

Getting Started Guide for IoT Core: SDT Smart Hub

2021-10-11

Table of Contents

1	<i>Document Information</i>	1
2	<i>Overview</i>	1
3	<i>Hardware Description</i>	2
4	<i>Set up your Development Environment</i>	3
5	<i>Set up your hardware</i>	4
6	<i>Setup your AWS account and Permissions</i>	5
7	<i>Create Resources in AWS IoT</i>	5
8	<i>Provision the Device with credentials</i>	7
9	<i>Build the demo</i>	7
10	<i>Run the demo</i>	9
11	<i>Debugging</i>	10
12	<i>Troubleshooting</i>	12

1 Document Information

1.1 Naming Conventions

1.2 Glossary

1.3 Revision History (Version, Date, Description of change)

Version: 1.0.0, Date: 2021. 09. 01. Description of Change: Initial version

2 Overview

This Smart Hub device is a microcontroller that transmits the measured value obtained from the sensor to the AWS server or executes the command received from the AWS IoT Core server.

This manual guides how to use the provided Out-of-Box (OOB) package. Further details can be found on the company's website.

3 Hardware Description

3.1 DataSheet

Model Name	SDT Smart Hub
Model Number	SDT-SH-MANHOLE 401010-001
MCU	STM32L496ZGT (ST)
Connectivity	LTE Cat. M1 (Quectel BG96)
GPS	GNSS (Quectel BG96)
I/O Port	USB Type-C - 1EA Battery IN - Sensor - 3EA
Sensors	Temperature (Sensirion) Relative Humidity (Sensirion) Water Level Magnetic Door Contact CO (ELT)
Power	DC 5V/500mA
Battery	3.7V, 32Ah((Lithium-thionyl Chloride)
Size	Main: W 130 x L 180 x H 30 mm Sensor: W 100 x L 120 x H 35mm 2 x Battery Assembly: W 130 x L 180 x H 30 mm

Please refer to our product datasheet here:

https://sdt.inc/bbs/board.php?bo_table=resources&wr_id=24

3.2 Standard Kit Contents

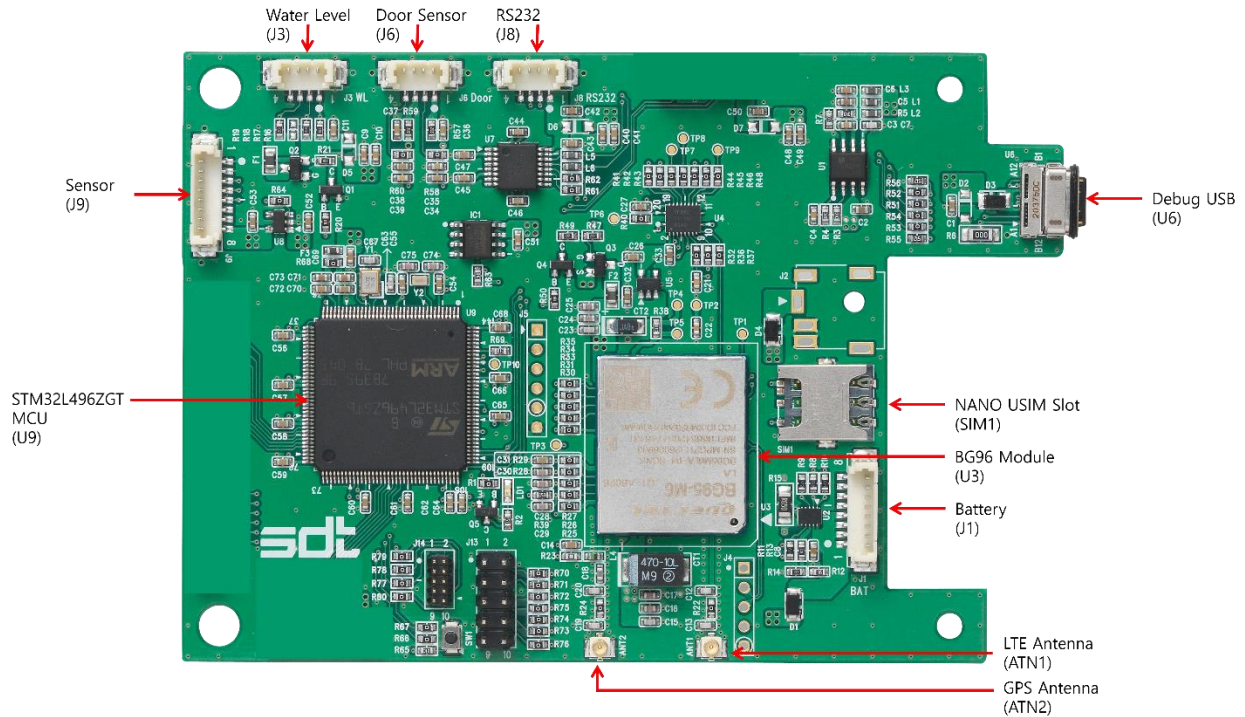
The product package contains the complete SDT Smart Hub and all components, including a main board and enclosure, the sensor board enclosures, 2 batteries packs and enclosures, the connecting cables, and the water sensor cables. Serial communication uses USB-C and may be accessed with standard USB-C products.

Please refer to our store page for more details:

https://sdt.inc/shop/item.php?it_id=1631585332

Additional battery packs can be viewed here:

https://sdt.inc/shop/item.php?it_id=1631610416



3.3 User Provided items

Full installation kit and cables are provided to user.

3.4 3rd Party purchasable items

SDT-Downloader kit: <https://sdt.inc/>

DAPLink downloader : <https://daplink.io/>

ST-Link/V2 JTAG programmer:

https://www.st.com/content/dam/kms/Contents/Reflibrary/STM32_Nucleo_board_Introduction.pdf

3.5 Additional Hardware References

ST, STM32L496ZG: <https://www.st.com/en/microcontrollers-microprocessors/stm32l496zg.html>

ELT SENSOR, gas sensor: <http://www.eltensor.co.kr/main>

Sensirion, temp/humi sensor: <https://www.sensirion.com/en/>

Quectel, BG96: <https://www.quectel.com/product/lpwa-bg96-cat-m1-nb1-egprs>

4 Set up your Development Environment

4.1 Tools Installation (IDEs, Toolchains, SDKs)

The SDT Smart Hub runs firmware based on MbedOS 6.9. The development environment for writing firmware is as follows:

1. IDE based.

- A. Please refer to the MbedOS website for information on how to install the integrated development environment for MbedOS 6.9.
<https://os.mbed.com/docs/mbed-os/v6.9/build-tools/install-and-set-up.html>
- B. Example source code for developers may be downloaded from the URL below.
https://github.com/SigmaDeltaTechnologiesInc/SDT_Manhole_Official
- C. Please enter the following in the Terminal Command window.
git submodules init
git submodules update

4.2 Other software required to develop and debug applications for the device

Teraterm: <https://tssh2.osdn.jp/index.html.en>

Putty: <https://www.putty.org/>

Mbed Studio: <https://os.mbed.com/docs/mbed-os/v6.15/quick-start/index.html>

VSCode: <https://code.visualstudio.com/>

4.3 Other pre-requisites

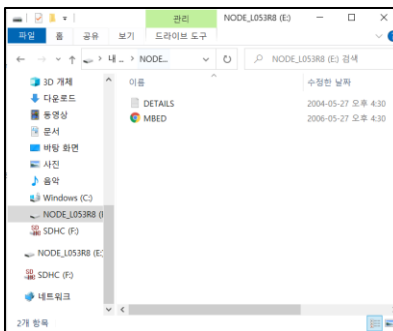
4.4 Additional Software References

Mbed forum: <https://forums.mbed.com/>

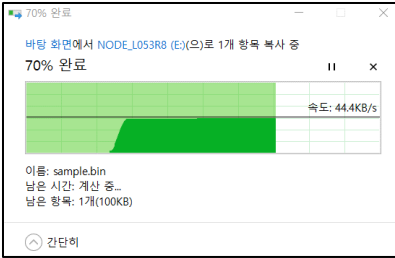
5 Set up your hardware

The SDT Smart Hub runs firmware based on the MbedOS operating system. The firmware has been installed on the device out-of-box. Simply power up the device and the device will automatically connect to our Smart Hub dashboard. However, *[you can download the installed firmware at SDT's GitHub](#)*. You may also follow the directions below to install other firmware on the device.

1. To make changes to the SDT Smart Hub firmware, you must first purchase a downloader at <https://sdt.inc>. You may also use a 3rd party downloader such as ST-Link/V2 JTAG.
2. Connect the downloader to the SDT Smart Hub.
3. To install the firmware on the SDT Smart Hub:
 - A. Copy the firmware file to the device.



Once you connect to the device via USB, the device will be identified as a removable disk.



Copy the compiled firmware file to the device folder.

4. Use serial communication to check the firmware operation status.

6 Setup your AWS account and Permissions

Refer to the instructions at [Set up your AWS Account](#). Follow the steps outlined in these sections to create your account and a user and get started:

- Sign up for an AWS account and
- Create a user and grant permissions.
- Open the AWS IoT console

Pay special attention to the Notes.

For instructions on how to create an AWS account, refer to the address below.

<https://docs.aws.amazon.com/iot/latest/developerguide/setting-up.html>

7 Create Resources in AWS IoT

Refer to the instructions at [Create AWS IoT Resources](#). Follow the steps outlined in these sections to provision resources for your device:

- Create an AWS IoT Policy
- Create a thing object

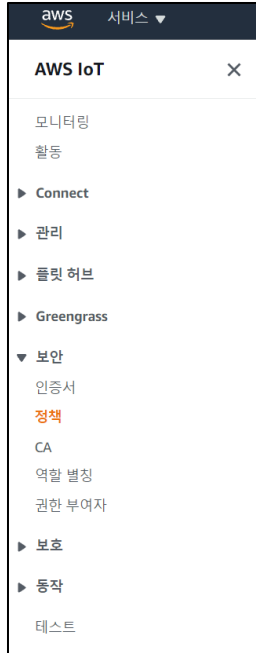
Pay special attention to the Notes.

Please refer to the following address for instructions on how to create a resource in AWS IoT. Additionally, you may also refer to the guide below.

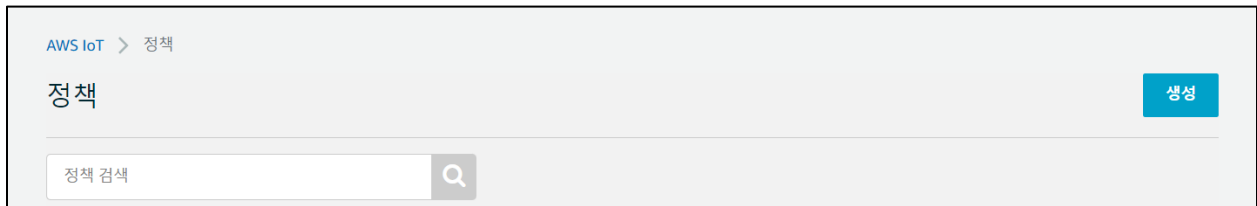
<https://docs.aws.amazon.com/iot/latest/developerguide/create-iot-resources.html>

How to create an AWS IoT Policy:

Navigate to the menu in AWS IoT



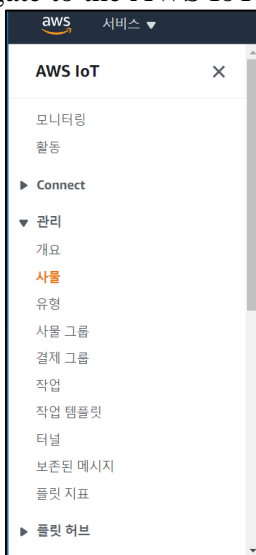
In the menu under “Security”, select “Policies”.



Click “Create” to create a Security Policy

How to create a thing object in AWS IoT:

Navigate to the AWS IoT menu.



In the menu under “Manage”, select “Things”.

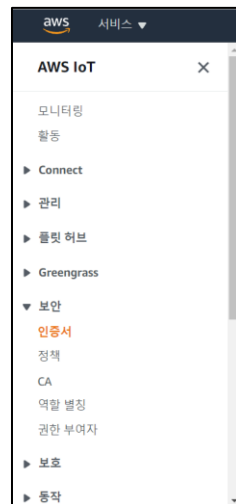


Click “Create” to create a Thing.

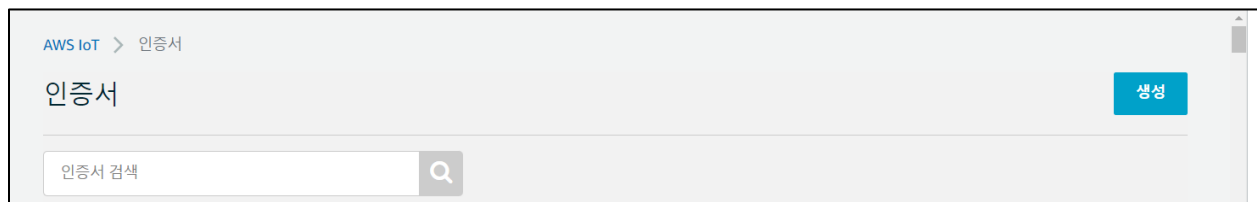
8 Provision the Device with credentials

How to generate a certificate in AWS IoT:

Navigate to the AWS IoT menu.



In the menu under “Security”, click “Certificates”.



Click the “Create” button to generate and download the certificate.

9 Build the demo

The firmware can be downloaded from SDT's GitHub. If you would like to modify the source code, you can download Mbed-os-example-for-aws from the link below and modify it.

SDT Smart Manhole: https://github.com/SigmaDeltaTechnologiesInc/SDT_Manhole_Official

mbed-os-example-for-aws: <https://github.com/ARMmbed/mbed-os-example-for-aws>

Please refer to the following instructions for modifying the SDT Smart Hub device information.

1. Modify the `AWS_Client_Identifier` device name in the file `Source/main/sdt_common.h`

```
13 #ifdef MBED_CONF_APP_AWS_CLIENT_IDENTIFIER
14     #undef MBED_CONF_APP_AWS_CLIENT_IDENTIFIER
15     #define MBED_CONF_APP_AWS_CLIENT_IDENTIFIER "SDT-MH-MAIN"
16 #endif
```

2. Set the address of the HTTP registration server that registers the device to AWS IoT Core. The address of the server is provided by SDT below:

```
48 //New HTTP_SERVER, HTTP_DESTINATION ADDRESSES, for device registration
49 // LTE Cannot use IPv4!, If using WiFi IPv4 is ok.
50 #define HTTP_SERVER "2406:da12:ea8:f000:1452:bb1c:5e5a:c763"
51 #define HTTP_DESTINATION "https://2406:da12:ea8:f000:1452:bb1c:5e5a:c763"
52 #define HTTP_PORT 8080
53 #define DATA "manholes"
```

3. Modify the `AWS_Endpoint` of AWS IoT in the `Mbed_config.h` file. The AWS IoT address is provided by SDT below:

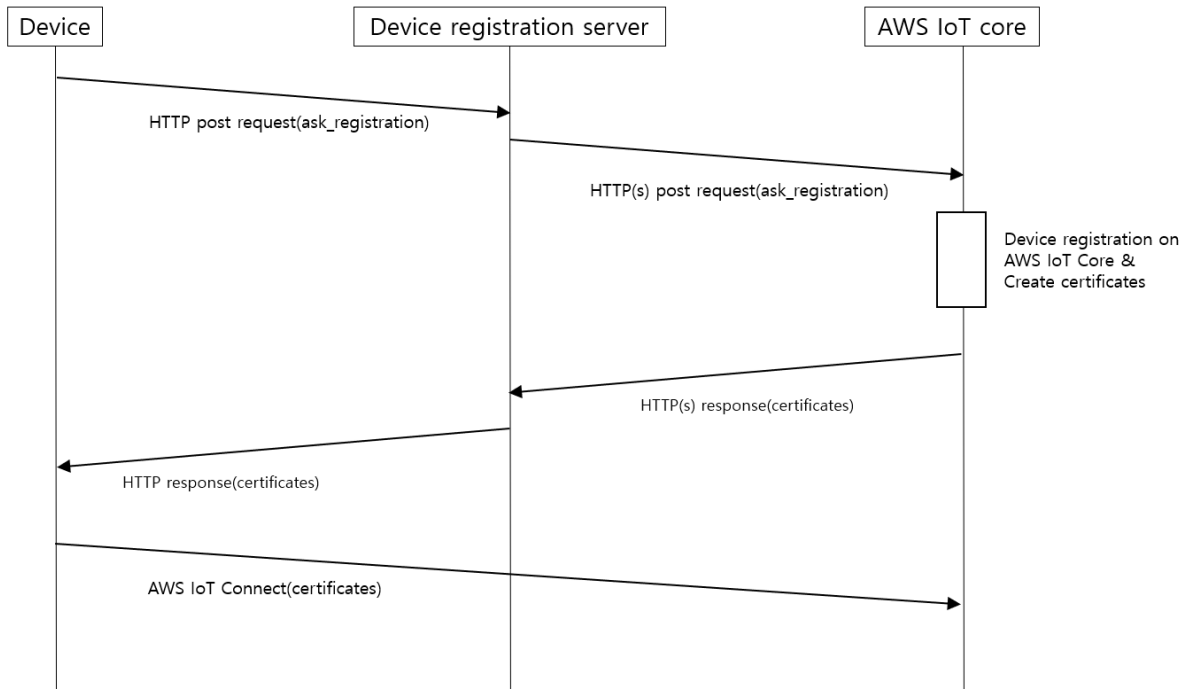
```
#define MBED_CONF_APP_AWS_ENDPOINT "avk03ee629rck-ats.iot.ap-northeast-2.amazonaws.com"
```

4. Modify the communication protocol and the pin number of the device to be used in `Mbed_app.json`. For SDT Smart Hub, the pin number is provided by SDT below:

```
"SDT_MANHOLE": {
  "target.network-default-interface-type" : "CELLULAR",
  "lwip.ipv4-enabled" : false,
  "lwip.ipv6-enabled" : true,
  "lwip.ethernet-enabled" : false,
  "lwip.ppp-enabled" : false,
  "lwip.tcp-enabled" : true,
  "nsapi.default-cellular-plmn" : 0,
  "nsapi.default-cellular-sim-pin" : "\"0000\"",
  "nsapi.default-cellular-apn" : "\"\"",
  "nsapi.default-cellular-username" : 0,
  "nsapi.default-cellular-password" : 0,
  "cellular.debug-at" : false,
  "cellular.use-apn-lookup" : true,
  "QUECTEL_BG96.provide-default" : true,
  "spif_driver_wp" : "PB_6",
  "spif-driver.SPI_CLK" : "PB_3",
  "spif-driver.SPI_MISO" : "PB_4",
  "spif-driver.SPI_MOSI" : "PB_5",
  "spif-driver.SPI_CS" : "PB_9",
  "spif-driver.debug" : true
}
```

SDT Smart Hub Complete Pin Numbers

Description of demo firmware operation



Demo firmware action sequence diagram

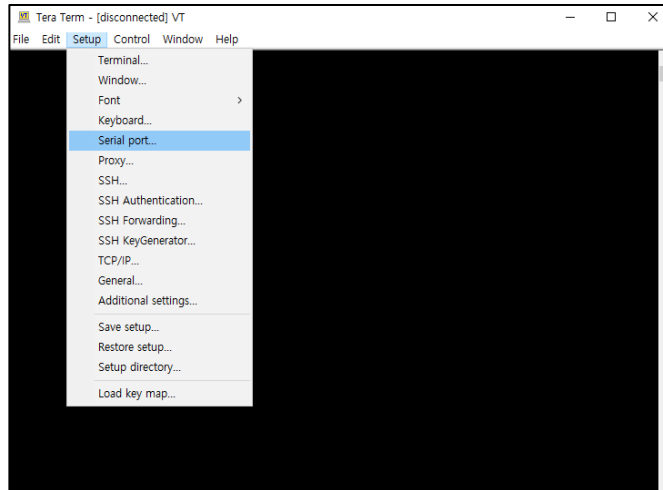
1. The SDT Smart Hub operates in the following order after bootup:
 - A. A device registration request is made to SDT's registration server.
 - B. The registration server makes a device registration request to AWS IoT core.
 - C. AWS IoT core creates Thing and Certificates for the device.
 - D. AWS IoT core returns the thing creation results and certificates to the registration server.
 - E. The device uses the certificate to perform AWS IoT shadowing.

When implementing firmware different from the firmware provided by SDT, refer to the [mbed-os-example-for-aws github](#).

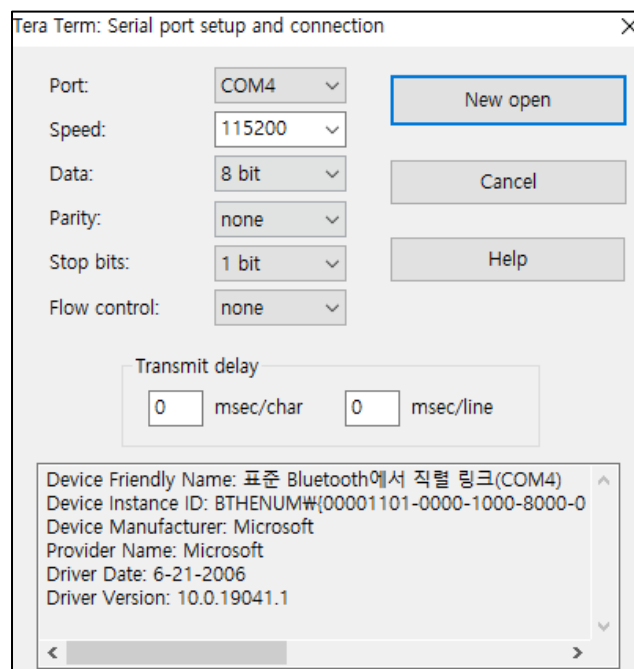
11 Debugging

To check for errors and receive the output messages from the device via a terminal program:

1. Install a terminal program such as Teraterm or PuTTY.
2. Connect the device to the computer using a USB-A to USB-C cable.
 - A. Check the serial output of the device using the Teraterm terminal

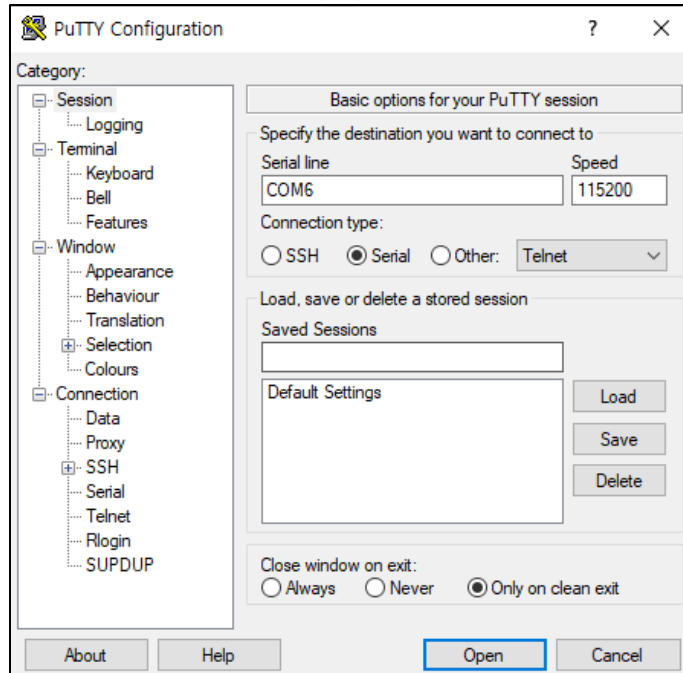


In the Teraterm menu, select Setup→Serial port.



Select the Port to which the device is connected, set the Speed to “115200” and click “New open.”

B. Check the serial output of the device using PuTTY



Enter the port to which the device is connected under “Serial Line” and set the “Speed” to “115200”.

3. Restart the device.

12 Troubleshooting

Tip 1: After installing the compiled firmware file on the device, restart the terminal.

Tip 2: Check that the opened SIM card is inserted correctly.

Tip 3: Check if the antenna is connected correctly.

Tip 4: If connection is still not possible, check the AWS IoT Core's certificate verification and security policy.