RA2E1, RA2L1, RA0E1

LoRaWAN® Sensor Demo

Introduction

This application note describes a LoRaWAN® sensor network solution and introduces how to visualize sensor data transmitted by the RA2E1, RA2L1 or RA0E1 Sensor Node to the Cloud (AWS/Azure) via LoRaWAN® networks.

The LoRaWAN is a Low Power, Wide Area (LPWA) networking protocol designed to wirelessly connect battery operated 'things' to the internet, and targets key IoT requirements such as end-to-end security.

The application example provided in the demo package uses a Kerlink LoRaWAN Gateway, a Loriot LoRaWAN Network Server, and a Cloud provider. This document shows the basic steps to configure these services.

Target Device

RA2E1, RA2L1, RA0E1

SEMTECH SX1261/SX1262

Hardware

- Renesas FPB-RA2E1 (RTK7FPA2E1S00001BE) with Digilent Pmod USBUART
- Renesas EK-RA2L1 (RTK7EKA2L1S00001BE) with Digilent Pmod USBUART
- Renesas FPB-RA0E1 (RTK7FPA0E1S00001BJ) with Digilent Pmod USBUART
- SEMTECH SX1261/SX1262 Shield
- Renesas HS3001 Humidity and Temperature Sensor Module (US082-HS3001EVZ)
- Kerlink iFemtoCell (LoRaWAN Gateway)

Development Tools

- Renesas Flash Programmer V3.14.00 or later.
- Flexible Software Package (FSP) for Renesas RA MCU Family, version 5.3.0.
- e^2 studio 2024-04 with GCC Arm Embedded toolchain.
- Terminal emulator (such as TeraTerm)

Contents

1. Getting started

1.1 Demo Overview

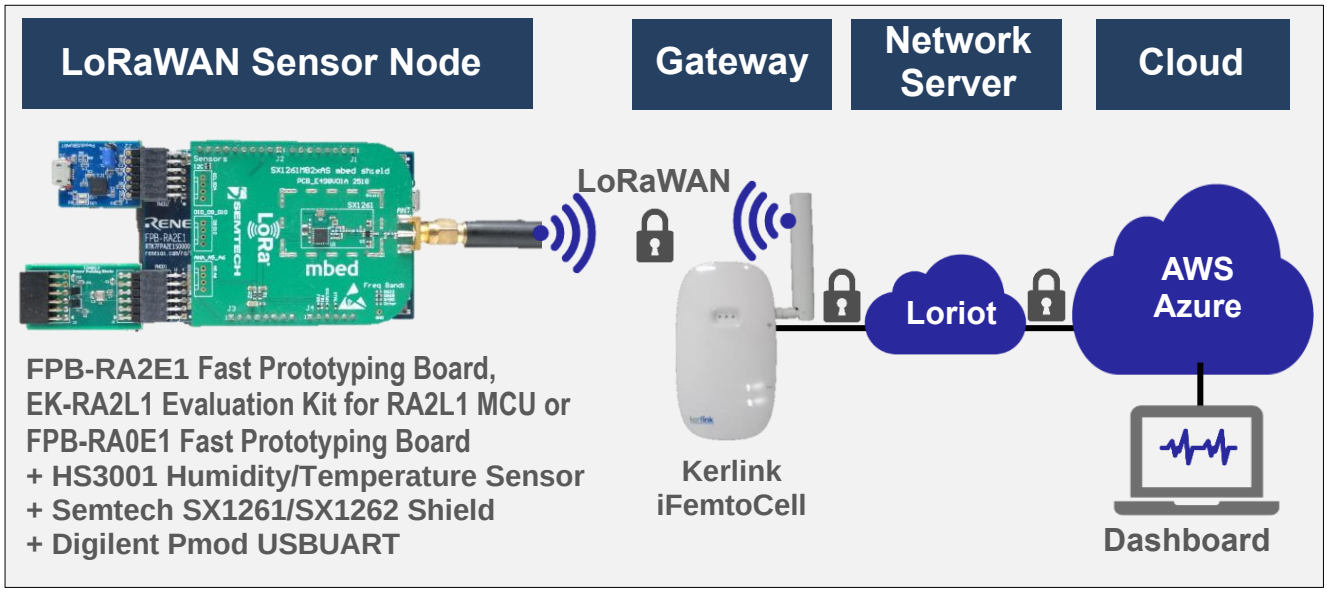


Figure 1. Demo Overview

IoT demo firmware running on RA2E1, RA2L1 or RA0E1 sends the sensor data to the LoRa modem periodically. The LoRa modem transmits the sensor data with LoRaWAN. The LoRaWAN Gateway (Kerlink iFemtocell) forwards the received LoRaWAN packets to the LoRaWAN Network Server (Loriot). The Loriot publishes the sensor data to the Cloud server (AWS/Azure).

In the case of AWS, the sensor data (Cayenne LPP format) received by AWS IoT Core is converted to the JSON by the demo Lambda Function (Python scripts). The demo Lambda Function stores the sensor data to the Amazon Timestream database. You can visualize the sensor data by accessing the timestream database with Amazon QuickSight or Grafana.

In the case of Azure, the sensor data (Cayenne LPP format) received by Azure IoT Hub is converted to the JSON by the demo Function App (C# scripts). To analyze and visualize the sensor data, demo Function App sends the sensor data to the Azure Data Explorer Clusters via Azure Event Hub.

VIDEOS:

- RA LoRa®-based Solution
  https://www.renesas.com/us/en/video/ra-lora-based-solution
- RA LoRaWAN® Sensor Demo Tutorial
  https://www.renesas.com/us/en/video/ra-lorawan-sensor-demo-tutorial

1.2 How to Setup the Boards

LoRaWAN Sensor Demo supports following board configurations:

FPB-RA2E1 Fast Prototyping Board + Semtech SX1261/SX1262 shield + HS3001 Sensor + USBUART

EK-RA2L1 Evaluation Kit for RA2L1 + Semtech SX1261/SX1262 shield + HS3001 Sensor + USBUART

FPB-RA0E1 Fast Prototyping Board + Semtech SX1261/SX1262 shield + HS3001 Sensor + USBUART

For more detail, please refer to *RA2E1, RA2L1, RA0E1 LoRa®-based Wireless Software Package* (R11AN0596).

IMPORTANT:

The use of wireless receivers and transmitters is restricted by international standards and domestic regulations. Wireless receivers and transmitters must therefore be used in accordance with the applicable laws and regulations of the country in which they are being used.

1.3 How to Setup the Demo Application

[How to program and debug the demo application]

Please refer to *RA2E1, RA2L1, RA0E1 LoRa®-based Wireless Software Package* (R11AN0596).

[How to configure and run the demo application]

1. Connect to the FPB-RA2E1, EK-RA2L1 or FPB-RA0E1 with Terminal emulator (such as TeraTerm).
   Terminal Settings: 115200 bps, 8 bit, no parity, 1 stop bit, no flow control, CR-LF, local echo ON
2. Set LoRaWAN specific parameters with AT-commands.
   For more detail of AT-commands, please refer to
   LoRaWAN® Stack Sample Application (R11AN0231)

   Example commands (do not use this as your actual settings):

```
AT+REGION=6                                  // Region: AS923-Group1
AT+CLASS=0                                   // Class A
AT+ACTMODE=1                                 // Activation: OTAA
AT+DEVEUI=749050FFFE000C26                   // DevEUI
AT+APPEUI=0123456789ABCDEF                   // AppEUI
AT+APPKEY=55555555555555555AAAAAAAAAAAAAAAA  // AppKey
AT+SAVE                                      // Save settings
```

3. Set the sensor specific parameters with AT-command.
   `AT+SENSOR=REJOIN,MEASURE,MODE:`
   - REJOIN: Join retry interval after join failure [sec] (default 30, Must be greater than zero)
   - MEASURE: Next measurement after Tx [sec] (default 30, Must be greater than zero)
     HS3001 requires a measurement time of 4 [sec] before transmitting sensor data.
     If 30 is specified for MEASURE, the sensor data will be sent at an interval of 30+4 [sec].
   - MODE:
     - 1: Auto start mode,
     - 0 or not 1: Manual start mode (default 0)
       If auto start mode is specified, sensor data will be sent automatically after reset.
       If manual start mode is specified, you need to invoke “AT+SENSOR” to start.

Example 1 (Manual start mode):

```
AT+SENSOR=60,60,0                    // Set parameters
AT+SENSOR                            // Start manually
(AT+DEBUG=0)                         // Optional: see NOTE2
```

Example 2 (Auto start mode):

```
AT+SENSOR=60,60,1                    // Set parameters
AT+SAVE                              // Save settings to NVM
AT+RESET=1                           // Auto start after reset
```

Note 1: If a small time is specified for the `REJOIN` and `MEASURE` parameters, `AT+SENSOR` will report a duty cycle limitation error and skip sending the sensor data until ADR enables a faster data rate.

Note 2: To reduce power consumption, `AT+SENSOR` can be followed by `AT+DEBUG=0`, which puts the MCU+SX126x to sleep when it is idle and ignores further command input until the hardware is reset.

Note 3: `AT+SENSOR` will send data with unconfirmed data type regardless of `AT+MTYPE` settings.

Note 4: `AT+SENSOR` will output log messages ***SENSOR:xxxxx**.
You can disable log message by deleting or undefining the following macro (re-build is required).
`#define APP_ENABLE_DEBUG_PRINTF in lorawan_sensor_sample.c`

Note 5: `AT+SENSOR` will send a sensor data as following format based on Cayenne LPP 2.0.
0x01 0x68 humidity(uint8_t): Channel 0 as humidity (unit 0.5%)
0x02 0x67 temperature(int16_t): Channel1 as temperature (unit 0.1 Celsius)

Cayenne will recognize any payload based on the Cayenne LPP 2.0 format. The demo function on the AWS and Azure support only above format, but you can extend the format by editing the demo functions.
For more detail on the Cayenne LPP 2.0 format, please refer to the following link:
https://docs.mydevices.com/docs/lorawan/cayenne-lpp

2. LoRaWAN Gateway

2.1 How to setup Kerlink LoRaWAN Gateway

This IoT demo uses Kerlink iFemtoCell as the LoRaWAN Gateway and also uses Loriot as the LoRaWAN Network Server. LoRaWAN Gateway is tightly coupled with LoRaWAN Network Server using the Network Server specific LoRaWAN Packet Forwarder. This means you should install the Loriot specific LoRaWAN Packet Forwarder into the Kerlink iFemtoCell LoRaWAN Gateway.

1. Get **Login Account** for Kerlink Website (https://www.kerlink.com/).
2. Get **Installation Manual** for iFemtocell from Kerlink Website.
3. Turn on the **Kerlink iFemtoCell**.
4. (Optional) Update **Kerlink firmware**.
5. Get **Login Account** for Loriot Website (https://www.loriot.io/).
6. Get **setup guide** for Kerlink iFemtoCell.
7. Download the **Loriot software package** (included Packet Forwarder).
8. Install and run the Loriot software package on the Kerlink iFemtoCell.

3. LoRaWAN Network Server

3.1 How to setup the Loriot LoRaWAN Network Server

Loriot provides the LoRaWAN Network Server. You should register the LoRaWAN Gateway and LoRaWAN Devices on the Network Server. You can register the cloud server as the output of the Network Server.

[Register LoRaWAN Gateway]

1. Login to Loriot LoRaWAN Network Server for your region (https://www.loriot.io/login.html)
2. Go to your Loriot Dashboard.
3. Click **Networks**> click **New Network** > fill in **Name** field > click **Create new network**.
4. Click **+Add Gateway** > click **Kerlink iFemtocell (OS V4.x.x ..)** > set MAC Address and Location.
5. Click **Configure**> select **Channel Plan** > select plan (for example, **AS923-1** in Japan)

[Register LoRaWAN Device]

1. Login to Loriot LoRaWAN Network Server for your region (https://www.loriot.io/login.html)
2. Go to your Loriot Dashboard.
3. Click **APPLICATIONS**> click **New Application** > fill in the **Name** field.
4. Enable **OTTA** in the **Features** section
5. Click **+Enroll Device** > select **LoRaWAN 1.0.x** and **OTAA**
6. Fill in the **Title**, **Device EUI**, **Join EUI** (Application EUI), **Application Key** > Click **Enroll**
   Device EUI is the same as the MAC address labeled on your LoRa Module.
   Example settings (do not use this as your actual settings):
   - **Device EUI** is 749050FFFE000C26
   - **Join EUI** (Application EUI) is 0123456789ABCDEF
   - **Application Key** is 5555555555555555AAAAAAAAAAAAAAAA

[Register the Cloud]

1. Login to the **Loriot LoRaWAN Network Server** for your region (https://www.loriot.io/login.html)
2. Go to your **Loriot Dashboard** > click **Applications**> click **APPLICATIONS**
3. Click app name (for example, **Sample App**) > click **Output** > click **Add new output** > select **Cloud.**
4. Enter your cloud credential information using the instructions on the screen.

[AWS IoT]

Endpoint Random, Region: You can find in **AWS** > **AWS IoT** > **Settings** > **Device data endpoint**

Access ID, Key: You can find in **AWS** > **IAM** > **Users** > **loriot**> security credentials

[Azure IoT]

IoT Hub Name: your Azure IoT Hub name (for example, **demoIotHubRliot**)

Primary Key: **Azure** > your IoT Hub (for example, **demoIotHubRliot**) > **Secure Settings** > **Shared access policies** > **device** > **Shared access key** > **Primary Key**

4. AWS Cloud Server

4.1 How to setup AWS Cloud Server

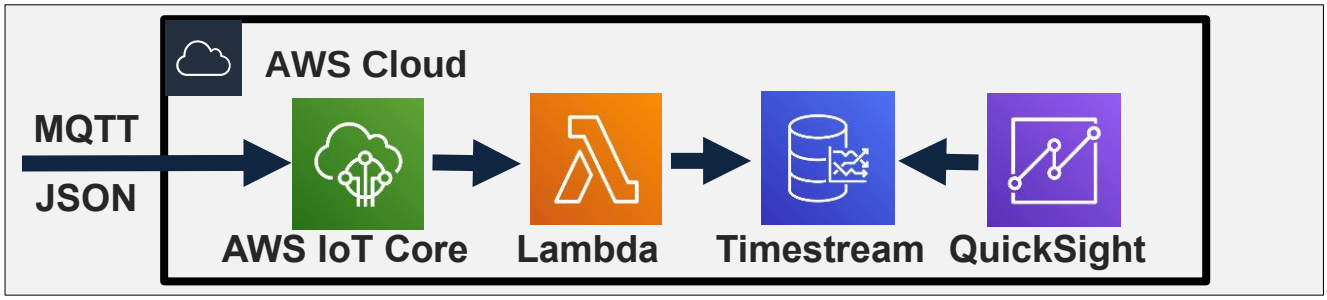


Figure 2. AWS Cloud Server Setup

AWS IoT Core is the MQTT message broker. When you configure the AWS IoT Core to subscribe to the MQTT messages published by Loriot LoRaWAN Network Server, the JSON message including sensor data on MQTT payload will be routed to the demo application (Lambda).

Demo Application will decode the individual sensor value as JSON (for example, `{'humidity':46, 'temperature':24.9}`) from the hex-ascii data stream formatted as Cayenne LPP (for example , **01685D026700F9**). The demo application stores sensor data and associated meta information (for example, **DevEUI**) in real time to Amazon Timestream.

You can visualize the sensor data with the Amazon QuickSight or 3rd party tools (for example, Grafana).

[Configuration Steps and References Documents]

1. Get Login Account for **AWS** (https://aws.amazon.com/)
2. Setup **AWS IoT Core** (https://docs.aws.amazon.com/iot/latest/developerguide/iot-gs.html)
3. Setup **Lambda** (https://docs.aws.amazon.com/timestream/latest/developerguide/Lambda.html)
4. Setup **Timestream** (https://docs.aws.amazon.com/timestream/index.html)
5. Setup **QuickSight** (https://docs.aws.amazon.com/timestream/latest/developerguide/Quicksight.html) or setup **Grafana** (https://docs.aws.amazon.com/timestream/latest/developerguide/Grafana.html)

Note: All resources (things endpoint on the AWS IoT Core, AWS Lambda function, Amazon Timestream databases/tables, and Amazon QuickSight dataset) should be created in the same region. Amazon Timestream is available in US East (Virginia), US East (Ohio), US West (Oregon), Europe (Ireland) and Europe (Frankfurt). If you use the Amazon QuickSight with Amazon Timestream, we recommend using the US East (Virginia) region.

Note: AWS IoT Core for LoRaWAN has been released in December 2020 but is not covered in this document.

4.2 How to setup AWS IoT Core

1. Login to **AWS** > Select **IoT Core** from services menu.
2. Click **Security** > **Policies** > **Create**, and fill out the fields as follows, and then save the policy.
   — Policy name: demoPolicy
   — Policy document: click Advanced mode (enter the following policy description for demo use only.)

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

   Note: This policy allows full access to the IoT Core, so do not use it for production use.

3. Click on **Manage** > **Things** > **Create things** > **Create single thing** > **Next**, and fill out the fields as follows:
   — Thing Name: DevEUI of your device (for example, **749050FFFE000C26**)
   — Device Shadow: Select **No shadow**

   Then, click on **Next**. Select `Auto-generate a new certificate(recommended)`. Click on **Next.** Select `demoPolicy` as a policy to attach to this certificate. Click on **Create thing**. Download certification files and key files. Click on **Done.**

4. Click on **Message Routing** > **Rules** > **Create**, and fill out the fields as follows, and the save the rule.
   — Rule Name: **demoRule**
   — Rule query statement (SQL version 2016-03-23):

```
SELECT current.state.reported
FROM '$aws/things/+/shadow/update/documents'
WHERE current.state.reported.cmd = "rx"

```

   Note: This query statement is for the Loriot network server only, because the MQTT topic and contents are network server specific. Loriot sends the message on the `$aws/things/DevEUI/shadow/update/documents` as topic. LoRaWAN uplink packet is located in the `current.state.reported` when ‘cmd’ is “rx”.

5. Click **Add rule action** on Rule actions. Select `Send a message to a Lambda function` (not **Timestream**).
   Then, click **Create a Lambda function**. After creating the Lambda function, you should select `demoFunction` as Lambda function associated with this rule. Click **Add action** > **Add Rule**.

4.3 How to Setup AWS Lambda (Demo Application)

Note: To use the Amazon Timestream from Lambda (Python), Python SDK (boto3) v1.15.9 or later is required.

To set up the AWS Lambda demo application, use the following steps:

1. Continuing from previous section step 5, click **Create a new Lambda function** or select **Lambda** from services menu > click **Functions** > **Create function** > select **Author from scratch**
   - Function name: **demoFunction**
   - Runtime: Python 3.8
   - Architecture: x86_64

   Click **Create function**.

2. Click **Upload from** on `Code` tab > select **Upload a .zip file**.
   Upload the demoFunction.zip located in the release `package\samples\cloud\aws\demoFunction.zip`.
   Click **Edit** on **Runtime Settings** > Set `lambda_function.lambda_handler` as handler > click **Save**.

3. Attach the timestream access permission to the AWS Lambada function by following steps.
   Click on **Permissions** in **Configuration** tab.
   Click role name (for example, `demoFunction-role-jjagv5pu`) on `Execution role` card.
   Click **Attach policies** > Search **AmazonTimestreamFullAccess**.
   Click check box on **AmazonTimestreamFullAccess**> Click **Attach Policy** for demo use only.
   Note: This policy allows full access to the Amazon Timestream, so do not use it for production use.

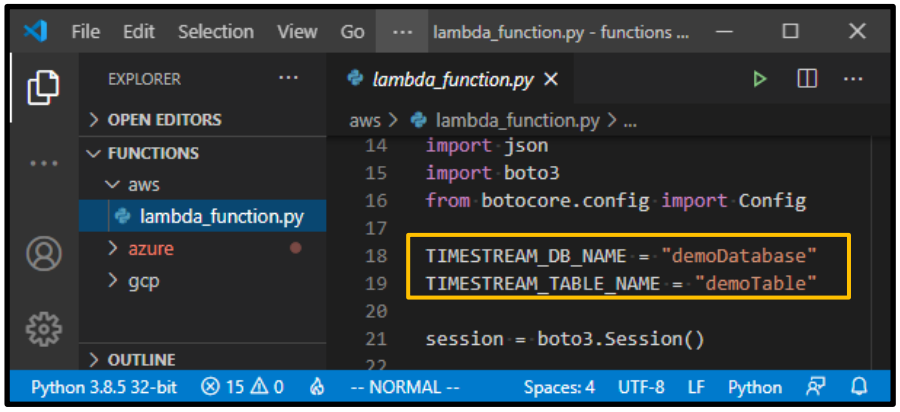
4.4 How to setup Amazon Timestream

1. Select **Amazon Timestream** from services menu > click **Databases** > click **Create database**.
   - Configuration: **Standard database**
   - Database Name: **demoDatabase**
   - Encryption Master Key: empty (after creating database, **aws/timestream** will be set).

   Click **Create database.**

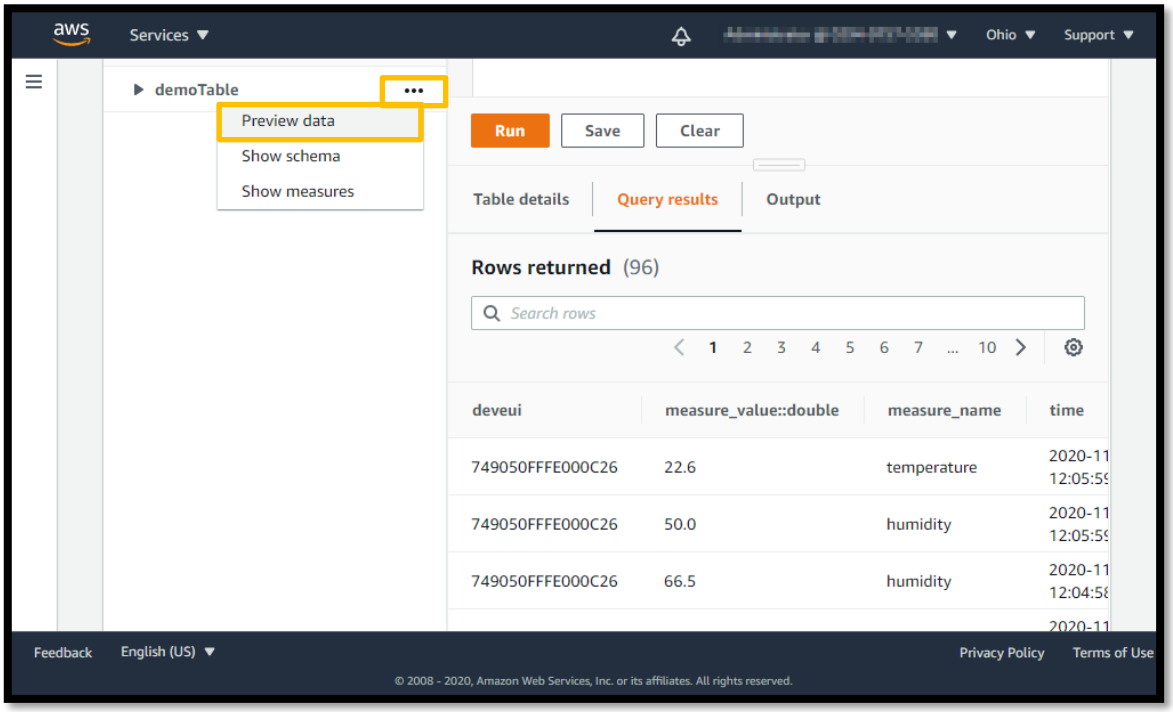
2. Click **Tables** > click **Create table**.
   - Database Name: **demoDatabase**
   - Table Name: **demoTable**
   - Memory store retention: 2 hours
   - Magnetic store retention: 1 day

   Click **Create table**.



Note: If you change the Database Name and Table Name, you should change the `lambda_function.py`.

3. Click **Query editor** > Click "- - -" and select **Preview data** > click **Run.**



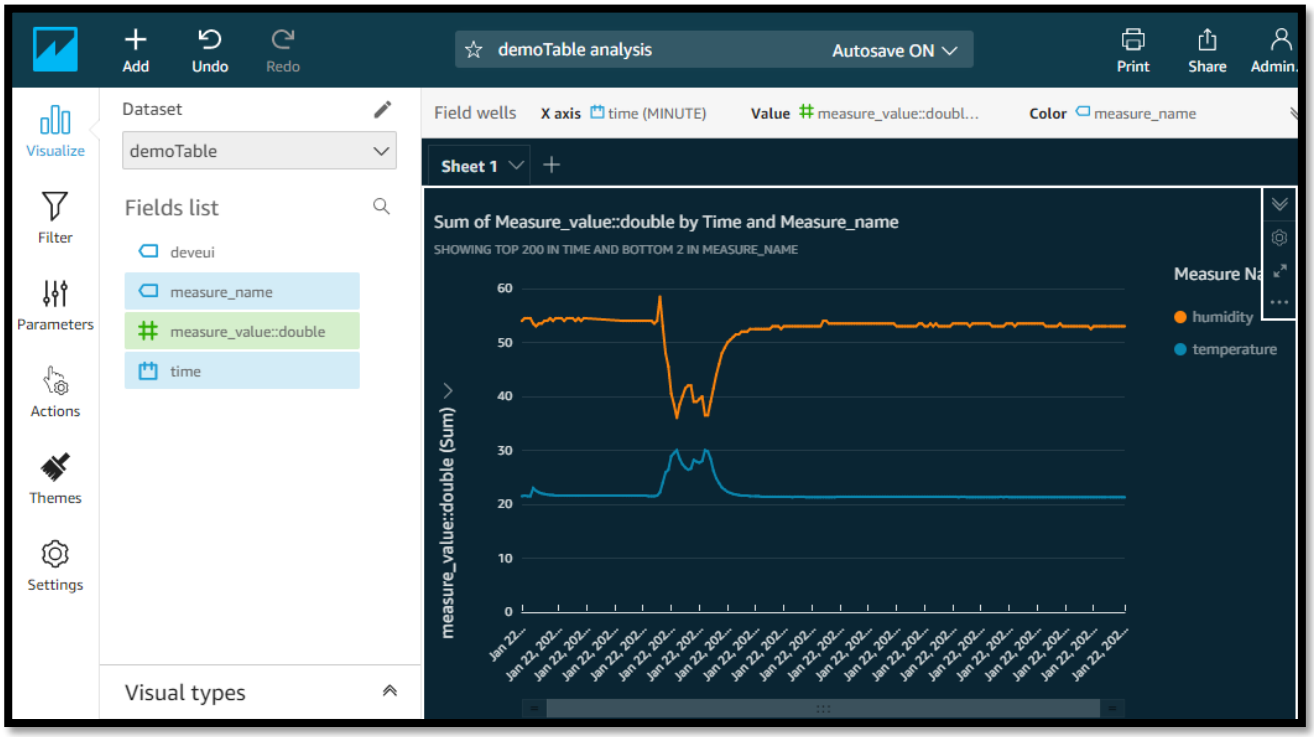
IMPORTANT:

You will continue to be billed until you stop the services and delete the resources you have created.

4.5 How to Set up Amazon QuickSight

1. Login to **AWS** > select **QuickSight** from services menu.
2. Sign into QuickSight (you need to select the same region as Timestream (for example, US East (Virginia))
   Note: By default, Amazon Timestream is not available in `QuickSight access to AWS services` list. To enable Amazon Timestream on user account, **Manage QuickSight** > **Security & permissions** > **QuickSight access to AWS services** > **Manage.**
3. Click **Datasets**> click **New dataset** > select **Timestream** card.
4. Enter **demoDatabase** as Data source name > click **Validate connection** > click **Create data source** > select **demoTable**> click **Select** > select **Directly query your data** > click **Visualize**.
5. Visualize Humidity and Temperature by using the following steps:
   A. Select **Line Chart** on visual styles menu.
   B. Click **time** (on **Fields list**)
   C. Click **measure_value** (on **Fields list**)
   D. Click **measure_name** (on **Fields list**)
   E. Click **time**(on **Field wells**) > select **Minute** as aggregate.
   F. Click **Themes** (on **Left Menu**) > select **Midnight** > click "…" > click **Apply**

   Finally, you can see the following charts.



IMPORTANT:

You will continue to be billed until you unsubscribe from Amazon QuickSight.

To unsubscribe from Amazon QuickSight, click your name on top bar > select **Manage QuickSight** > click **Account settings** > click **Delete account**.

4.6 How to set up Grafana

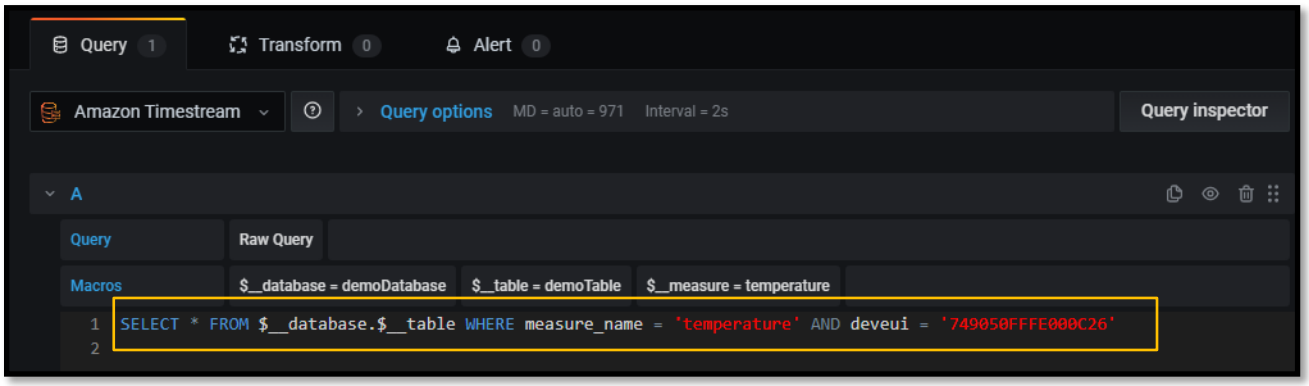
1. Install Grafana (https://grafana.com/) on your PC or **Log in to Grafana Cloud**.
2. Install plugin for Timestream and add IAM policy for accessing the Timestream. For more detail, please refer to https://grafana.com/grafana/plugins/grafana-timestream-datasource

   Click **Gear mark** > click **Data Sources** > click **Add data source** > select **Amazon Timestream**.

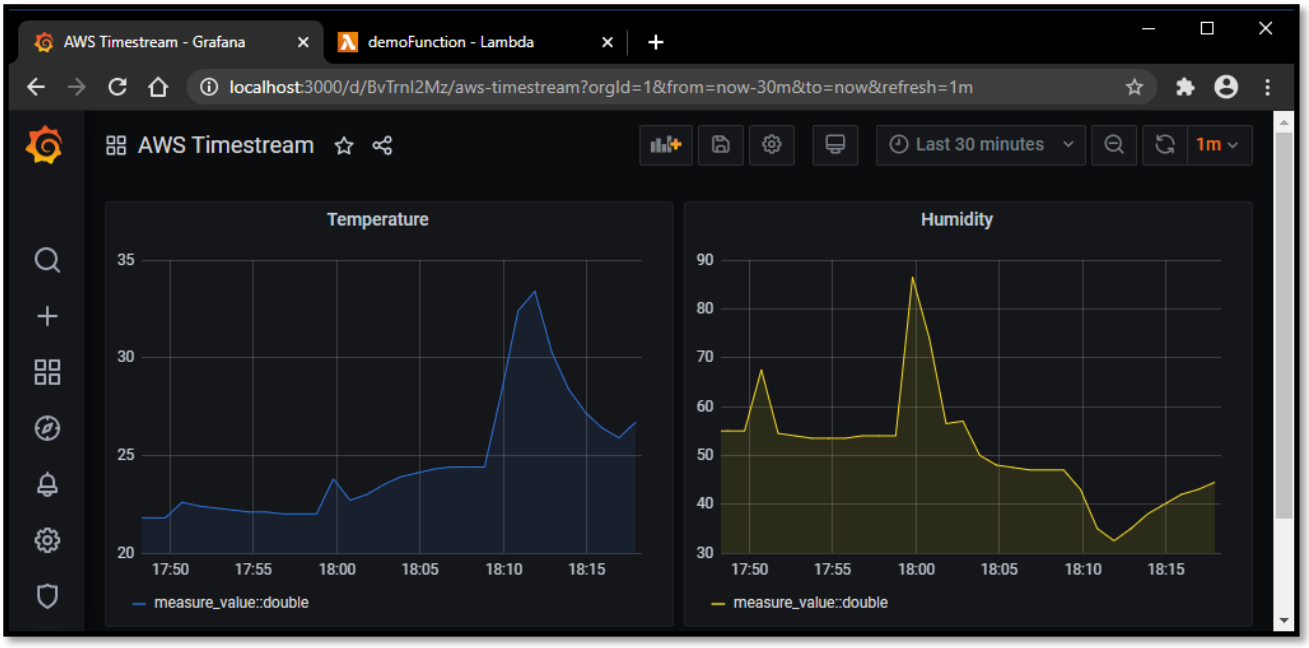
   — Auth Provider: **Access & secret key**
   — Access Key ID: Your AWS Access Key (**AWS** > **IAM** > **Users** > **Security Credentials** > **Create Access Key**)
   — Secret Key: Your AWS Secret Key (**AWS** > **IAM** > **Users** > **Security Credentials** > **Create Access Key**)
   — Default Region: **us-east-1**

   The following default query macros settings are optional:
   Default Query Macros: `$__database:demoDatabase, $__table:demoTable, $__measure:temperature`

3. Click “+” mark (add Dashboard) > click **Add New Panel** > set query commands (SQL) as shown below.



Finally, you can see the temperature and humidity data on the dashboard.



Note: Amazon Managed Grafana has been released in 2020 but is not covered in this document.

5. Azure Cloud Server

5.1 How to Set up Azure Cloud Server

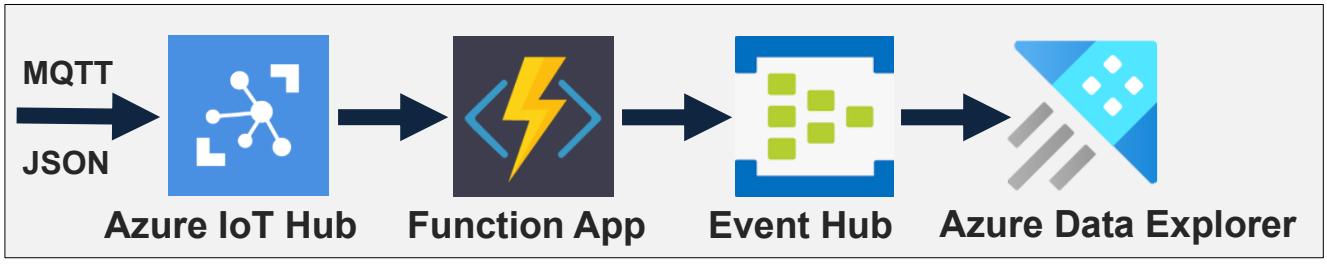


Figure 3. Azure Cloud Server Setup

Azure IoT Hub is the MQTT message broker. When you configure the Azure IoT Hub to subscribe to the MQTT messages published by LoRaWAN Network Server, the JSON message including sensor data on MQTT payload will be routed to the built-in endpoint (IoT Event Hub) which will invoke the demo Function App.

The demo Function App will decode the messages and convert the sensor data from the hex-ascii formatted as Cayenne LPP (for example, **01685D026700F9**) to the JSON (for example, `{'humidity':46, 'temperature':24.9}`).

To analyze and visualize the sensor data, the demo Function App will send the sensor data to the Azure Data Explorer Clusters via Azure Event Hub.

Overview:

1. Create an instance of the **Azure IoT Hub**. Register your sensor node as IoT device.
2. Create an instance of the **Azure Event Hub** as event source of the **Azure Data Explorer Clusters**.
3. Create the demo **Function App**. Bind the Event Hub as the output of the **demoFunction**.
4. Create the instance of the **Azure Data Explorer Clusters** and connect to the Event Hub.
5. Analyze and visualize the sensor data with ADX (Azure Data Explorer) Web UI.

IMPORTANT:

To run this Azure demo, you must upgrade from a Free-Trial plan to a Pay-per-use plan.

5.2 How to set up Azure IoT Hub and Event Hub

[Configure Azure IoT Hub]

1. Get **Login Account** for Azure (https://azure.microsoft.com/)
2. Go to your **Azure Portal**.
3. Click **All Services** > click **Internet of things** > click **IoT Hub** > click **+Create** > fill in all fields.
   Note: In this demo, following parameters are used as example (Do not use this as your actual settings).
   - Subscription: **Pay-per-use**
   - Resource Group: **demoResouceGroup**(Create New) Required to create new storage account)
   - IoT Hub Name: **demoIotHubRliot**
   - Region: **Japan East**
   - Connectivity Configuration: **Public access**
   - Pricing and scale tier: **F1: Free tier** (max 8,000 messages per day for free)

   Click **Create**> click **Go to resource** (go to created IoT Hub **demoIoTHubRliot**).

4. Click **Devices** > click **Add Device** > fill in **Device ID** as Device EUI
   In this demo, following parameters are used as example (Do not use this as your actual settings).
   - Device ID: **749050FFFE000C26**
   - Authentication type: **Symmetric key**
   - Auto-generate keys: **Checked**
   - Connect this device to an IoT hub: **Enable**
   - Parent device: **No parent device**

Click **Save**.

[Configure Azure Event Hub]

1. Login to Azure.
2. Go to your **Azure Portal**.
3. Click **All Services** > click **Internet of things** > click **Event Hubs** > click **+Create** > fill in all fields.
   In this demo, following parameters are used as example (Do not use this as your actual settings).
   - Subscription: **Pay-per-use**
   - Resource Group: **demoResourceGroup**
   - Namespace name: **demoEventHubRliot**
   - Location: **Japan East**
   - Pricing tier: **Basic**
   - Throughput Units: **1**

Click “Review + create” > Click “Create”

Click “Go to resource” (Go to created Event Hub) > Click “+Event Hub” > Fill in all field.

In this demo, following parameters are used as example (Do not use this as your actual settings).
- Name: **demoEventHub**
- Partition Count: 2

Click **Review + create** > click **Create**

5.3 How to set up Azure Function App (Demo Application)

[Configure Azure Function App]

1. Login to Azure.
2. Go to your **Azure Portal**.
3. Click **All services** > **Compute**> **Function App** > Click **+Create** > fill in all fields.
   In this demo, following parameters are used as example (do not use this as your actual settings).
   - Subscription: **Pay-per-use**
   - Resource Group: Select **demoResourceGroup**
   - Function App name: **demoFunctionRliot**
   - Publish: **Code**
   - Runtime stack: **.Net**
   - Version: **3.1**
   - Region: **Japan East**
   - Operating System: **Windows**
   - Plan type: **Consumption (Serverless**)
   - Storage account: **demofunctionrliotsa** (create new)
   - Enable Application Insights: **Yes**
   - Application Insights: **demoFunctionRliot (Japan East)**

   Click **Create**> click **Go to resource** (go to created Function App **demoFunctionRliot**).

4. Click **Start** if the Function app is stopped. Click **Functions**> click **+ Create** > fill in all field > click **Create**.
   In this demo, following parameters are used as example (do not use this as your actual settings):
   - Development environment: Select **Develop in portal**
   - Template: Select **IoT Hub (Event Hub)**
   - New Function: **IoTHub_EventHub1** (default)
   - Event Hub connection: Click **New**> select **IoT Hub** > fill all shown as below > click **OK**
   - **demoIotHubRliot** as Event Hub connection, **Events (built-in)** as Event Hub connection
   - Event Hub consumer group: **$Default**

5. Click function name (for example, **IoTHub_EventHub1**) > click **Code+Test**. Copy-paste the C# Script referring `(package top)\samples\cloud\azure\run.csx` > click **Save.**
6. Click **Integration**> click **+Add output** > fill in all filed as follow > click **OK.**
   In this demo, following parameters are used as example (do not use this as your actual settings):
   - Binding Type: Select **Azure Event Hubs**
   - Event Hub connection: Click **New** > select **Event Hub** > select as follows > click **OK**.
     - 1st: **demoEventHubRliot**, 2nd: **demoeventhub**, 3rd: **RootManagesSharedAccessKey**
   - Event parameter name: **outputEventHubMessage**(it depends on the `run.csx`)
   - Event Hub Name: **demoeventhub**

7. Click function name (for example, **IoTHub_EventHub1**) > click **Enable** if it is not enabled.

5.4 How to set up Azure Data Explorer Clusters

[Configure Azure Data Explorer Clusters]

1. Login to Azure.
2. Go to your **Azure Portal**.
3. Click **All services** > **Analytics**> **Azure Data Explorer Clusters** > click **+Create** > fill in all fields as follows.
   In this demo, following parameters are used as example (do not use this as your actual settings):
   - Subscription: **Pay-per-use**
   - Resource Group: Click **Create new** > enter **demoAdxResourceGroup**
   - Cluster name: **demoadxcluster**
   - Region: **Japan East**
   - Workload: **Dev/test**
   - Availability zones: **(none)**

   Click **Review + create** > After creating, click **Go to resource**.
4. Click **+Add Database** > fill in all fields as follows > click **Create.**
   In this demo, following parameters are used as example (Do not use this as your actual settings).
   - Database name: **demoAdxDatabase**
   - Retention period (in days): **7**
   - Cache period (in days): **1**
5. Click created database name (for example, **demoAdxDatabase**) > click **Query.**

IMPORTANT:

Once you create the Azure Data Explorer Clusters, you will continue to be billed until you clean up the resources regarding the Azure Data Explorer Clusters.

To clean up the resources regarding Azure Data Explorer Clusters:

- Click **Resource groups** icon on portal menu > click the resource group (for example, **demoAdxResourceGroup**)
- Click **Delete resource group** on top menu > enter resource group name > click **Delete.**

You should also perform a cleanup of the **demoResourceGroup** when you no longer need to use the IoT Hub Function App and Event Hub.

[Configure Azure Data Explorer (Web UI)]

1. Start sending your sensor node and wait about 5 minutes until the first data is stored to the database.
2. Click **Open in Web UI** on the tab menu > click **Data** on the left menu bar.
3. Click **Ingest data** > fill in all fields as follows.
   In this demo, following parameters are used as example (do not use this as your actual settings).
   — Cluster: **demoadxcluster.japaneast**
   — Database: **demoAdxDatabase**
   — Table: Select **New table**, enter **demoAdxTable**

   Click **Next: Source.**

   — Source type: **Event Hub**
   — Subscription: **Pay-per-use**
   — Event Hub namespace: **demoEventHubRliot**
   — Event Hub: **demoadxeventhub**
   — Data connection name: **demoAdxDatabase-demoadxeventhub**
   — Consumer group: **$Default**
   — Compression: **None**
   — Event system properties: **empty**

   Click **Next: Schema** > wait until some sensor data appear in the table.

   — Data format: **JSON**

   Click **Next: Start Ingestion** > click **Close**> click **Dashboards** (preview)

   Click **New Dashboard** > enter **demoAdxDashboard**> click **Create**> click **+Add Tile**

   Source: Chose **+New data source** > fill in all fields as follows.

   In this demo, following parameters are used as example (do not use this as your actual settings):

   — Data source name: **demoAdxDataSource**
   — Cluster URI: https://demoadxcluster.japaneast.kusto.windows.net

   Click **Connect.**

   — Database: **demoAdxDatabase**
   — Query results cache max age: **Disabled**

   Click **Apply.**

4. Replace sample KQL (Kusto SQL) commands with following KQL commands.

```
['demoAdxTable']
| where deveui == "749050FFFE000C26"
| take 10
```

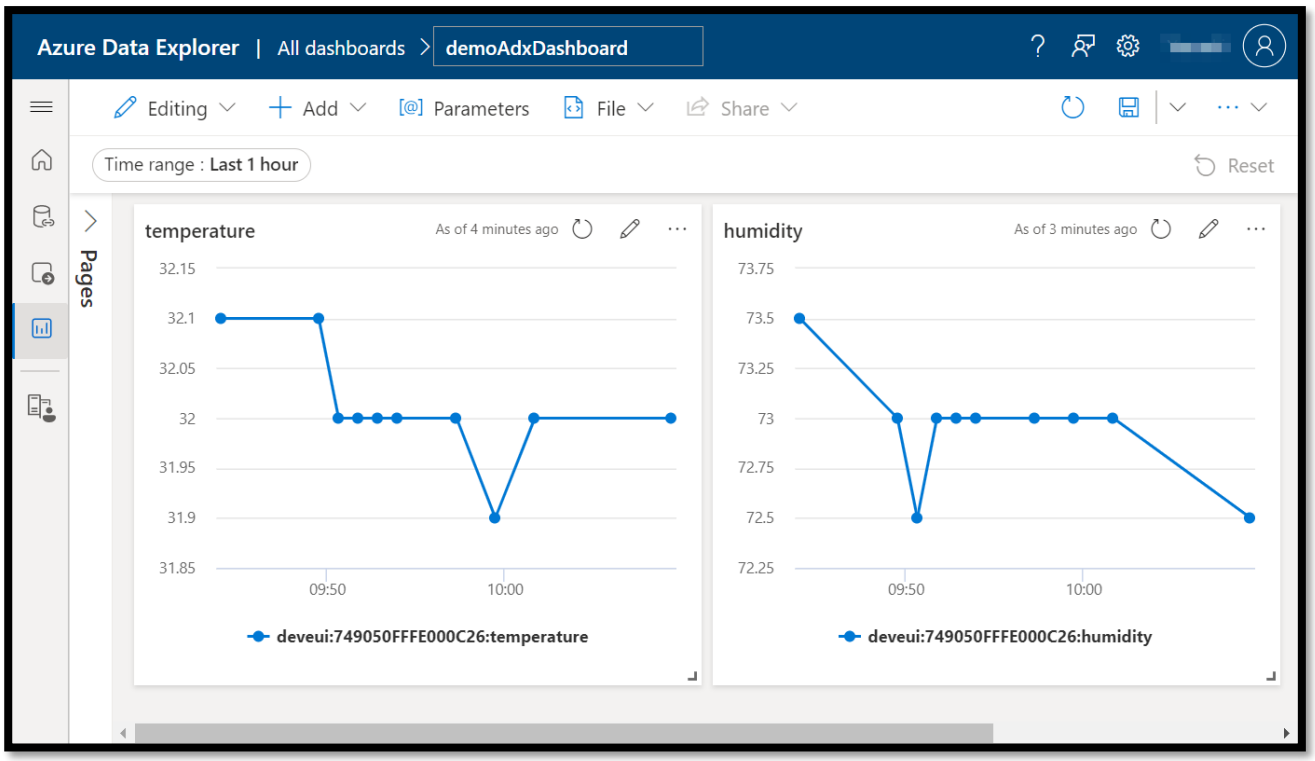
Click **Run**> click **+Add Visual.**

- Tile name: **temperature**
- Visual type: **Line chart**
- Data Y Columns: **temperature (double)**

Click **Apply changes**.

5. Click **+Add** > Repeat step 4 for humidity.

The following chart will be displayed.



Revision History

Rev.	Date	Description	
		Page	Summary
2.00	Aug.29.22	All	Initial version for RA2E1
2.10	Nov.29.22	1 5	Changed supported IDEs and toolchains. Corrected a paragraph error.
2.20	Mar.31.23	1 1, 3, 4	Changed supported IDEs and toolchains. Supports RA2L1.
2.40	Dec.22.23	1 All	Changed version of development tools GCP support has been discontinued.
2.50	May.24.24	1 1, 3, 4 5 All	Changed version of development tools. Supports RA0E1. Changed reference link of the Cayenne LPP 2.0 format. Cayenne services has been discontinued.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.

Trademarks

Arm® and Cortex® are registered trademarks of Arm Limited. Semtech, the Semtech logo, LoRa, LoRaWAN and LoRa Alliance are registered trademarks or service marks, or trademarks or service marks, of Semtech Corporation and/or its affiliates. Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.