Getting Started Guide for AWS IoT Core Device Location By using the ZeroCarbon LoRa Evaluation Board and TE Antenna

Table of Contents

1 Document Information

1.1 Naming Conventions

The term "downlink device" or "endpoint device" is used in this document to refer to a LoRaWAN device that connects to a LoRaWAN "Gateway". The "Gateway" in turn, connects to AWS IoT Core for LoRaWAN.

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

Version: v0.1
Date: Mar. 11, 2024
Description of the change: First Release.

2 About AWS IoT Core Device Location

Refer to the AWS online documentation for information about AWS IoT Core Device Location and how it works.

3 Overview

The ZeroCarbon LoRa evaluation board is a tracker device manufactured by Tachibana Electronic Solutions Co., Ltd. It operates with ultra-low power consumption and can resolve location information. BRICK eIoT is a product that combines the ZeroCarbon LoRa evaluation board (tracker device) with his Dragino company (gateway) to quickly set up and evaluate your system using AWS IoT Core device location.

We are using a LoRaWAN indoor gateway manufactured by Dragino.

This document shows how to setup tracker device and gateway combination to start using AWS IoT Core Device Location.

AWS IoT Core Device Location is a feature of AWS IoT Core for LoRaWAN. We will first show the steps to connect to AWS IoT Core for LoRaWAN and then check AWS IoT Core Device Location.

4 Hardware Details

4.1 DataSheet

ZeroCarbon LoRa Evaluation Board:
https://tachibana-denshi-solutions.co.jp/lora_document/Zero_Carbon_EVB_HUM_Ver100_Preliminary_en.pdf
Dragino GW:
LIG16 Indoor LoRaWAN Gateway (dragino.com)
TE Antenna:
ANT-GNRM-L1A-3 : TE Connectivity

4.2 Standard Kit Contents

- ZeroCarbon LoRa Evaluation Board(PC-1570001)
- Dragino GW(LIG16 Indoor LoRaWAN Gateway 923)

Please refer to the BRICK eIoT LoRaWAN + AWS Starter Kit web site for more information.

4.3 User Provided items

- PC (Running Windows 10 or later)
- Micro USB cable (A-MicroB)
- USB type-C cable
- LAN cable (RJ45 cable)
- GPS Active TE Antenna(ANT-GNRM-L1A)

4.4 3rd Party purchasable items

Please contact us for details.
URL: https://www.nds-osk.co.jp/topics/672.html
E-mail: brick@nds-osk.co.jp

4.5 Additional Hardware References

Documentation for ZeroCarbon LoRa Evaluation Board
https://tachibana-denshi-solutions.co.jp/lora_document.php

5 Setup AWS

If you don't have an AWS account, refer to the instructions in the guide here. The relevant sections are Sign up for an AWS account and Create an administrative user.

If you have an AWS account and administrative user already set up, proceed with the steps below.

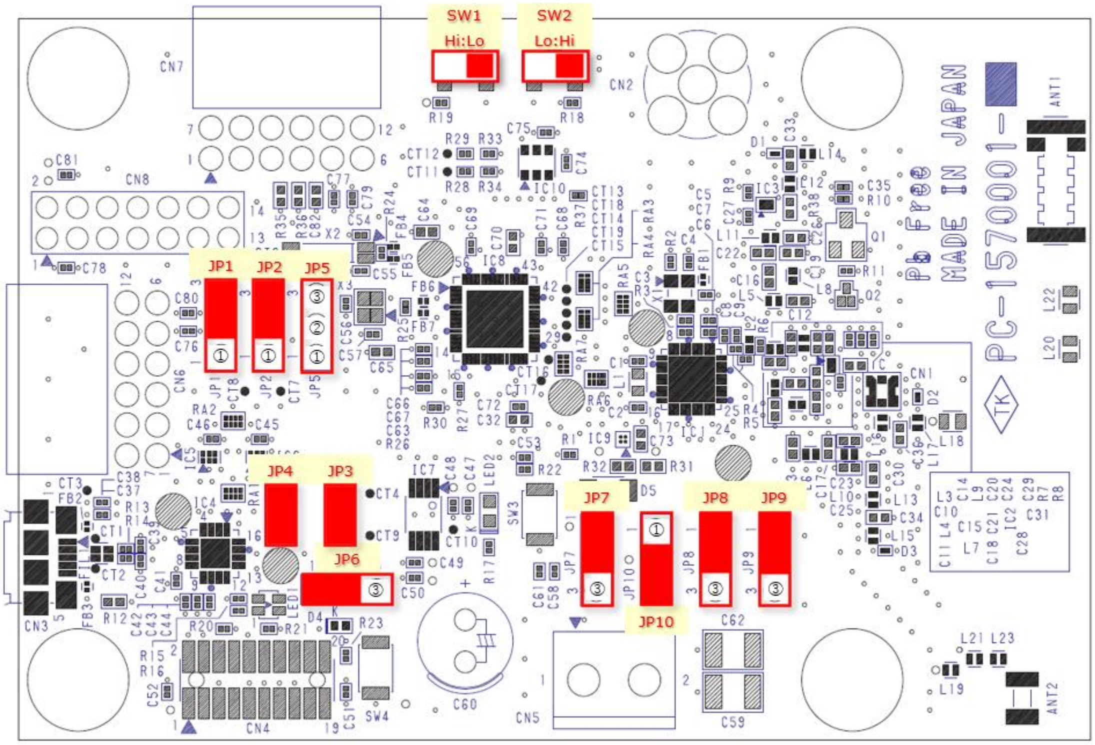
6 Set up and onboard your gateway to AWS

Set up your qualified gateway following documentation provided for the gateway. Once the gateway has been set up, onboard it to AWS IoT Core for LoRaWAN following the online instructions.

7 Set up your location tracker device

7.1 Jumper and Switch Setting

Set the jumpers and switches on the ZeroCarbon LoRa® Evaluation Board for USB communication and power supply as follows.



7.2 Connecting GPS Active TE Antenna.

Connect the GPS Active TE Antenna to the ZeroCarbon LoRa® Evaluation Board.

7.3 Connect the device to PC with a USB cable.

Connect to a PC with a USB cable to supply power to the tracker device.
You can view the operation status by connecting with terminal software using the method described in End Device Debugging.

8 Provision the location tracker device in AWS

8.1.1 Prerequisites

There should be enough information available about the device to provision it into the AWS IoT Core for LoRaWAN service. Steps to provision the tracker are detailed below. For additional details, refer to the AWS LoRaWAN developer guide.

For the list of valid device classes and versions supported by AWS IoT Core for LoRaWAN, refer to LoRaWAN version, Activation modes and Device classes. You can choose any combination that is valid for your device.

8.1.2 Create a Device Profile

Device profiles define the device capabilities and boot parameters that the network server uses to set the LoRaWAN radio access service. It includes selection of parameters such as LoRa frequency band, LoRa regional parameters version, and MAC version of the device.

The instructions below use the AWS CLI. If you prefer, you can use the AWS console to create a device profile. Refer to Add device profiles.

Ensure that the RF region (e.g., AS923-1) of your device matches the gateway's RF region.

First create the file *device-profile.json* with the appropriate settings. An example is shown below for an OTAA 1.0.3 Class A device

```
% cat device-profile.json
{
    "LoRaWAN": {
        "FactoryPresetFreqsList": [],
        "MacVersion": "1.0.3",
        "MaxDutyCycle": 10,
        "MaxEirp": 15,
        "RegParamsRevision": "RP002-1.0.1",
        "RfRegion": "AS923-1",
        "RxDataRate2": 8,
        "RxDelay1": 1,
        "RxDrOffset1": 0,
        "RxFreq2": 9233000,
        "Supports32BitFCnt": true,
        "SupportsClassB": false,
        "SupportsClassC": false,
        "SupportsJoin": true
     },
 "Name": "testDeviceProfile"
}
```

Use the AWS CLI to create the device profile.

```
% aws iotwireless create-device-profile --cli-input-json file://device-profile.json
```

Sample output:

```
{
    "Arn": "arn:aws:iotwireless:your-account-id:your-account-id:DeviceProfile/your-device-profile-id",
    "Id": "your-device-profile-id"
}
```

Save the value of *your-device-profile-id* for later use.

8.1.3 Create a Service Profile

Create a service profile with AddGwMetadata=true so that you will receive additional gateway metadata such as RSSI and SNR for each payload.

```
% aws iotwireless create-service-profile --lorawan AddGwMetadata=true --name "testServiceProfile"

Sample output:
{
    "Arn": "arn:aws:iotwireless:your-region:your-account-
id:ServiceProfile/your-service-profile-id",
    "Id": "your-service-profile-id"
}
```

Save the value of *your-service-profile-id* for later use.

8.1.4 Create uplink and location destinations

An AWS IoT Core for LoRaWAN destination allows you to either define an AWS IoT rule that processes a device's messages for use by AWS services, or send the message directly to an MQTT topic. In the steps below, we will send the message directly to an MQTT topic. Refer to the online guide for more details.

Create the following two destinations:

1. An uplink destination to route the raw messages from the device and publish it to the topic *testUplinkTopic*
2. A location destination to route the solved location data from the service to the topic *testLocationTopic*.

To create the destinations, you first need an IAM role – follow the instructions in the section Create an IAM role below.

8.1.4.1 *Create an IAM role*

AWS IoT Core for LoRaWAN destinations require IAM roles (you can use the same role for both destinations) that give AWS IoT Core for LoRaWAN the permissions necessary to send data to the destination topic.

To create this IAM role, follow the instructions in the sections **To create an IAM policy for your AWS IoT Core for LoRaWAN destination role**, and **To create an IAM role for an AWS IoT Core for LoRaWAN destination** as detailed online at Create an IAM role for your destinations.

For **Role name**, use *testDestinationRoleName.* Once the role has been created, save the ARN associated with this role for later use.

8.1.4.2 *Create the uplink and location destinations*

Using the AWS CLI, create the uplink destination:

```
% aws iotwireless create-destination --name "testUplinkDestination" --
expression-type "MqttTopic" --expression "testUplinkTopic" --role-arn
"arn:aws:iam::your-account-id:role/testDestinationRoleName"
```

Sample output:

```
{
    "Arn": "arn:aws:iotwireless:your-region:your-account-
id:Destination/testUplinkDestination",
    "Name": "testUplinkDestination"
}
```

Using the AWS CLI, create the location destination, using the same role:

```
% aws iotwireless create-destination --name "testLocationDestination" --
expression-type "MqttTopic" --expression "testLocationTopic" --role-arn
"arn:aws:iam::your-account-id:role/testDestinationRoleName"
```

Sample output:

```
{
    "Arn": "arn:aws:iotwireless:your-region:your-account-
id:Destination/testLocationDestination",
    "Name": "testLocationDestination"
}
```

8.1.5 Create the device in AWS IoT

The device can now be created using the above resources.

While creating the device, you have to specify the configuration for the tracker and specify the geolocation frame port (Fport), which is used by AWS IoT Core for LoRaWAN to pass the Wi-Fi / GNSS scan data to the AWS IoT Core Device Location service.

Create a file with the name *wireless-device.json* and specify the configuration in the file as shown below. Note that “Positioning” must be set to “Enabled”, and FPort must be 199.

```
% cat wireless-device.json
{
   "ClientRequestToken": "1234",
   "Description": "Device qualification testing",
   "DestinationName": "testUplinkDestination",
   "LoRaWAN": {
      "DevEui": "your-DevEui",
      "DeviceProfileId": "your-device-profile-id",
      "ServiceProfileId": "your-service-profile-id"
      "FPorts": {
         "Applications": [
            {
               "DestinationName": "testLocationDestination",
               "FPort": 199,
               "Type": "SemtechGeolocation"
            }
         ]
      },
      "OtaaV1_0_x": {
         "AppEui": "...", // received after claiming the device
         "AppKey": "..." // received after claiming the device
      }

   },
   "Name": "Qualification Test Device",
   "Positioning": "Enabled",
   "Type": "LoRaWAN"
}
```

Using the AWS CLI, create the wireless device.

```
% aws iotwireless create-wireless-device --cli-input-json file://wireless-
device.json
```

Sample output:

```
{
    "Arn": "arn:aws:iotwireless:your-region:your-account-
id:WirelessDevice/your-wireless-device-id",
    "Id": "your-wireless-device-id"
}
```

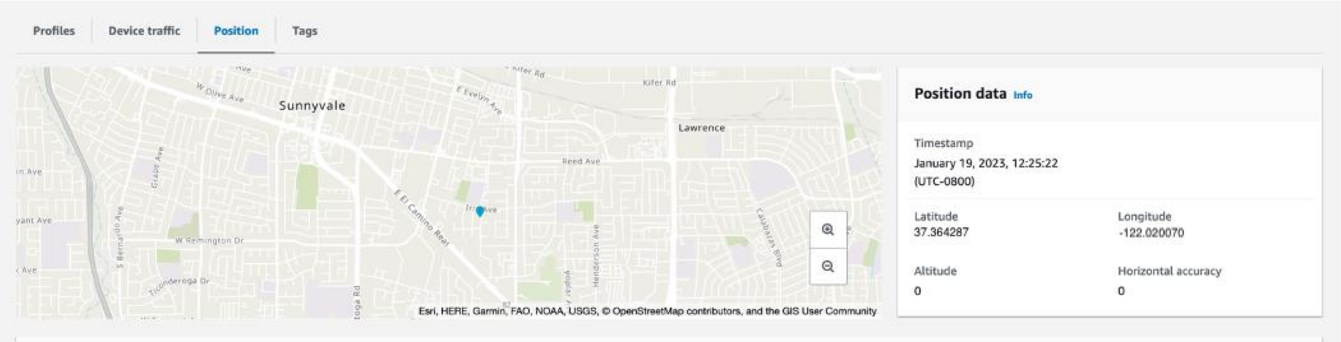
Save the value of *your-wireless-device-id* for later use.

Check that the device has connected successfully using the instructions at Check device connection status using the console.

9 Verify Operation with AWS IoT Core Device Location

9.1 Check the location via AWS Console map

The device location can be visualized on the AWS IoT Wireless console in the Position tab of the device page:


Profiles
Device traffic
Position
Tags
Sunnyvale
Lawrence
Esri, HERE, Garmin, FAO, NOAA, USGS, © OpenStreetMap contributors, and the GIS User Community
Info

Position data

Timestamp
January 19, 2023, 12:25:22
(UTC-0800)

Latitude
37.364287

Longitude
-122.020070

Altitude
0

Horizontal accuracy
0

9.2 Check the location using AWS CLI

Once the device location is solved, the last known position of the device is updated and invoking the get-resource-position command will yield the coordinates, along with any accuracy value.

The command should specify the resource-identifier of your wireless device. Use the value of *your-wireless-device-id* from the section Create the device or determine it using the list-wireless-devices command.

```
% aws iotwireless list-wireless-devices
```

Sample output:

```
{
    "WirelessDeviceList": [
        {
            "Name": "myLoRaWANDevice",
            "DestinationName": "IoTWirelessDestination",
            "Id": "your-wireless-device-id",
            "Type": "LoRaWAN",
            "LoRaWAN": {
                "DevEui": "ac12efc654d23fc2"
            },
            "Arn": "arn:aws:iotwireless:us-east-
1:123456789012:WirelessDevice/your-wireless-device-id"
        }
    ]
}
```

The output of the get-resource-position command will be in GeoJSON format and will need to be stored in an output file.

```
% aws iotwireless get-resource-position --resource-identifier your-wireless-
device-id --resource-type WirelessDevice outputfile.geojson

% cat outfile.geojson | jq
{
  "coordinates": [
    -122.020070,
    37.364287,
    0
  ],
  "type": "Point",
  "properties": {
    "verticalAccuracy": 0,
    "horizontalAccuracy": 0,
    "timestamp": "Thu Jan 19 20:25:22 UTC 2023"
  }
}
```

10 Troubleshooting

Please contact us for more information.
URL: tachibana-denshi-solutions.co.jp
E-mail: tcs_info@tachibana.co.jp

11 Additional Hardware References

Please contact us for more information.
URL: tachibana-denshi-solutions.co.jp
E-mail: tcs_info@tachibana.co.jp