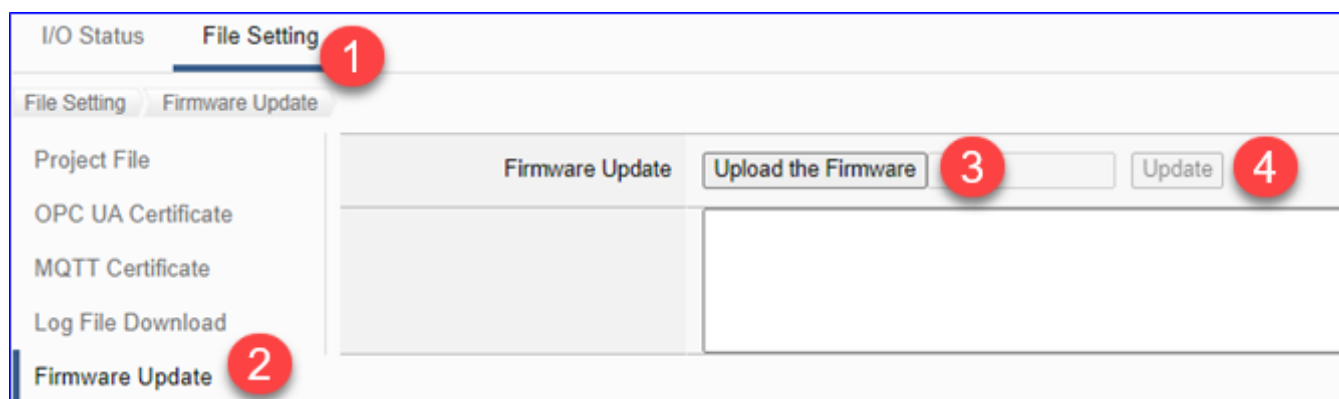
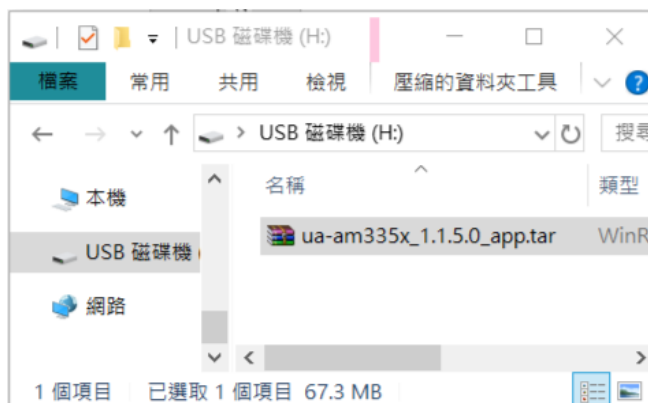
The user can update firmware via the Web UI function of the UA controller.

Before updating, please download the latest Firmware software file from ICP DAS UA series download center, save it to your computer, and then can upload the file to the UA controller.

### UA series Download Center:

<https://www.icpdas.com/en/download/index.php?nation=US&kind1=&model=&kw=ua->

**Note: unzip the zip file to “.tar”, DO NOT decompress the “.tar” file again.**



File Setting > Firmware Update	
Upload the Firmware	Go to the UA series download center of the ICP DAS website to download the latest Firmware software file, save it to your computer. <b>Note: unzip the zip file to “.tar”, DO NOT decompress the “.tar” file again.</b> Click “Upload the Firmware” button and select the firmware file.
Update / Information Box	Select the Firmware file and click the “Update” button, it will update the Firmware version automatically. In the update operation, the information box below will display the updating status, and a final message will show if the updating succeeded or failed.

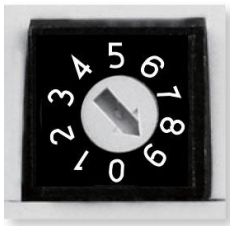
## 6.3 Updating Firmware B - via USB (Rotary Switch: 9)

Turn the Rotary Switch of UA series to “9” can update the Firmware 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 hardware, and turn the Rotary Switch to “9”.



2. Download the Firmware package file of the UA hardware corresponding model.  
UA Download Center:  
<https://www.icpdas.com/en/download/index.php?nation=US&kind1=&model=&kw=ua->
3. Save the Firmware package file into an empty FAT32 format USB drive and put to the UA USB port.
4. Reboot the UA and wait a long buzzer sound that means of doing the version updating.
5. Wait about **three** minutes until **two** long buzzer sounds, and then turn the Rotary Switch to “0”.  
**Note:**  
If the buzzer makes **4 short** beeps, it means the USB may not connected properly. Please check the USB again.
6. Reboot the UA again, and the system will update to the version of the package file.

**Note:**

**If the updating Firmware via USB still fails, please refer to nest section for using the MicroSD card to manually update the Firmware version.**



## 6.4 Updating Firmware C - via MicroSD Card

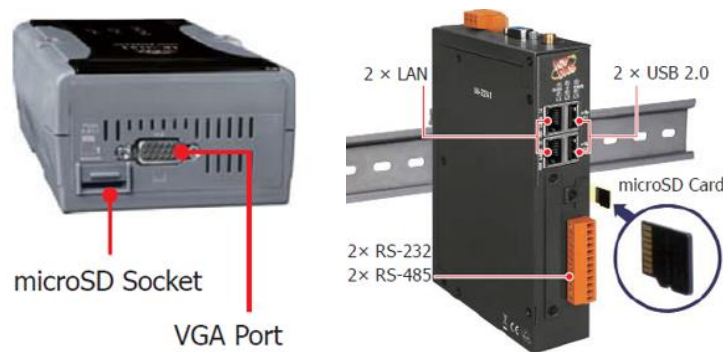
If the updating Firmware (UA version file) via USB still fails, please refer to the following steps for using the microSD card to manually update the Firmware version.

### ● Preparations:

- ✓ PC \* 1
- ✓ SSH / Telnet Software, Ex: PuTTY
- ✓ UA Series \* 1, Ex: UA-5231 (**Must wire with a networked device, ex: PC or Switch**)
- ✓ MicroSD Card Reader \* 1
- ✓ CA-0910 Cable \* 1 (In the shopping box of the UA Series)
- ✓ Power Supply \* 1 (10 ~ 30 VDC)

### ● The Steps to Update Firmware via MicroSD Card :

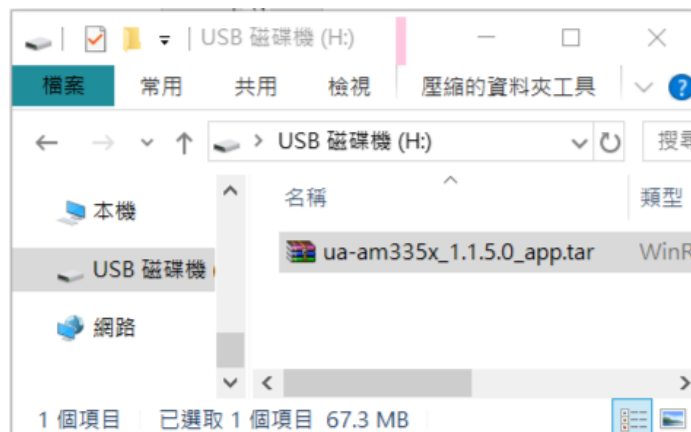
1. Take the microSD card from the socket of the UA, and connect the card with PC via the card reader.



2. Download the Firmware file and save to the microSD card. (**Save one Firmware file only**)  
UA Download Center:

<https://www.icpdas.com/en/download/index.php?nation=US&kind1=&model=&kw=ua->

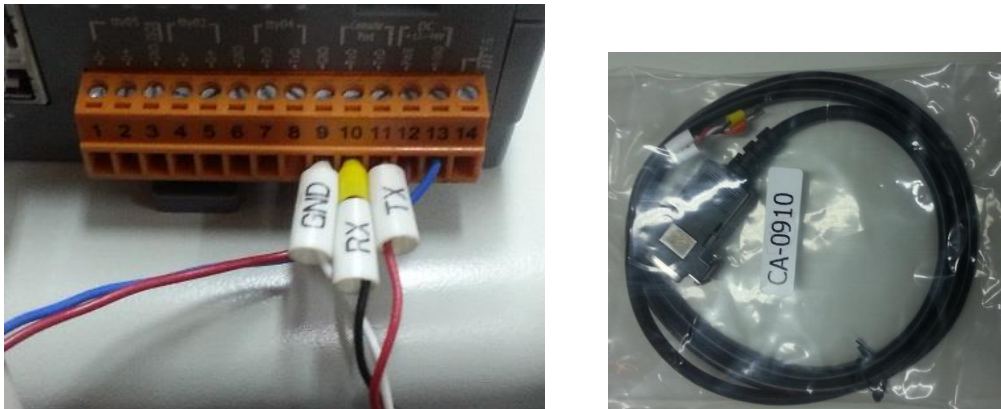
**Note: unzip the zip file to .tar, ex: ua-am335x\_x.x.x.x\_app.tar DO NOT decompress again.**



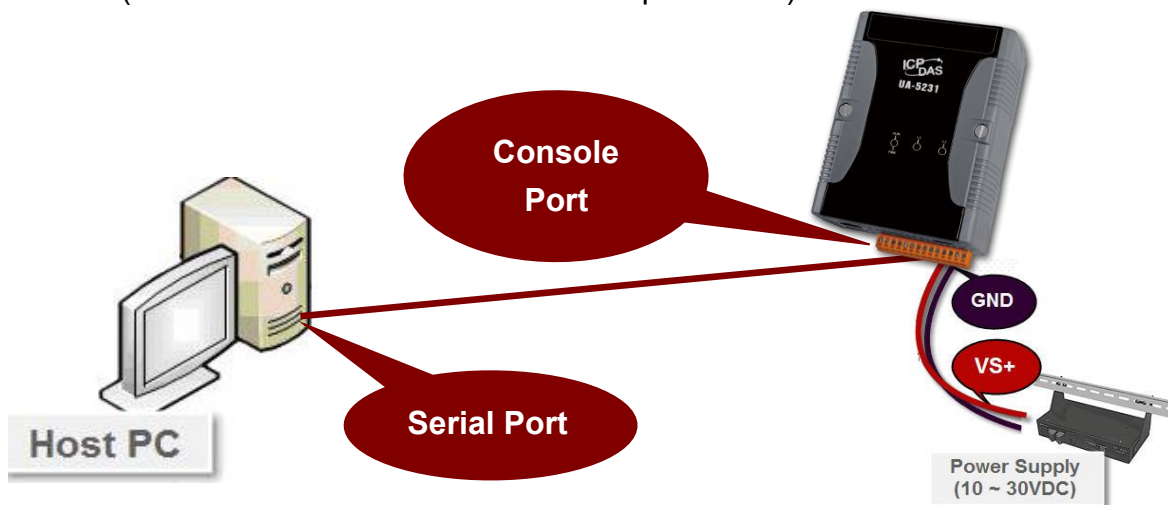
3. Insert the microSD card into the UA again.



4. Wire the female head of CA-0910 cable to the network PC, and the other head to the "Console Port" of UA.



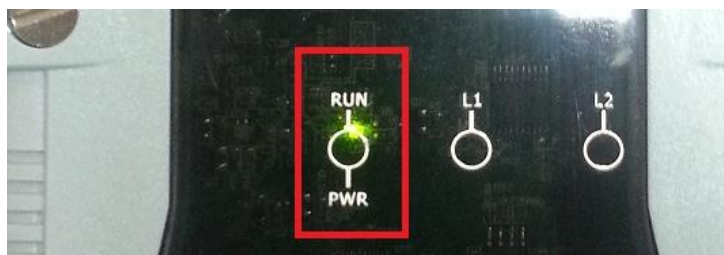
(Wire CA-0910 cable to the Console port of UA)



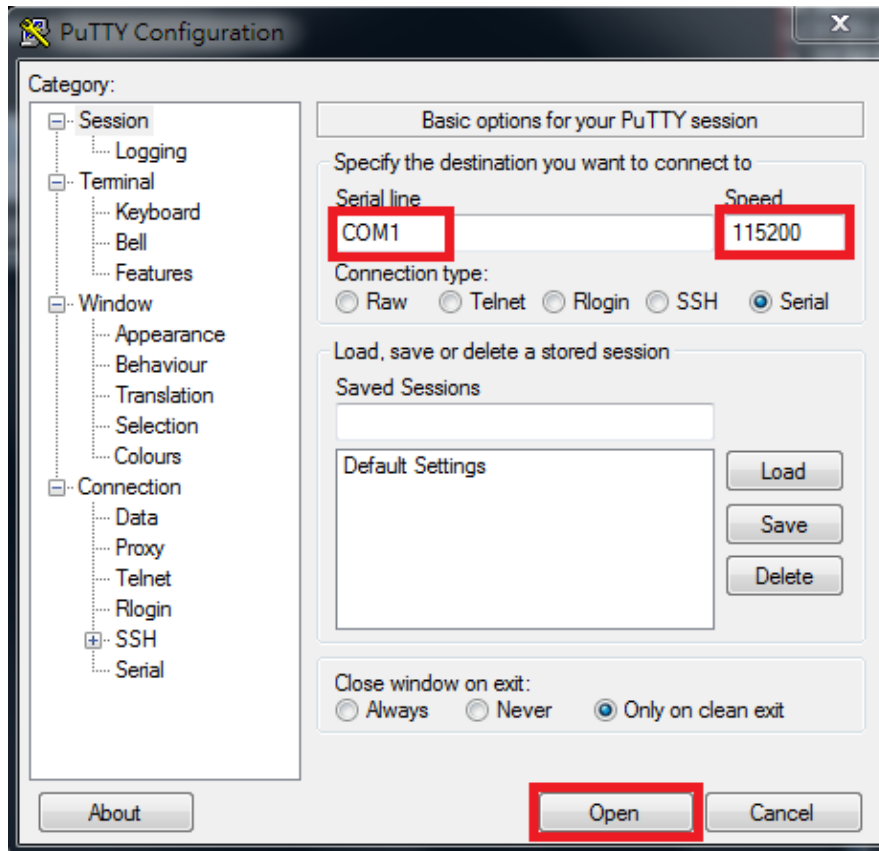
5. Turn on the power of the UA. When the lights are left with **ONE** LED, the boot is completed.

**Note:**

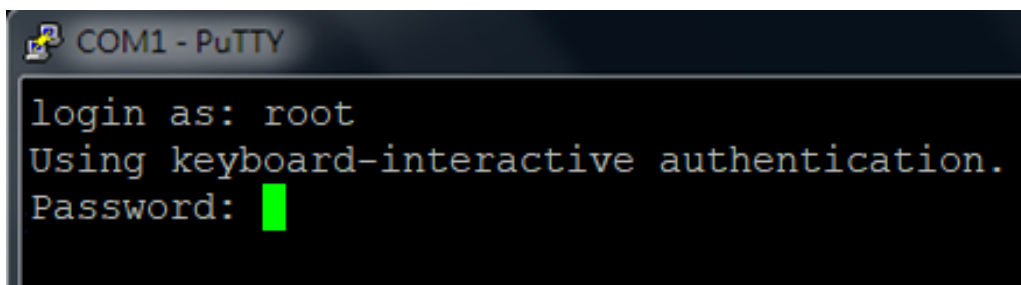
If the buzzer makes **4 short** beeps, it means the network is not connected properly. Please check the network cable again.



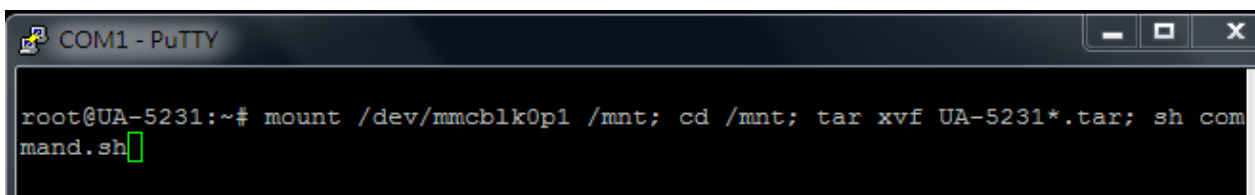
- Use an SSH/Telnet software, e.g. PuTTY, to connect to UA via the Serial connection. Input your Serial line (default: COM1) and Speed (115200 for UA). And then click “Open” button.



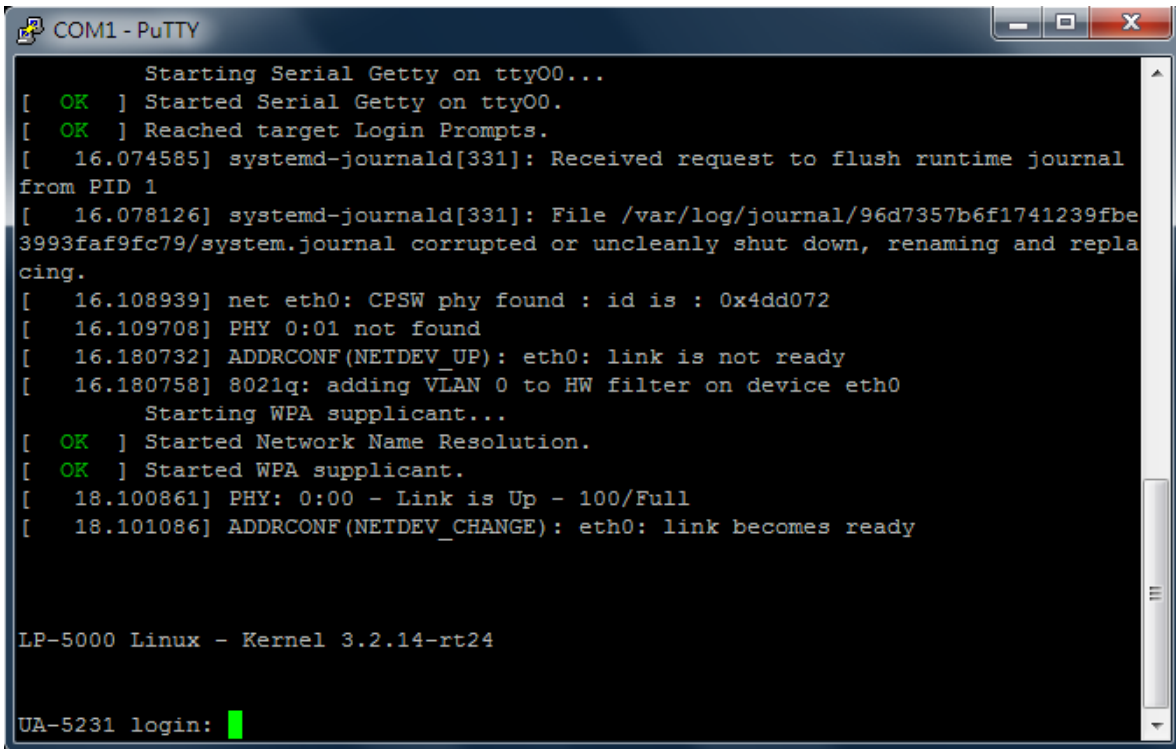
- After the login message, enter the default **username (root)** and **password (icpdas)**.



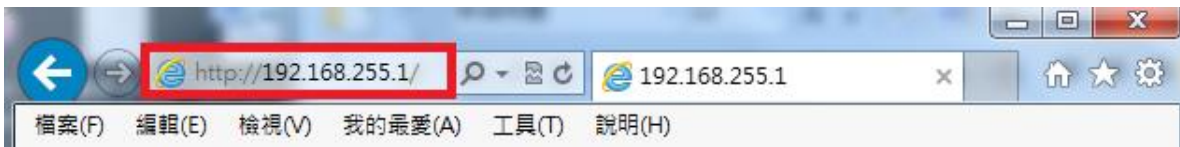
- Configure the UA environment:  
Copy the following red strings “**mount /dev/mmcbk0p1 /mnt; cd /mnt; tar xvf ua-am335x\*.tar; sh command.sh**” to the PuTTY screen and press the Enter key.



- Please wait a while for the UA controller configuration until the login screen appears again.




- Open a web browser on the PC (ex: Google Chrome, IE...) and enter "192.168.255.1" in the address bar.



- The web browser will run and go to the UA login web site as below. Please enter the default username/password and click the "Login" button.

**Default Username: root**

**Default Password: root**

 <p><b>UA-2200/5200</b> IIoT Communication Server ICP DAS CO., LTD.</p>	<p>Username : <input type="text" value="root"/></p> <p>Password : <input type="password" value="...."/></p> <p>Language : <input type="text" value="English"/></p> <p><input type="button" value="Login"/></p>
--	--

12. When login to the web interface, the UA home page will be displayed as below. If the Firmware Version number is the same as your download version, the updating is successful.

The screenshot displays the web interface for the UA-2200/5200 IIoT Communication Server. The top navigation bar includes 'System Setting', 'Module Setting', 'IoT Platform Setting', 'Convert Setting', 'Advanced Setting', and 'Logger Setting'. A 'Function Wizard (Click here)' button is highlighted with a red box. The left sidebar lists various settings: 'I/O Status', 'File Setting', 'System Setting', 'Controller Service Setting', 'Time Setting', 'Network Setting', 'Account Setting', 'Boot', and 'COM Port Interface Setting'. The main content area shows 'Version Information' with a red box around the 'Firmware Version' row, which displays 'Version 1.3.0.0'. Other rows include 'Main Program' (Version 1.1.41), 'Web Interface' (Version: 6.3.0, Date: 2019/08/06), and 'Install Information' (2020/01/15-17:43:52\_Factory\_InstallSuccess). Below this is a 'System Setting' section with descriptive text for each setting category. The footer contains the copyright notice: '© ICP DAS Co., Ltd. All Rights Reserved'.

Version Information	
Firmware Version	Version 1.3.0.0
Main Program	Version 1.1.41
Web Interface	Version : 6.3.0 Date : 2019/08/06
Install Information	2020/01/15-17:43:52_Factory_InstallSuccess

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

## 7 Security Certificate: Download / Upload

**UA communication security** includes the **username/password protection**, **SSL/TLS secure communication** (Secure Socket Layer / Transport Layer Security), and **OPC UA Server / MQTT Client certificate mechanism** to protect data transmission security.

**OPC UA certificate** provides the certificate exchange with **Server and Client** side to authenticate each other **through the x.509 certificate** to protect the data transmission security. OPC UA **default enables** encryption and security mechanisms with advanced processing, **including: Authentication, Authorization, Confidentiality and Integrity.**

**MQTT certificate** provides three certificate types: **Trusted Certificate, Certificate, and Private Key.** Depending on the type of certificate obtained to perform the **Broker verification** or **Broker/Client mutual verification.** It supports identity verification and data encryption, and provides a secure connection. mechanism.

- **OPC UA Server Certificate management**

**UA controller supports OPC UA Server security connection, including identity authentication, data encryption, data signature. Server and Client authenticate each other through x.509 certificate.**

There are 3 items in OPC UA Certificate: Remove, Upload and Download the Certificate file.

Download is for providing the OPC UA Server certificate to the other side.

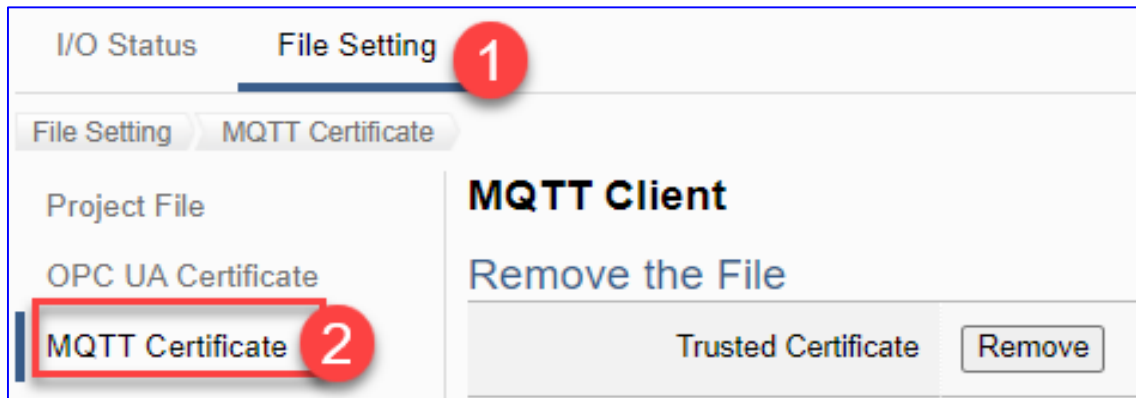
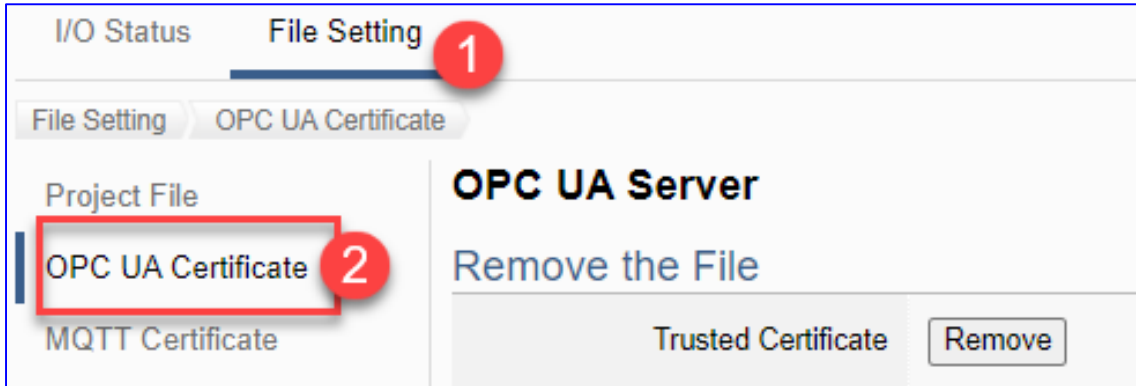
Upload is to save the other's trust certificate into the UA. So before uploading, please get the certificate file of the other side device first. Please refer to 7.2.1 for details.

- **MQTT Client Certificate management**

**UA controller supports MQTT Client secure encrypted certificate file management. There are three types of files: Trusted Certificate, Certificate, and Private Key.** The users upload the file to the UA controller according to the type of obtained certificate. If you want to perform **Broker authentication**, you need to upload the **Trusted Certificate.** If you want to perform the **Broker/Client two-way authentication**, you need to upload the **Credential and Private Key additionally.**

Therefore, before uploading, you must confirm the required verification method, obtain the required certificate file, and store it in your computer before you can configure the upload certificate. Please refer to 7.2.2 for details.

The upload and download of OPC UA and MQTT Certificates are in the main menu [**File Setting**]. Before setting this function, you need to download or upload the relevant certificates. This chapter is divided into two sections, download/upload, to explain the setting steps and notices.



7.1 Download the Certificate from UA Controller

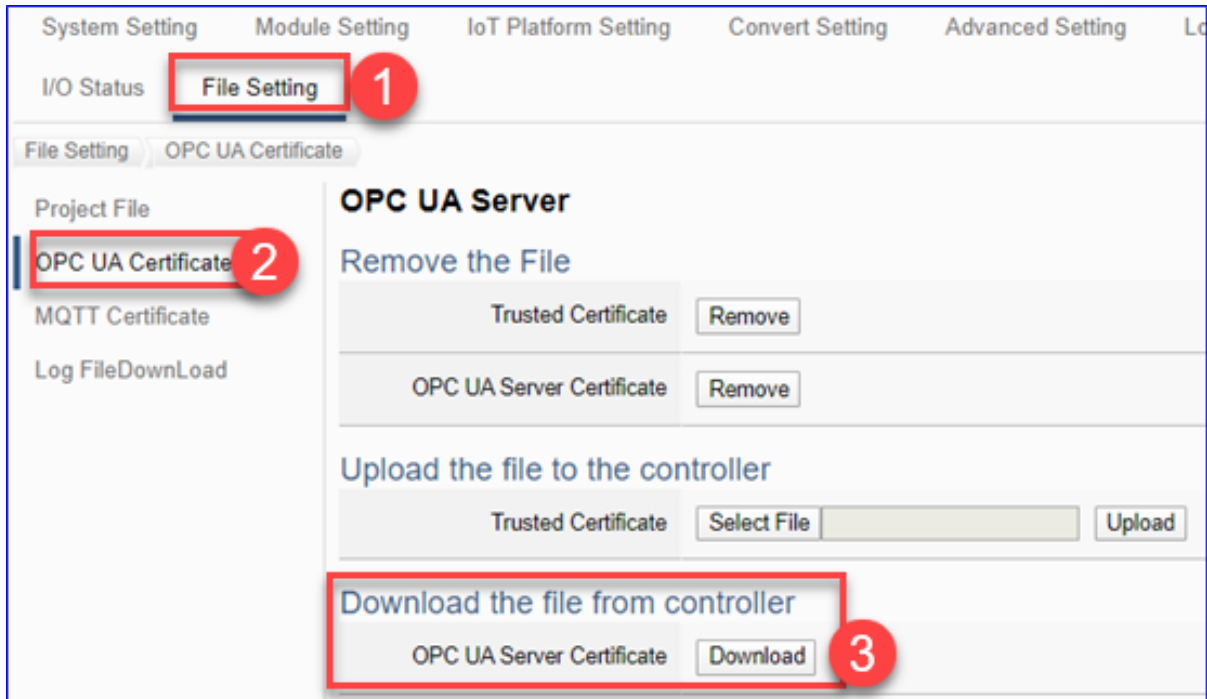
7.2 Upload the Certificate to UA Controller

7.2.1 OPC UA Certificate

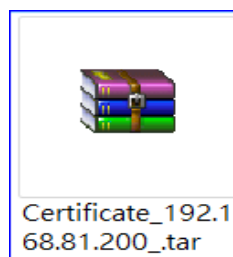
7.2.2 MQTT Certificate

## 7.1 Download the Certificate from UA Controller

Download is for providing the OPC UA Server certificate to the other side.



1. Click the main menu [ ① **File Setting** > ② **OPC UA Certificate** > **Download the file from the controller** – OPC UA Server Certificate] and then click on the button ③ [**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.

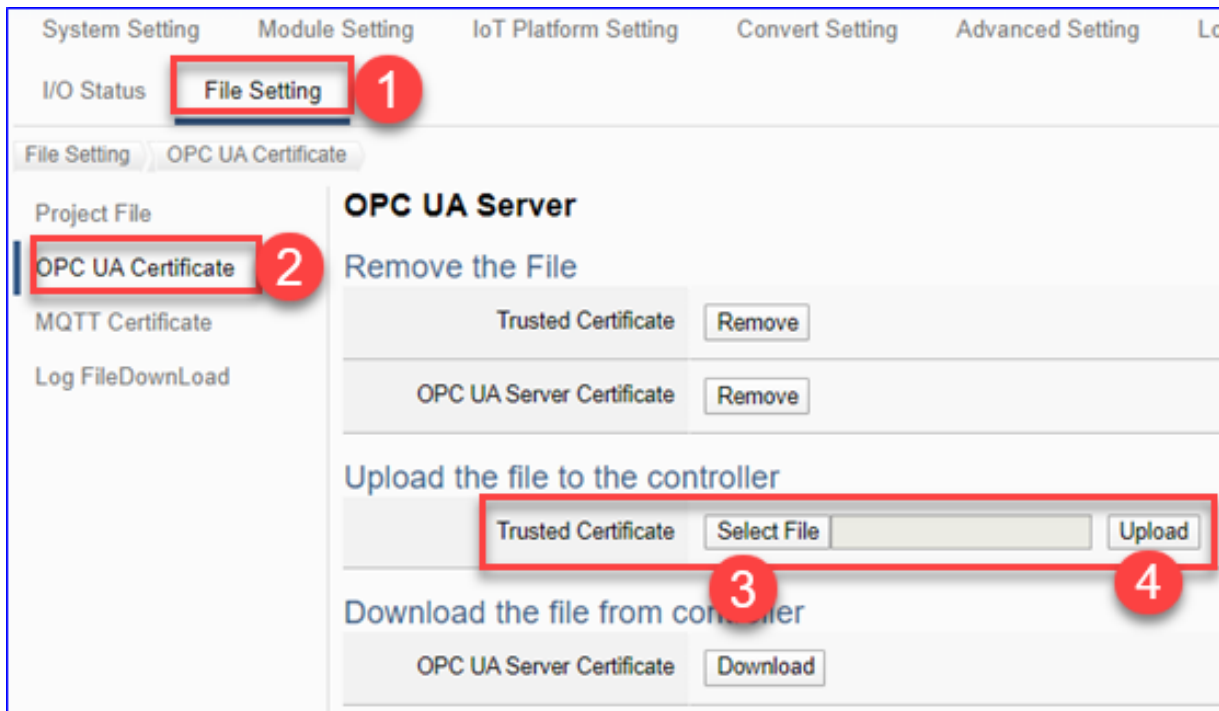




## 7.2 Upload the Certificate to UA Controller

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


### 7.2.1 OPC UA Certificate



1. Get the trusted certificates from OPC UA Client and save in the PC.
2. Click the main menu [①File Setting > ② OPC UA Certificate > Upload the file to the controller – Trusted Certificate], click on the button ③ [Select File] to select the trusted certificates from OPC UA Client.
3. Click the button ④ [Upload], then can upload and exchange the certificate authentication.

#### Notes for OPC UA Certificate:

- The supported code format: “**DER**”.
- The supported file extension name: “**\*.der / \*.cer / \*.crt**”.
- The OPC UA Server Certificate downloaded from UA series:

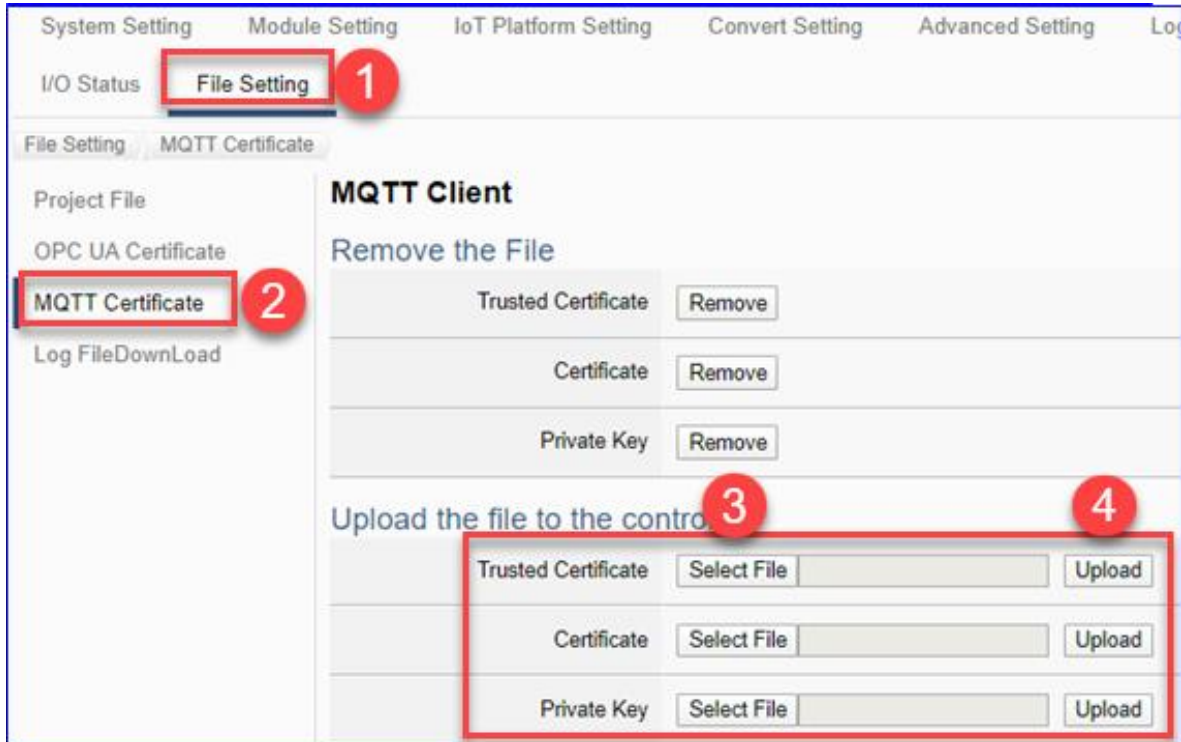
File name: **Certificate\_IP-address\_.tar**, e.g.  Certificate\_192.168.255.102\_.tar

Before using, decompress to **icpdasuserver.der**, e.g.  icpdasuserver.der

- Refer to [5.8.2](#) for detail parameter descriptions.

## 7.2.2 MQTT Certificate

If you want to perform **Broker authentication**, you need to upload the **Trusted Certificate**. If you want to perform the **Broker/Client two-way authentication**, you need to upload the **Credential and Private Key** additionally.



1. Get the trusted certificates from MQTT Client and save in the PC.
2. Click the main menu [ ① **File Setting** > ② **MQTT Certificate** > **Upload the file to the controller – Trusted Certificate / Certificate / Private Key**], click on the button ③ [**Select File**] to select the Trusted Certificate, Certificate or Private Key from MQTT Client.
3. Click the button ④ [**Upload**], then UA system can exchange the certificate authentication.

### Notes for MQTT Client Certificate:

- The supported code format: **“PEM”**.
- The supported file extension name for Certificates: **“\*.pem / \*.cer / \*.cert”**.
- The supported file extension name for Private Key: **“\*.key”**.
- Refer to [5.8.3](#) for detail parameter descriptions.

## 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. Protocol Technical Reference

- **OPC UA**

<https://opcfoundation.org/>

- **MQTT**

<http://mqtt.org/>

- **Modbus**

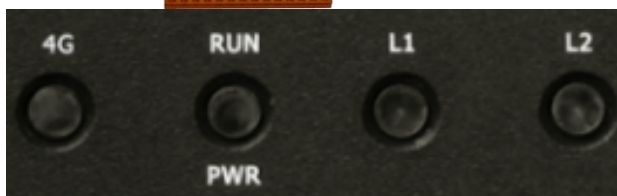
<http://modbus.org/>

## Appendix C. LED Indicators

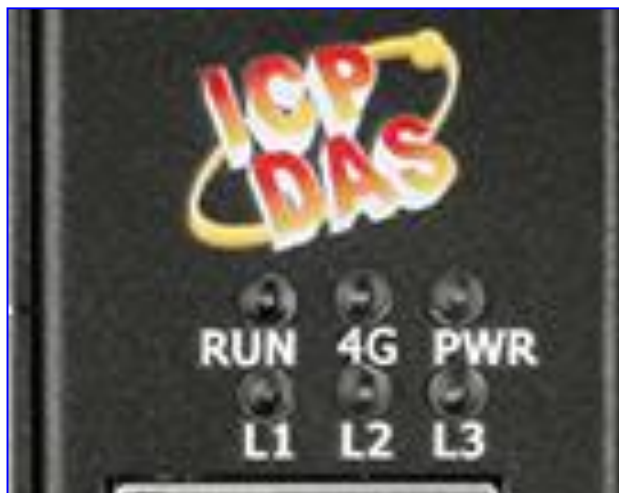
LED indicators of UA Series provide a very convenient way of status indications for faster, easier diagnostics.



UA-5231 / UA-5231M		
LED	LED Status	Module Status
PWR	Green: ON	The module is powered on.
RUN	Red: Blinking	The module is functioning normally. <b>When power on UA, please wait about one minute for completing the start-up procedure until the "RUN" LED starts blinking.</b>
L1	OFF	Function reserved
L2	Green: Blinking, then ON, and then OFF.	When install or update the Firmware, L2 will blinking. When complete the process, L2 will steady ON to notify user and then OFF.



UA-5231M-3GWA / UA-5231M-4GE / UA-5231M-4GC		
LED	LED Status	Module Status
PWR	Green: ON	The module is powered on.
RUN	Red: Blinking	The module is functioning normally. <b>When power on UA, please wait about one minute for completing the start-up procedure until the "RUN" LED starts blinking.</b>
3G	Green: ON	The modem is functioning normally, and SIM card inside.
	OFF	The modem is not functioning, or no SIM card inside.
4G	Green Light Flash: ON for 2 seconds, OFF for 1 sec, and flashing alternately.	The modem is functioning normally, and the SIM card inside.
	Green Light Flash: ON for 1 second, OFF for 2 secs, flashing alternately, or OFF .	The modem is not functioning, no SIM card inside, or the PIN is still locked.
	Green Light Flash: Quickly	Data is transmitting.
L1	OFF	Function reserved
L2	Green: Blinking, then ON, and then OFF.	When install or update the Firmware, L2 will blinking. When complete the process, L2 will steady ON to notify user and then OFF.



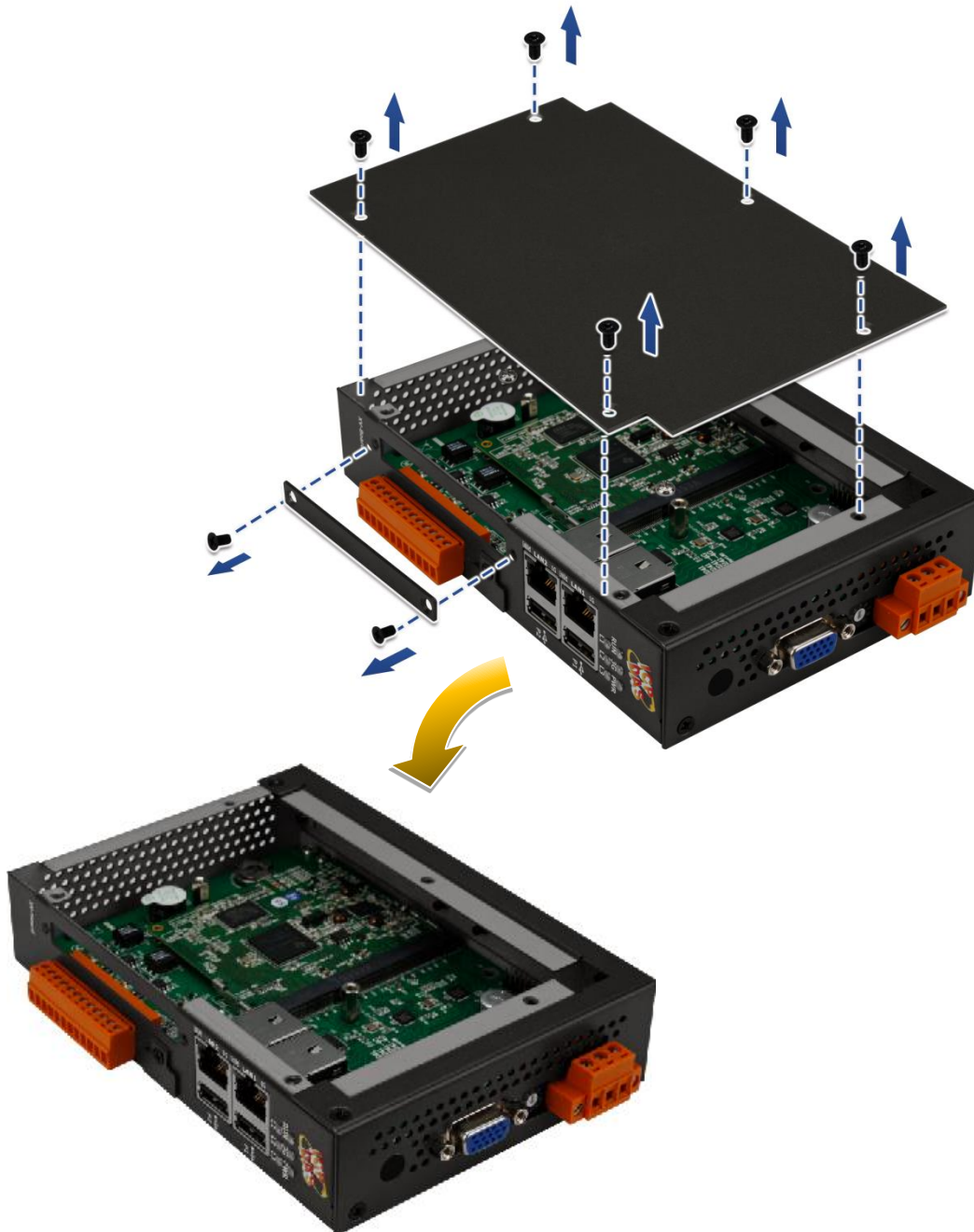
UA-2241M / UA-2241MX-4GE / UA-2241MX-4GC		
LED	LED Status	Module Status
PWR	Green: ON	The module is powered on.
RUN	Red: Blinking	The module is functioning normally. <b>When power on UA, please wait about one minute to complete the start-up procedure until the "RUN" LED starts blinking.</b>
L1 / L3	OFF	Function reserved
L2	Yellow Light: Blinking, then ON, and then OFF.	When install or update the Firmware, L2 will blinking. When complete the process, L2 will steady ON to notify user and then OFF.
4G (UA-2241MX-4GE UA-2241MX-4GC)	Orange Light: Flashing per 2 seconds, and then ON.	The modem is functioning normally, and the SIM card is functioning normally.
	Orange Light: Flashing per 2 seconds, and then OFF. Or OFF.	The modem is not functioning, no SIM card inside, or the PIN is still locked.
	Light Flash: Quickly	Data is transmitting.

## Appendix D. Mounting the XV-board for UA-2241M

UA-2241M has one I/O expansion bus to expand the functions by insert one optional XV-board. The supported XV-boards are listed after the mounting steps below.

### Mounting Steps:

**Step 1: Remove stripped screws and then remove the cover**



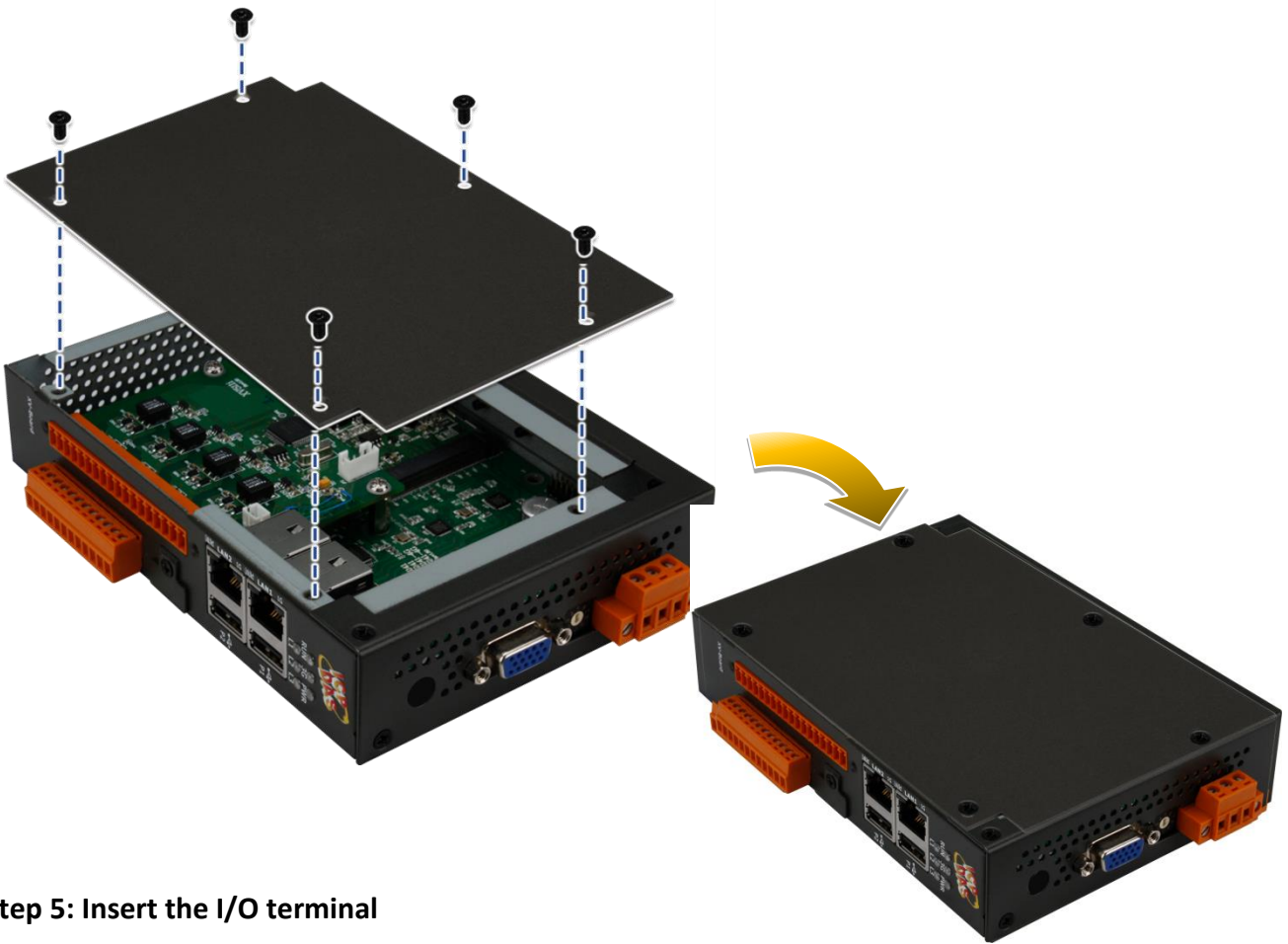


**Step 2: Hold the XV-board vertically and align the socket, and then carefully press the XV-board onto the I/O expansion bus**

**Step 3: Fasten the XV-board using the screws supplied**



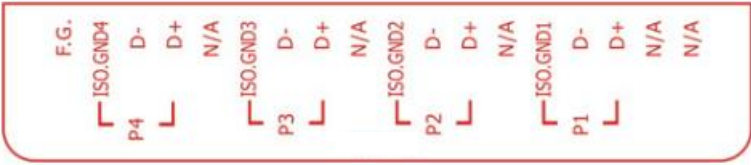
**Step 4: Close the cover and then fasten the screws**



**Step 5: Insert the I/O terminal**



**Supported XV-board for UA-2241M:**

Model	Descriptions
XV511i	<p>4-channel RS-485 I/O Expansion Board (RoHS)</p> <p style="text-align: center;">PIN Assignment:</p>  <p style="text-align: center;">P4: ttyO9    P3: ttyO8    P2: ttyO7    P1: ttyO6</p>

For more detailed information about the **XV511i** board specifications, please refer to the XV-Board Modules.

<https://www.icpdas.com/en/product/XV511i>

For all XV-Board:

[https://www.icpdas.com/en/product/guide+Remote+\\_I\\_O\\_Module\\_and\\_Unit+PAC\\_%EF%BC%86amp;+\\_Local+\\_I\\_O\\_Modules+XV-board](https://www.icpdas.com/en/product/guide+Remote+_I_O_Module_and_Unit+PAC_%EF%BC%86amp;+_Local+_I_O_Modules+XV-board)