Getting started with AIROC™ IFW56810 Single-band Wi-Fi 4 Cloud Connectivity Manager

About this document

Scope and purpose

This document guides the user to set up and connect the AWS IoT ExpressLink IFW56810 Cloud Connectivity Manager module to the AWS IoT Core in a few simple steps without involving any firmware development. This document also provides an overview of the module, the evaluation kit, steps to connect to the AWS Cloud and usage of AT commands.

Intended audience

Customers who are new to IoT and/or require a turnkey solution to get their products connected to the cloud.

Table of contents

Application Note
www.infineon.com
Please read the Important Notice and Warnings at the end of this document
page 1 of 22
002-34322 Rev. *A
2021-12-14

Table of contents

1   Overview

AIROC™ IFW56810 Single-band Wi-Fi 4 Cloud Connectivity Manager Module (CCM) is a configurable Wi-Fi connectivity module that provides a simple, secure, plug-and-play solution for connecting products to AWS IoT cloud services. The IFW56810 CCM module is preprogrammed with a tested secured firmware and supports an easy-to-use AT command interface for configuration, and provides end-to-end security.

- The device identity certificate is built into the module and can run only Infineon-signed firmware.
- The module connects to the cloud using secure connections.
- Devices are managed securely in the cloud.

The IFW56810 CCM module features a 1x1 single-band (2.4 GHz) device operating at 20-MHz channels supporting IEEE 802.11 b/g/n. See the **datasheet** for details.

The host processor system interacts with the IFW56810 CCM module through AT commands over UART. The IFW56810 CCM module handles all networking-related operations to connect to the AWS IoT Core through MQTT over Wi-Fi.
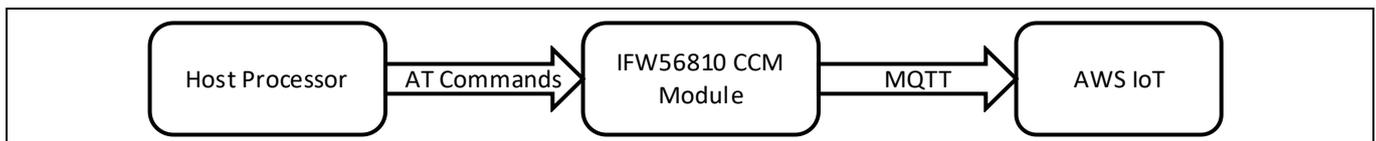
Figure 1   System architecture for products using the IFW56810 CCM module

2 Kit contents

- IFW956810 Single-band Wi-Fi 4 Cloud Connectivity Manager evaluation kit to evaluate the IFW56810 Single-band Wi-Fi 4 Cloud Connectivity Manager Module
- USB Type-A (male) to Type-C (female) cable

3 Hardware

The IFW956810 CCM evaluation kit consists of an IFW56810 single-band Wi-Fi 4 Cloud Connectivity Manager module with powerful processing capability, a PCB antenna, an FTDI chip for the USB to serial interface, and an 8x2 pin header.
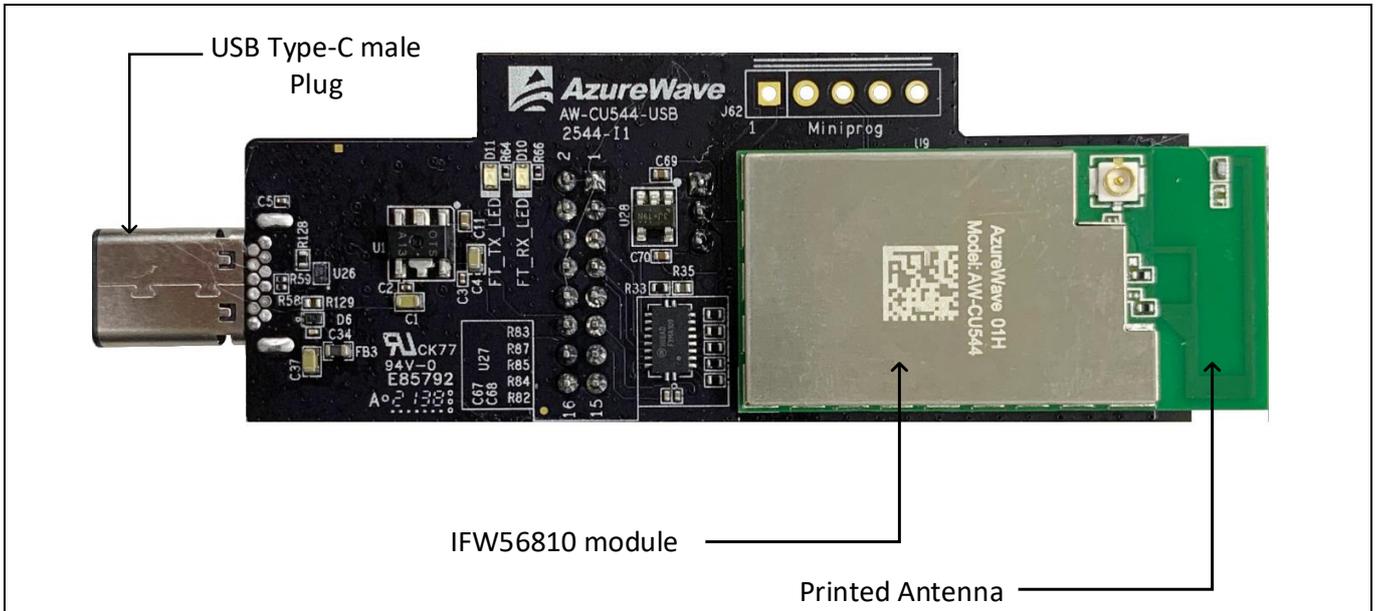
Figure 2   Top view
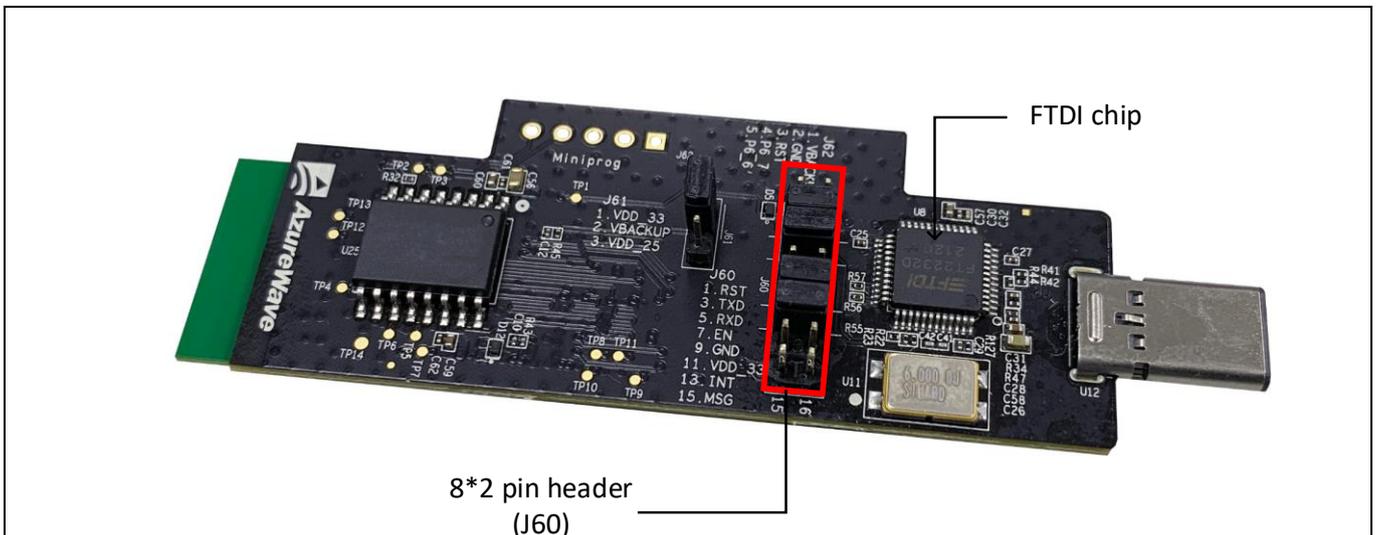
Figure 3   Bottom view

Ensure that pin 3 to pin 4, pin 5 to pin 6, pin 9 to pin 10 and pin 11 to pin 12 of header J60 are closed as shown in **Figure 3** before connecting USB dongle to the PC. The PC can be used as a host for evaluation. AT commands can be sent through a serial terminal on the PC to the IFW956810 CCM kit.

4 Getting started

4.1 Connect the kit to the PC

Connect the IFW956810 Single-band Wi-Fi 4 Cloud Connectivity Manager evaluation kit to the PC using either the Type-C connector or Type-A male to Type-C female cable.

Figure 4   **Connect the USB dongle to the PC**

4.2 Set up the host machine

Run a serial terminal to communicate with the kit over USB by choosing higher of the two enumerated COM ports with the following configuration.

- Baud rate: 115,200
- Bits: 8
- Parity: None
- Stop: 1
- Flow control: None
- Local Echo: Yes
- End of Line: Line Feed

See Section **7** for more details.

Once you open the serial terminal, type `AT+CONF? About` in the serial terminal.
You should see a response "OK Infineon IFW56810 CCM".

Steps in sections **4.3** to **4.5** are required only for the first time you connect. The IFW56810 module will remember its configuration and will be ready to connect to your AWS account automatically.

Jump to Section **4.6** if the kit has already been configured to connect to AWS.

4.3 Set up your AWS account and permissions

Follow the instructions in Amazon documentation (**Set up your AWS Account**) to create your account and get started.

1. Sign up for an AWS account.
2. Create a user and grant permissions for access to AWS IoT resources.
3. Open the AWS IoT Console.

4.4 Register the kit to your AWS development account

You can register your kit to your AWS development account using either Infineon's CIRRENT cloud flow or AWS flow.

The CIRRENT cloud flow uses the Cloud ID solution to simplify the process of connecting the kit to your AWS developer account by eliminating several manual steps that need to be performed in the AWS flow.

Learn more about the Cloud ID solution **here**.

At a high level, CIRRENT cloud flow performs the following functions to connect the kit to your developer account:

1. Binds your kit to your CIRRENT account.
2. Provisions the AWS resources for the kit on your AWS account.
   The resources include the Thing (cloud representation of your physical device),
   device certificate (associates the device certificate from the CIRRENT cloud to your Thing),

   Policies (creates and associates AWS IoT Thing access policies to your Thing)
3. Enables the kit to connect to your AWS account.

   The endpoint of your AWS development account is required for the kit to connect to your AWS account.

   The CIRRENT cloud pulls the AWS endpoint required from your AWS account and automatically pushes it to the kit. This enables the kit to connect to your AWS development account.

CIRRENT cloud flow is very close to the production flow and demonstrates the ease of securely connecting your products to the product cloud. CIRRENT cloud is preloaded with the device certificate of all IFW56810 modules, thus offloading the effort of managing device certificates from product manufacturers.

See Section **4.4.1** for CIRRENT cloud flow or jump to Section **4.4.2** for AWS flow.

4.4.1 CIRRENT cloud flow

4.4.1.1 Bind your device to your CIRRENT account

Your kit is already bound to your CIRRENT account when you registered your kit through the Quick Start Guide provided along with the kit.

If you landed on this page without following the Quick Start Guide, see the document provided along with your kit.

Do the following to verify the kit binding:

1. Log in to your CIRRENT account at **https://console.cirrent.com/**.
2. On the panel on the left of the screen, select **Device Management** > **Cloud ID**, and then select the **Binding** tab.
   Alternatively, click this **link** to go directly to this page.
3. Choose your account number as follows:
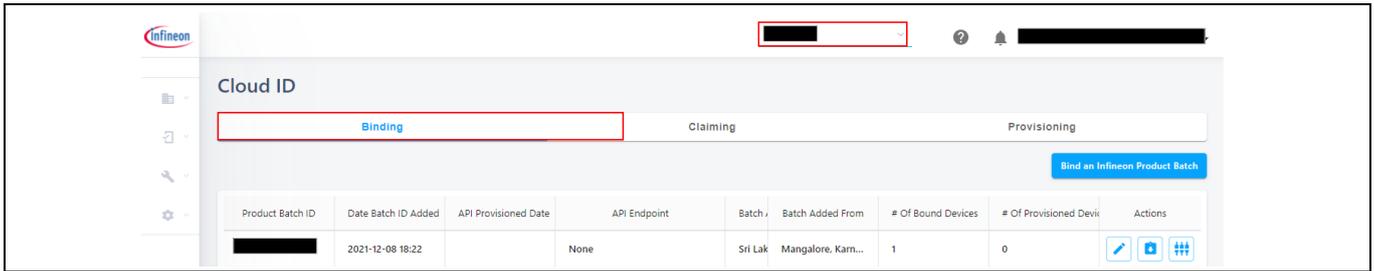
4. Locate your device listed on the page.

4.4.1.2 Execute the CloudFormation template

CloudFormation is an AWS service that helps in setting up the required resources in AWS through a template (JSON or YAML file). Executing a CloudFormation template creates a stack in the AWS CloudFormation service. A stack is a collection of AWS resources.

The template for creating AWS resources required for connecting the IFW956810 evaluation kit to AWS IoT Core is already created by Infineon and stored in Amazon S3 storage. The stack created by this template provides certain outputs that can be used to establish a channel of back-end cloud communication between your CIRRENT account and your AWS account.

You need to execute the CloudFormation template only once per AWS account in a region. The same stack can be reused to provision multiple kits to the AWS account in that region.

Do the following to execute the Infineon-provided CloudFormation template:

1. Click on the following link to execute the CloudFormation template. By default, the link uses the **us-west-1** region.

   https://us-west-1.console.aws.amazon.com/cloudformation/home?region=us-west-1#/stacks/create/template?stackName=infineon-iot-quickstart&templateURL=https://cirrent-quickstarts.s3.us-west-2.amazonaws.com/infineon-iot-quickstart.yaml

   You can change the region in which you want to execute this template by changing the `region=us-west-1` in this link to your required region. See **Choosing a Region** on AWS documentation.

2. On the **Create Stack** page in the AWS CloudFormation service with the Infineon CloudFormation template preloaded, click **Next**.
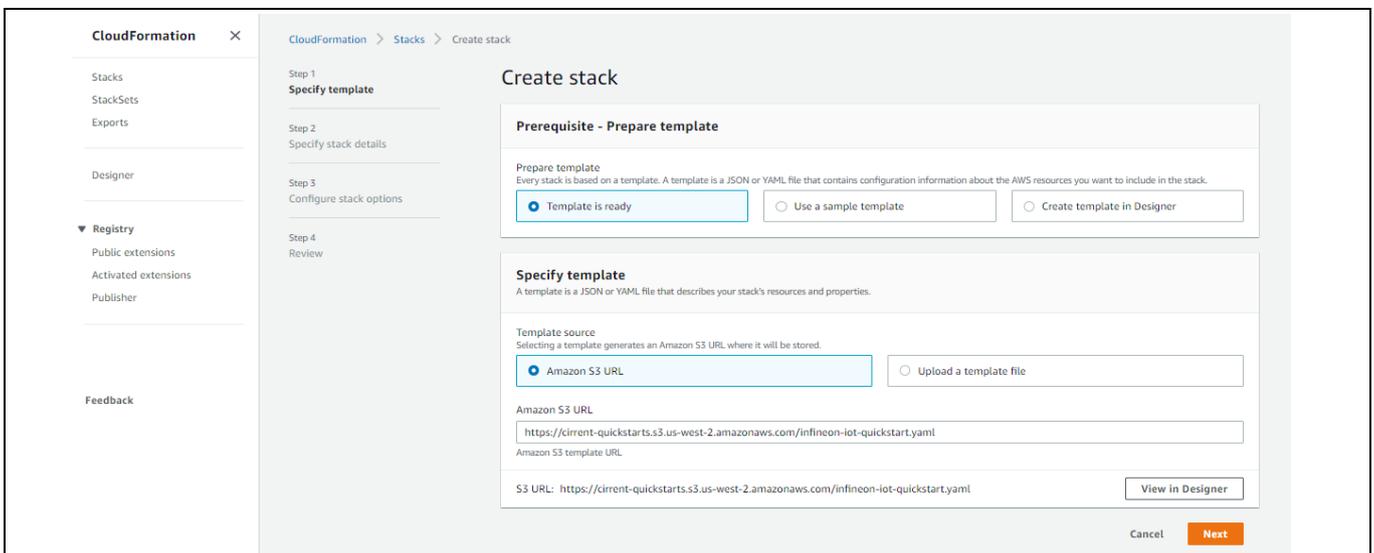
3. Click **Next** in Step 2.

4. Click **Next** in Step 3.

5. Select the boxes to provide permissions to access the resources required by the CloudFormation template.
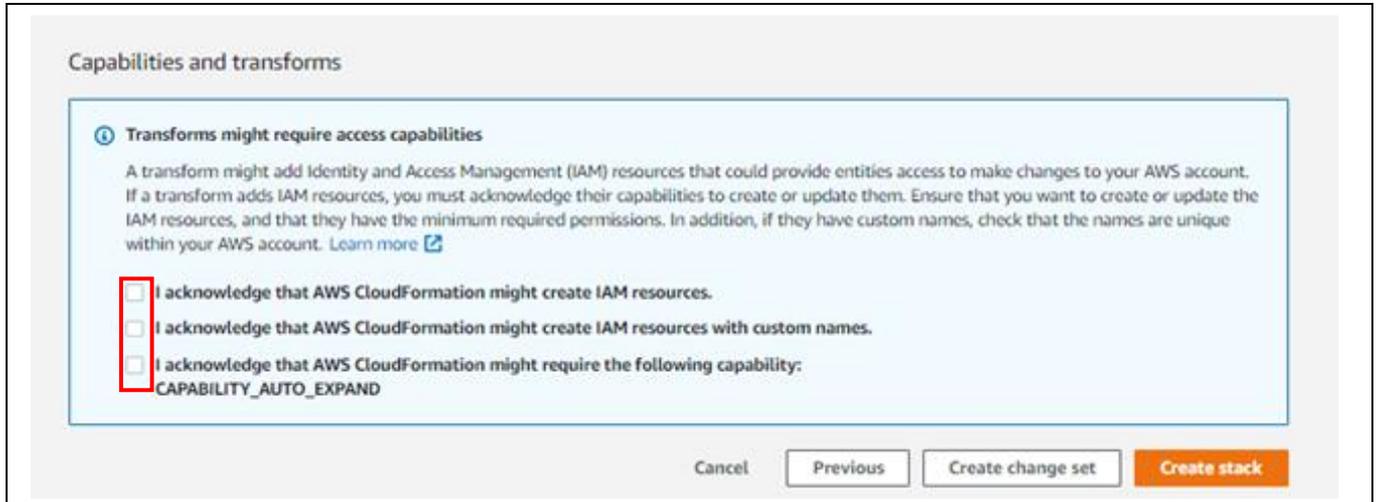
6. Click **Create stack**.

Wait for up to five minutes for the stack creation to complete.
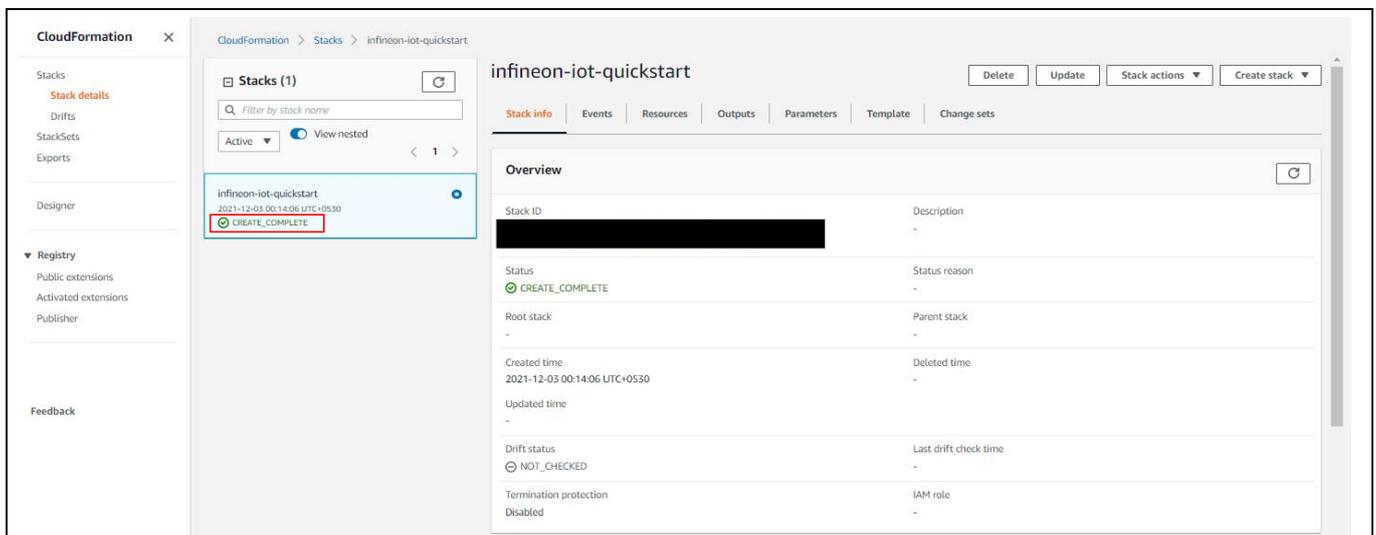
4.4.1.3   Create Product Cloud API

Product Cloud API is used to communicate with your AWS account. This API must be executed once per CIRRENT account. The same API can be used to provision multiple kits from your CIRRENT account to your AWS account.

1. Navigate to your CIRRENT console and go to **Provisioning** tab or use this **link** directly.

2. Click **Add Cloud API**.
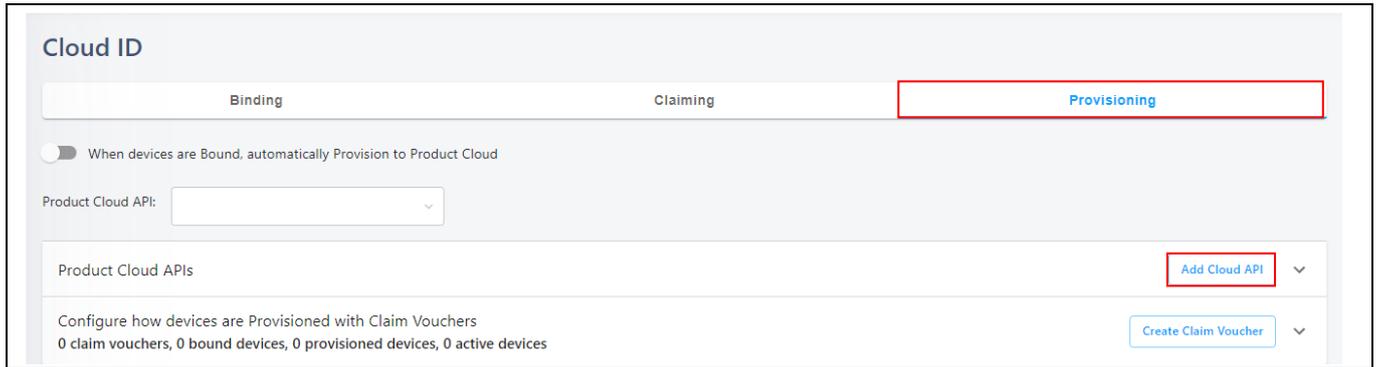
3. When prompted, enter your AWS account parameters.
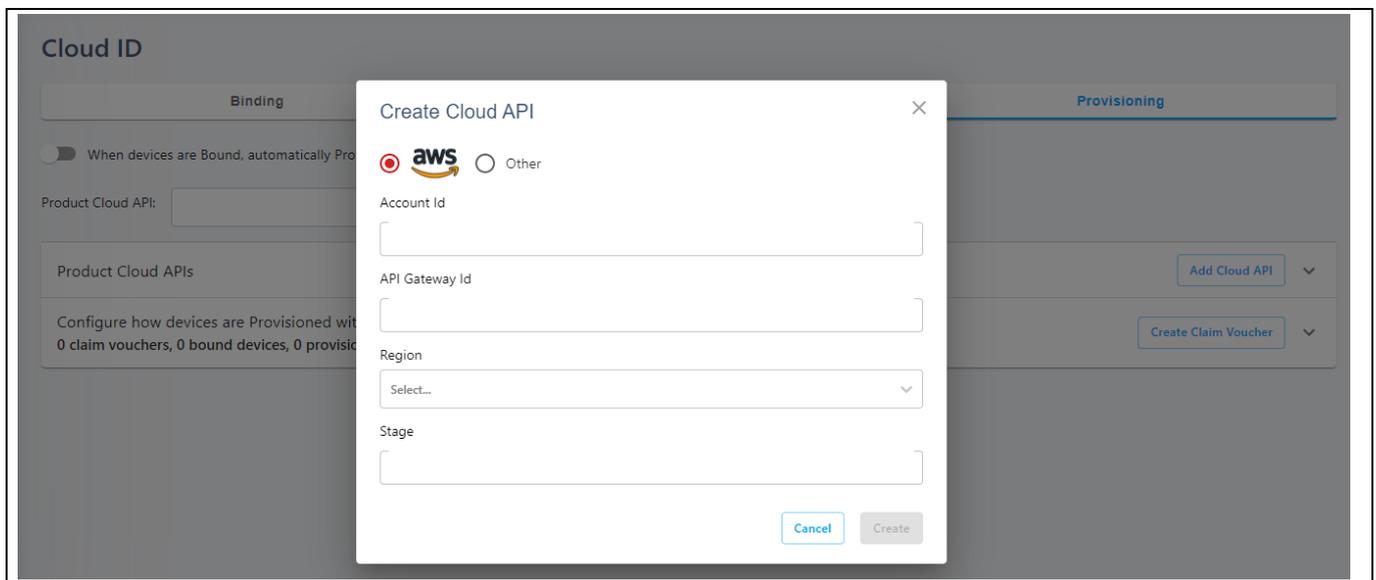
4. Navigate back to your **AWS CloudFormation** page.
5. Click **Outputs**.
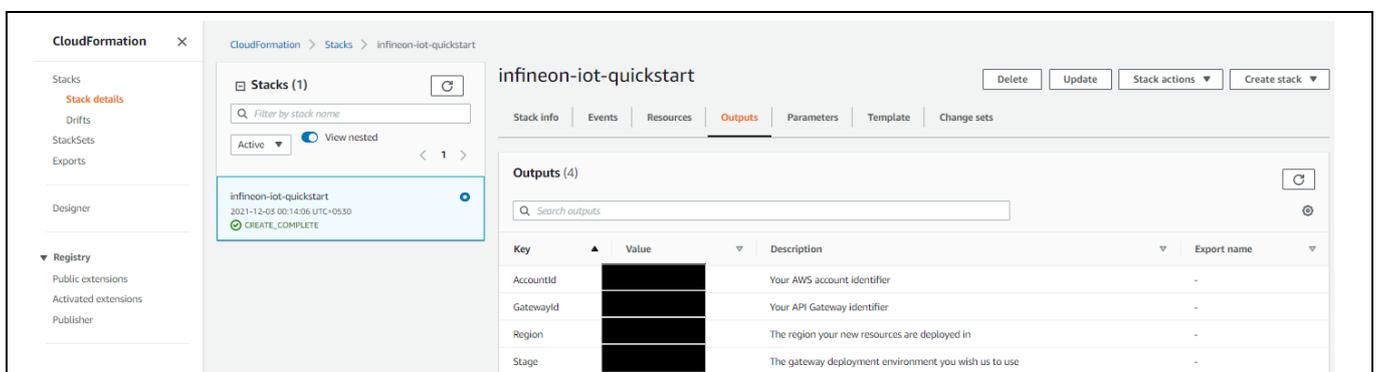
6. Copy the `AccountId`, `GatewayId`, `Region`, and `Stage` values from your AWS Console to the CIRRENT console.
7. In the CIRRENT console's Create Cloud API window, click **Create**.
8. Select the created Product Cloud API from the dropdown list next to **Product Cloud API**.
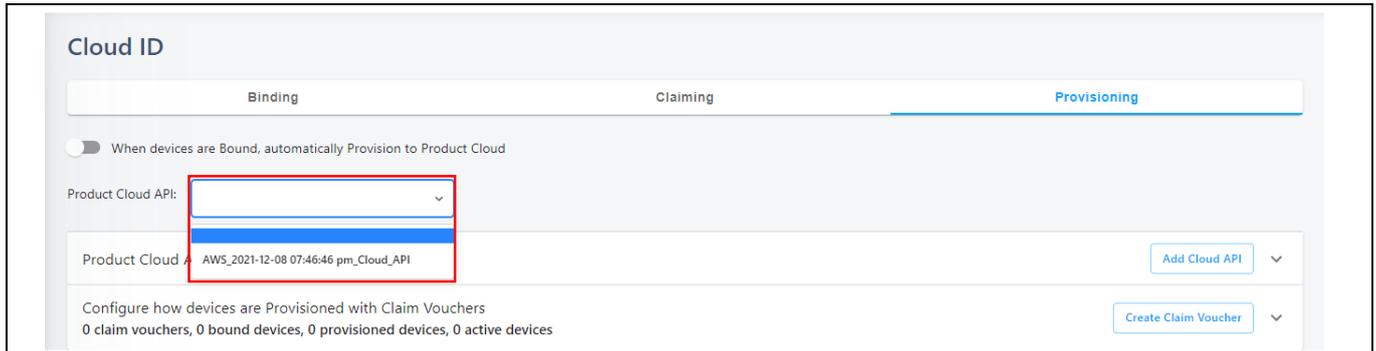
4.4.1.4 Provision and connect the kit to your AWS account

1. In the CIRRENT console, navigate to the **Binding** tab.
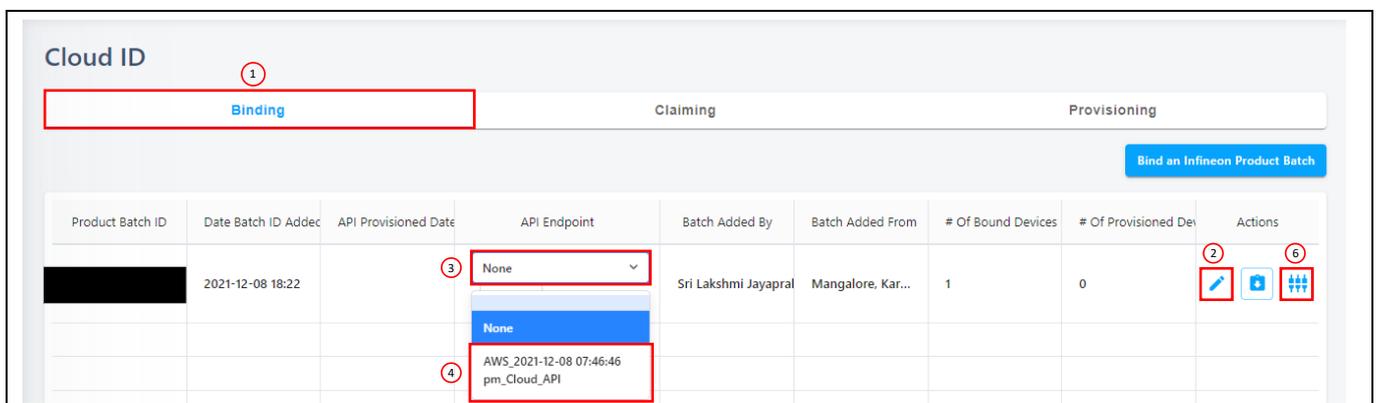
2. Click **Edit**.
3. Click the drop-down list in the **API Endpoint** column.
4. Choose the Product Cloud API created in Section **4.4.1.3**.
5. Click **Save**.
6. Click **Provision now**. This performs two functions:
   a. Creates the Thing for your kit in the AWS Console and attaches the related policy and device certificate.
   b. Pulls the AWS endpoint required by the kit to connect to your AWS account and pushes it to the device.

   When the kit is connected to Wi-Fi network, it will pull this endpoint from the CIRRENT cloud and connect to the AWS cloud automatically.

7. Jump to Section **4.5** to connect your kit to Wi-Fi.

Note: *To connect one more kit using the CIRRENT cloud flow, skip section **4.4.1.2** if you are using the same AWS account and Section **4.4.1.3** if you are using the same CIRRENT cloud account. Follow only Section **4.4.1.1** and Section **4.4.1.4**.*

4.4.2 AWS flow

1. Open the **AWS IoT Console**.
2. Select **Manage**, and then select **Things**.
3. Choose **Create Things**.
4. Select **Create single thing**, and then click **Next**.
5. In the terminal application, type the following command:

```
AT+CONF? ThingName
```

6. Copy the returned string (a sequence of alphanumeric characters) from the terminal.

7. On the console, on the **Specify thing properties** page, paste the copied string from the terminal into the **Thing name** field under **Thing properties**.

8. Leave other fields at their default values, and then click **Next**.

9. In the terminal application, type the following command:

```
AT+CONF? Certificate
```

   You will receive the device certificate in PEM format as part of the response.

10. Copy the returned string (a longer sequence of alphanumeric symbols), and save it into a text file on your host machine as "ThingName.cert.pem".
    Replace ThingName with the name of the Thing obtained in Step **5.**

11. On the **Configure device certificate** page, select **Use my certificate**, and then choose **CA is not registered with AWS IoT**.

12. Under **Certificate**, select **Choose file**.

13. Double-click the *ThingName.cert.pem* file created in Step **10**.
    Replace ThingName with the name of the Thing obtained in Step **5.**

14. Under **Certificate Status**, select **Active**.

15. Click **Next** to attach policies to the certificate.

16. Click **Create** to **Create a policy**. This opens a new tab.

17. Enter the policy name (e.g., *IoTDevPolicy*) and click **Advanced mode**.

18. Copy the following section into the console.

```
{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action":
"*", "Resource": "*" } ] }
```

Note: The examples in this document are intended only for development environments. All devices in your end product must have credentials with privileges that authorize only intended actions on specific resources. The specific permission policies can vary for your use case. Identify the permission policies that best meet your business and security requirements.

19. Click **Create** to create the policy.

20. Go back to the Thing creation tab, and choose the policy created in the earlier step.

21. Click **Create Thing**.

22. In the AWS IoT Console, choose **Settings**, and then copy your account **Endpoint** string under **Device data endpoint**.

23. Type the following AT command in the serial terminal to configure the endpoint:

```
AT+CONF EndPoint=<endpoint copied in step 22 >
```

24. Type the following AT command in the serial terminal to disable CIRRENT cloud ID error logs that will be visible from the serial terminal when using the AWS flow.
```
AT+CONF CloudID=Disable
```

4.5   Connect the kit to Wi-Fi

You can either follow Section **4.5.1** or Section **4.5.2** to connect the IFW956810 CCM evaluation kit to Wi-Fi.

A Wi-Fi onboarding mechanism is required for IoT products that do not have a display to enter Wi-Fi SSID and password. Use one of the following methods to connect the IFW956810 CCM evaluation kit to Wi-Fi.

Note: *Connect the kit only to a 2.4-GHz Wi-Fi network.*

4.5.1   Using Cirrent Wi-Fi onboarding app

The Cirrent cloud solution from Infineon provides an effective Wi-Fi onboarding service through the Cirrent Wi-Fi onboarding app for any IoT product. The IFW956810 CCM evaluation kit supports onboarding through either Bluetooth® LE or SoftAP.

1. Download and install the Cirrent Wi-Fi onboarding app from **Google Play Store for Android** or **iOS App Store for iOS on your mobile phone**. Scan the following QR code relevant to your mobile device:

Android	iOS

2. Type the following command in the serial terminal on the PC.

   `AT+CONFMODE`

   You should receive the "OK CONFMODE Enabled" response from the module.
3. Open the Cirrent Wi-Fi onboarding app.

   You can skip the sign in to perform Wi-Fi onboarding.
4. From the menu, select **Configuration** and then choose either **BLE** or **Soft AP** onboarding.
5. Follow the onscreen instructions and enter the Wi-Fi router's SSID and password.

4.5.2   Using AT commands

AT commands provide a simple method of Wi-Fi onboarding in a development environment.

- Type the following commands in sequence in the terminal application:

  `AT+CONF SSID=<your router ssid>`

  `AT+CONF Passphrase=<your router passphrase>`

Note: *Your local router's SSID and passphrase are stored securely inside the IFW56810 CCM module. While the SSID can be retrieved later (i.e., for debugging purposes), any attempt to retrieve the passphrase will return an error.*

4.6   Connect and interact with the AWS Cloud

Use the MQTT client in the AWS IoT Console to monitor the communication between your evaluation kit and the AWS Cloud.

1. Navigate to the AWS IoT Console (**https://console.aws.amazon.com/iot/**).
2. In the navigation pane, choose **Test** and then click **MQTT Test Client**.
3. In **Subscribe to a topic** panel, enter #, and then click **Subscribe**.

Skip Section **4.6.1** if you used CIRRENT Wi-Fi onboarding app to connect the kit to Wi-Fi.

You will receive "OK 1 CONNECTED" in the serial terminal once the device is connected to the AWS IoT Core.

4.6.1　　Connect

- Enter the following command in the serial terminal to establish a secure connection:

  ```
  AT+CONNECT
  ```

After a short time, you will receive the message "OK 1 CONNECTED".

4.6.2　　Send data to the AWS Cloud

- Enter the following command in the serial terminal to send the "Hello World!" message:

  ```
  AT+SEND data Hello World!
  ```

After a short time, you will receive the message "OK". You should see the "Hello World!" message appear on the AWS IoT Console under the topic "data".

4.6.3　　Receive data and commands from the cloud

By default, the device subscribes to a topic called "state".

1. Do the following on the AWS IoT Console:

   a) Select the MQTT client, and then select **Publish to a topic.**

   b) Type **state** in **Topic name** field. Keep "Hello from the AWS IoT Console" message.

   c) Click **Publish**.

2. Type the following command in the serial terminal:

   ```
   AT+GET
   ```

   You will receive the message "OK Hello from the AWS IoT Console".

5 Advanced AT commands

Wait for a response from the IFW956810 CCM evaluation kit in the serial terminal after entering each AT command.

5.1 Publish to a non-default topic

- Enter the following commands in sequence in the serial terminal:

```
AT+CONF Topic1=/MyPubTopic
AT+SEND1
```

5.2 Subscribe to a non-default topic

1. Enter the following commands in sequence in the serial terminal:

```
AT+CONF Topic2=/MySubTopic
AT+SUBSCRIBE2
```

2. Do the following on the AWS IoT Console:

   a) Select the MQTT client, and then select **Publish to a topic**.

   b) Type `MySubTopic` in the **Topic name** field. Keep the "Hello from the AWS IoT Console" message.

   c) Click **Publish**.

3. On your serial terminal, enter the following command:

```
AT+GET2
```

   You will receive the message "OK Hello from the AWS IoT Console".

   Click **here** for a complete list of AT commands supported by the IFW956810 evaluation kit.

6 Performing firmware over-the-air update

6.1 Prerequisites

1. Get the signed firmware image from Infineon. Contact Infineon Sales team for the firmware.

    The firmware will be provided along with the signature hashing algorithm used and signature encryption algorithm used. This information is required in subsequent steps.

2. Create an OTA update role in your account using the steps outlined **here**.

6.2 Create a firmware update job in AWS IoT

1. Open **AWS IoT Console**.
2. Click **Manage**, and then click **Jobs**.
3. Click **Create job.**
4. Select **Create FreeRTOS OTA Update Job**, and then click **Next**.
5. Provide a job name which is unique within your AWS account. Optionally, provide a description, and then click **Next**.
6. From the **Devices to update** drop-down list, choose the Thing name with which the IFW56810 CCM evaluation kit is registered in the account.
7. Select **MQTT** as the transfer protocol, and deselect **HTTP** if selected.
8. Select **Use my custom signed file.**
9. On the form that appears, enter the details from the **Prerequisites** section. Do the following:
    - In the **signature** field, provide the base64-encoded signature for the image.
    - From the **Original hashing algorithm** drop-down list, select the hashing algorithm provided by Infineon.
    - From the **Original encryption algorithm** drop-down list, select the encryption algorithm provided by Infineon.
    - In the **Path name of code signing certificate on device**, field, enter NA.
10. Select **Upload a new file**.
11. Click **Choose file** and upload the image received from Infineon.
12. Do one of the following:
    - Click **Create S3 bucket** to create a new bucket for the new uploaded image.
    - Click **Browse S3** to select an existing bucket in your account.
13. Under **Path Name of file on device**, enter NA if the image is not targeted as an executable file within a filesystem.
14. From the **File type** drop-down list, select a value "101" to signify that it is an IFW56810 CCM firmware update, and not a host firmware update.
15. Choose the OTA update role created above from the **Role** drop-down list under the **IAM role** section, and then click **Next**.
16. Click **Create Job**.

If successful, the job will be listed with the state as "in progress".

6.3 Monitor and apply the new firmware update for the IFW56810 module

The IFW56810 CCM module polls for firmware update jobs, receives and validates a job, and then enters a state waiting for the update to be accepted. The host application receives an OTA event indicating that a new firmware image is available for the IFW56810 CCM module.

The host application or the user can perform the following sequence by entering appropriate commands in the serial terminal:

1. Query the state of the job:

   ```
   AT+OTA?
   ```

   You will receive a response "OK 1 New OTA proposed."

2. Accept the new firmware update:

   ```
   AT+OTA ACCEPT
   ```

   The IFW56810 CCM module starts downloading the firmware update from the cloud.

3. Query the state of the job:

   ```
   AT+OTA?
   ```

   Downloading the image takes a few minutes to complete. During the OTA image download, this command returns "OK OTA in progress". You will receive an OTA event when the download is completed the image signature is verified.

4. Check whether the received image is verified:

   ```
   AT+EVENT?
   ```

   You will receive the response "OK 5 OTA event New Image verified".

5. Apply the new image received through OTA:

   ```
   AT+OTA APPLY
   ```

   The IFW56810 CCM module now reboots and boots up with the new image.

6. Connect back to the AWS IoT:

   ```
   AT+CONNECT
   ```

   The IFW56810 CCM module should now connect to AWS IoT, complete the self-test and mark the image as valid. This prevents further rollback to the old image.

You can check the job status by going back to the AWS IoT Console. You should see the job status as completed.

7 Setting up a serial terminal on the PC

Note: *The following instructions are only for a Windows PC.*

The IFW956810 CCM evaluation kit should be recognized by the PC when connected to it. If the device is recognized, COM ports will be available in the Device Manager.

If the device is not recognized, you need to install the FTDI USB to UART Bridge Virtual Communication Port drivers from this **link**. For more information, see **FAQs**.

7.1 Determine the COM port number

- Determine the COM port number from the Device Manager.

Figure 5   **COM port**

7.2 Serial terminal settings

1. Open a terminal such as Tera Term.
2. Choose the higher of the COM port numbers for the IFW956810 CCM evaluation kit.
3. Select **Set Up** > **Serial port.**
4. Select the settings as follows:
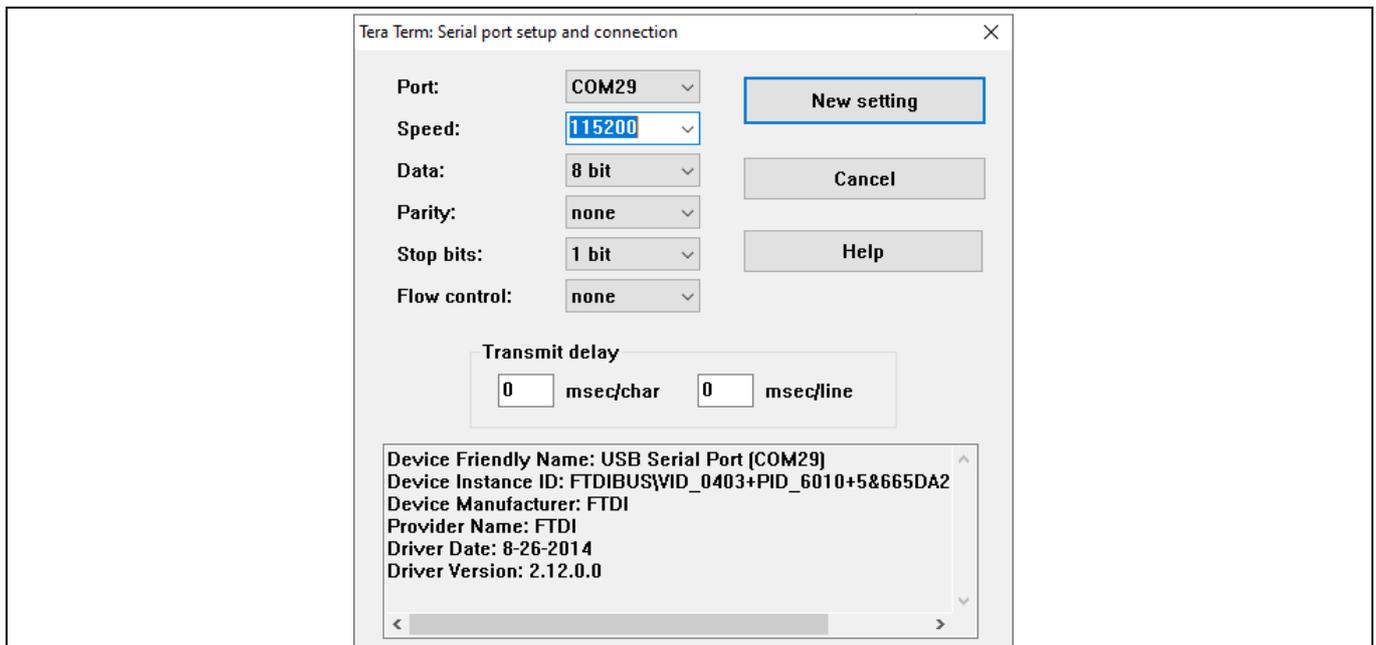
5. Select **Set Up** > **Terminal**.
6. Do the following:
   - Set **End of Line** as **Line Feed**.
   - Enable **Local Echo** to view the commands that you type on the terminal.
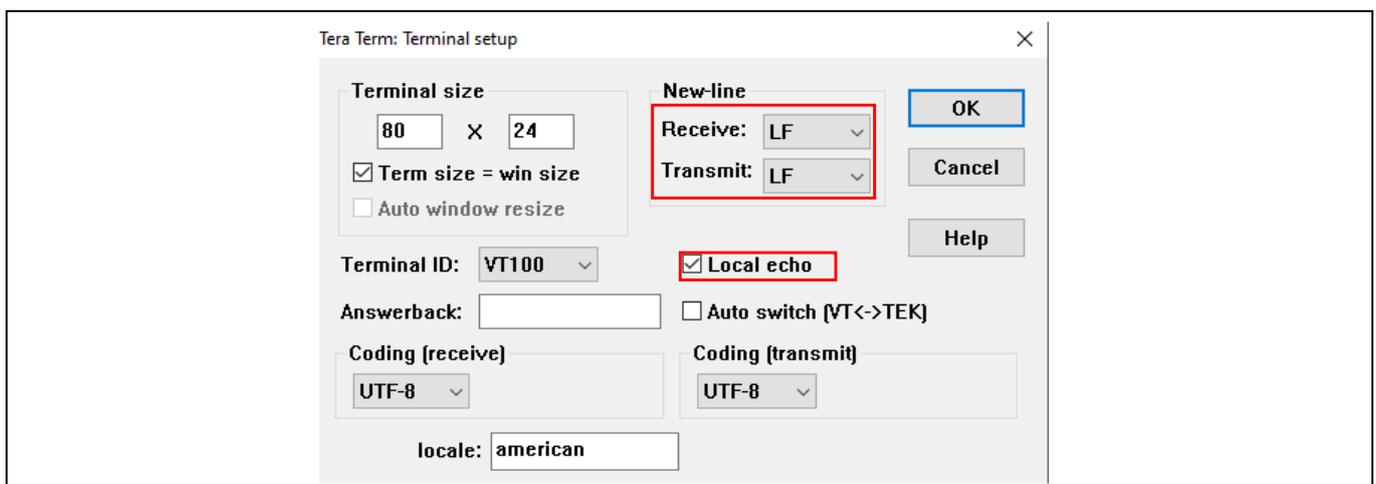
8   FAQs

1. Two COM ports enumerated when the kit is connected

- The IFW956810 CCM evaluation kit has a **FT2232H chip** capable of supporting USB to dual-channel UART (USB serial converter A and USB serial converter B). Only USB serial converter B is configured in the kit for USB-to-UART conversion. Therefore, use the higher-number COM port among the enumerated COM ports to communicate with the kit.

2. Errors appear on the terminal when commands are entered

For example:

```
AT+SUBCRIBE
ERR01 Invalid cmd
```

- Make sure that you have typed the command correctly.
- Note the error codes and refer to the **AWS IoT ExpressLink Programmer's Guide** for details of the error code and to determine the cause.

3. BLE/SoftAP onboarding does not work with the Cirrent app

- Check if `CONFMODE` was enabled using the `AT+CONFMODE` command.

4. Executing the AT+CONNECT command results in ERR 14 UNABLE TO CONNECT

The `AT+CONNECT` command first connects to Wi-Fi if not already connected and then connects to the AWS IoT Core.

- Check the Wi-Fi connection.
- Check the entered Wi-Fi credentials.
- Type the following command to verify whether the kit connects to Wi-Fi:

  ```
  AT+DIAG PING 8.8.8.8
  ```

  If the connection is successful, the device will respond with "OK Received ping response in <ping latency ms>".

- If the Wi-Fi connection test is successful, check the AWS IoT connection.
- Check the device certificate uploaded to the AWS IoT Core in Section **4.4** .

5. How to change the Wi-Fi network connected

- Execute `AT+DISCONNECT` on the serial terminal to disconnect from the current Wi-Fi network.
- See Section **4.5** to configure the required Wi-Fi credentials.

Revision history

Document version	Date of release	Description of changes
**	2021-11-29	Initial release
*A	2021-12-14	Added **CIRRENT cloud flow** to register the kit to the AWS account

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.