Quickstart - Queclink to Azure IoT Hub
Quickly integrate Queclink devices to Azure IoT Hub

Queclink to Azure IoT Hub

This guide will walk you through integrating a Queclink device to Azure IoT Hub using an available solution template.

Pre-requisites

1. Account on Tartabit IoT Bridge.
2. Access to a Microsoft Azure subscription.
3. Queclink GL500 or GV500 devices.

Details of this guide

▸ In this guide you will:
  ▸ Create a new Azure IoT Hub.
  ▸ Retrieve the credentials needed to connect to your Azure IoT Hub.
  ▸ Import a solution template, this will create all of the required triggers, services, and endpoints required to connect your device.
  ▸ Connect a Queclink device
  ▸ Verify that the device is properly reporting into the Azure IoT Hub.

1. Create an IoT Hub in Azure

If you already have an IoT Hub, you can skip this step.

For a detailed walkthrough from Microsoft, check out the following link:
Quickstart - Queclink to Azure IoT Hub | Tartabit IoT Bridge

A: Select an existing resource group or create one.
B: Provide a unique name for your IoT Hub.
C: You can skip the other tabs and create a new Hub.

2. Retrieve your IoT Hub connection string

1. Navigate to your new Azure IoT Hub.
2. Click Shared Access Policies.

3. Select iothubowner (this is required because the IoT Bridge will automatically provision devices in the IoT Hub as they are needed).
4. Copy the **Connection string - primary key** value and save it for later.

![Connection string](image)

3. Prepare a Queclink device

Ensure your devices is configured according to the instructions below to ensure accurate reporting.

**GV500 device**

1. Configure backend server registration

   - Report Mode: **UDP mode**
   - Main Server IP: `queclink-us.tartabit.com` (or the correct URL for your server).
   - Main Server Port: **10000**
   - Buffer Mode: **0: Disable**
   - Protocol Format: **HEX**
Follow the steps below to import the solution template:

1. AT+GTSRI=gv500,4,,0,queclink-us.tartabit.com,10000,,0,,0,0,1,0,,FFFF$

2. Ensure the correct minimum hex report masks are present, the RSP Mask and EVT Mask must have the following fields enabled:
   - Check Device Type
   - Check Length
   - Uncheck Device Name (highly recommended to use IMEI instead of custom name)

1. AT+GTHRM=gv500,,,6F,FDF7FF,FC37FF,FD7D,EF,7D,7D,,FFFF$

**GL500 device**

1. Configure backend server registration
   - Report Mode: UDP mode
   - Main Server IP: queclink-us.tartabit.com (or the correct URL for your server).
   - Main Server Port: 10000
   - Buffer Mode: 0: Disable
   - Protocol Format: HEX

1. AT+GTSRI=gl500m,4,1,0,queclink-us.tartabit.com,10000,,0,,0,0,0,1,,FFFF$

2. Ensure the correct minimum hex report masks are present, the RSP Mask and EVT Mask must have the following fields enabled:
   - Check Device Type
   - Check Length
   - Uncheck Device Name (highly recommended to use IMEI instead of custom name)

1. AT+GTHRM=gl500m,,,7F,FF6F,FF6F,F77F,7F,,,,,FFFF$

**4. Import the solution template**

Follow the steps below to import the solution template:
1. Click **Solution Templates** at the bottom of any page.

2. Find the **Queclink to Microsoft Azure IoT Hub** template and click **Import**.

3. Follow the instructions below to import the template:
   
   ▶ **A**: Enter the IMEI or device name of your device.
   
   ▶ **B**: Enter the Azure IoT Hub connection string that you previously saved in Step #1.
   
   ▶ **C**: Select the option to **Start Triggers** this will automatically start all of the new triggers upon import.
   
   ▶ **D**: Click import when finished.
5. Verify the import was successful

You should receive an import result like this:

![Import Result](image)

6. Connect your Queclink device

Now that everything has been imported, you can connect your first Queclink device.

Verify that the client is connected by navigating to **Endpoints -> List** and checking the status.

![Endpoints List](image)

7. Check your device in Azure IoT Hub

With the device connected, you should now see it connected in the IoT Hub, and data being updated in the device twin.

1. Navigate to your Azure IoT Hub
2. Click IoT Devices

3. You should see your newly created device, click on it.
4. Click on **Device Twin**
5. You will see the LWM2M object data in your twin.

```json
{
  "device": {
    "batteryLevel": 0,
    "batteryStatus": "normal",
    "currentDateTime": "2020-10-08T15:12:19Z",
    "deviceType": "Cellular Module",
    "errorCode": {},
    "firmwareVersion": "20.06.11.0203",
    "hardwareVersion": "Rev 0",
    "manufacturer": "Tellit",
    "memoryFree": 12669152,
    "memoryTotal": 12864892,
    "modelNumber": "Not Set",
    "serialNumber": "Not Set",
    "softwareVersion": "1.0.2",
    "supportedBindingAndNodes": "UQ",
    "timezone": "EDT",
    "uTCoFFset": "-04:00"
  },
  "firmwareUpdate": {
    "g": {
      "firmwareUpdateDeliveryMethod": 2,
      "firmwareUpdateProtocolSupport": {
        "0": 0,
        "1": 1,
        "2": 2
      }
    }
  }
}
```
6. You can use the Azure IoT Explorer to view the telemetry as well.

You are done!

Powered by Wiki.js