

-Microsoft Azure IoT Starter Kit-



泓格科技股份有限公司

Warning

ICP DAS Inc., LTD. assumes no liability for damages consequent to the use of this product. ICP DAS Inc., LTD. reserves the right to change this manual at any time without notice. The information furnished by ICP DAS Inc. is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS Inc., 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 2017 by ICP DAS Inc., 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.

Table of Contents

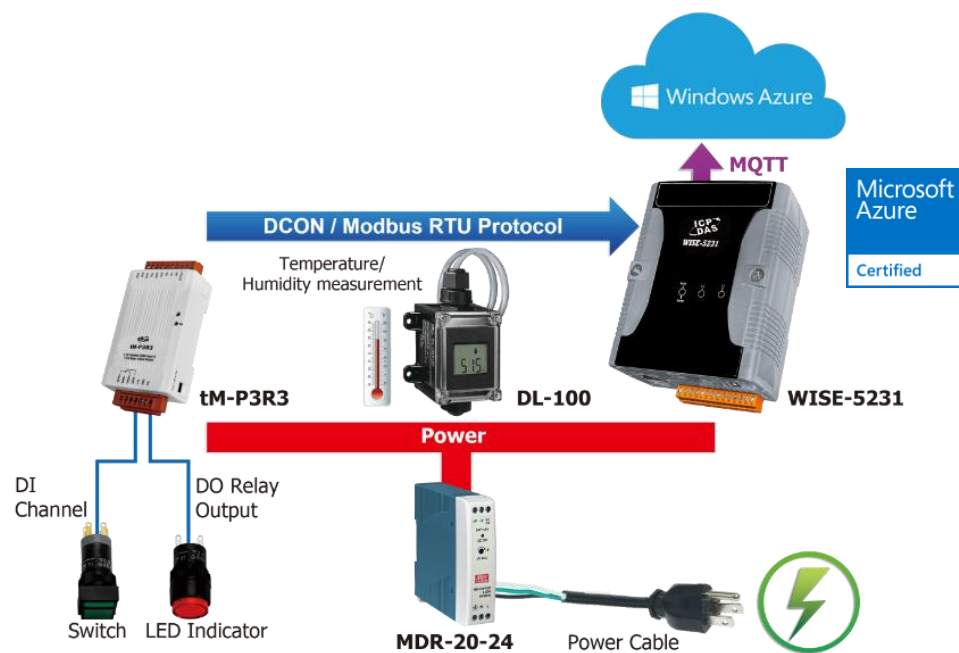
1	Introduction.....	1
2	Create an IoT Hub	4
3	Register a device for WISE-5231 in the IoT Hub	7
4	Setup WISE Monitoring IoT Kit	9
5	Connect WISE-5231 to Azure IoT Hub	11
6	Resource	16

1 Introduction

Microsoft and ICP DAS have teamed up to bring you the easy way to implement the IoT (Internet of Things) Cloud system. The WISE Monitoring IoT Kit has been designed to help you seamlessly connect the Sensors and I/O modules to the cloud with the Microsoft Azure IoT. This kit includes an ICP DAS WISE-5231, a Temperature/Humidity module, an 3-channel DI/3-channel Relay Output module, and a 24W Industrial Power Supply. There are also a LED Indicator, Switch and wires to help you set up your Temperature/Humidity monitoring system. Once your WISE-5231 is connected to Microsoft Azure you can start visualizing and analyzing your data.

Microsoft Azure is a leading provider of cloud computing and Microsoft Azure IoT Hub enables secure, reliable bi-directional communications between IoT endpoints such as sensors and the cloud. Azure IoT Hub supports a broad set of operating systems (Linux, Windows, RTOS etc.), protocols and common languages, so you can configure your connections to the devices.



WISE-5231 is a product developed by ICP DAS that functions as control units for use in remote logic control and monitoring in various industrial applications. WISE offers a user-friendly and intuitive web site interface that allows users to implement IF-THEN-ELSE control logic on controllers just a few clicks away; no programming is required. WISE-5231 provides flexible integration with the Sensor and I/O module, and features various functions such as: built-in IF-THEN-ELSE logic engine, Schedule/Timer operation, data logging, CGI command sending/receiving and Email alarm notification. In addition, WISE-5231 also supports powerful Network connection ability for seamless integration with the Microsoft Azure IoT. All of these make WISE-5231 not only a Real-time automation controller of I/O modules and Sensors at the field site; it is also a Concentrator/Gateway to collect/transfer the data of the Sensors and I/O modules to the Microsoft Azure IoT Cloud platform. WISE-5231 is a cost-effective Concentrator of the Sensors and I/O modules for the Microsoft Azure IoT Cloud platform.



Features:

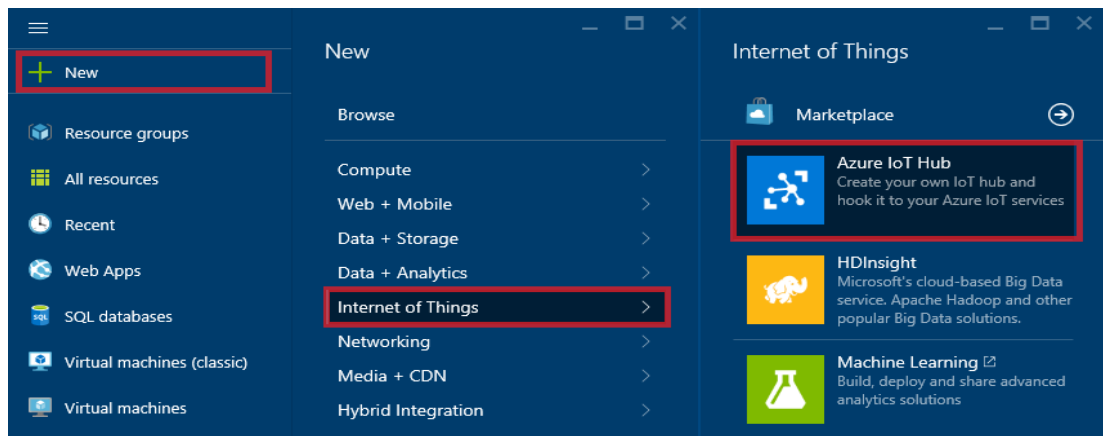
- ◆ Runs on browsers, no extra software tool is required.
- ◆ No more programming, user-friendly web pages are provided for building the IoT Cloud system.
- ◆ Ready-to-run IoT Solutions: Includes an Intelligent Sensor Concentrator, a Temperature/Humidity module, an I/O module, and Microsoft Azure service..
- ◆ Completed Application Scenario: sensor data collection and Real-time automation control can be performed at the field-site, and the data can be transferred to Microsoft Azure IoT platform for analysis.
 - ✧ Flexible integration with the Sensor and I/O module by Modbus protocol.
 - ✧ Powerful automation control, data logger and alarm notification functions at field site.
 - ✧ Seamless integration with Microsoft Azure IoT service without programming.

What’s in the Box?

ICP DAS WISE-5231 Industrial IoT Concentrator	ICP DAS DL-100 Temperature and Humidity Module	ICP DAS tM-P3R3 3-channel Digital Input and 3-channel Relay Output Module	ICP DAS MDR-20-24 24W Industrial Power Supply
			
LED Indicator (RED)	Switch	Power cable	
			

2 Create an IoT Hub

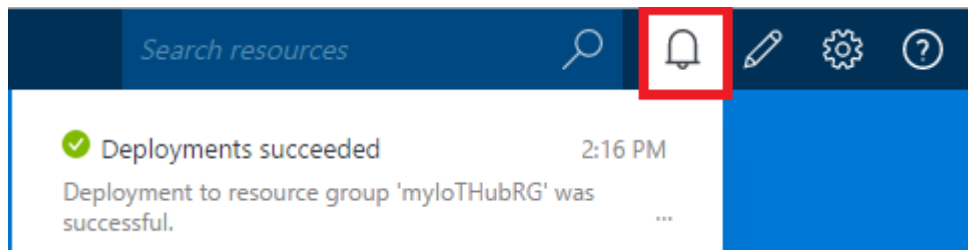
- i. In the Azure portal, click **New > Internet of Things > IoT Hub**.



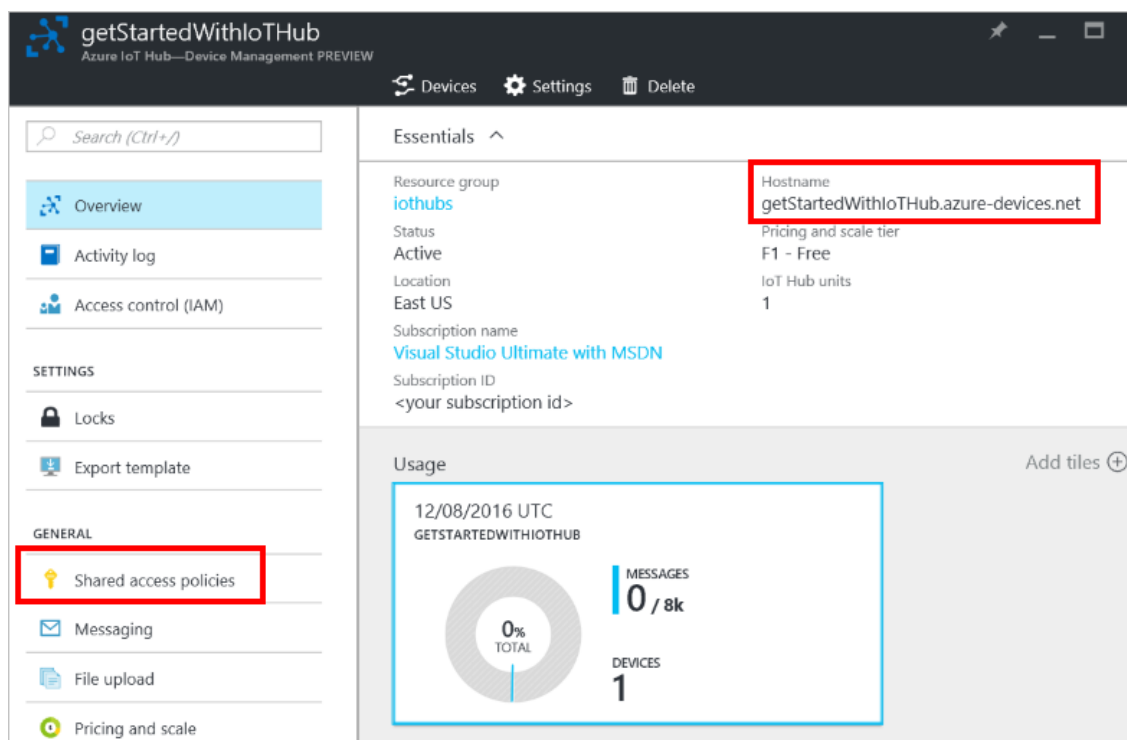
- ii. In the IoT hub pane, enter the following information for your IoT hub:

- In the **Name** box, enter a name to identify your IoT hub. When the **Name** is validated, a green check mark appears in the **Name** box.
- Change the **Pricing and scale tier** as desired. The getting started samples do not require a specific tier.
- In the **Resource group** box, create a new resource group, or select an existing one. For more information, see [Using resource groups to manage your Azure resources](#).
- Use **Location** to specify the geographic location in which to host your IoT hub.

- iii. Once the new IoT hub options are configured, click **Create**. It can take a few minutes for the IoT hub to be created. To check the status, you can monitor the progress on the Startboard. Or, you can monitor your progress from the Notifications section.



- iv. After the IoT hub has been created successfully, open the blade of the new IoT hub, take note of the hostname URI, and click **Shared access policies**.



- v. In the **Shared access policies** pane, click the **iothubowner** policy, and then copy and make a note of the **Connection string** of your IoT hub. For more information, see [Control access to IoT Hub](#).

The screenshot shows the 'iothubowner' policy configuration in the Azure IoT Hub portal. The left sidebar shows the 'Shared access policies' option under 'SETTINGS'. The main area displays a table of policies and their permissions. The 'iothubowner' policy is selected, and its permissions are listed on the right. The 'Connection string—primary key' field is highlighted with a red box.

POLICY	PERMISSIONS
iothubowner	registry write, service connect, device connect, registry read, registry write

iothubowner

Access policy name: iothubowner

Permissions:

- ☒ Registry read
- ☒ Registry write
- ☒ Service connect
- ☒ Device connect

Shared access keys:

Primary key: fky+kg960fVX19XDOJ02WjNMPb6DaLhG

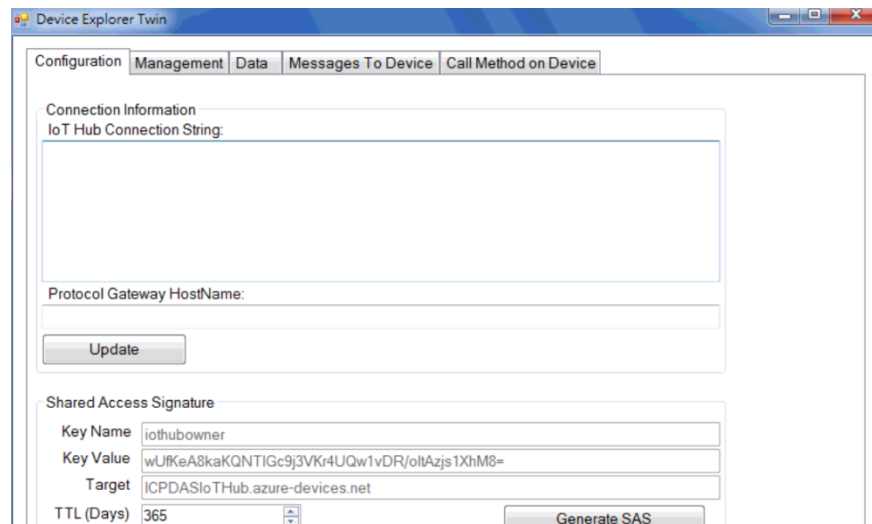
Secondary key: bPFekUT+b/QGNdl/B/pYWs4xJnMFpJCOJ

Connection string—primary key: HostName=ioTGetStarted.azure-devices.r

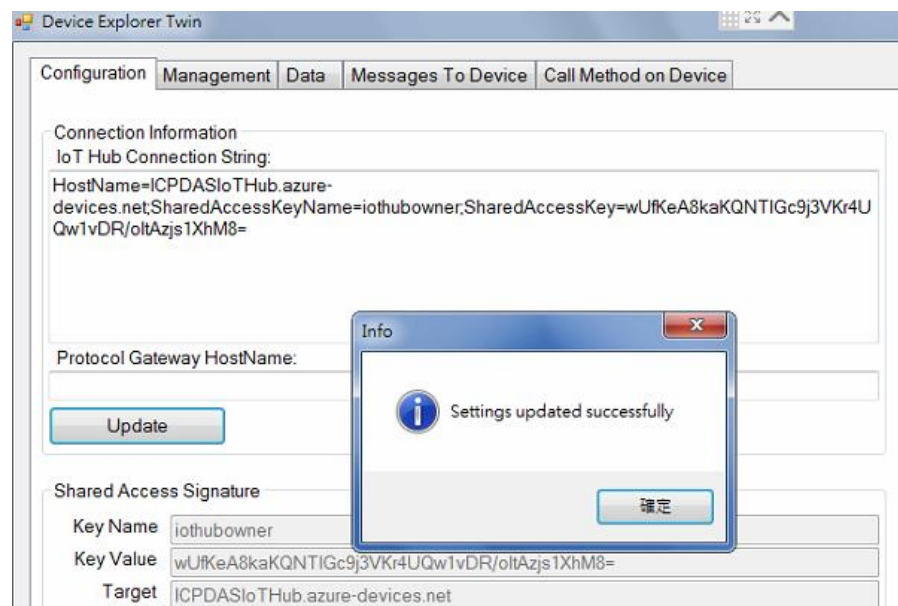
3 Register a device for WISE-5231 in the IoT Hub

- i. Download **SetupDeviceExplorer.msi** like link as below and install it.

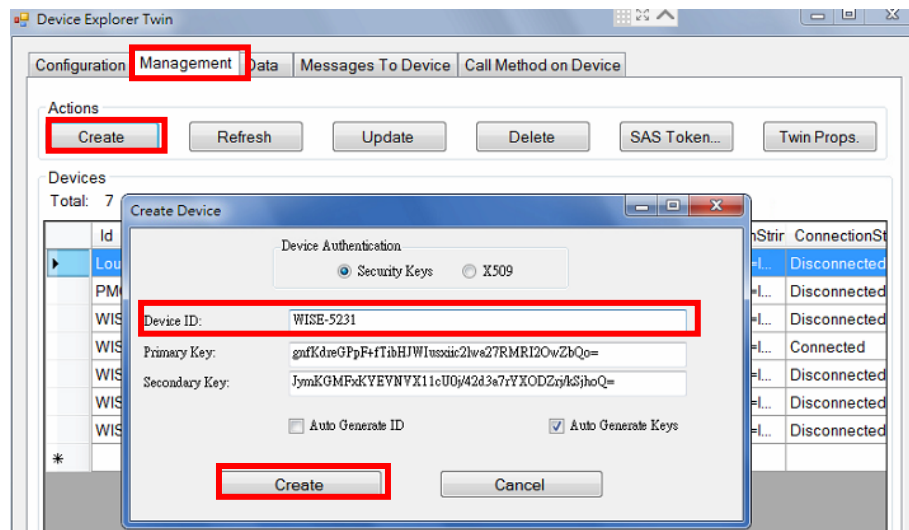
<https://github.com/Azure/azure-iot-sdks/releases>



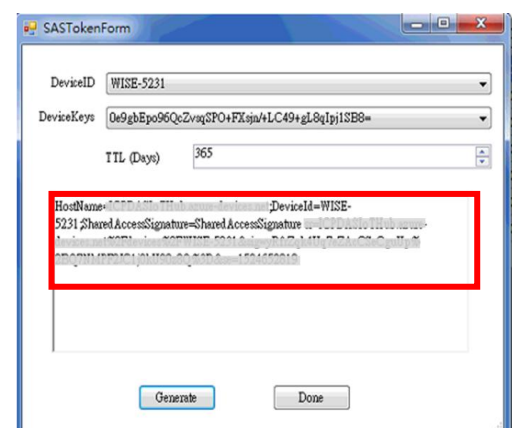
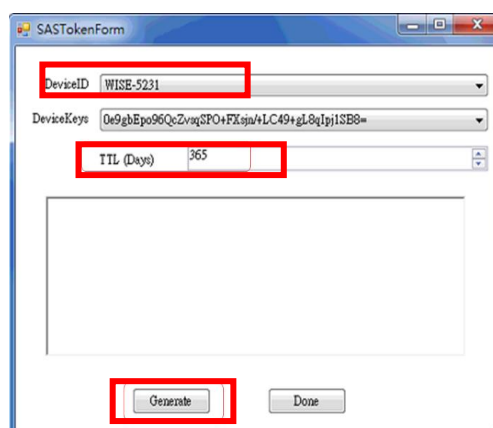
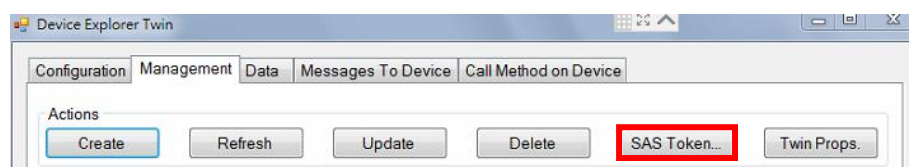
- ii. Open and go to **Configuration** window, paste the **Connection String** of your IoT hub, and click the Update button, and then the Device Explorer connects to your IoT hub successfully.



- iii. Switch to the **Management** window and click the Create button to add the device. Key in the Device ID and press the Create button to create a new device in your IoT hub.



- iv. Click the **SAS Token** button to get SAS Token of the new device:
- Select the Device ID of WISE-5231.
 - Set the TTL (Days) to 365. The TTL (Days) means the Time-To-Live days of this SAS Token.
 - Press the **Generate** button.
 - Copy and make a note of this SAS Token.

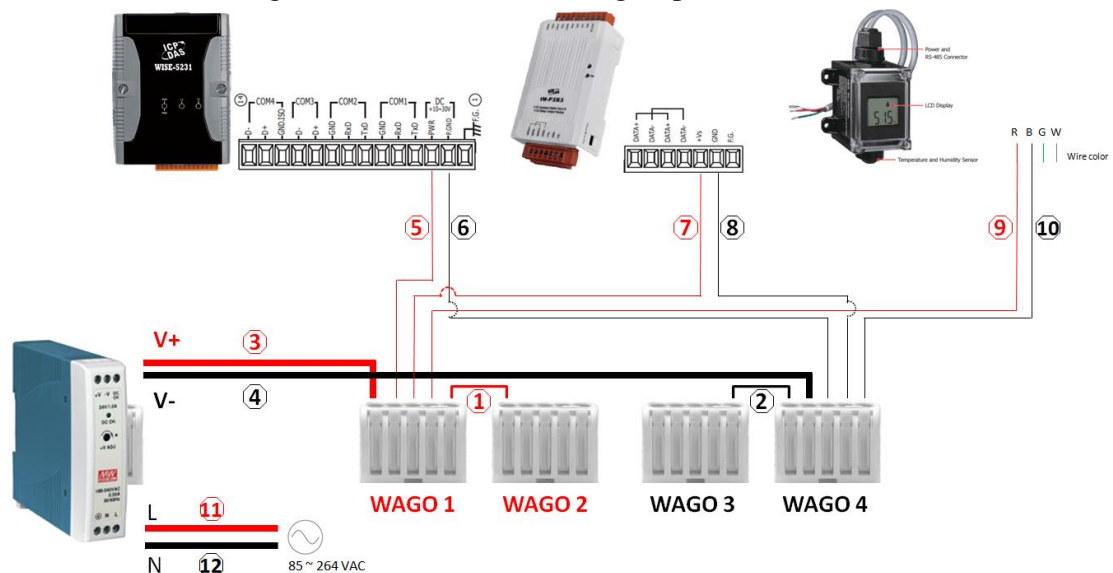


4 Setup WISE Monitoring IoT Kit

Connect the modules as bellow provided by the IoT Kit.

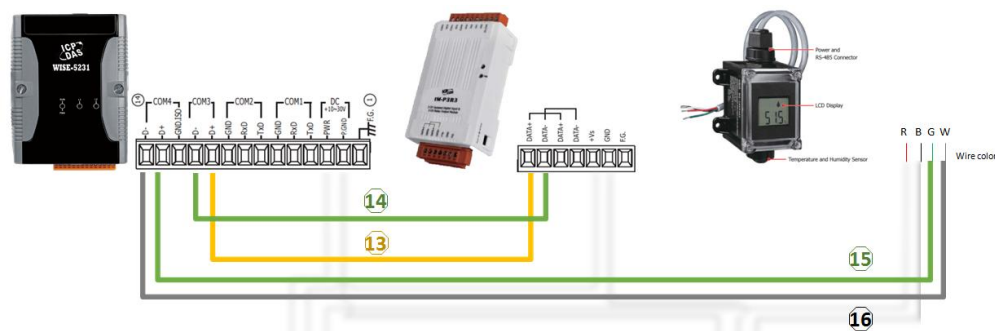
- WISE-5231
- tM-P3R3
- DL-100T485
- MDR-20-24
- LED Indicator (Red)
- Switch
- Power cable

i. Please refer to the figure as below for the wiring of power.



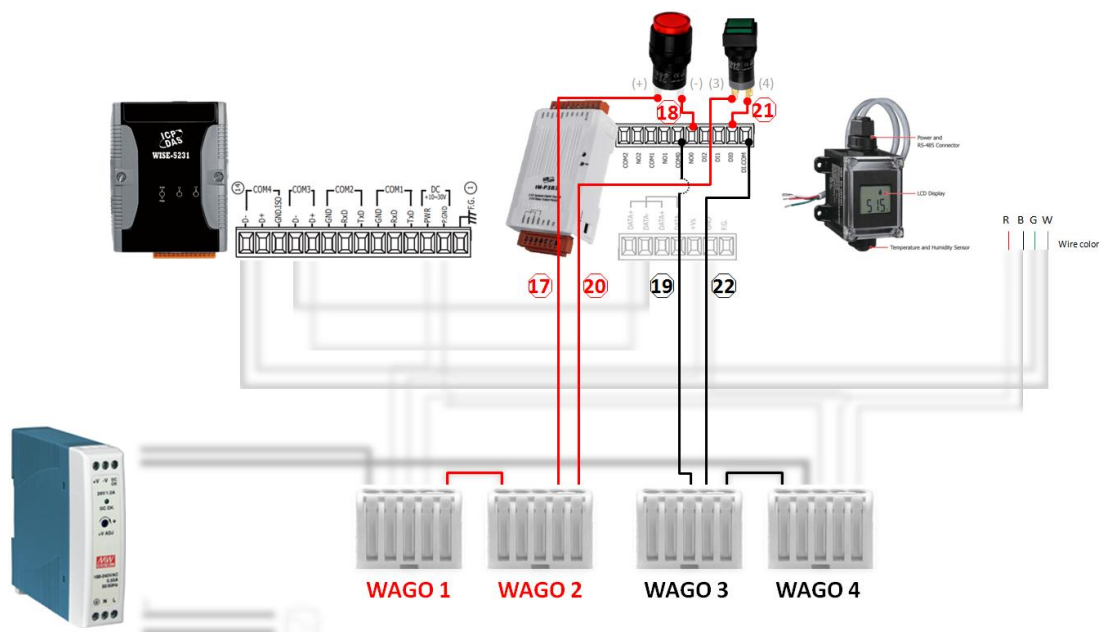
Step	Decription
1	Use the red wire(5CM) to connect two WAGOs(WAGO 1, WAGO 2)
2	Use the black wire(5CM) to connect two WAGOs(WAGO 3, WAGO 4)
3	Use the red wire(30CM) to connect MDR-20-24 DC V+ with WAGO 1
4	Use the black wire(30CM) to connect MDR-20-24 DC V- with WAGO 4
5	Use the red wire(15CM) to connect WISE-5231 PWR with WAGO 1
6	Use the black wire(15CM) to connect WISE-5231 P.GND with WAGO 4
7	Use the red wire (15CM) to connect tM-P3R3 +Vs with WAGO 1
8	Use the black wire(15CM) to connect tM-P3R3 GND with WAGO 4
9	Connect DL-100's red wire with WAGO 1
10	Connect DL-100's black wire with WAGO 4
11,12	Connect the Power Cable with MDR-20-24 AC end L,N pin

ii. Please refer to the figure as below for the wiring of RS-485 communication.



Step	Decription
13	Use the yellow wire(15CM) to connect tM-P3R3 Data+ with WISE-5231 COM3 D+
14	Use the green wire(15CM) to connect tM-P3R3 Data- with WISE-5231 COM3 D-
15	Connect DL-100's green wire with WISE-5231 COM4 D+
16	Connect DL-100's white wire with WISE-5231 COM4 D-

iii. Please refer to the figure as below for the wiring of LED and Switch.



Step	Decription
17	Connect LED Pin+ with WAGO 2
18	Connect LED Pin- with tM-P3R3 NO0
19	Use the black wire(30CM) to connect tM-P3R3 COM0 with WAGO 3
20	Connect switch Pin3 with WAGO 2
21	Connect switch Pin4 with tM-P3R3 DO0
22	Use the black wire(30CM) to connect tM-P3R3 DI COM with WAGO 3

5 Connect WISE-5231 to Azure IoT Hub

Step1: Prepare your Device

- Follow the instruction described in this [Quick Start](#) to Connect to the Web interface of WISE-5231.
- Follow the instruction described in this [User Manual](#) to set tM-P3R3 and Module parameters following table.

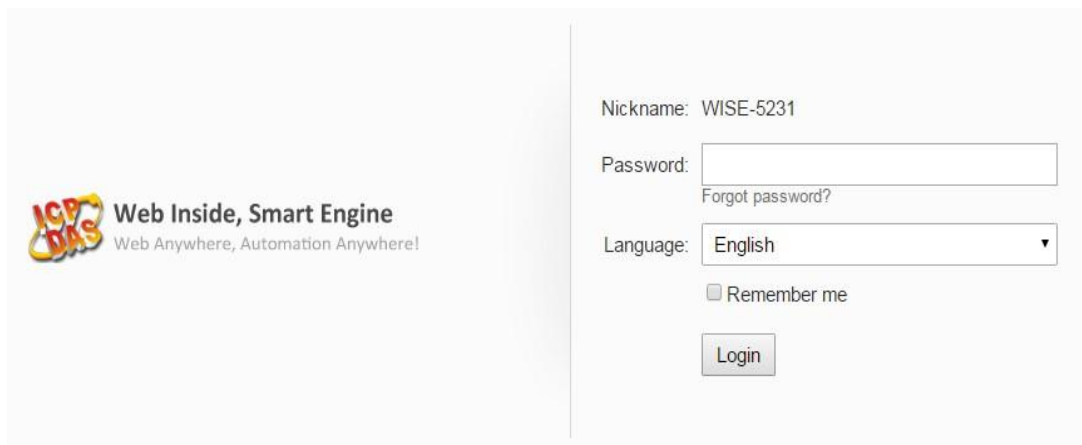
Module Name	Serial port parameters	Modbus Address
tM-P3R3	9600 N,8,1 (Default)	1 (Default, Connect to WISE-5231 COM3)

- Follow the instruction described in this [Quick Start](#) to set DL-100T485 and Module parameters following table.

Module Name	Serial port parameters	Modbus Address
DL-100T485	9600 N,8,1 (Default)	1 (Default, Connect to WISE-5231 COM4)

Step 2: Build the sample

- Connect to WISE-5231's webpage server via Web browser, login with the default password "**Admin**".



- Go to the "**System setting >> COM Port Interface Setting**" page to complete the setting of COM3(Modbus RTU Master) and COM4(DCON Master).

Web Inside, Smart Engine
Web Anywhere, Automation Anywhere!

WISE-5231

3694.2MB(Approx. 3619 Days) Instant Message

System Setting Module Setting Logger Setting IoT Platform Setting Advanced Setting Rule Setting Channel Status

System Setting COM Port Interface Setting

COM Port Interface Setting Page

COM2 COM3 COM4

Function: Modbus RTU Master

Baudrate: 9600 bps

Parity: ☒ None ☐ Odd ☐ Even

Stop bits: ☒ 1 ☐ 2

Silent Interval: 200 milliseconds(s)

COM Port Interface Setting Page

COM2 COM3 COM4

Function: DCON Master

Baudrate: 9600 bps

Parity: ☒ None ☐ Odd ☐ Even

Stop bits: ☒ 1 ☐ 2

Timeout: 1000 milliseconds(s)

Checksum: ☒ Disable ☐ Enable

- iii. Go to the "Module Setting >> Remote I/O Module Setting" page to add tM-P3R3 in COM3, and add DL-100 in COM4.

System Setting Module Setting Logger Setting IoT Platform Setting Advanced Setting Rule Setting Channel Status IP Camera Status

Module Setting Remote I/O Module Setting

Modbus RTU Module List

COM3 COM4 LAN

No.	Address	*Module Name / Nickname	Polling Timeout(ms)	Retry Interval(secs)
2	2	tM-P3R3	1000	5

DCON Module List

COM3 COM4 LAN

No.	Address	*Module	DI	DO	AI	AO	Nickname
2	2	DL-100	0	0	3	0	

- iv. Complete the setting and download the setting to WISE-5231, and then go to the "Channel Status" page to check the module communication status.

Web Inside, Smart Engine
Web Anywhere, Automation Anywhere!

WISE-5231

3688.2MB(Approx. 2852 Days) The system is busy, please try again later.

System Setting Module Setting Logger Setting IoT Platform Setting Advanced Setting Rule Setting Channel Status

Channel Status DL-100(1)

COM3 689ms

tM-P3R3(1)

COM4 238ms

DL-100(1)

Other

Internal Register

Event List

Log File List

CGI File List

Humidity	Temperature(°C)	Temperature(°F)
46.240 %	24.300 °C	75.720 °F

© ICP DAS Co., Ltd. All Rights Reserved

- v. Go to the "Microsoft Azure Platform Setting" page.

The screenshot shows the 'Microsoft Azure Setting Page' in the WISE-5231 interface. The page has a sidebar with 'Microsoft Azure Platform Setting', 'IBM Bluemix Platform Setting', and 'MQTT Setting'. The main content area includes a 'Function Status' section with a checked 'Enable' checkbox. Below this is a large text input field for the '*SAS Token'. Further down are 'Keep Alive Time' (set to 60 seconds) and 'Periodical Publish Interval' (set to 5 seconds). At the bottom, there is a 'Connection Testing' section with a 'Testing' button.

- vi. Input the SAS Token generated by Device Explorer. (please refer previous section)

This screenshot shows the 'SASTokenForm' dialog box overlaid on the 'Microsoft Azure Setting Page'. The dialog box contains fields for 'DeviceID' (WISE-5231), 'DeviceKey' (0e9gbEpo96Qc2vqSFO+FXjsu+LC49+g18q1p1SB8=), and 'TTL (Days)' (365). A red box highlights the 'HostName' field, which contains a long alphanumeric string. Below the dialog box, the 'Connection Testing' button in the background is highlighted with a red box and labeled 'Testing Connect successfully'.

- vii. Complete the Publish Message editing.

The screenshot displays the 'Publish & Subscribe Setting' page. It includes a 'Publish' button and a 'Subscribe' button. Below these are two rows of settings for 'value_for_temperature' and 'value_for_humidity', each with a corresponding message template. At the bottom, there are 'Setting', 'Copy', and 'Remove' buttons.

Nickname	Message
value_for_temperature	{ "temperature": "DL-100 Temperature(°C)" }
value_for_humidity	{ "humidity": "DL-100 Humidity" }

- viii. Complete the Subscribe Message editing and click the **"Save"** button to save the settings.

Keep Alive Time second(s)

Periodical Publish Interval second(s)
Input 0 represent disable periodical publish.

Connection Testing

Publish & Subscribe Setting

Variable Name

- ix. Go to the **"Rule Setting"** page to add a rule to turn the relay on when receive the message from Azure, then remember to download the setting to WISE-5231.

Web Inside, Smart Engine
Web Anywhere, Automation Anywhere!

WISE-5231

System Setting Module Setting Logger Setting IoT Platform Setting Advanced Setting **Rule Setting** Channel Status

Rule Setting Add new rule

+ Add new rule

Rule Information Setting

*Nickname

Description

Status ☒ Enable ☐ Disable

Rule Content Setting

IF

Add a new Condition:

Microsoft Azure Subscribe Message(LED) = ON

THEN

Add a new Action:

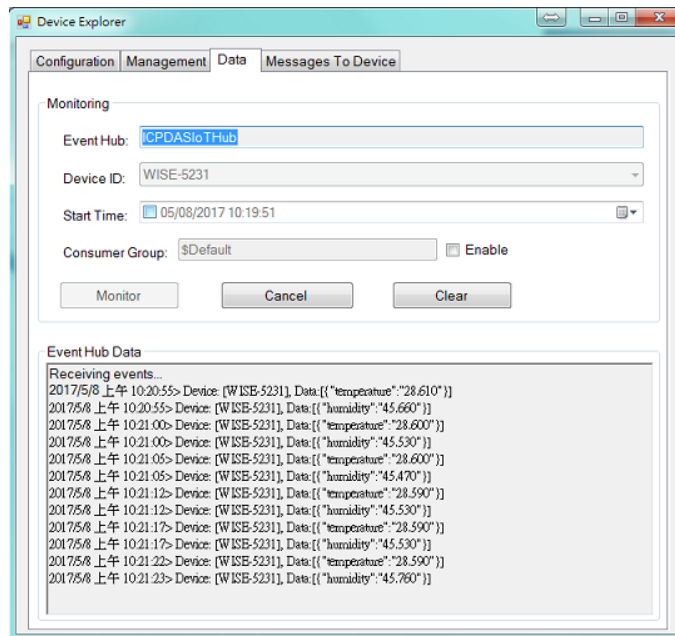
COM3 tM-P3R3(1) DO0 = ON

ELSE

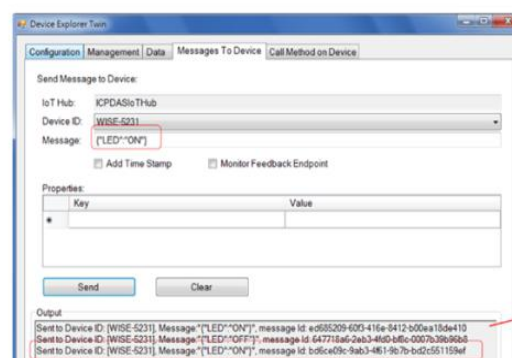
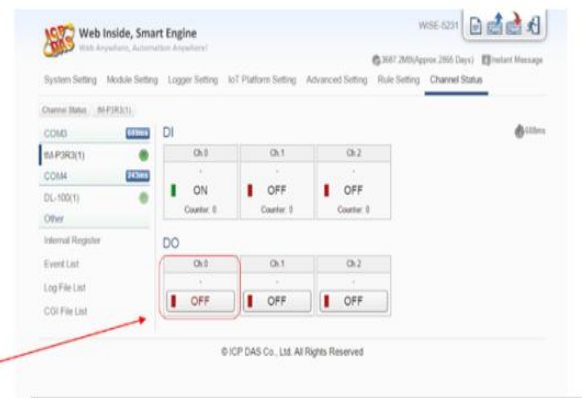
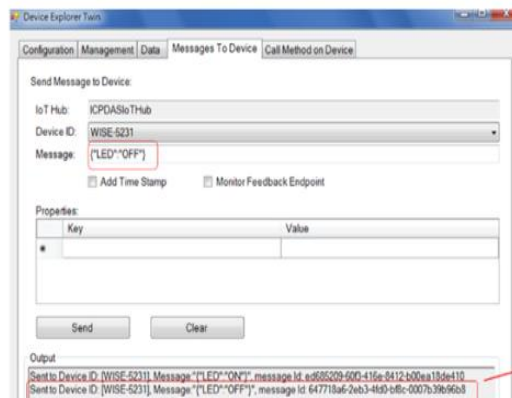
Add a new Action:

COM3 tM-P3R3(1) DO0 = OFF

- x. Use the Device Explorer utility to verify if the IoT Hub receives the messages from WISE-5231. Go to the **"Data"** window, select the Device ID of WISE-5231 and press the **"Monitor"** button. And then Device Explorer would receive the messages send by WISE-5231.



- xi. Go to the "**Channel Status**" page to observe that the LED status changes when WISE-5231 gets the message send by Device Explorer. Go to the "Messages To Device" window of Device Explorer, select the Device ID of WISE-5231, and input the Messages `{"LED": "ON"}` or `{"LED": "OFF"}` in the "Message" field and then press the "Send" button to change the status of the LED indicator.



6 Resource

- [ICP DAS WISE Monitoring IoT Kit URL: http://wise.icpdas.com/products/WISE_IoTKit_01.html](http://wise.icpdas.com/products/WISE_IoTKit_01.html)
- [Microsoft Azure IoT Starter Kits URL: http://aka.ms/iotstarterkitss](http://aka.ms/iotstarterkitss)