**User Manual**

Getting Started with I-CUBE-TENCENT IoT Cloud Software expansion for STM32Cube

Introduction

This user manual describes the content of the STM32Cube software expansion package for TENCENT® IoT cloud platform.

The TENCENT® IoT cloud software expansion package (I-CUBE-TENCENT) for STM32Cube provides application examples that connect STMicroelectronics boards to TENCENT® IoT cloud platform. It uses the TENCENT® **qcloud-iot-explorer-sdk-embedded-c** as middleware which is compiled and running on the STM32 device.

I-CUBE-TENCENT runs on the NUCLEO-L476RG board.

Implementation examples are included for device-to-cloud telemetry reporting, cloud-to-device messages for notifications to the connected devices. I-CUBE-TENCENT offers the following features:

• Ready to run firmware example using Wi-Fi® to support quick evaluation and development of device applications connected to TENCENT® IoT cloud platform.

• Board configuration interface

• Wi-Fi® connection

• Connection to the TENCENT® IoT cloud platform

• The sensor expansion board X-NUCLEO-IKS01A3 measures and report any one of the following values:

– Temperature (external)

– Humidity

– Pressure

– 3D Accelerometer data

– 3D Gyroscope data

– 3D Magnetometer data



Contents

[1 General information 6](#_Toc35614509)

[2 TENCENT® IoT cloud platform 7](#_Toc35614510)

[3 Package description 8](#_Toc35614511)

[3.1 General description 8](#_Toc35614512)

[3.2 Architecture 10](#_Toc35614513)

[3.3 Folder structure 12](#_Toc35614514)

[3.4 X-NUCLEO-IKS01A3 board sensors 12](#_Toc35614515)

[3.5 ESP-WROOM-02D Wi-Fi® expansion board 13](#_Toc35614516)

[3.6 Reset push-button (Black button) 13](#_Toc35614517)

[3.7 User push-button (Blue button) 14](#_Toc35614518)

[3.8 User LED 14](#_Toc35614519)

[4 Hardware and software environment setup 15](#_Toc35614520)

[5 Interacting with the boards 16](#_Toc35614521)

[6 Getting started with TENCENT® IoT explorer platform 18](#_Toc35614522)

[6.1 TENCENT® IoT explorer account creation 18](#_Toc35614523)

[6.2 Device creation on the TENCENT® IoT explorer platform 18](#_Toc35614524)

[6.3 Application build and first launch 25](#_Toc35614525)

[6.4 Application First Launch 26](#_Toc35614526)

[6.5 Application Runtime 27](#_Toc35614527)

[7 Frequently asked questions 30](#_Toc35614528)

[8 Revision history 31](#_Toc35614529)

List of tables

[Table 1. List of acronyms 6](#_Toc35614530)

[Table 2. Document revision history 31](#_Toc35614531)

List of figures

[Figure 1. TENCENT® IoT cloud ecosystem 7](#_Toc35614532)

[Figure 2. NUCLEO-L476RG board 8](#_Toc35614533)

[Figure 3 ESP-WROOM-02D WIFI module 8](#_Toc35614534)

[Figure 4. X-NUCLEO-IKS01A3 board 9](#_Toc35614535)

[Figure 5. I-CUBE-TENCENT software architecture 11](#_Toc35614536)

[Figure 6. Project Folder Structure 12](#_Toc35614537)

[Figure 7. Download firmware to ESP-WROOM-02D 13](#_Toc35614538)

[Figure 8. Hardware and software setup environment 15](#_Toc35614539)

[Figure 9. Terminal Setup 16](#_Toc35614540)

[Figure 10. Serial port setup 17](#_Toc35614541)

[Figure 11. TENCENT® IoT explorer account creation 18](#_Toc35614542)

[Figure 12.TENCENT® IoT explorer project creation 18](#_Toc35614543)

[Figure 13. Create a Product using Common Data Template 19](#_Toc35614544)

[Figure 14. Create a Product using Customized Data Template 19](#_Toc35614545)

[Figure 15. Create a Property 20](#_Toc35614546)

[Figure 16. Create an Event 20](#_Toc35614547)

[Figure 17. Data Template Information 21](#_Toc35614548)

[Figure 18. Export Data Template to JSON file 22](#_Toc35614549)

[Figure 19. Generate Data Template C code 22](#_Toc35614550)

[Figure 20. Create a New Device 22](#_Toc35614551)

[Figure 21．Mini Program control panel 23](#_Toc35614552)

[Figure 22. Config shortcut for Mini Program 23](#_Toc35614553)

[Figure 23. WIFI® configuration for device 24](#_Toc35614554)

[Figure 24. Device information 24](#_Toc35614555)

[Figure 25. Update device information 25](#_Toc35614556)

[Figure 26. Device provisioning 25](#_Toc35614557)

[Figure 27. Pins connection between NUCLEO board and WIFI® module 26](#_Toc35614558)

[Figure 28. Hardware connection 26](#_Toc35614559)

[Figure 29. Default WIFI® configuration 27](#_Toc35614560)

[Figure 30. WiFi® provisioning 27](#_Toc35614561)

[Figure 31. Run the icube\_data\_template\_sample application 28](#_Toc35614562)

[Figure 32. Report properties 28](#_Toc35614563)

[Figure 33. Post events when blue button is pressed 28](#_Toc35614564)

[Figure 34. Send and receive data in the debug window at real-time 28](#_Toc35614565)

[Figure 35. Events posted by device 28](#_Toc35614566)

[Figure 36. The latest property status reported 29](#_Toc35614567)

[Figure 37. Logs on cloud 29](#_Toc35614568)

[Figure 38. Pop-up when the IAR™ IDE version is not compatible 30](#_Toc35614569)

# General information

The I-CUBE-TENCENT package for the TENCENT® IoT cloud platform runs on STM32 32-bit microcontrollers based on the Arm® Cortex®-M processor.

*Table 1* presents the definition of acronyms that are relevant for a better understanding of this document.

Table 1. List of acronyms

|  |  |
| --- | --- |
| Term | Definition |
| API | Application programming interface |
| BSP | Board support package |
| CA | Certification authority |
| DHCP | Dynamic host configuration protocol |
| DNS | Domain name server |
| HAL | Hardware abstraction layer |
| IDE | Integrated development environment |
| IoT | Internet of things |
| IP | Internet protocol |
| LED | Light-emitting diode |
| SDK | Software development kit |
| RTC | Real-time clock |
| UART | Universal asynchronous receiver/transmitter |

# TENCENT® IoT cloud platform

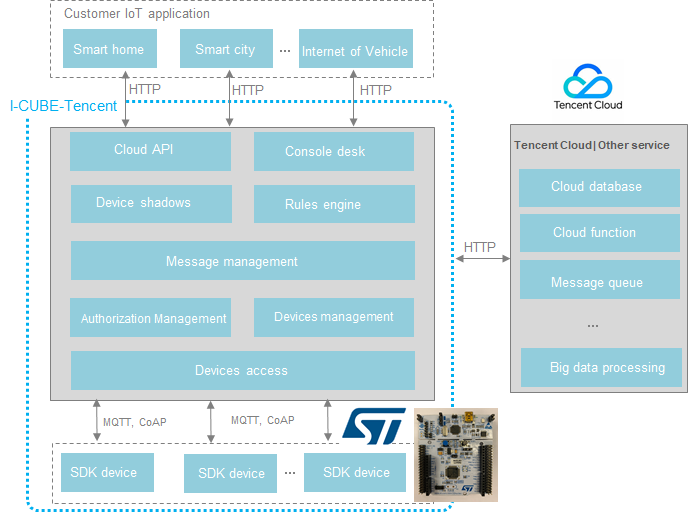
The I-CUBE-TENCENT package implements the TENCENT qcloud-iot-explorer-sdk-embedded-c which allows the board to securely connect to the TENCENT® IoT cloud platform.

TENCENT® IoT explorer platform provides development resources with customizable data protocol templates.

Users can connect to the cloud with a smartphone or personal computer and have access to the information provided by the board at any time and from any location.

*Figure 1* presents the TENCENT® IoT cloud ecosystem targeted by the I-CUBE- TENCENT® package.

Figure 1. TENCENT® IoT cloud ecosystem



# Package description

This chapter details the I-CUBE-TENCENT package content and the way to use it.

## General description

The I-CUBE-TENCENT package provides a TENCENT® qcloud-iot-explorer-sdk-embedded-c stack middleware for STM32 microcontrollers.

It is built to run on the NUCLEO-L476RG board and connects to the Internet through the ESP-WROOM-02D Wi-Fi® network interface:

• ESP-WROOM-02D supports Wi-Fi® connectivity with an on-board ESPRESSIF® Wi-Fi® module.

Figure 2. NUCLEO-L476RG board

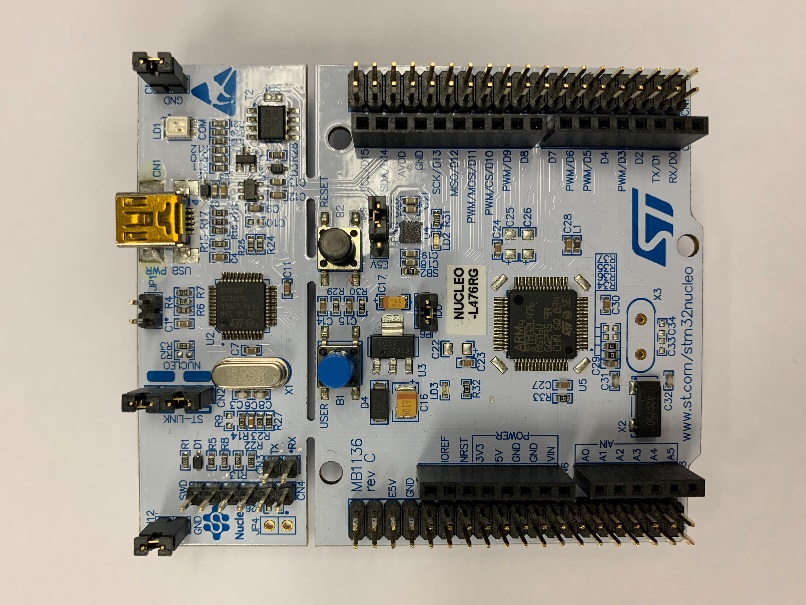
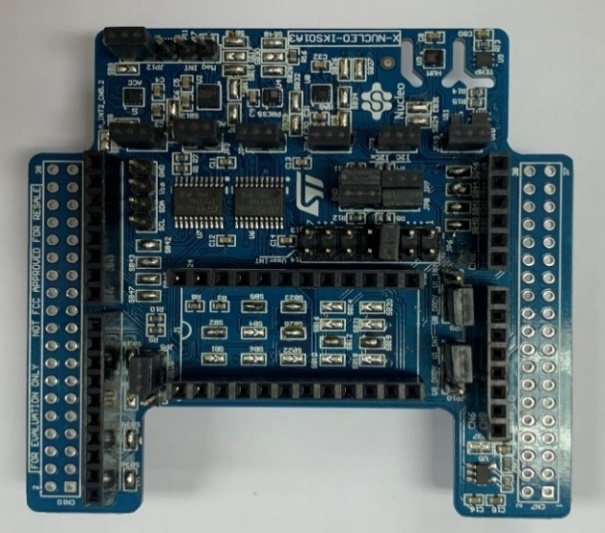


Figure 3 ESP-WROOM-02D WIFI module

图片包含 电子产品

描述已自动生成

Figure 4. X-NUCLEO-IKS01A3 board



The package is split into the following software components:

• TENCENT® qcloud-iot-explorer-sdk-embedded-c for connecting to TENCENT® IoT cloud platform.

• Wi-Fi® drivers (currently, it is included the qcloud-iot-explorer-sdk-embedded-c)

• Sensor drivers for the X-NUCLEO-IKS01A3 board

• STM32L4 Series HAL

• TENCENT® application examples

The software is provided as a zip archive containing source code. The following integrated development environments are supported:

• IAR Embedded Workbench® for Arm® (EWARM)

• Keil® Microcontroller Development Kit (MDK-ARM)

• STM32CubeIDE

Note: Refer to the release note available in the package root folder for information about the IDE versions supported.

## **Architecture**

This section describes the software components of the I-CUBE-TENCENT package.

The I-CUBE-TENCENT software is an expansion for the STM32Cube. Its main features and characteristics are:

• Fully compliant with STM32Cube architecture

• Expands STM32Cube in order to enable the development of applications accessing and using the TENCENT® IoT cloud platform

• Based on the STM32CubeHAL, which is the hardware abstraction layer for STM32 microcontrollers

The software components used by the application software to access and use the TENCENT® IoT cloud platform are the following:

1. STM32Cube HAL

The HAL driver layer provides a generic multi-instance simple set of APIs (Application Programming Interfaces) to interact with the upper layers (application, libraries and stacks).

It is composed of generic and extension APIs. It is directly built around a generic architecture and allows the layers that are built upon, such as the middleware layer, to implement their functionalities without dependencies on the specific hardware configuration for a given microcontroller unit (MCU).

This structure improves the library code reusability and guarantees an easy portability onto other devices.

1. Board Support Package (BSP)

The software package needs to support the peripherals on the STM32 boards apart from the MCU. This software is included in the board support package (BSP). This is a limited set of APIs which provides a programming interface for certain board specific peripherals such as the LED and the user button.

1. TENCENT® **qcloud-iot-explorer-sdk-embedded-c** software development kit (SDK)
2. MbedTLS
3. *Figure 5* outlines I-CUBE-TENCENT software architecture.

Figure 5. I-CUBE-TENCENT software architecture

**Application**

**Tencent sample application**

**Middleware**

**Tencent**

**qcloud-iot-explorer-sdk-embedded-c**

**FreeRTOS**

**MbedTLS**

**Driver**

**STM32L4 HAL**

**BSP**

**Hardware**

WiFi module

STM32L4

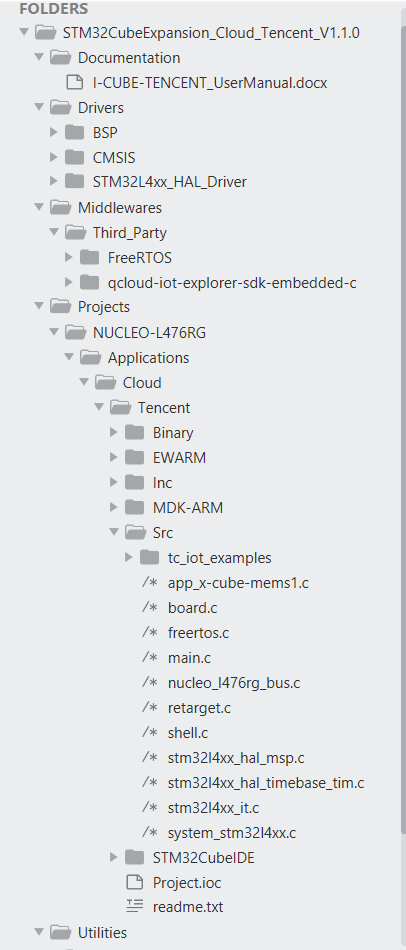
Sensors

*Note: Currently, the drive for WIFI module driver (ESP-WROOM-02D) is included in the qcloud-iot-explorer-sdk-embedded-c.*

## Folder structure

*Figure 6* presents the folder structure of the I-CUBE-TENCENT package.

Figure 6. Project Folder Structure



Tencent qcloud-iot-explorer-sdk-embedded-c

BSP drivers for NUCLEO-L476RG board

BSP drivers for X-NUCLEO-IKS01A3 board

STM32CubeIDE supported

Application and samples

KEIL supported

IAR supported

FreeRTOS

## X-NUCLEO-IKS01A3 board sensors

The sensors that are present on the board that can be used by the sample application are:

• Capacitive digital sensor for relative humidity and temperature (HTS221)

• High-performance 3-axis magnetometer (LIS303AGR)

• 3D accelerometer and 3D gyroscope (LSM6DSL)

• 260-1260 hPa absolute digital output barometer (LPS22HB)

For the application example, the internal temperature sensor data of STM32L is also sent to the cloud.

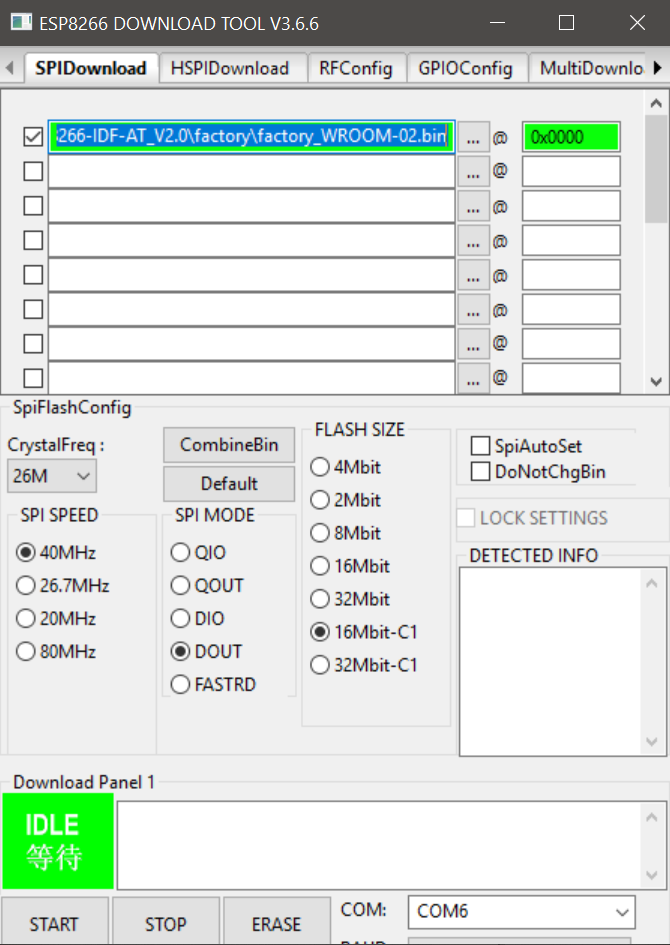
## ESP-WROOM-02D Wi-Fi® expansion board

The Wi-Fi® software is split over Wi-Fi abstraction, module specific high-level API and module specific low level I/O operation.

The ESPRESSIF WIFI module AT firmware version is “ESP8266 IDF AT Bin V2.0”.

Download link: <https://www.espressif.com/sites/default/files/ap/ESP8266-IDF-AT_V2.0_1.zip>

Figure 7. Download firmware to ESP-WROOM-02D



## Reset push-button (Black button)

The reset push-button (black) is used to reset the board at any time. This action makes the board reboot.

## User push-button (Blue button)

The user push-button (blue) is used in the following cases:

• Toggle user LED(LD2) on NUCLEO board and publish the state message to cloud.

The application configures and manages the user button via the board support package (BSP) functions. The BSP functions are in the Drivers\BSP\<board name> directory.

When using the BSP button functions with the BUTTON\_USER value, the application does not take in to account the way this button is connected from a hardware standpoint for a given platform. The mapping is handled by the BSP.

## User LED

The configuration of the user LED that is used by the applications is done via the board support package (BSP) functions.

The BSP functions are under the Drivers\BSP\<board name> directory.

Using the BSP button functions with the LED\_GREEN value, the application does not take in to account the way the LED is mapped for a given platform. The mapping is handled by the BSP.

The ON/OFF behavior of user LED has been selected to indicate the desired LED state during application run time.

# Hardware and software environment setup

To set up the hardware and software environment, the supported board must be plugged into a personal computer via a USB cable. This connection with the PC allows the user to:

• Flash the board

• Select the sample application

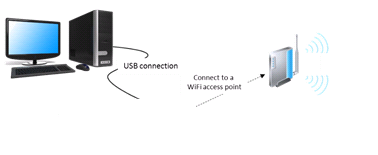
• Set the Wi-Fi® AP credentials

• Interact with the board via a UART console

• Debug

The NUCLEO-L476RG must be connected to a Wi-Fi® access point as illustrated in *Figure 8*.

Figure 8. Hardware and software setup environment



IoT Device

(NUCLEO-L476RG)

The prerequisites for running the examples are:

• A Wi-Fi® access point, with a transparent Internet connectivity meaning that neither a proxy, nor a firewall are blocking the outgoing traffic. It must run a DHCP server delivering the IP and DNS configuration to the board.

• A development PC for building the application, programming through ST-Link, and running the terminal console.

• A TENCENT® developer account. Once registered a Product\_ID, Device Name and Device Secret for TENCENT® IoT cloud platform will be provided.

• To register and create a developer account, go to: <https://cloud.tencent.com/register?s_url=https%3A%2F%2Fcloud.tencent.com%2F>

• A TENCENT® IoT explorer account (see Chapter 6)

# Interacting with the boards

A serial terminal is required to:

• Configure the board

• Display locally the sent/received TENCENT® IoT cloud device-to-cloud/cloud-to-device messages

The example in this document is illustrated with the use of Tera Term. Any other similar tool can be used instead.

• Determine the STM32 ST-LINK Virtual COM port used on the PC for the Discovery board. On a Windows® PC, open the Device Manager.

• Open a virtual terminal on the PC and connect it to the above virtual COM port.

Note: The information provided below in this chapter can be used to configure the UART terminal as an alternative to using the Tera Term initialization script.

Terminal setup is illustrated in *Figure 9*, which shows the terminal setup and the New-line recommended parameters.

The virtual terminal New-line transmit configuration must be set to Linefeed (\n or LF) in order to allow copy-paste from UNIX type text files. The Local echo option makes copy- paste visible on the console.

Figure 9. Terminal Setup



The serial port must be configured with:

• COM port number

• 115200 baud rate

• 8-bit data

• Parity none

• 1 stop bit

• No flow control

Serial port setup is illustrated in *Figure 10*.

Figure 10. Serial port setup



Once the UART terminal and the serial port are set up, press the board reset button (black). Follow the indications on the UART terminal to upload Wi-Fi® provisioning data. Those data remain in Flash and are reused the next time.

# Getting started with TENCENT® IoT explorer platform

This section introduces how to register and log on the TENCENT® IoT explorer platform, how to run TENCENT® IoT explorer sample application in the I-CUBE-TENCENT® package.

## TENCENT® IoT explorer account creation

* + To register and create an IoT explorer account, go to: <https://cloud.tencent.com/product/iotexplorer>

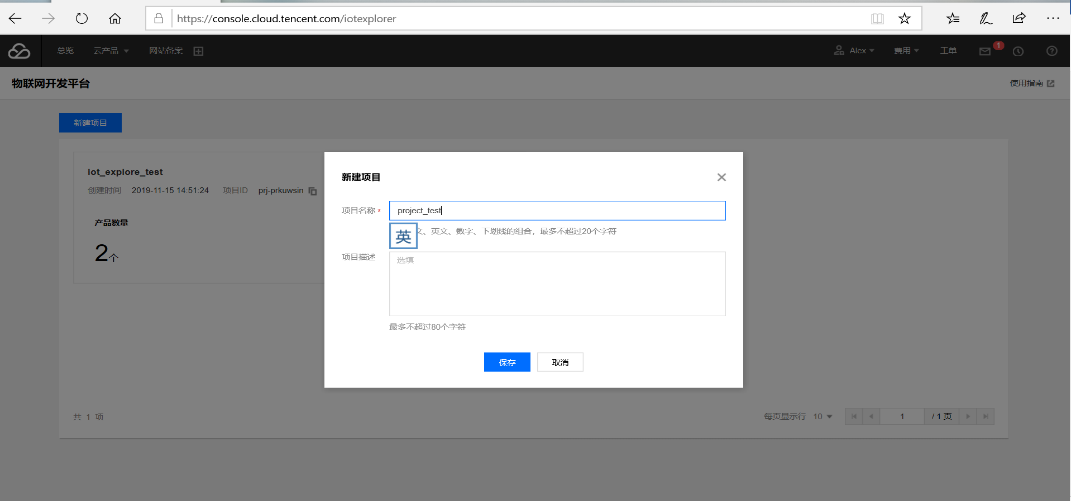
Figure 11. TENCENT® IoT explorer account creation



## Device creation on the TENCENT® IoT explorer platform

1. Create a new project: <https://console.cloud.tencent.com/iotexplorer>

Figure 12.TENCENT® IoT explorer project creation



1. Create a new product: <https://console.cloud.tencent.com/iotexplorer/project/prj-cz86er0i/product/list>

·There are two approaches to create data template for a product:

* 1. Choose a common data template and then modify it
  2. Create a new customized data template.

·Developers can find more details from below links:

<https://cloud.tencent.com/document/product/1081/34739>

<https://cloud.tencent.com/document/product/1081/34916>

Figure 13. Create a Product using Common Data Template

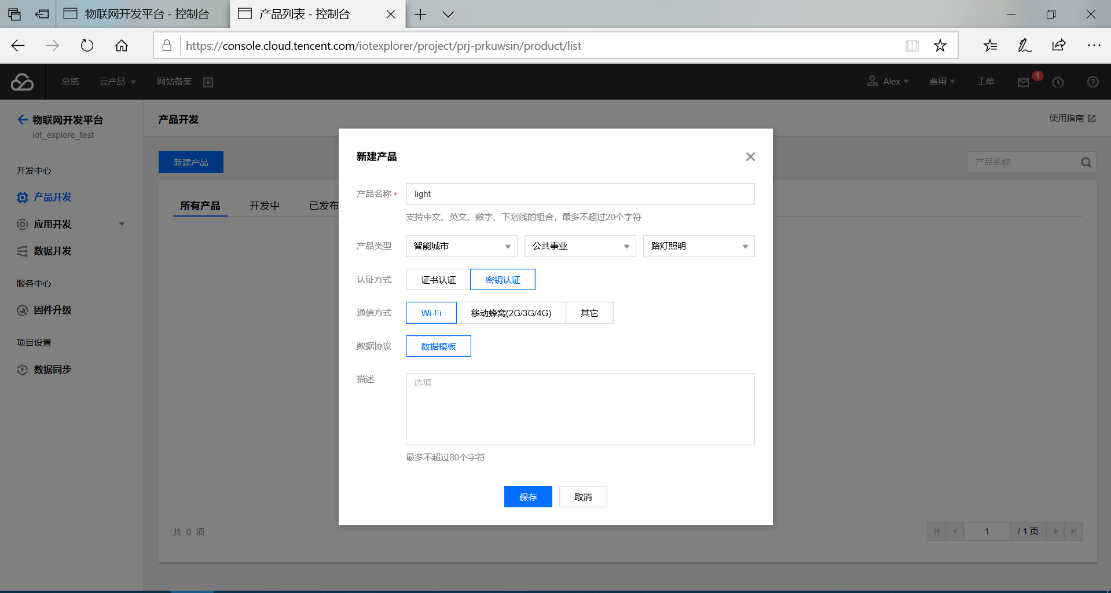
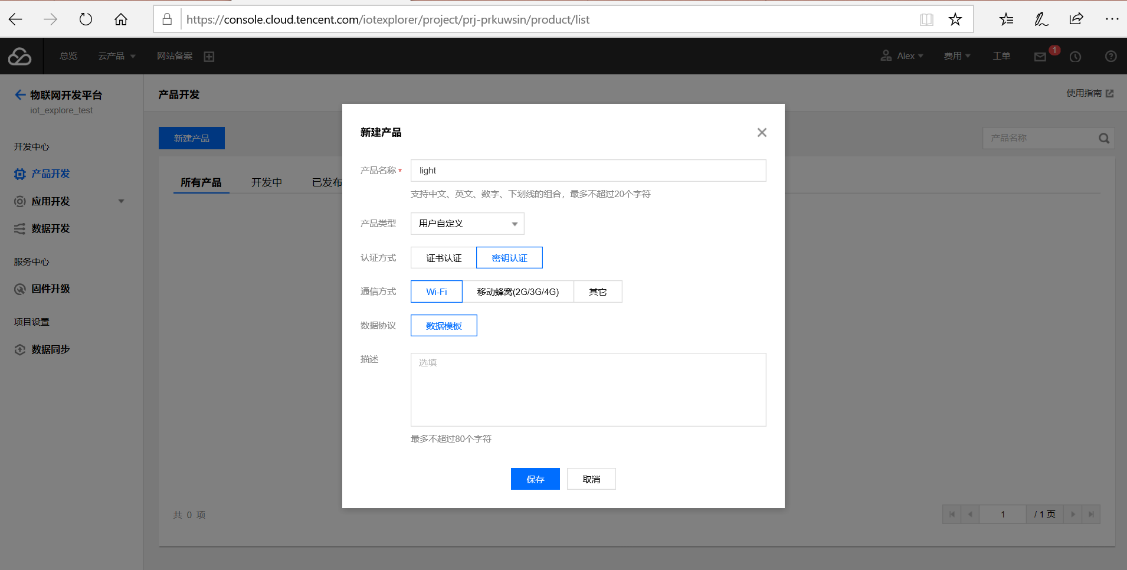


Figure 14. Create a Product using Customized Data Template



1. Modify data template according to actual demand

Figure 15. Create a Property



Figure 16. Create an Event

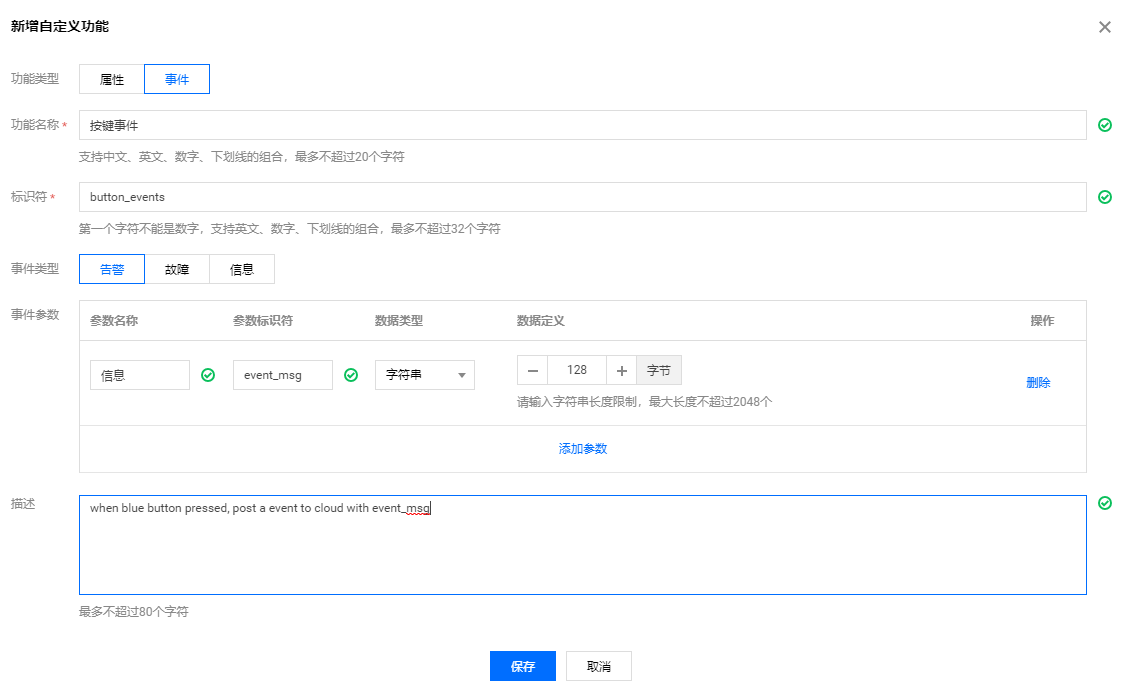
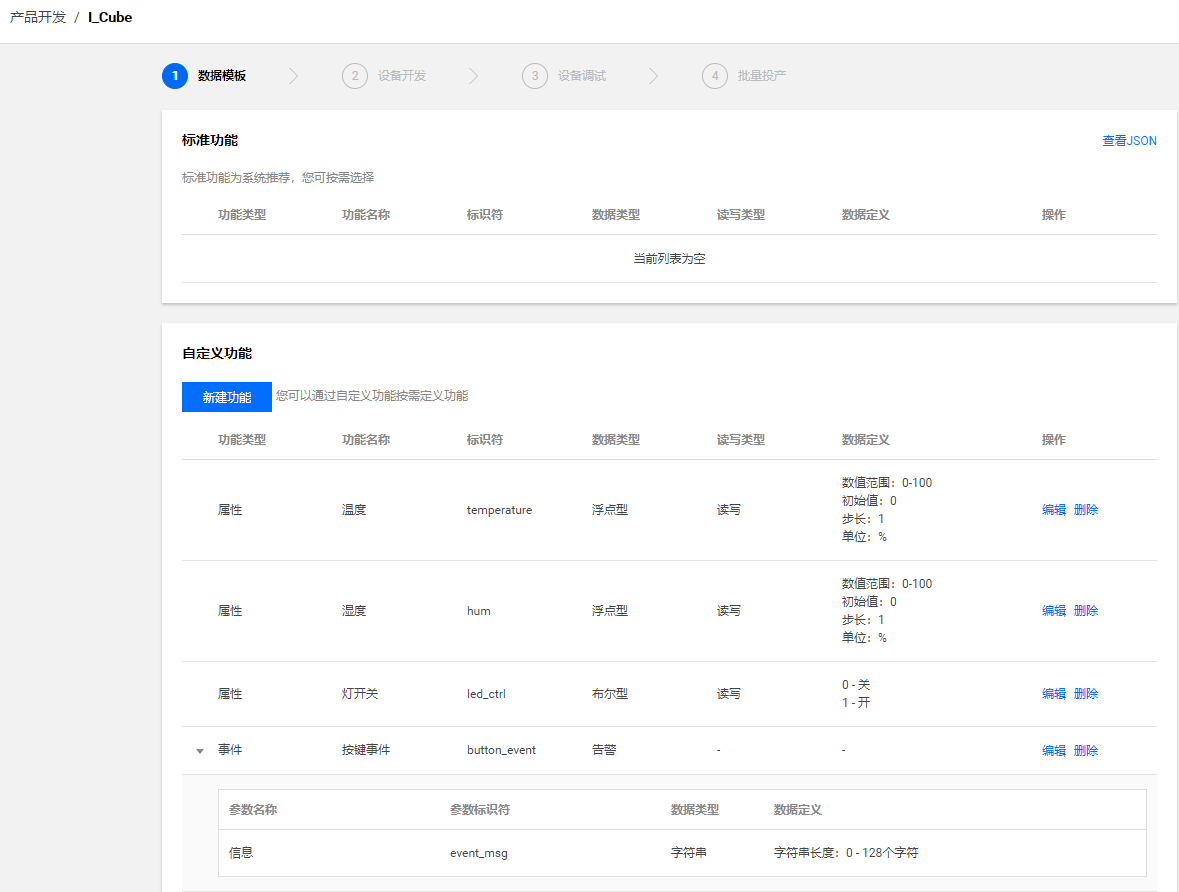


Figure 17. Data Template Information



1. Steps to generate data template configuration code

* Export the data template to JSON file
* Convert the JSON file to C code using codegen.py located in Middlewares\Third\_Party\qcloud-iot-explorer-sdk-embedded-c\tools.
* Copy the generated data\_config.c and events\_config.c into data\_template\_sample.c in the same directory
* The data\_template\_sample.c provides a general frame to show how to use data template, users can develop their own logic based on it.

Figure 18. Export Data Template to JSON file

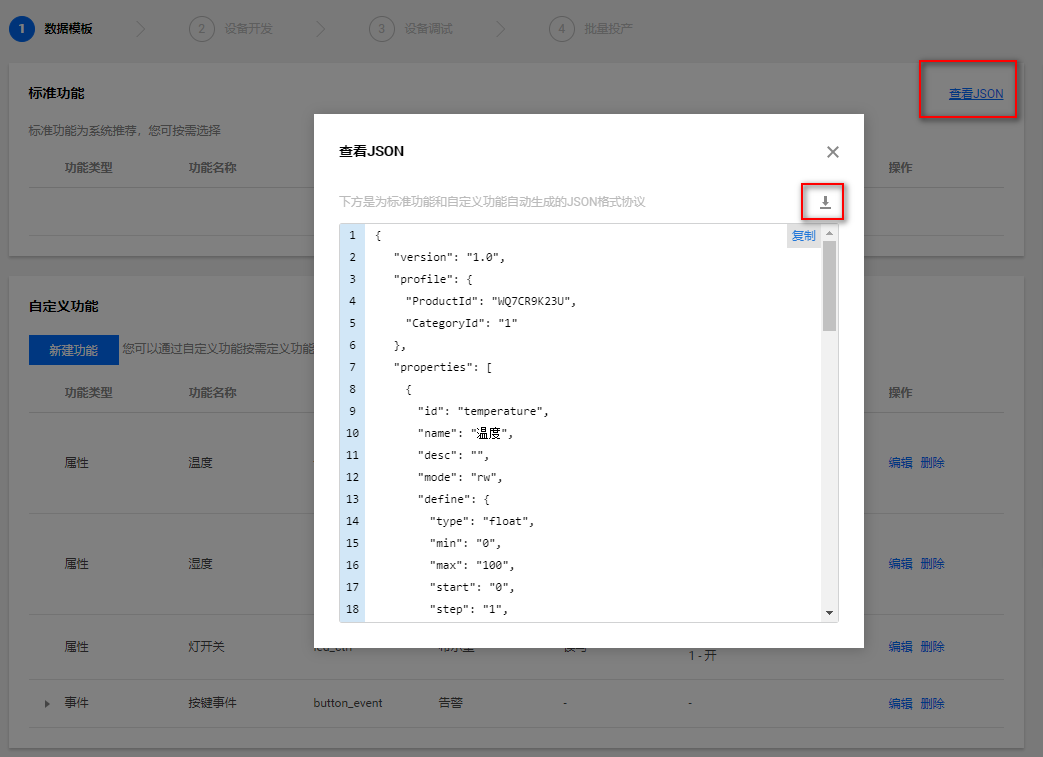
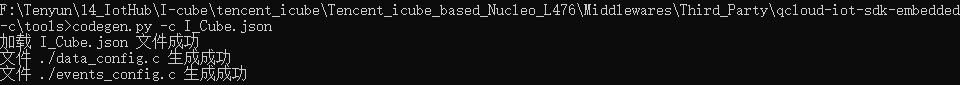


Figure 19. Generate Data Template C code



1. Create a new device on TENCENT® IoT explorer platform

Figure 20. Create a New Device

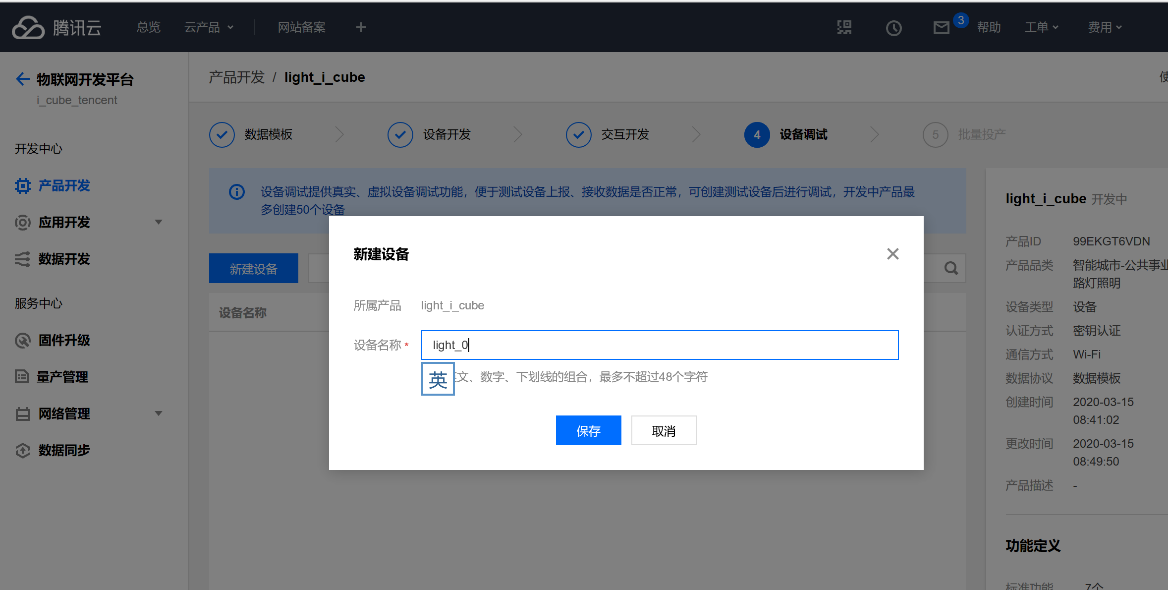


Figure 21．Mini Program control panel



Figure 22. Config shortcut for Mini Program

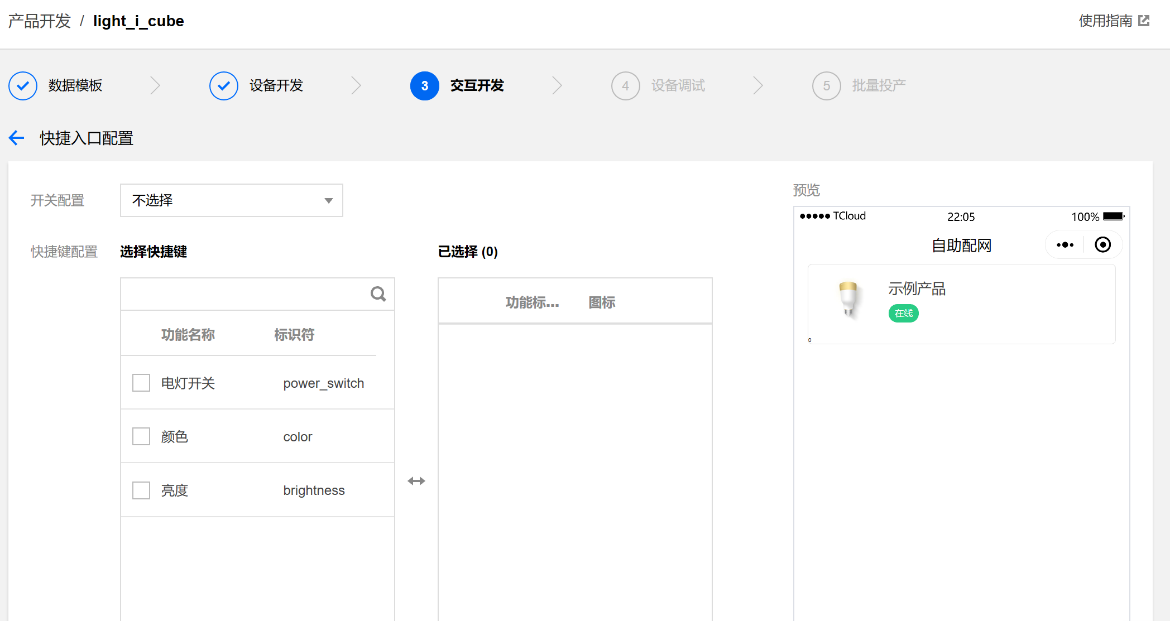
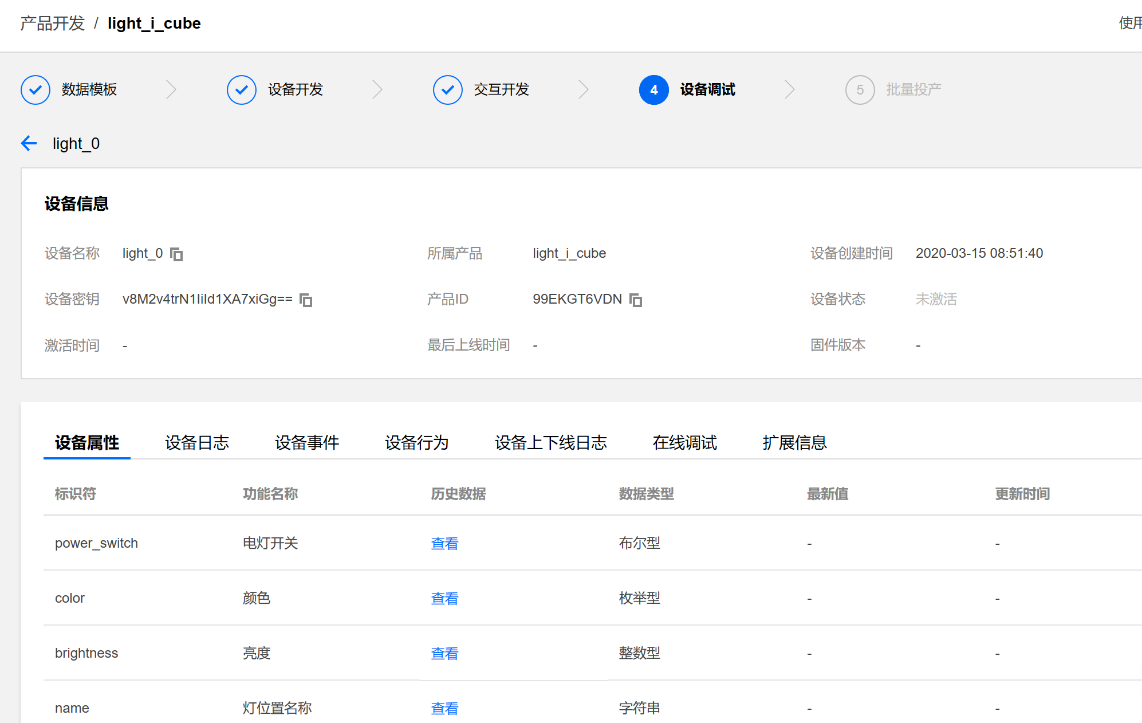


Figure 23. WIFI® configuration for device



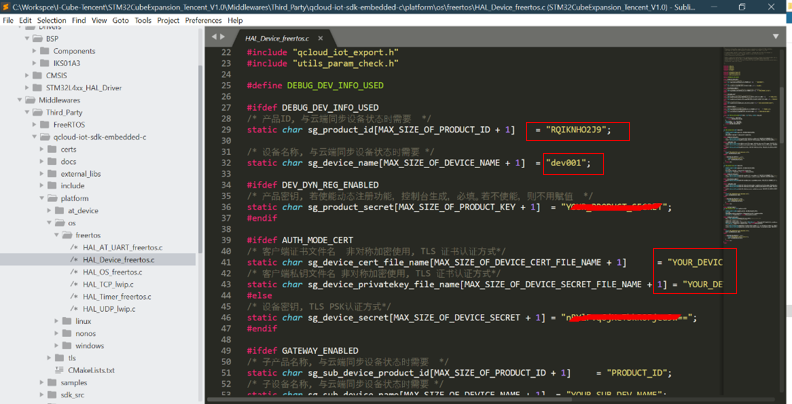
Figure 24. Device information



1. Update device information

Update the device information to the corresponding macro definition in \Middlewares\Third\_Party\qcloud-iot-explorer-sdk-embedded-c\platform\os\freertos\ HAL\_Device\_freertos.c.

Figure 25. Update device information

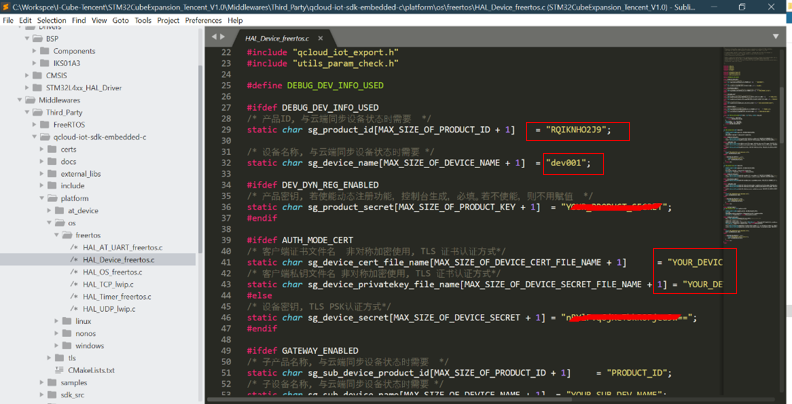


## Application build and first launch

**Caution:** Before opening the project with any tool chain, **make sure that the folder installation path is not too deep,** otherwise the tool chain may report errors after the build.

Enter the Productid, Device Name and Device Secret at the specified position in HAL\_Device\_freertos.c, which is located at Middlewares\Third\_Party\qcloud-iot-explorer-sdk-embedded-c\platform\os\freertos**.**

Figure 26. Device provisioning



Open and build the project with one of the supported development tool chains (see the release note for detailed information about the version requirements).

Program the firmware on the STM32 board: you can copy (or drag and drop) the generated bin file to the USB mass storage location created when you plug the STM32 board to your PC.

Alternatively, you can program the STM32 board directly through one of the supported development tool chains.

## Application First Launch

1. Hardware connection

Figure 27. Pins connection between NUCLEO board and WIFI® module

|  |  |  |
| --- | --- | --- |
| NUCLEO-L476RG |  | ESP-WROOM-02D |
| D2(PA10) | connects to | 13(GPIO13) |
| D8(PA9) | connects to | 15(GPIO15) |
| GND | connects to | GND |
| 3V3 | connects to | 3V3 |

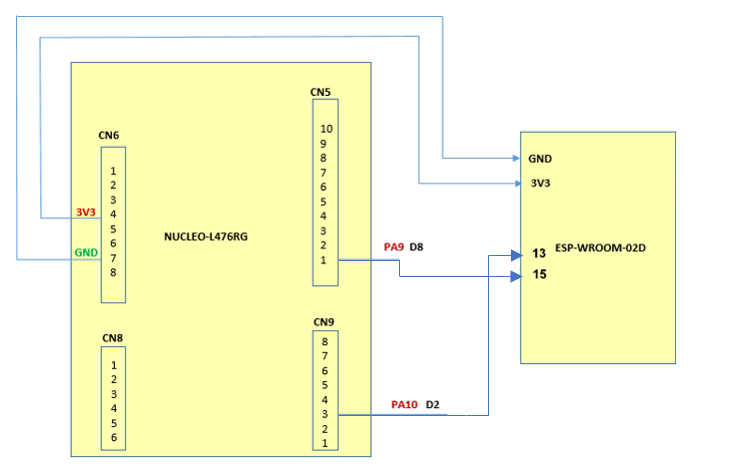
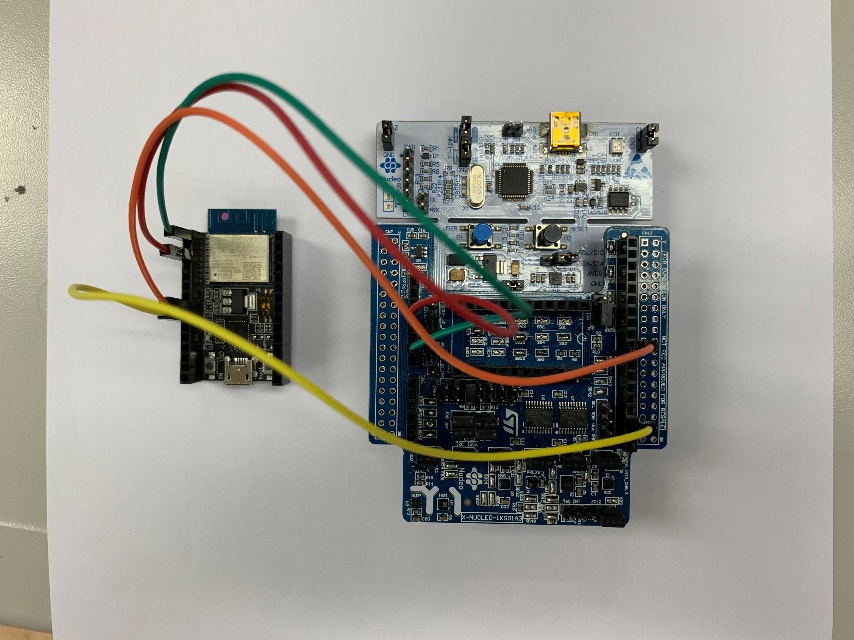


Figure 28. Hardware connection



1. The board must be connected to a PC through USB (ST-LINK USB port). Open the console through a serial terminal emulator (such as Tera Term), select the ST-LINK COM port of the board, and configure it with:

– 8N1, 115200 bauds, no HW flow control

– Line endings set to LF

For more details, see Chapter 5: Interacting with the boards.

1. After the system boot up, a shell menu is shown, users can use ‘wificonfig’ command to change the target WIFI® to which you want to connect. Default SSID and PASSWORD are defined in at\_device\_esp8266.c.

Figure 29. Default WIFI® configuration

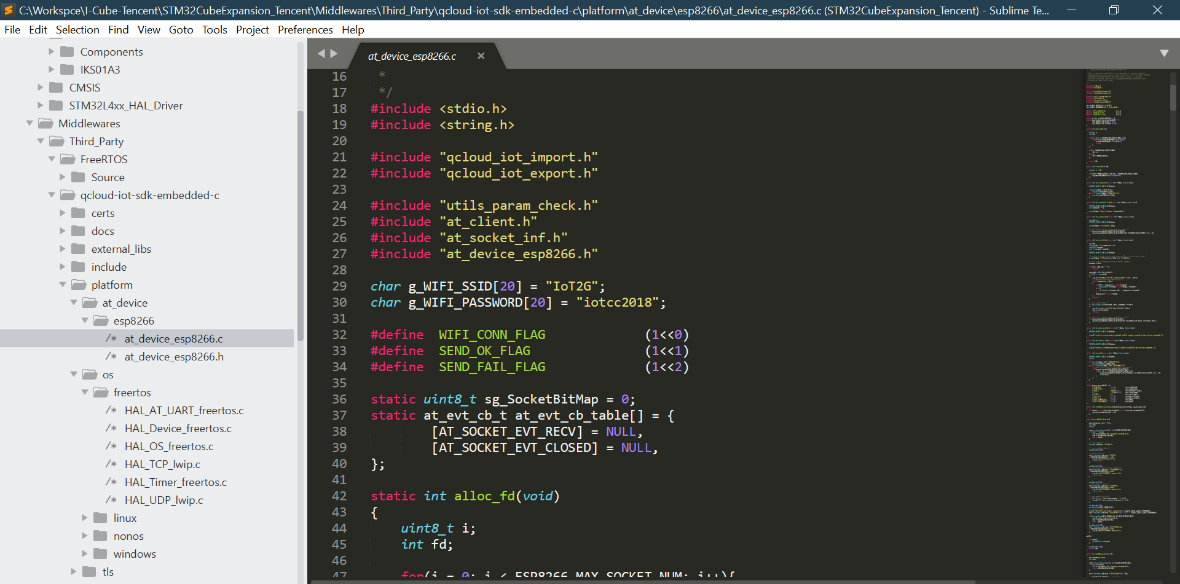
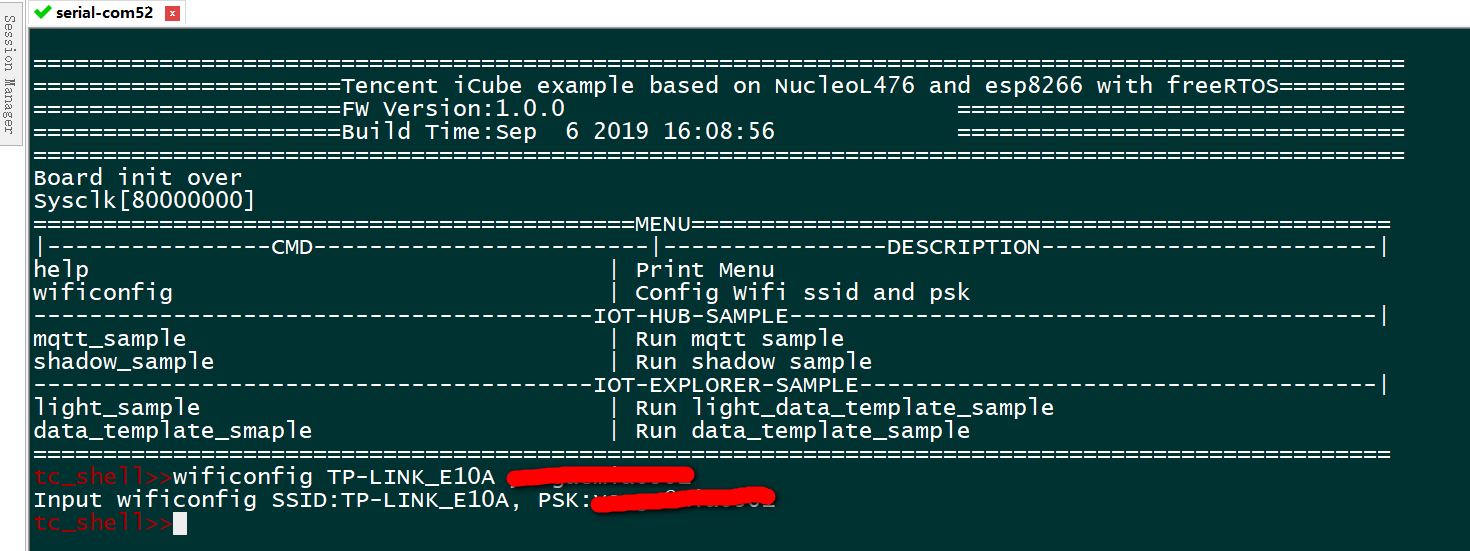


Figure 30. WiFi® provisioning



## Application Runtime

This section introduces how to run the icube\_data\_template\_sample application in the I-CUBE-TENCENT package. The icube\_data\_template\_sample application creates a MQTT connection between the NUCLEO board and TENCENT® IoT explorer platform using a customized data template protocol.

* Properties (e.g. temperature/humidity/led state) would be reported when the properties are changed.
* Events would be posted once the blue user button is pressed.

1. View running status on console

Figure 31. Run the icube\_data\_template\_sample application

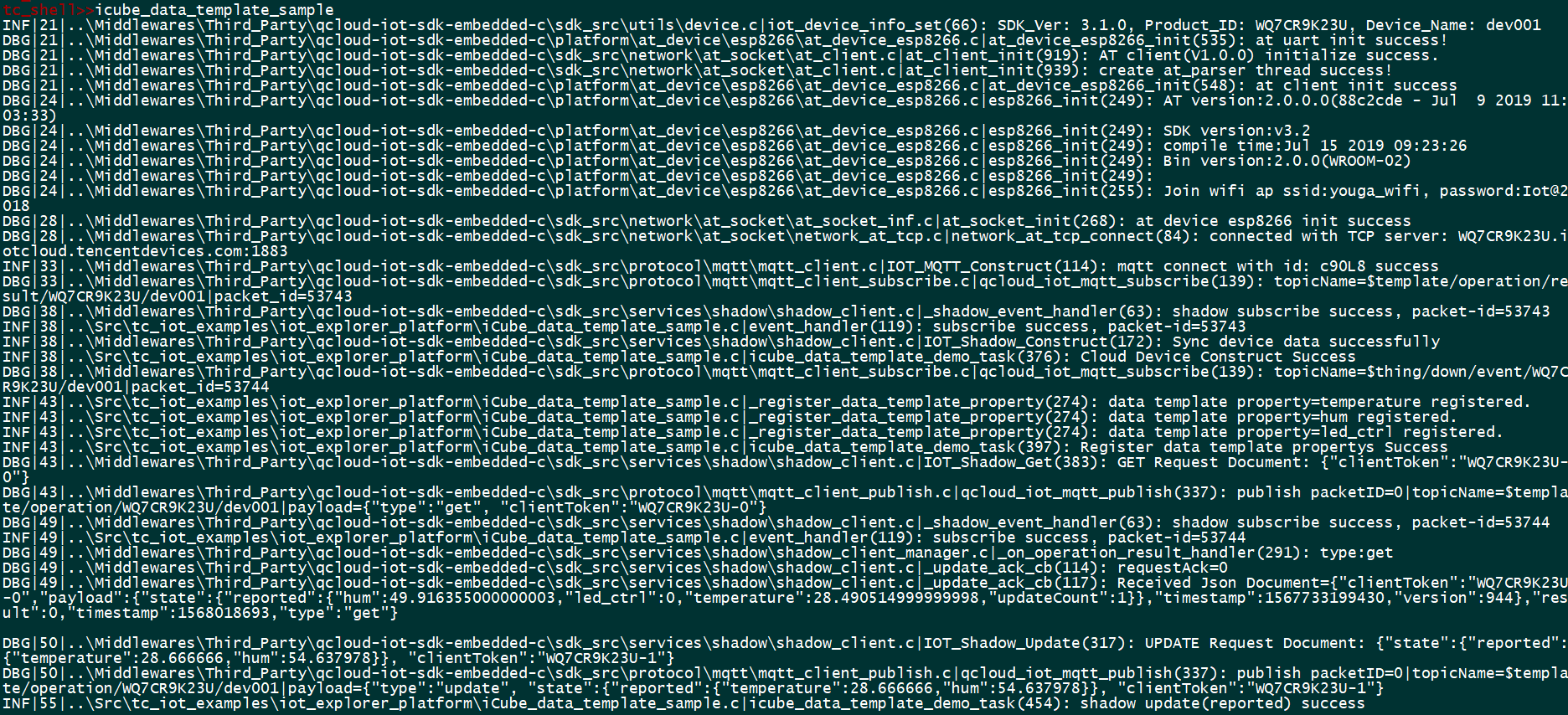


Figure 32. Report properties

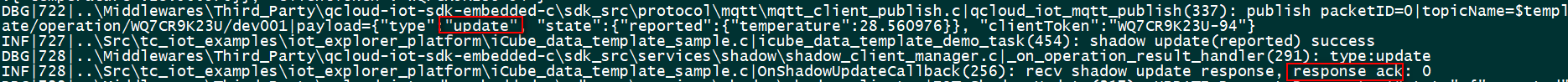


Figure 33. Post events when blue button is pressed

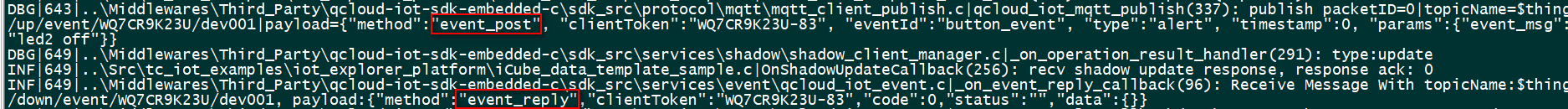


Figure 34. Send and receive data in the debug window at real-time



1. View running status on cloud

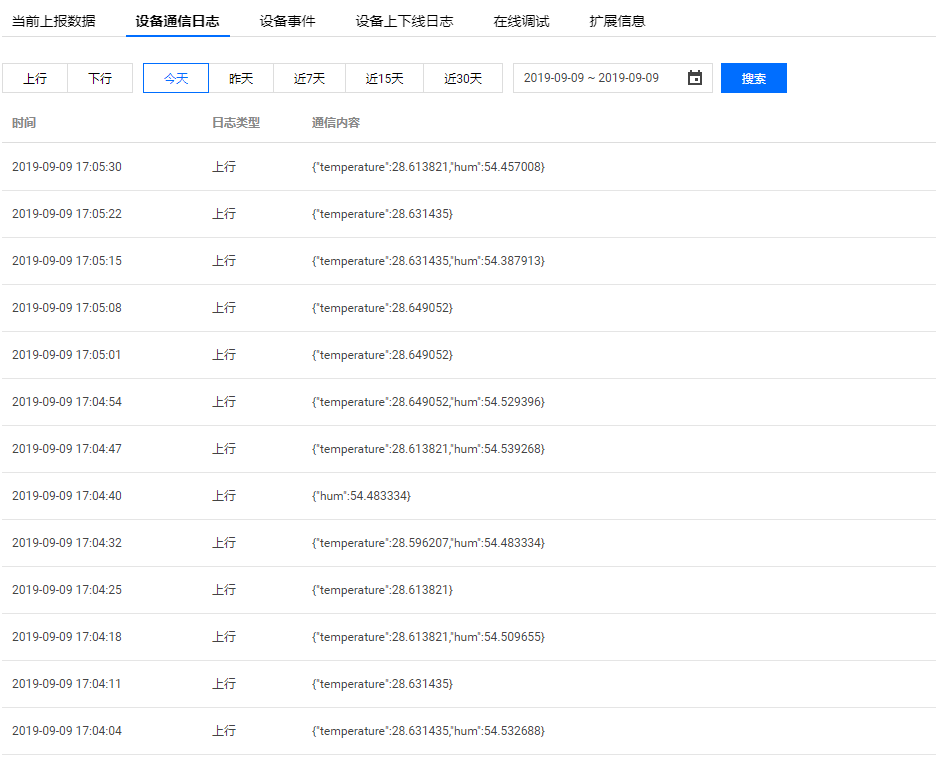
Figure 35. Events posted by device



Figure 36. The latest property status reported



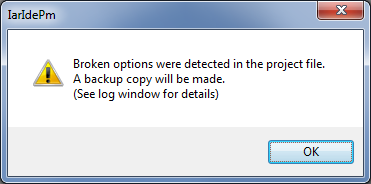
Figure 37. Logs on cloud



# Frequently asked questions

**Q:** Why do I get this pop-up (refer to *Figure 38*) when I open the project with IAR™?

Figure 38. Pop-up when the IAR™ IDE version is not compatible



**A:** It is very likely that the IAR™ IDE version is older than the one used to develop the package (refer to the release note available in the package root folder for the IDE versions supported), hence the compatibility is not ensured. In this case, the IAR™ IDE version needs to be updated.

**Q:** My device does not connect to the Wi-Fi® access point. How shall I proceed?

**A:** Make sure that another device can connect to the Wi-Fi® access point. If it can, input the ‘wificonfig’ command to enter the new Wi-Fi® credentials after board reset.

Note: you must run a sample application first, then the board will connect to the AP.

# Revision history

Table 2. Document revision history

|  |  |  |
| --- | --- | --- |
| **Date** | **Revision** | **Changes** |
| 20-Mar-2020 | 1 | Initial release. |