Green Grass Demo:

Vineyard / Agricultural IIoT

Setup Guide

Table of Contents

Revision History:

Revision	Date	Changes
0.1	08-Nov-2017	Initial Version

Introduction

Scope:

Smart Vineyard System built on top of DragonBoard™ 410c from Arrow Electronics integrated with sensors to Monitor and Optimize growing conditions by capturing data and processing it on the edge gateway, and making real time decisions about irrigation, grape temperature for frost prevention.

The data collected from the sensors shall be stored on the cloud and analytics performed on the data on hourly, daily and long term including the quality and Quantity of wine grapes.

Hardware and Software Requirements

Hardware components:

- DragonBoard™ 410c from Arrow Electronics with power supply.
- Host machine(Linux).
- MicroSD card with 8GB or more of storage.
- USB Mouse and keyboard.
- HDMI Monitor with HDMI cable and HDMI connector.
- Sensor Mezzanine board.
- Grove Moisture sensor v1.4, Grove Digital light sensor v1.1, Grove 10DoF IMU v2.0, Grove Mini fan v1.1, Grove Chainable RGB LED v2.0.

Software Components:

- Linaro aarch64 - Please follow the below instruction to install Linaro OS on dragon board 410c.

Installation of Linaro OS on Dragon-Board 410c

Download SD Card image:

To download “SD Card Image–Install and boot from eMMC” click here.

Choose Host Machine(Linux):

1. Prepare MicroSD Card,
    a. Ensure data from microSD card is backed up.
    b. Everything on microSD card will be lost by the end of this procedure.

2. Find SD Card Device name,
    a. Use host computer.
    b. Open “Terminal” application from your host machine or press `Ctrl+Alt+t`
    c. Remove SD card from host computer and run the following command from your terminal,

```
$ lsblk
```

    d. Note all recognized disk names.
    e. Insert SD card and run the following command (again),

```
$ lsblk
```

    f. Note the newly recognized disk. This will be your SD card.
    g. Remember your SD card device name, it will be needed in Step 7.

3. Recall Download Location,
    a. Locate SD card install file from Downloads page.
    b. This file will be needed for the next step.

4. Unzip SD Card Install Image,
    a. Command for Unzip file:

```
$ unzip <filename>
```

    b. When unzipped, you will have a folder with the following contents:
        i. Install Image(.img)
        ii. License.txt

5. Go to directory with SD Card Install Image folder using Terminal,
    a. Open “Terminal” application from your host computer.
    b. Change to the directory where you unzipped SD Card Install Image,

```
$ cd /home/username/Downloads
```

6. Locate SD Card Install Image,
   a. Make sure you are in the extraction directory.
   b. Inside this folder you will find install image "**db410c_sd_install_debian.img**".
   c. This .img file is what will be flashed to your SD Card.

7. Install Image onto SD Card. Checklist,
   a. SD card inserted into host computer.
   b. Recall SD Card device name from Step 2
   c. Within that downloaded folder you will find "**db410c_sd_install_debian.img**" and from terminal run the below command.

```
$ sudo dd if=db410c_sd_install_debian.img of=/dev/XXX bs=4M
oflag=sync status=noxfer
```

NOTE : Syntax for the command above,

```
$ sudo dd if=<name of image downloaded> of=/dev/<SD Card
deviceName without partition> bs=4M oflag=sync status=noxfer
```

   d. This command will take some time to execute. Be patient and avoid tampering with the terminal until process has ended.
   e. Once SD card is done flashing, remove from host computer.

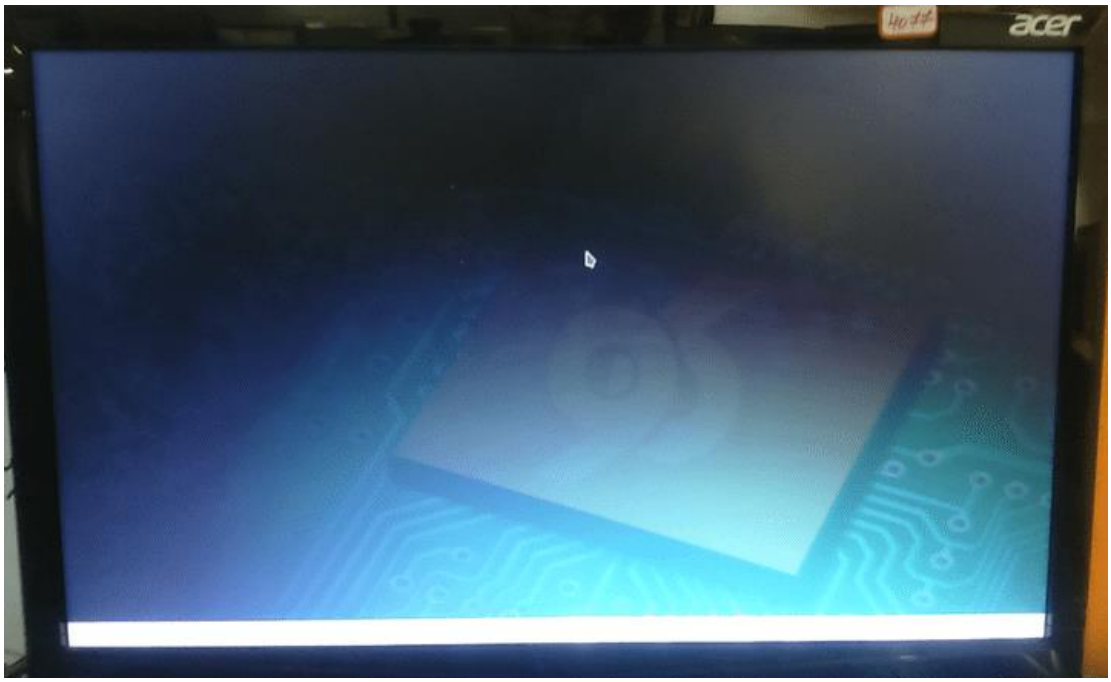
8. Prepare Dragonboard 410c with SD card,
   a. Make sure Dragonboard™410c is unplugged from power.
   b. Set S6 switch on Dragonboard™410c to **0-1-0-0**, "SD Boot switch" should be set to "**ON**". See S6 switch on the below diagram:

   c. Connect an HDMI monitor to the Dragonboard™410c with an HDMI cable, and power on the monitor.
   d. Plug a USB keyboard and/or mouse into either of the two USB connectors on the Dragonboard™410c.
   e. Insert the microSD card into the Dragonboard™ 410c.
   f. Plug power adaptor into Dragonboard™ 410c, wait for board to boot up.

9. Install image onto Dragonboard 410c. If Steps 1 - 8 were followed correctly, the below screen should be visible from your Dragonboard™410c,

   a. Select the image to install and click “Install icon”. OS will be installed into the eMMC memory.
   b. This process can take a few minutes to complete.
   c. Upon completion, “Flashing has completed and OS has installed successfully….” message will appear.
   d. **Before clicking “OK”**, Remove the SD Card.
   e. Set S6 switch on Dragonboard™ 410c to **0-0-0-0**, all switches should be set to “**OFF**”.
   f. Now click “**OK**” button and allow Dragonboard™410c to reboot.

g. It takes some time to reboot, After Some time, if screen is not appeared, then plug it off and again replug it of Dragonboard, After that Below Screen will appear.



Installation of Greengrass on Dragon-Board 410c

Run AWS Greengrass core software:

Enable symlink and hardlink protection:

The Debian release for the DragonBoard 410c has symlink and hardlink protection disabled by default. The following creates **sysctl** configuration file with the options to enable the protection. Open a terminal application from your dragonboard 410c or press **Ctrl+Alt+t** and type the below command,

```
$ sudo sh -c 'echo "fs.protected_hardlinks = 1" >> /etc/sysctl.d/local-ggc.conf'
$ sudo sh -c 'echo "fs.protected_symlinks = 1" >> /etc/sysctl.d/local-ggc.conf'
```

Reboot the system:

```
$ sudo reboot
```

After rebooting your Dragonboard 410c, you can verify the protection is working by checking:

```
$ sudo cat /proc/sys/fs/protected_{hardlinks,symlinks}
```

NOTE : You should see two 1s.

Use the following command to add a user for greengrass. Open a terminal from your dragon-board 410c or press **Ctrl+Alt+t** and type the below command,

```
$ sudo adduser --system ggc_user
$ sudo addgroup --system ggc_group
$ sudo apt-get update
```

Use the following command to install sqlite3,

```
$ sudo apt-get install sqlite3
```

Use the following command to install cmake,

```
$ sudo apt-get install cmake
```

Install AWS CLI on Linux (Dragonboard 410c):

Open your terminal application:-

1. Check your Python installation,

```
$ python --version
```

2. Download the AWS CLI Bundled installer from below command,

```
$ curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o
"awscli-bundle.zip"
```

3. Unzip the package,

```
$ unzip awscli-bundle.zip
```

4. Run the install executable,

```
$ sudo ./awscli-bundle/install -i /usr/local/aws -b
/usr/local/bin/aws
```

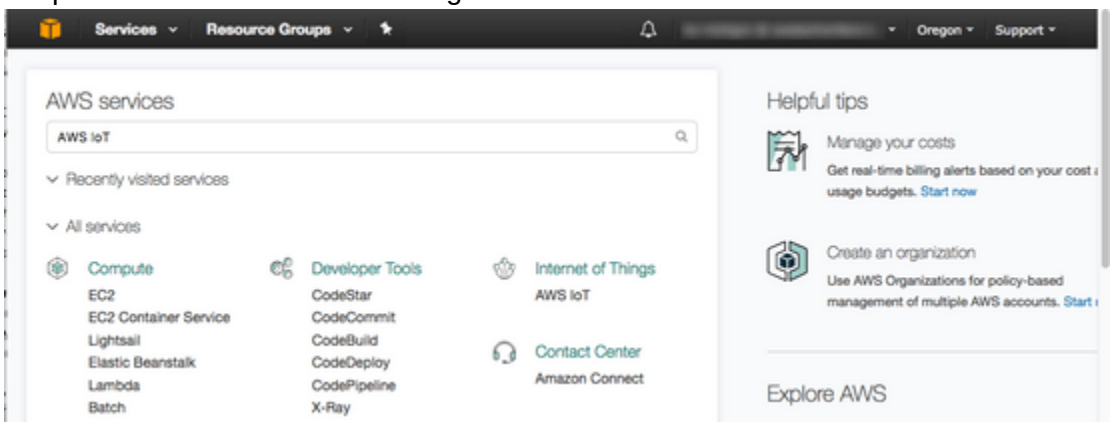
5. Reboot the system, open your terminal in dragon-board 410c and type,

```
$ sudo reboot
```

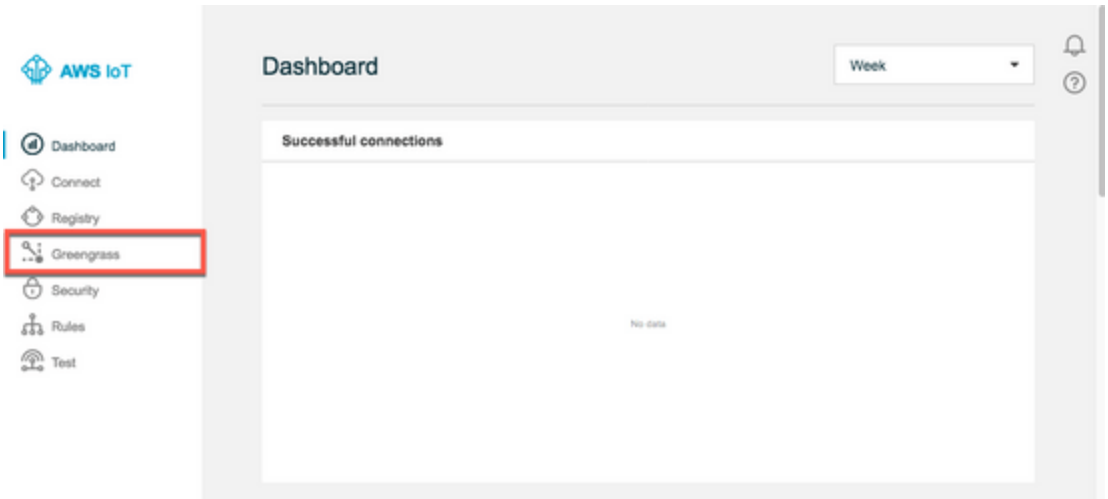
Create AWS Greengrass Group and AWS Greengrass Core:

Assuming you have AWS account.

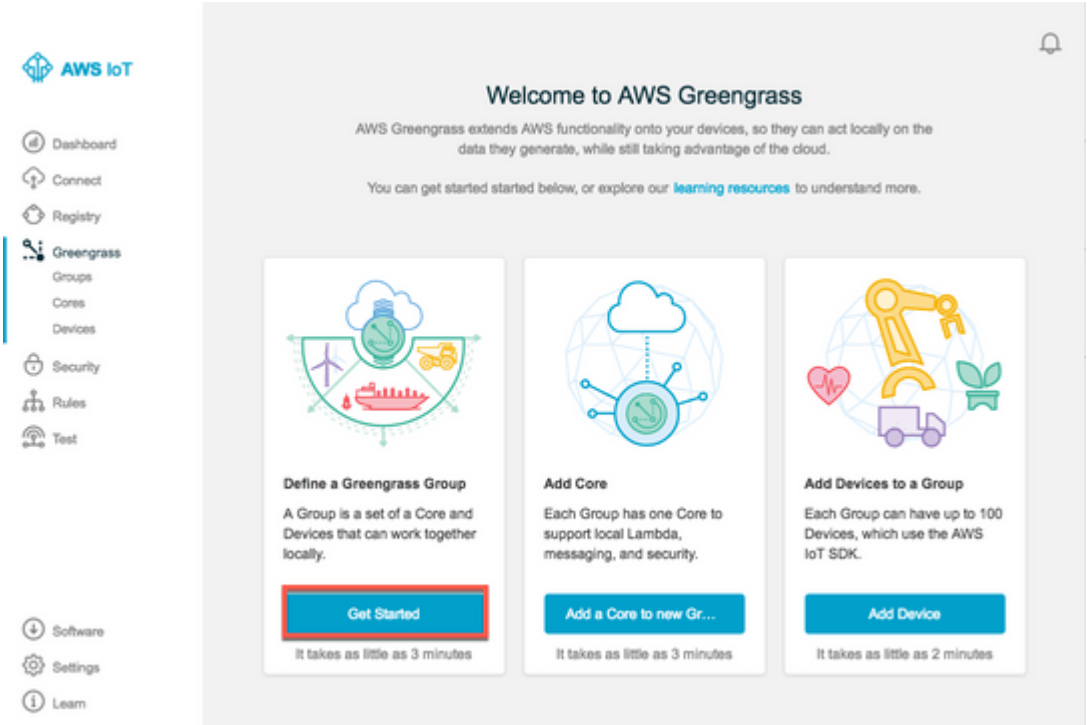
1. Sign in to the AWS Management console.
2. Open the AWS IoT console using the console search.



3. From the navigation pane, choose **Greengrass**.

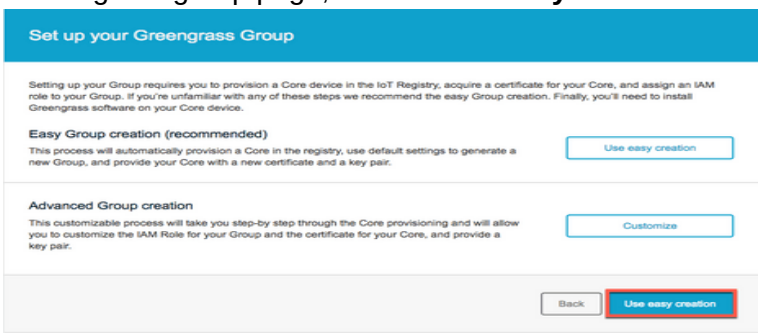


4. On the AWS Greengrass page, choose **Get Started**.

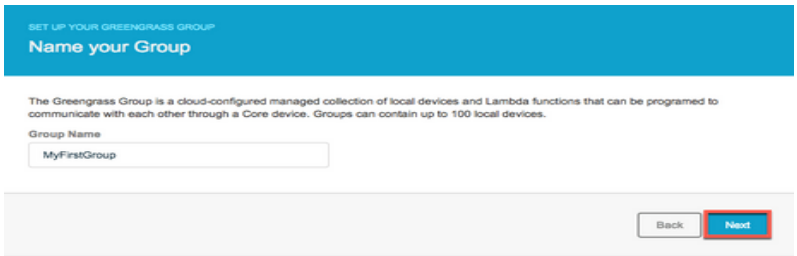


Create a Greengrass Group:

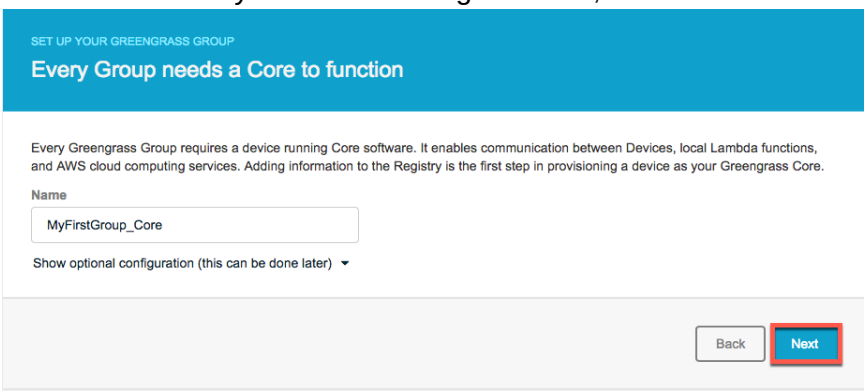
On the Set up your Greengrass group page, choose **Use easy creation**



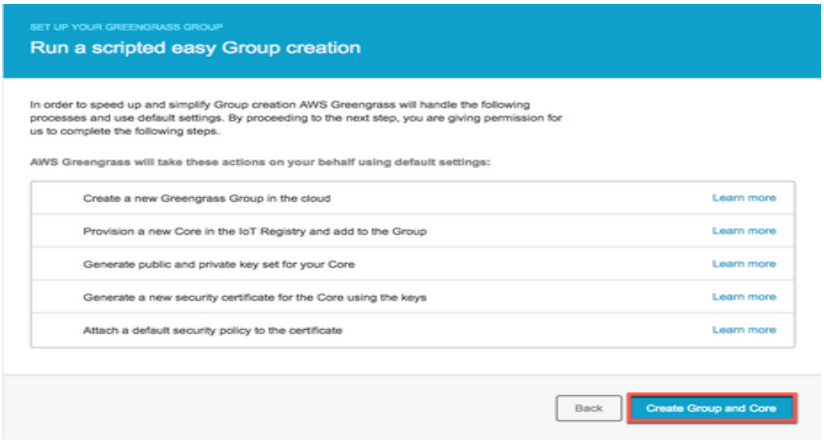
1. Type a name for your group, and then choose **Next**.



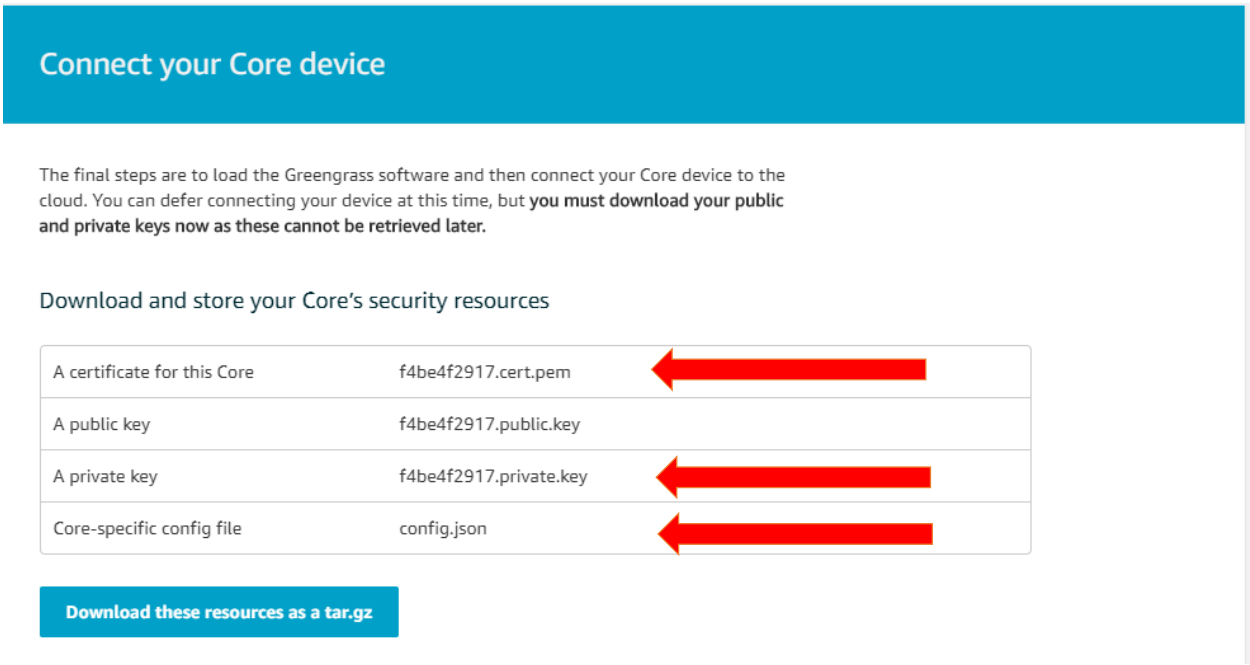
2. Use the default name for your AWS Greengrass core, and then choose **Next**.



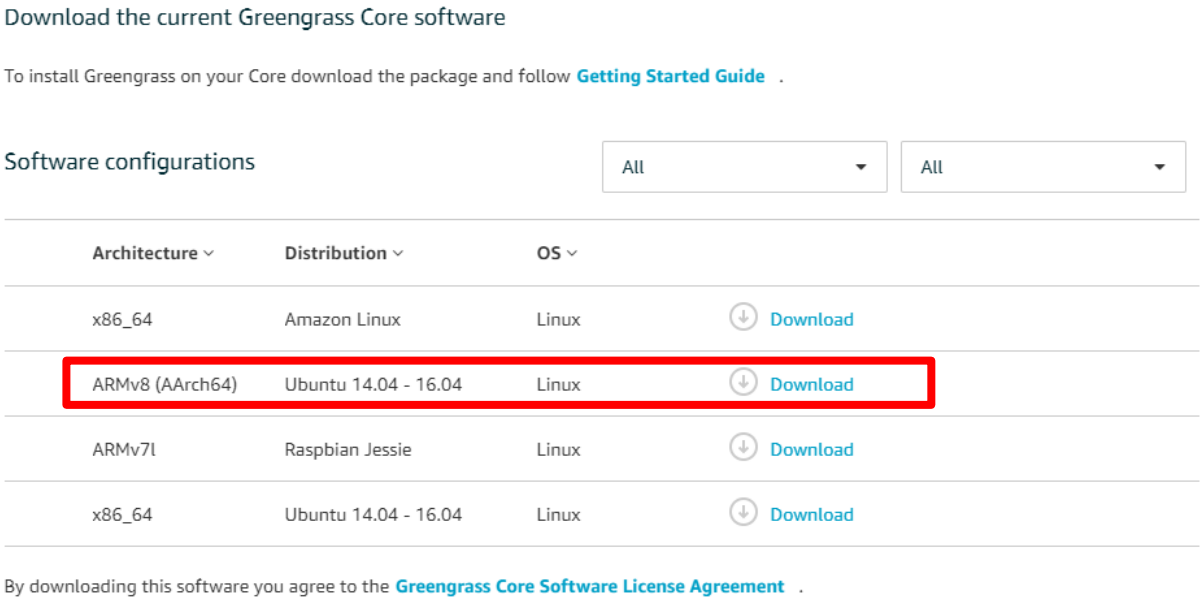
3. Choose **Create Group and Core**.



4. Click the links to download the private key, public key, and certificate for your AWS Greengrass core.



5. Choose the AArch64(ARMv8)(for your dragonboard 410c)and then choose **Download Greengrass** to download the Greengrass software package. When the download is complete, choose **Finish**.



Copy and Extract the Greengrass Software Package onto your core device(Dragon-board-410c):

1. Determine the IP address of the device where you will install the AWS Greengrass core software (Dragonboard 410c).
2. The name of Greengrass software package, "**greengrass-platform-version.tar.gz**"
3. Use the scp command to copy the Greengrass software package into the home directory of your AWS Greengrass core device. (Use the IP_ADDRESS of your Greengrass core device).

```
$ scp greengrass-platfrom-version.tar.gz linaro@IP_ADDRESS:~
```

4. Similarly copy the all the certificates & key into the AWS GreenGrass core device (Dragonboard 410c).
5. On your AWS Greengrass core device (Dragonboard 410c), use the following command to expand the package,

```
$ sudo tar -zxvf greengrass-<os-name>-<platform-type>-
<version>.tar.gz -C /
```

Install Certificates on your AWS Greengrass Core Device (Dragon-board 410c):

1. Download the AWS IoT root CA certificate from Symantec/Version. You can download from command-line of host machine,

```
$ wget
https://www.symantec.com/content/en/us/enterprise/verisign/roots/Veri
Sign-Class%203-Public-Primary-Certification-Authority-G5.pem
```

2. Move downloaded file into AWS GreenGrass core device (Dragonboard 410c) using scp command, (Use Dragonboard 410c ipaddress)

```
$ scp VeriSign-Class\ 3-Public-Primary-Certification-Authority-G5.pem
linaro@IP_ADDRESS:~
```

3. Move the file in device to /greengrass/certs directory. Command as follows,

```
$ sudo mv VeriSign-Class\ 3-Public-Primary-Certification-Authority-
G5.pem /greengrass/certs/root-ca.pem
```

4. Copy the AWS Greengrass core's private key and certificate and the AWS IoT root CA certificate into the /greengrass/configuration/certs directory of your AWS Greengrass core device. Use the following names,

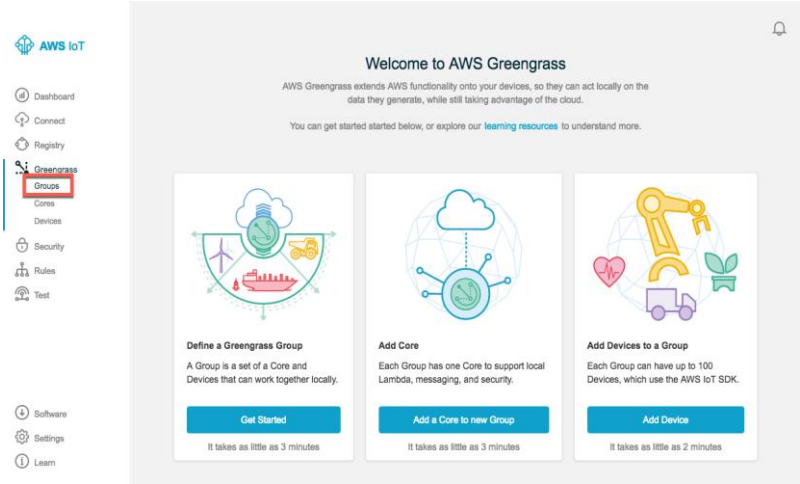
File	Description
cloud.pem.crt	The AWS Greengrass core certificate downloaded from the console (for example, 61970a435c-certificate.pem.crt or b01d32b3f6.cert.pem).
cloud.pem.key	The private key downloaded from the console (for example,61970a435c-private.pem.key or b01d32b3f6.private.key).
root-ca.pem	The AWS IoT root CA certificate from Symantec.

NOTE: You can use the scp command to copy the files to your AWS Greengrass core device. By default, you do not have permission to copy the files directly to greengrass directory. To work around this, copy the files to your user directory and then use move the files to the /greengrass/configuration/certs directory.

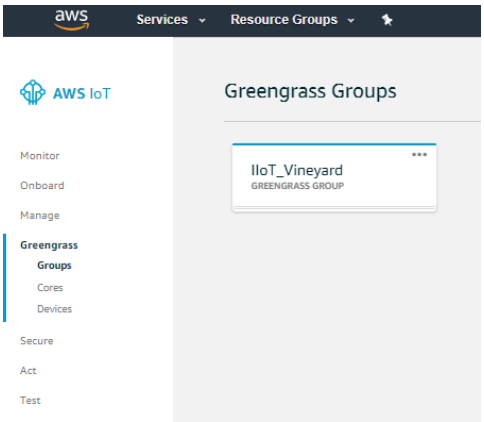
Configure your Greengrass core:

You need to associate your AWS Greengrass core with the AWS IoT thing you created for it. You do that by using the thing's ARN, the unique identifier assigned in the cloud to your AWS Greengrass core device when you provisioned it.

1. Find the thing ARN for your AWS Greengrass core. In the AWS Greengrass console, from the navigation page, choose **Groups**.

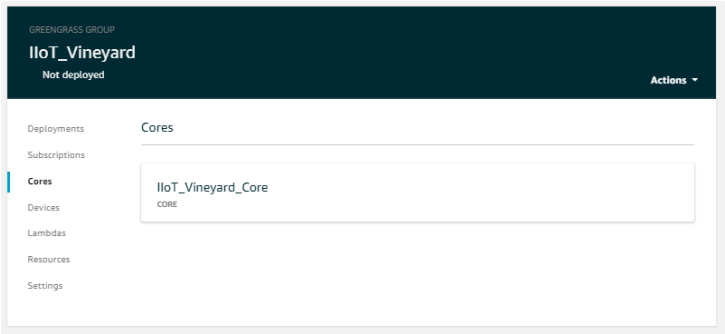


2. Choose your group to display its detail page.

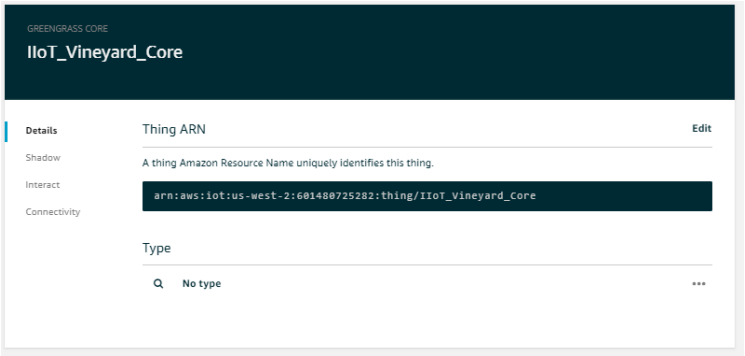


3. In the navigation pane, choose **Cores**.

4. Choose your AWS Greengrass core to display its detail page.



5. Copy the ARN of your AWS Greengrass core.



6. On your AWS Greengrass core device, open your terminal do following,

```
$ sudo vim /greengrass/configuration/config.json
```

Initially it will look like this,

```
    {
      "coreThing": {
        "caPath": "<ROOT_CA_PEM_HERE>",
        "certPath": "<CLOUD_PEM_CRT_HERE>",
        "keyPath": "<CLOUD_PEM_KEY_HERE>",
        "thingArn": "<THING_ARN_HERE>",
        "iotHost":
"<HOST_PREFIX_HERE>.iot.<AWS_REGION_HERE>.amazonaws.com",
        "ggHost": "greengrass.iot.<AWS_REGION_HERE>.amazonaws.com",
        "keepAlive": 600
    },
    "runtime": {
        "cgroup": {
            "useSystemd": "[yes|no]"
        }
    }
}
```

Update the file with following content,

```
{
  "coreThing": {
    "caPath": "root-ca.pem",
    "certPath": "cloud.pem.crt",
    "keyPath": "cloud.pem.key",
    "thingArn": "arn:aws:iot:us-west-
2:601480725282:thing/IIoT_Vineyard_Core",
    "iotHost": "your-AWS-IoT-endpoint,
    "ggHost": "greengrass.iot.us-west-2.amazonaws.com",
    "keepAlive": 600
  },
  "runtime": {
    "cgroup": {
      "useSystemd": "yes"
    }
  }
}
```

Attribute	Description
caPath	The path to the AWS IoT root CA certificate.
certPath	The path to the AWS Greengrass core certificate.
keyPath	The path to the AWS Greengrass core private key.
thingArn	The ARN of the AWS IoT thing associated with your AWS Greengrass core.
iotHost	Your AWS IoT endpoint. Can be obtained using the aws iot describe-endpoint CLI command or in the **Settings** section of the AWS IoT console.
ggHost	A fully qualified AWS Greengrass host address. Replace AWS_REGION with the AWS region you are using.

7. Start Your AWS Greengrass Core to connect It to the Cloud.
8. Change to /greengrass directory and run the following command to start your core and enable a cloud connection to AWS IoT.

```
$ sudo ./greengrassed start
```

If the Greengrass daemon started successfully, you should see the following output:

```
Setting up greengrass daemon
Validating execution environment
ggc_group:x:119:
Found cgroup subsystem: cpu
Found cgroup subsystem: cpuacct
Found cgroup subsystem: blkio
Found cgroup subsystem: memory
Found cgroup subsystem: devices
Found cgroup subsystem: freezer
Found cgroup subsystem: net_cls
Starting greengrass daemon
PID: 1306
Greengrass daemon started
```

If you see the error like, **The cgroup subsystem is not mounted: cpuset**
This means cgroup is not mount, to do you need to run a script. Click here for script.

Install Library to use GPIO of the 96Boards

Update the installed image:

Open your terminal from 96Boards and type below command:

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get dist-upgrade
```

Install Package dependencies:

```
$ sudo apt-get install git build-essential autoconf automake libtool
swig3.0 python-dev nodejs-dev cmake pkg-config libpcre3-dev
$ sudo apt-get clean
```

Install 96Boards configuration files:

```
$ git clone https://github.com/96boards/96boards-tools.git
$ sudo cp 96boards-tools/70-96boards-common.rules /etc/udev/rules.d/
$ sudo vim /etc/profile.d/96Boards-sensor.sh
```

Append the following lines into the above opened file,

```
export JAVA_TOOL_OPTIONS="-Dgnu.io.rxtx.SerialPorts=/dev/tty96B0"
export MONITOR_PORT=/dev/tty96B0
export PYTHONPATH="$PYTHONPATH:/usr/local/lib/python2.7/site-packages"
```

Copy the file,

```
$ sudo cp /etc/profile.d/96boards-sensors.sh /etc/X11/Xsession.d/96boards-
sensors
```

Install and build libsoc:

From your terminal type below command:

```
$ cd /home/linaro
$ git clone https://github.com/jackmitch/libsoc.git
$ cd libsoc
$ autoreconf -i
$ ./configure --enable-python2 --enable-board=dragonboard410c
$ make
```

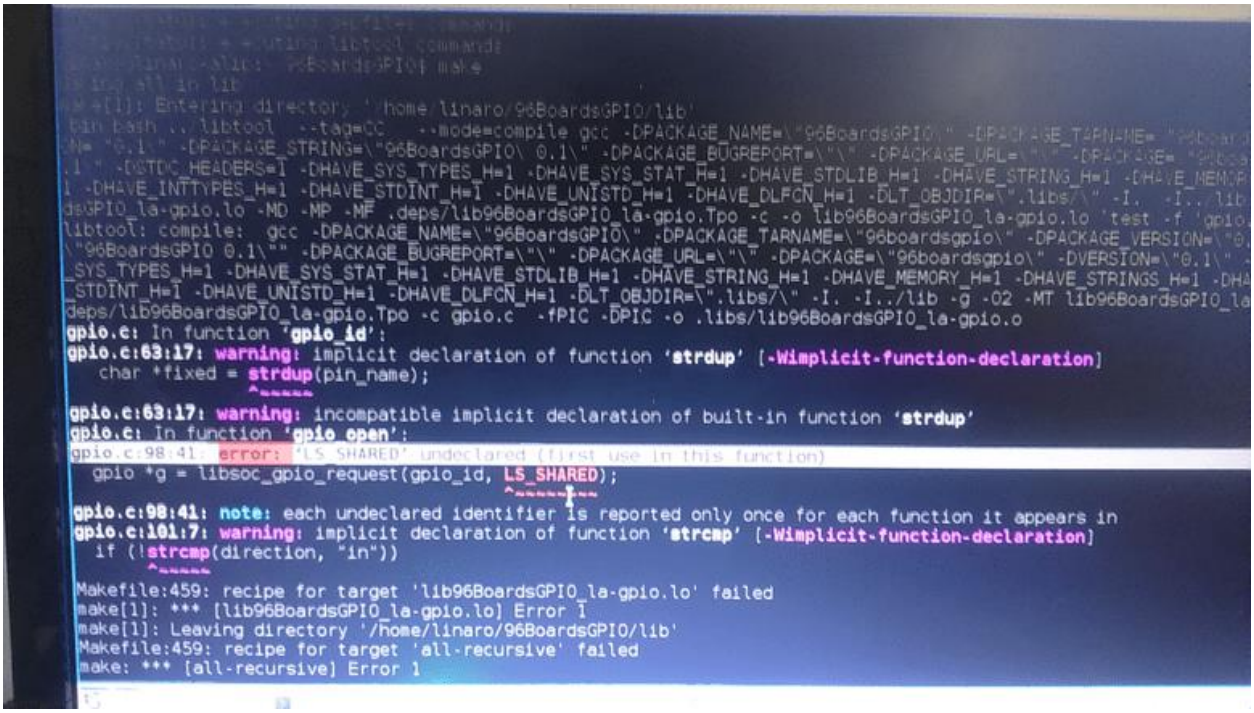
```
$ sudo make install
$ sudo ldconfig /usr/local/lib
```

Install and build 96BoardsGPIO:

This library requires libsoc to be installed first. From your terminal type below command:

```
$ cd /home/linaro
$ git clone https://github.com/96boards/96BoardsGPIO.git
$ cd 96BoardsGPIO
$ ./autogen.sh
$ ./configure
$ make
```

In case if the make command it shows the following error:



Please run the following procedure to fix,

```
$ cd /home/linaro/96BoardsGPIO/lib
```

In this Directory you will see **gpio.c** file. In that file go to line number **98** and modify as follow:

```
$ sudo vim gpio.c +98
```

before changing line inside file(gpio.c), It should looks like as below:

```
gpio *g = libsoc_gpio_request(gpio_id, LS_SHARED)
```

After changing line inside file(gpio.c), It should looks like as below:

```
gpio *g = libsoc_gpio_request(gpio_id, LS_GPIO_SHARED)
```

Add Header file in gpio.c for removing the warnings:

```
#include <string.h>
```

save the gpio.c file and proceed with following command,

```
$ cd /home/linaro/96BoardsGPIO
$ make
$ sudo make install
$ sudo ldconfig /usr/local/lib
```

Reset the system,

```
$ sudo reboot
```

Install extra tool packages:

From your terminal application (Dragonboard 410c) type below command for **ARDUINO IDE**,

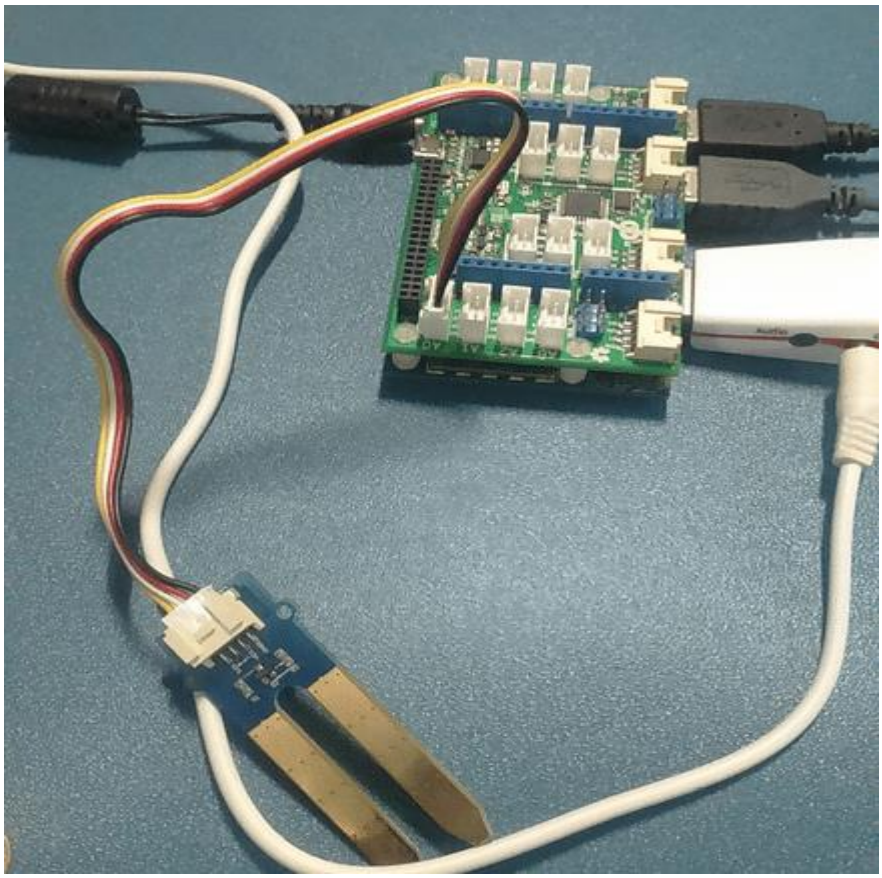
```
$ sudo apt-get install arduino-mk arduino git build-essential autoconf
libtool swig3.0 python-dev nodejs-dev cmake pkg-config libpcre3-dev
$ sudo apt-get clean
```

Attach Sensor Mezzanine Board to Dragon-Board 410c

The following figure shows how to attach both board,

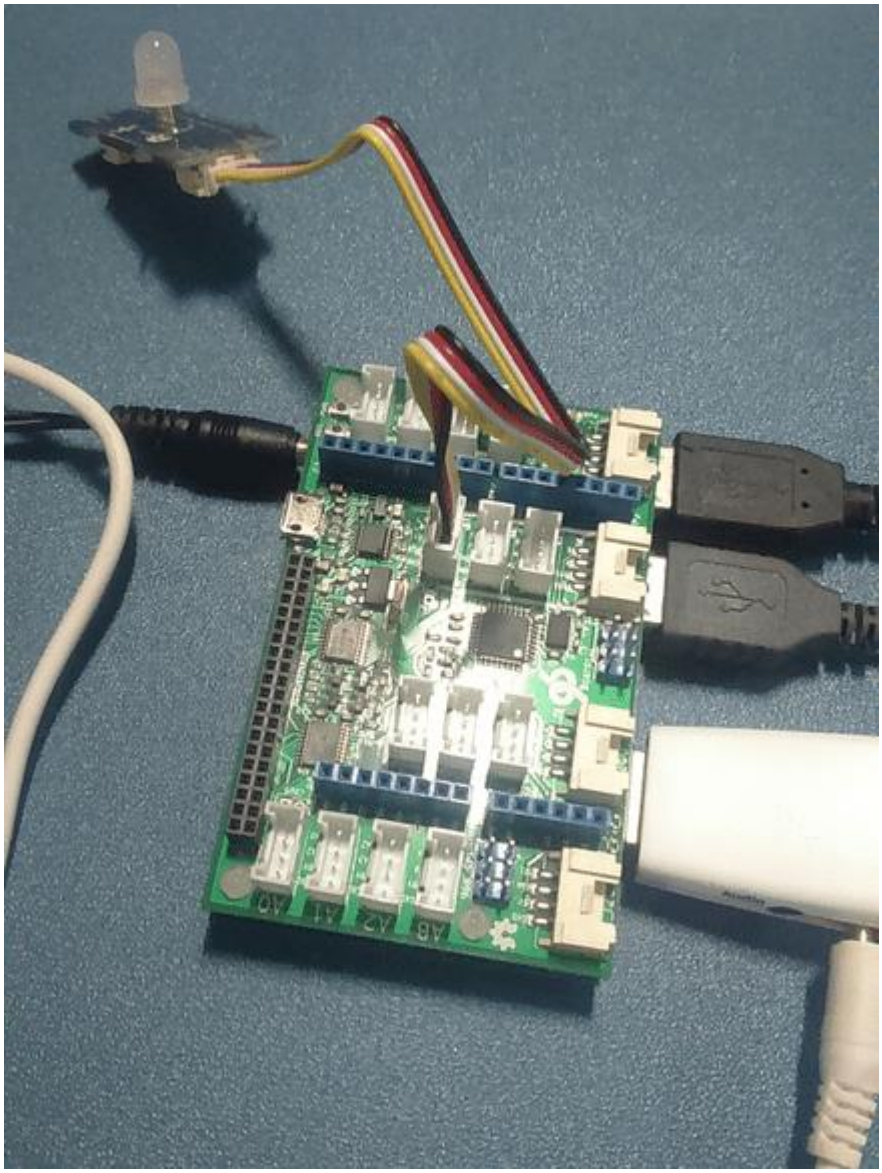
Connect Grove Moisture Sensor:

The figure shows how to connect grove moisture-sensor to A0 pin of seeed Sensor Mezzanine for 96Boards.



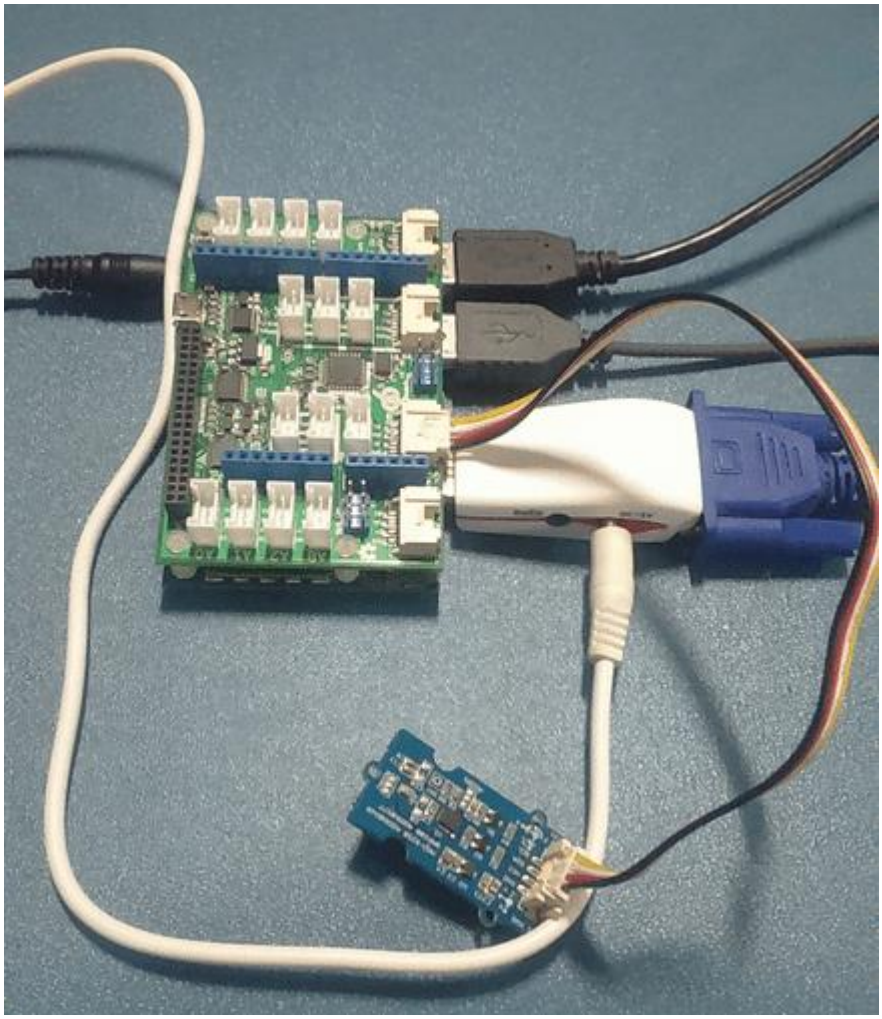
Connect Grove RGB LED Sensor:

The figure shows how to connect grove RGB_LED sensor to KL port pin of seeed Sensor Mezzanine for 96Boards.



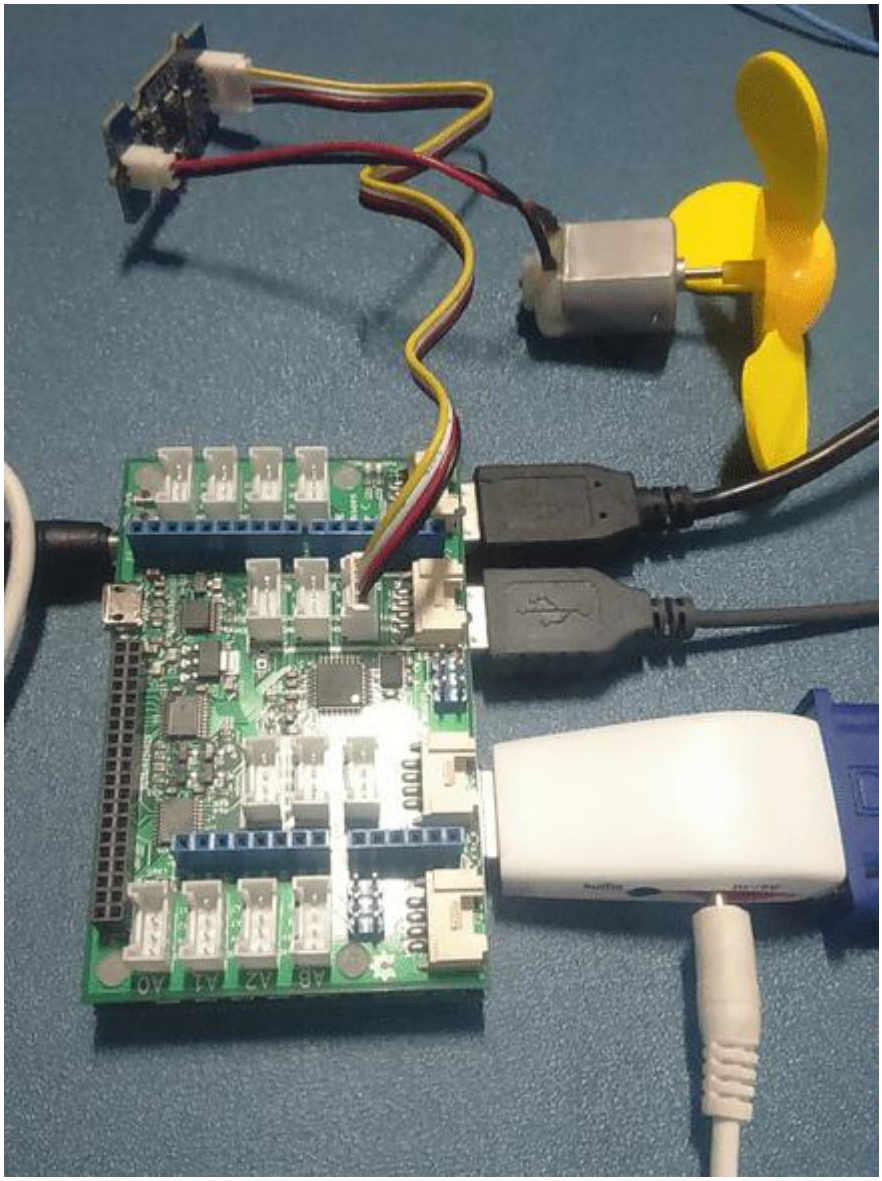
Connect Grove 10DoF IMU Sensor:

The figure shows how to connect grove 10DoF IMU sensor to i2c port(baseboard) of seeed Sensor Mezzanine for 96Boards.



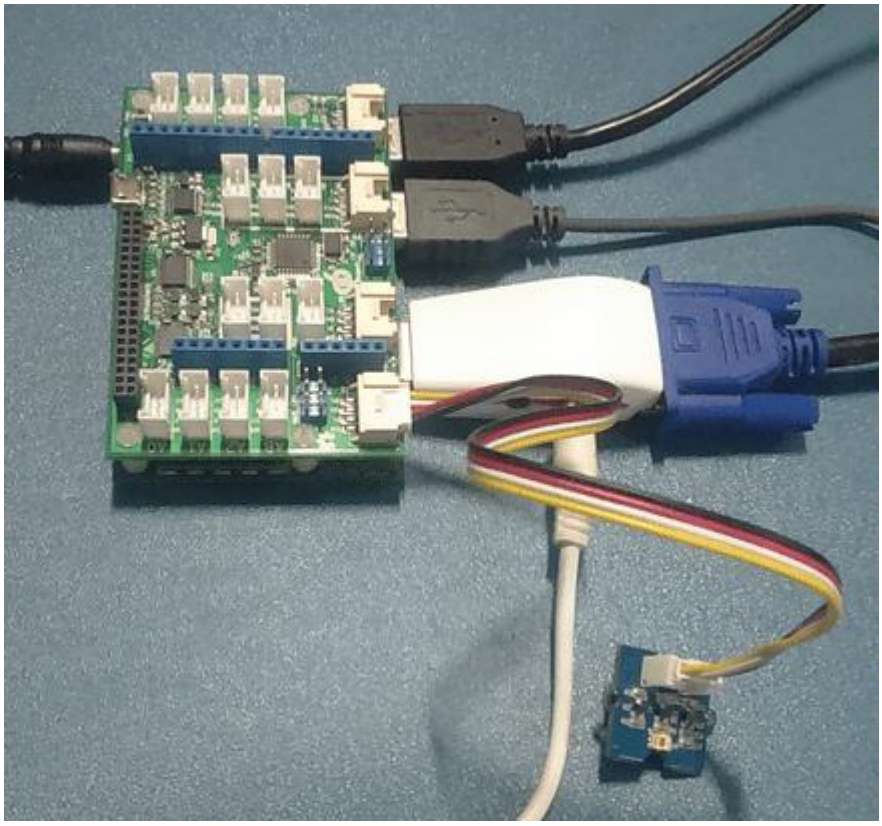
Connect Grove Mini Fan Sensor:

The figure shows how to connect grove Connect grove mini fan sensor to GH port of seeed Sensor Mezzanine for 96Boards.

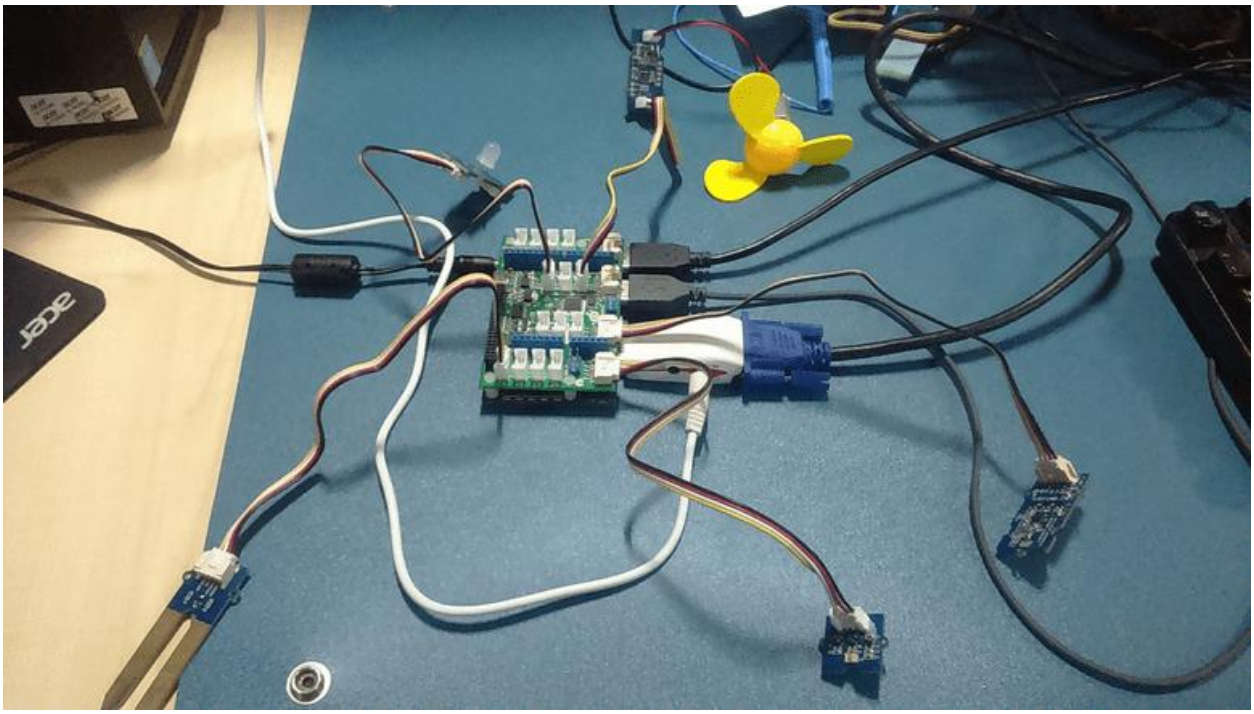


Connect Grove Digital Light Sensor:

The figure shows how to connect grove Connect grove digital light sensor to i2c port of seeed Sensor Mezzanine for 96Boards.



Connect All Sensor together:



Code Dump in Mezzanine Board

1. To download library click here.
2. Unzip the downloaded file,

```
$ cd /home/linaro/Downloads
$ unzip Grove_Digital_Light_Sensor-master.zip
```

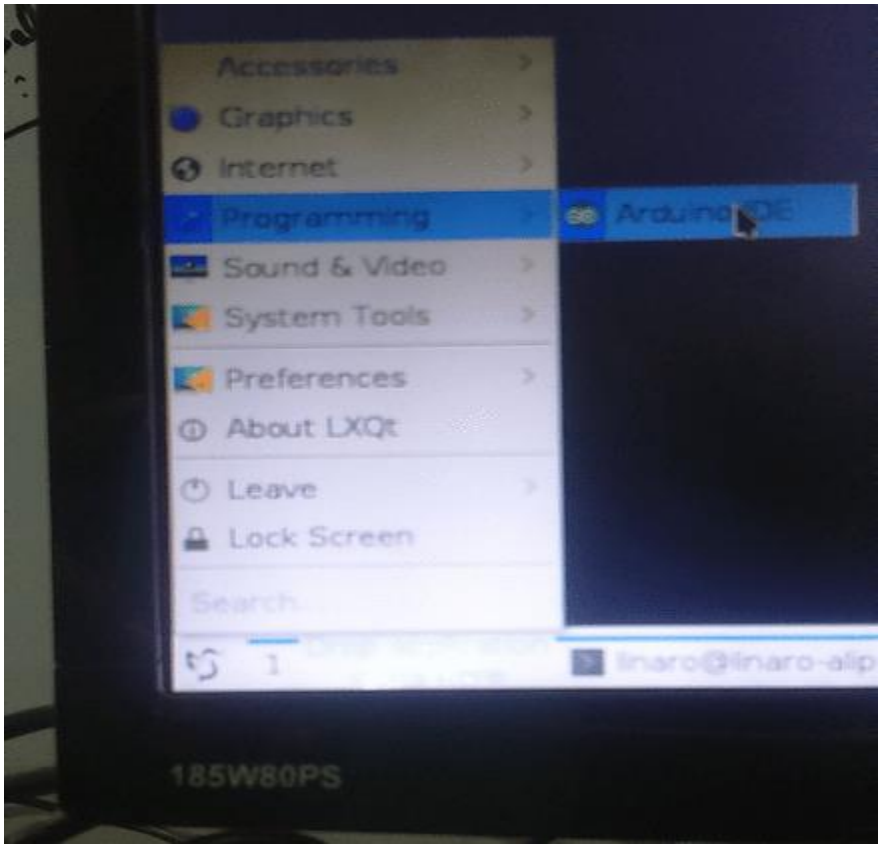
3. After unzip, rename the folder with Digital_Light_TSL2561(because In Arduino library its support file with name & number).

```
$ sudo mv Grove_Digital_Light_Sensor-master Digital_Light_TSL2561
```

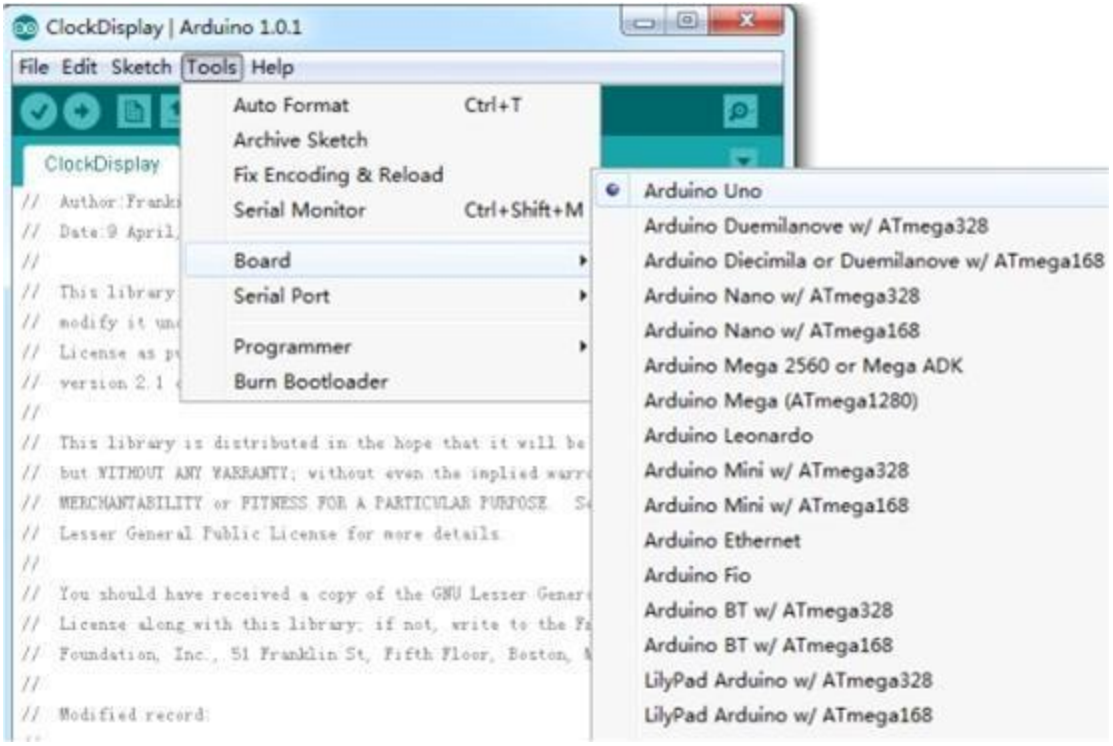
4. Place Digital_Light_TSL2561 library in /usr/share/arduino/libraries from your terminal application.

```
$ sudo cp -r Digital_Light_TSL2561 /usr/share/arduino/libraries
```

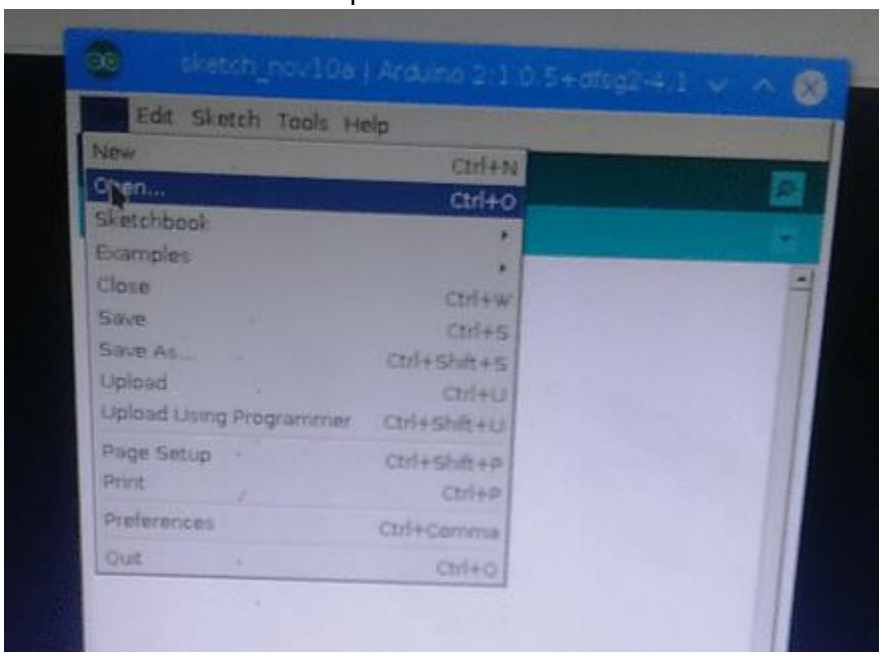
5. Open Arduino IDE.



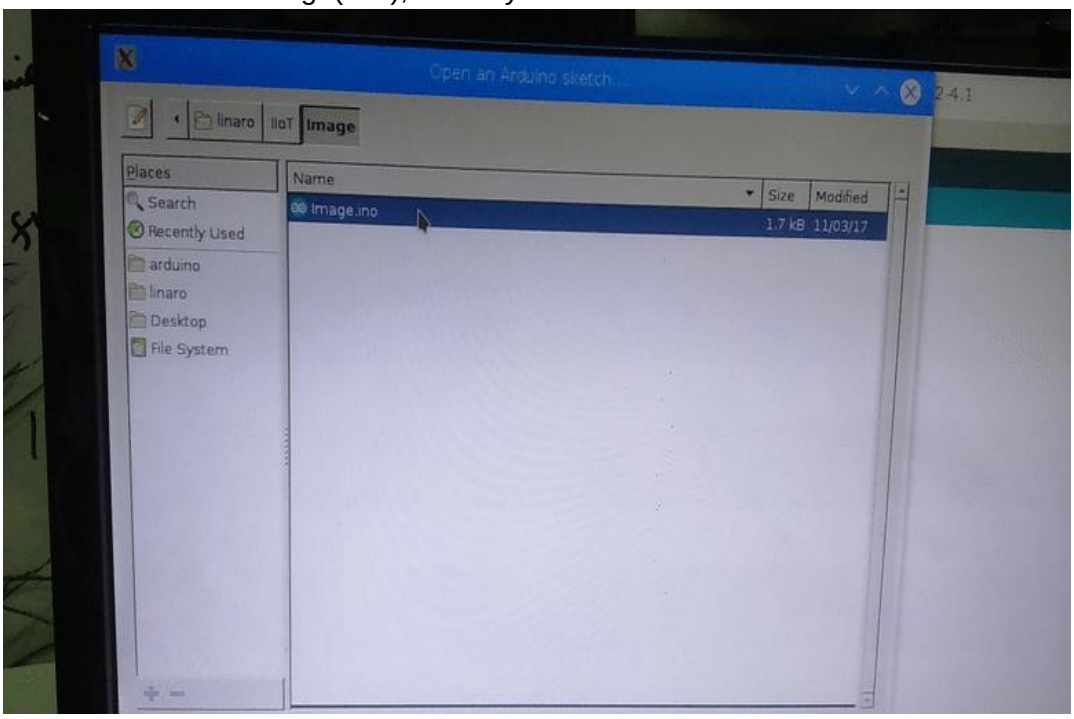
6. Select correct board from tools (in this case select Arduino Uno).



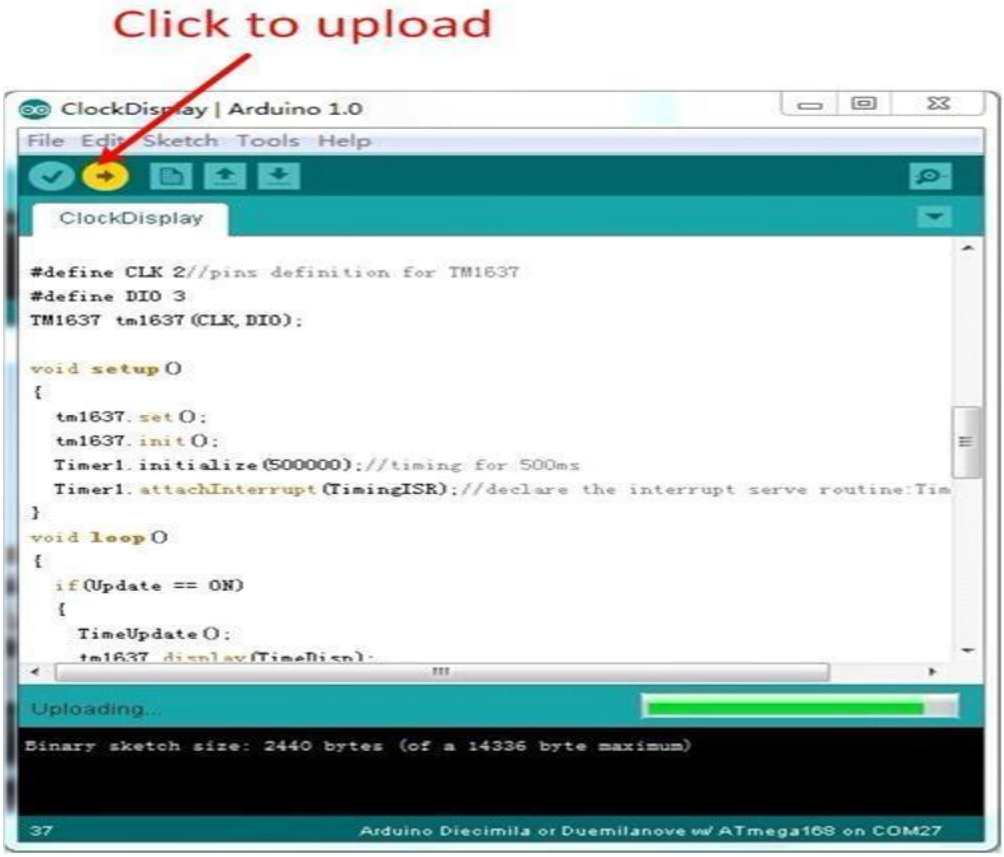
7. Click on ‘files’ and then click on ‘open’.



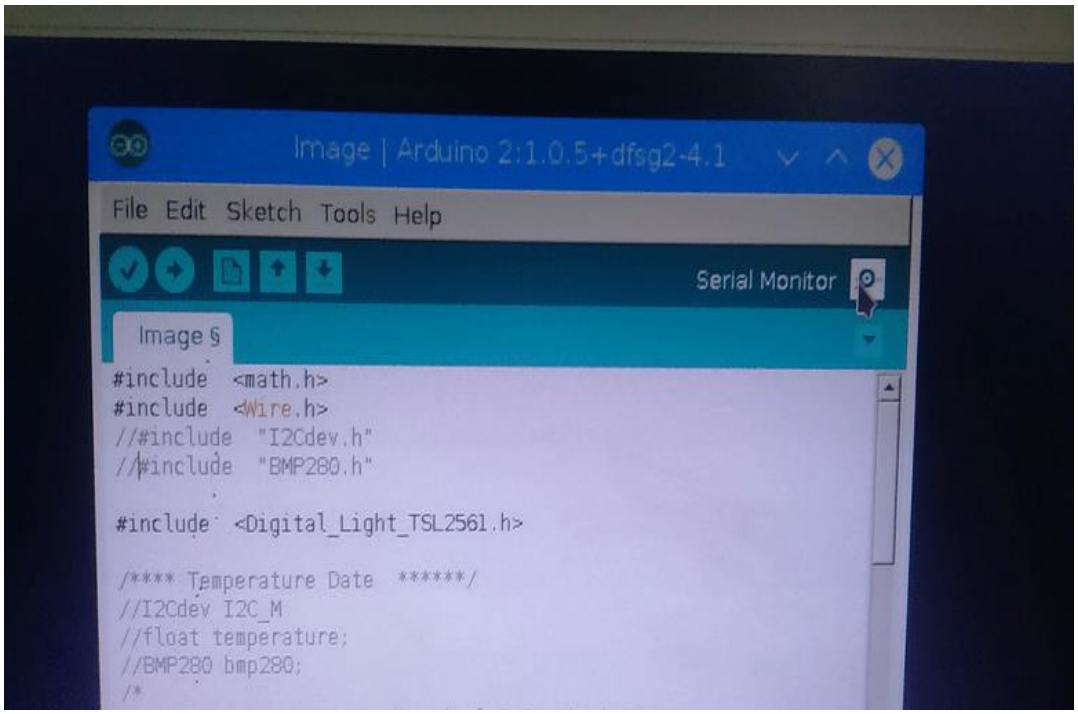
8. Select the arduino image(.ino), where your code located.



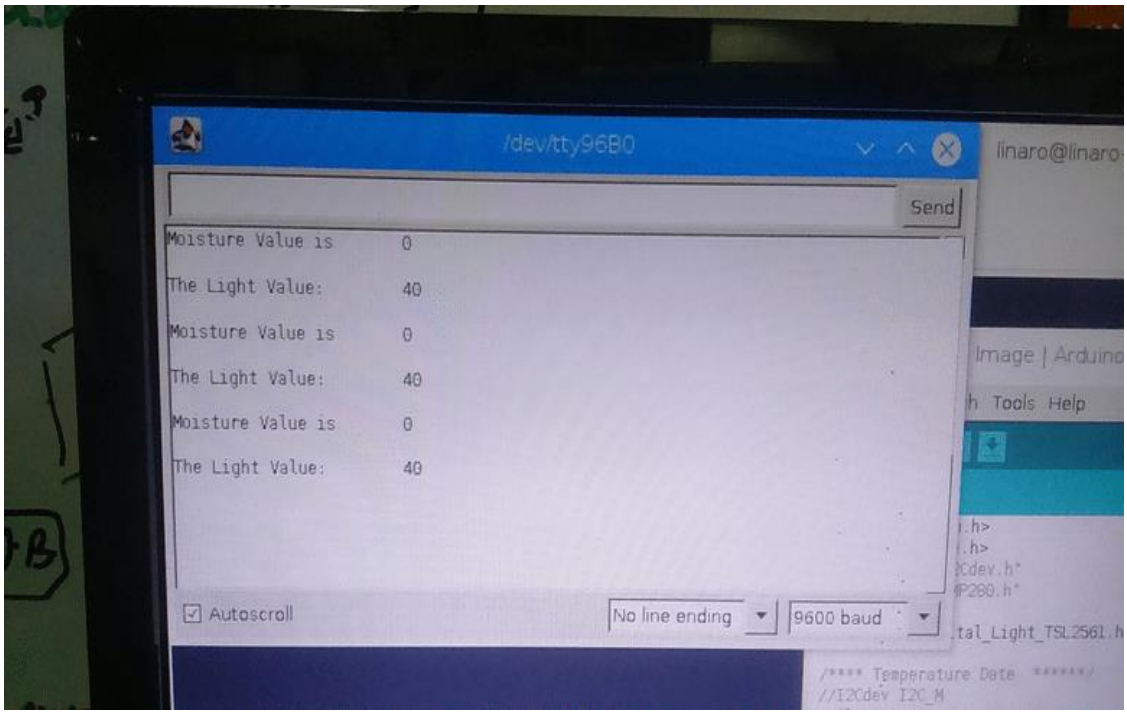
9. Upload the image,



After dumping the code in Arduino check whether its Working Properly or not. You can verify using Serial Monitor (icon present at right-most-corner) and click on it.



When you Click on Serial Monitor Icon it will look like as below,

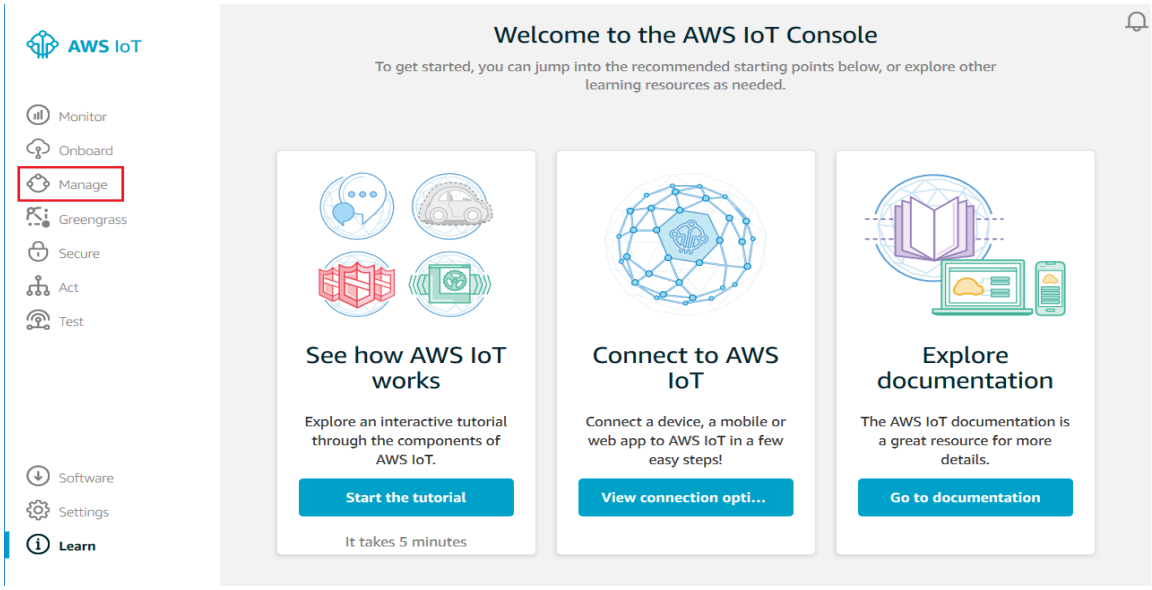


After that you close all arduino terminal.

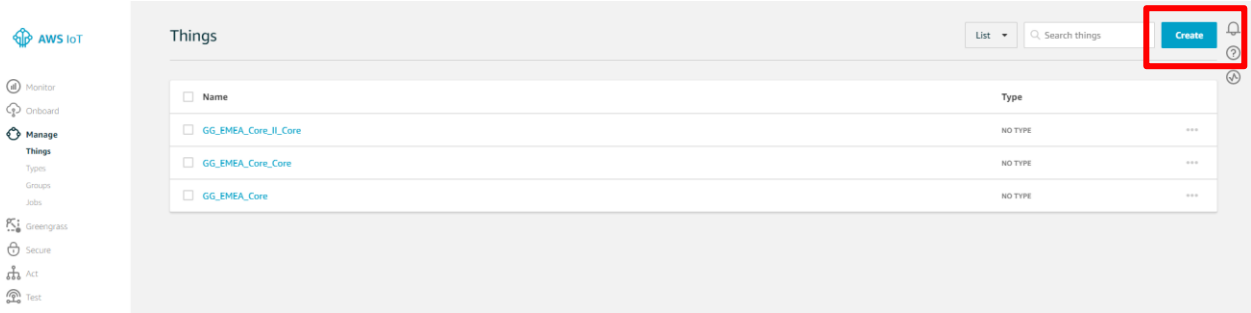
Register a Device in the Thing Registry

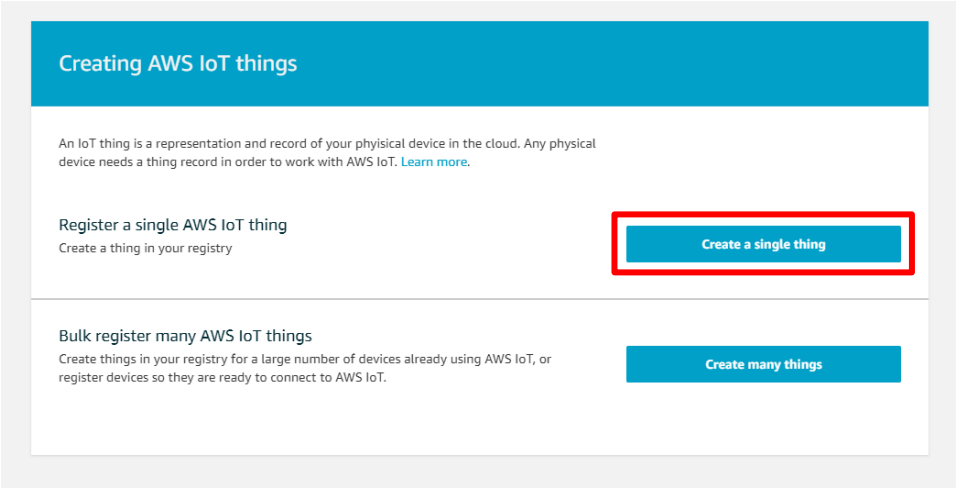
To register your device in the thing registry:

1. On the **Welcome to the AWS IoT Console** page, in the left navigation pane, choose **Manage** to expand the choices, and then choose **Things** (as necessary).

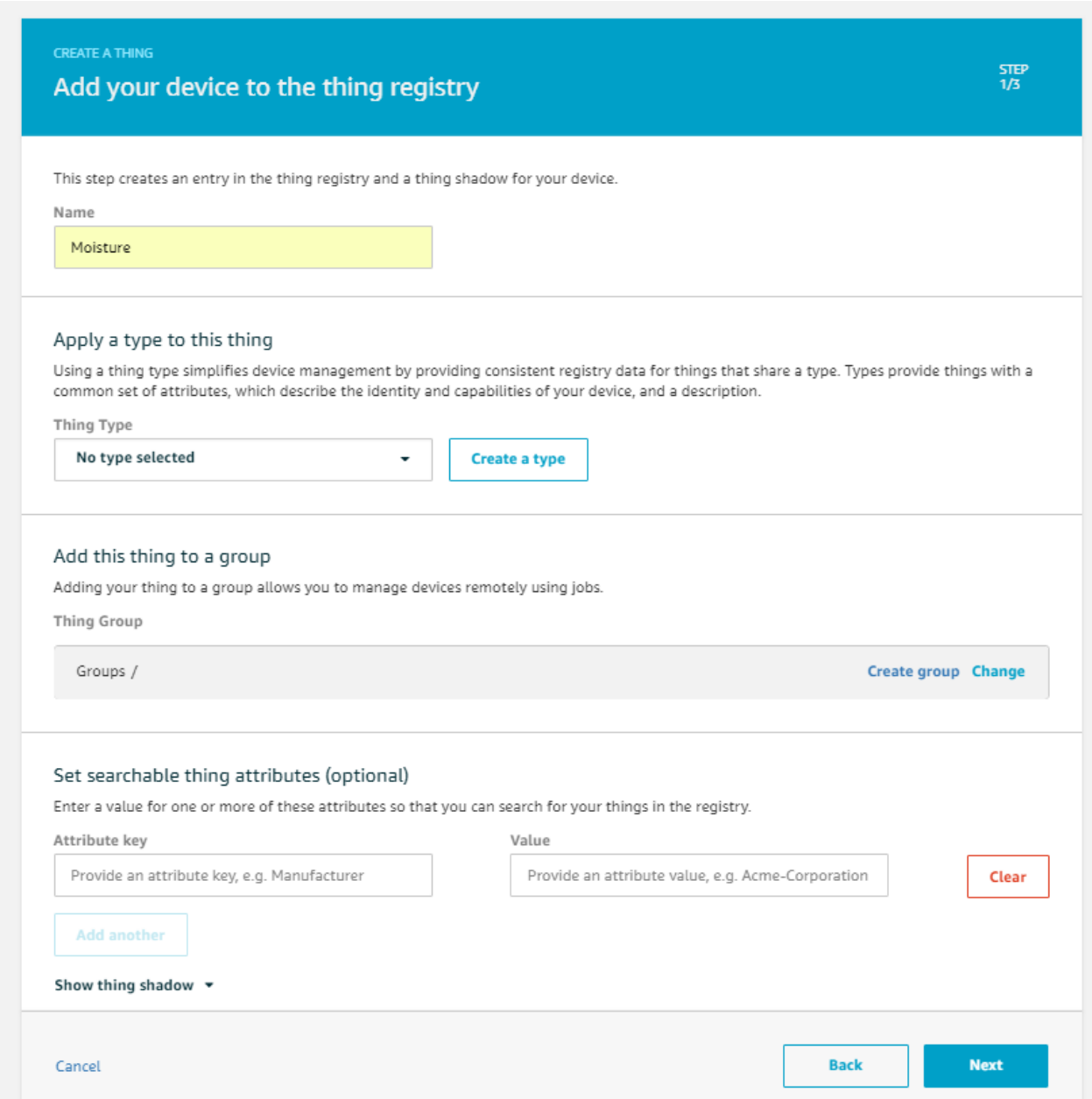


2. On the page that says **You don't have any things yet**, choose **Create a thing**.



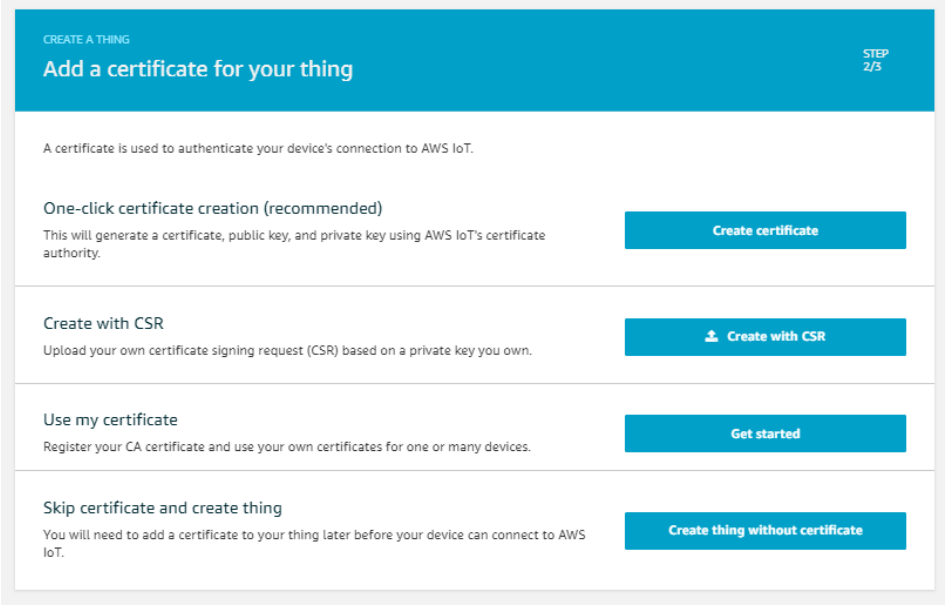


3. On the **Register a thing** page, in the **Name** field, type a name for your device, such as **Moisture**. Choose **Next** to add your device to the thing registry.

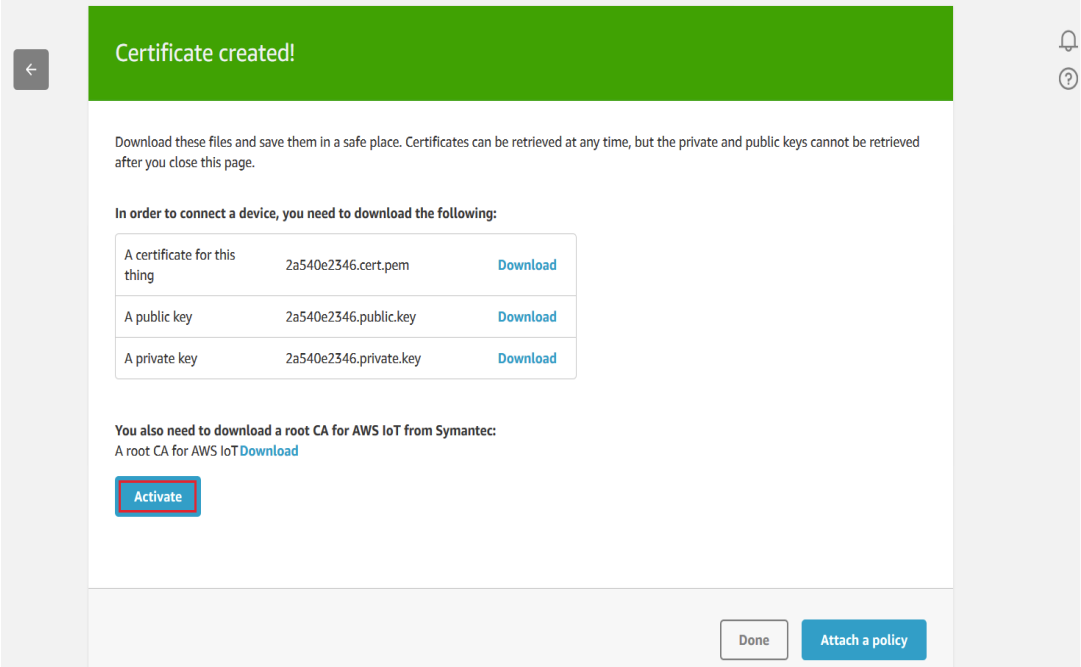


Create and Activate a Device Certificate:

1. Next Screen allows you to add a certificate to your thing. Select “One-click certificate creation”



2. On the **Certificate created!** page, choose **Download** for the certificate, private key, and the root CA for AWS IoT (the public key need not be downloaded). Save each of them to your computer, and then choose **Activate** to continue.



Be aware that the downloaded filenames may appear differently then those listed on the Certificate created! page. For example,

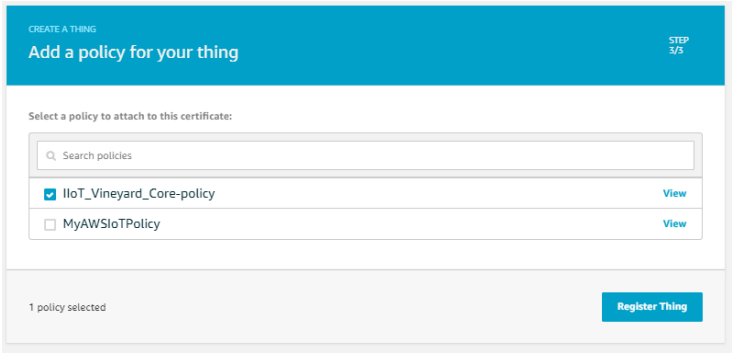
- 2a540e2346-certificate.pem.crt
- 2a540e2346-private.pem.key
- 2a540e2346-public.pem.key

After Downloading all keys rename as,

- Moisture.cert.pem
- Moisture.private.key
- Moisture.public.key

And rename root certificate as **root-CA.crt.** Also click on “Attach a policy” button below:

Select your appropriate policy like i choosen below,

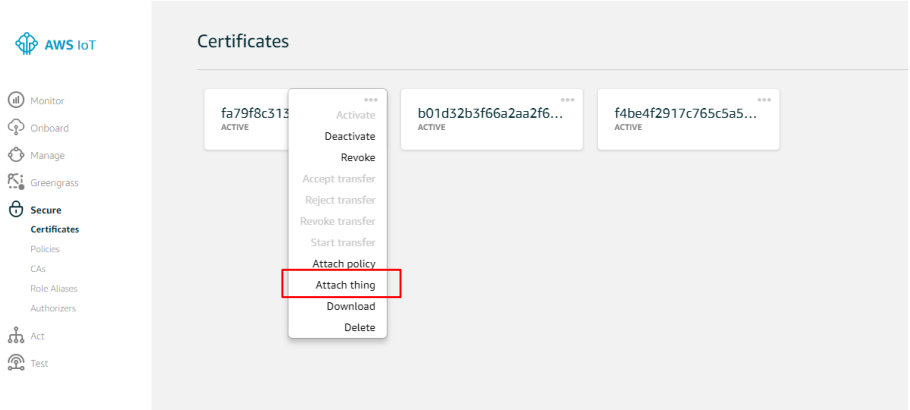


3. Choose **“Register Thing”**.

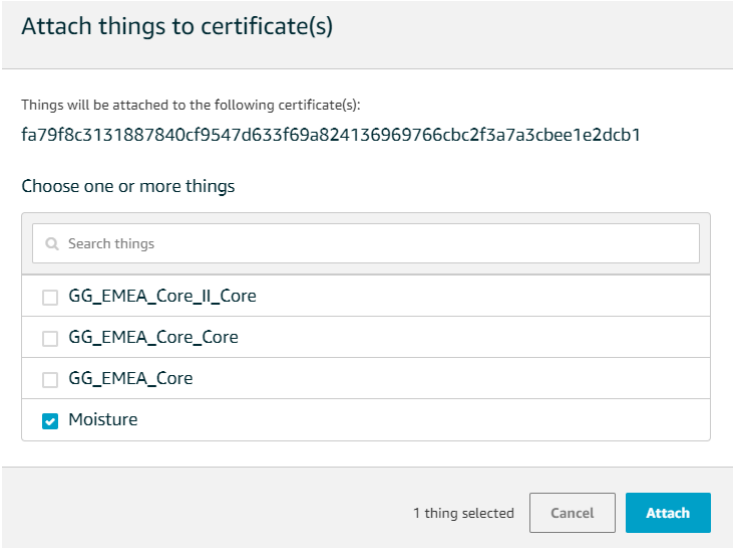
Attach a Certificate to a Thing:

To attach a certificate to the thing representing your device in the thing registry,

1. In the box for the certificate you created, choose ... to open a drop-down menu, and then choose **Attach thing**.

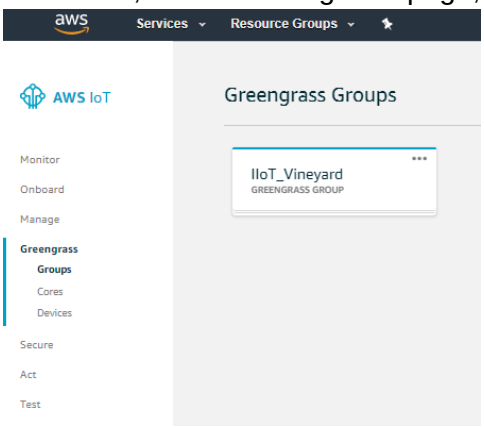


2. In the **Attach things to certificate(s)** dialog box, select the check box next to the thing you registered, and then choose **Attach** (Your thing name will be **Moisture** instead of MyIoTButton).

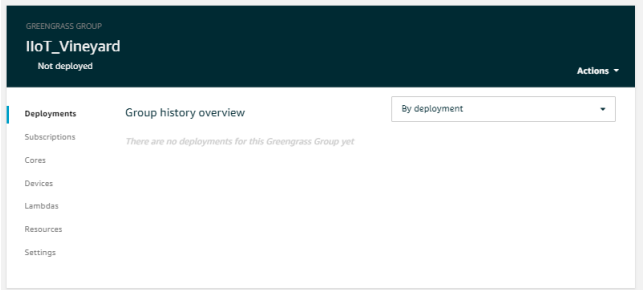


Add Your device to a Greengrass Group:

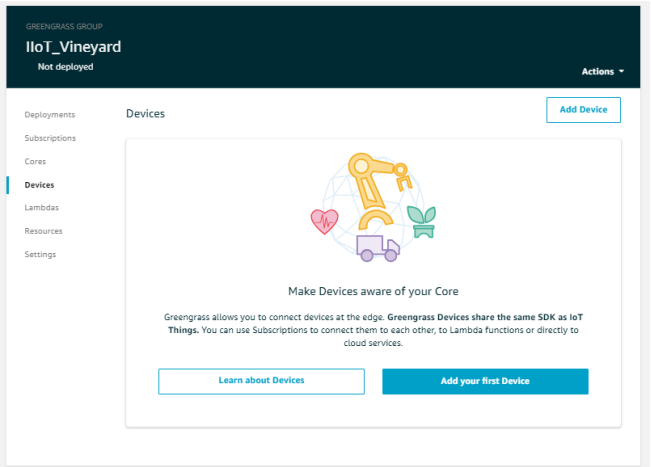
1. In the AWS Greengrass console, from the navigation page, choose **Groups**.



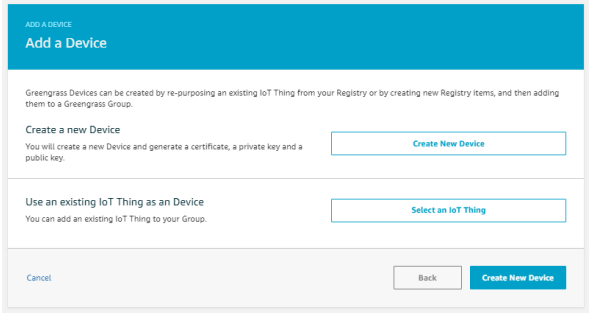
2. Choose your Greengrass Group.

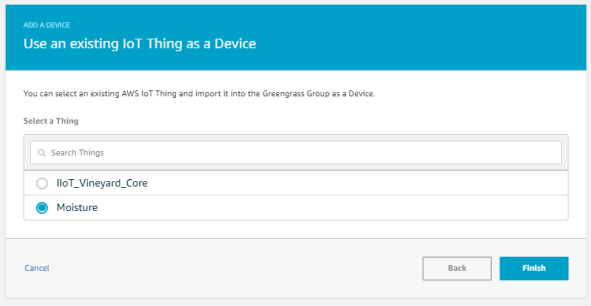


3. In the navigation pane, choose **Device** and select **Add Device**.



4. Choose **Select an IoT Thing** and select your device name(**Moisture**) and choose **finish**.

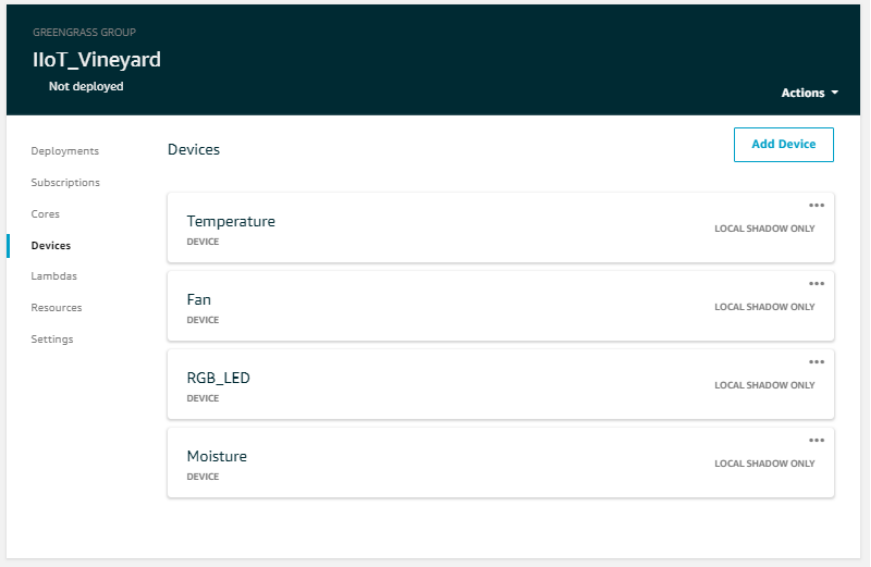




NOTE: Follow the above procedure (8.1 to 8.4) to create a thing (like Moisture) for Fan, Temperature, RGB_LED

Deploy all the device in Greengrass:

1. From AWS IoT console, go through **Greengrass** and choose **Groups** and select **GreenGrass_Group**.
2. From GreenGrass_Group, in navigation pane choose **Devices**.

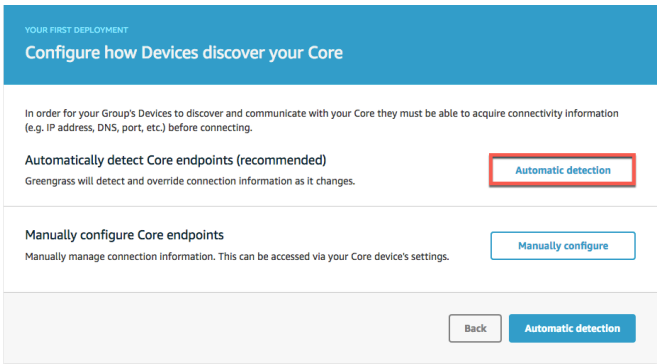


3. You will see all devices in GreenGrass_Group.
4. Before Deploying, open a terminal application from your core device (Dragonboard 410c), Go to /greengrass directory and start your greengrass to run.

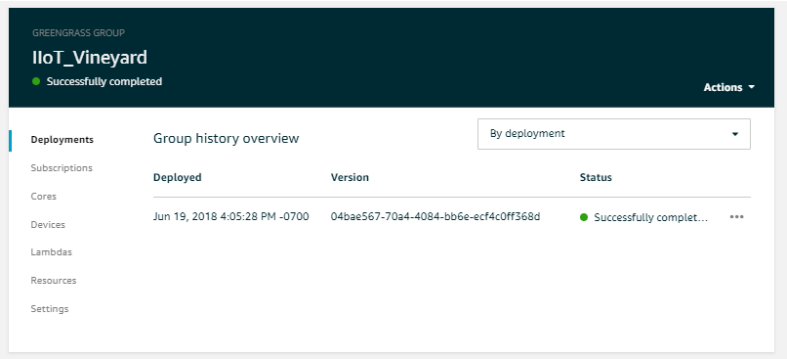
```
$ cd /greengrass
$ sudo ./greengrassed start
```

5. After running greengrass in core devices(DragonBoard410c).
6. From the **Actions** menu, choose **Deploy**.

7. On the **Configure how Devices discover your Core** page**,** choose **Automatic detection**.



8. Your deployment might take a couple of minutes. You will know the deployment was successful when a Deployment successfully completed message is displayed in the group details page.

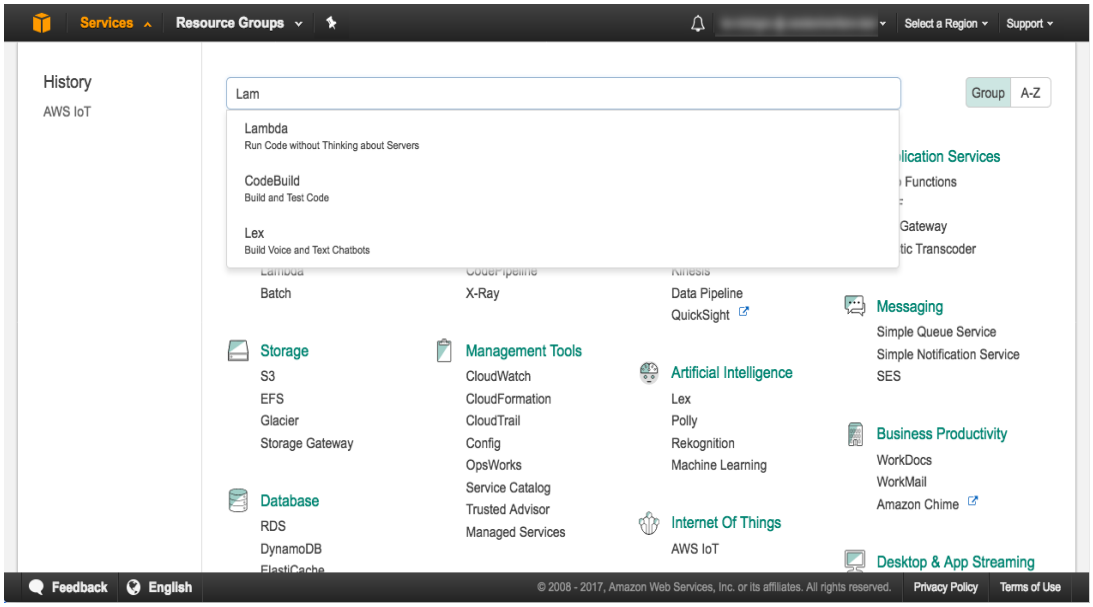


Create Lambda Function

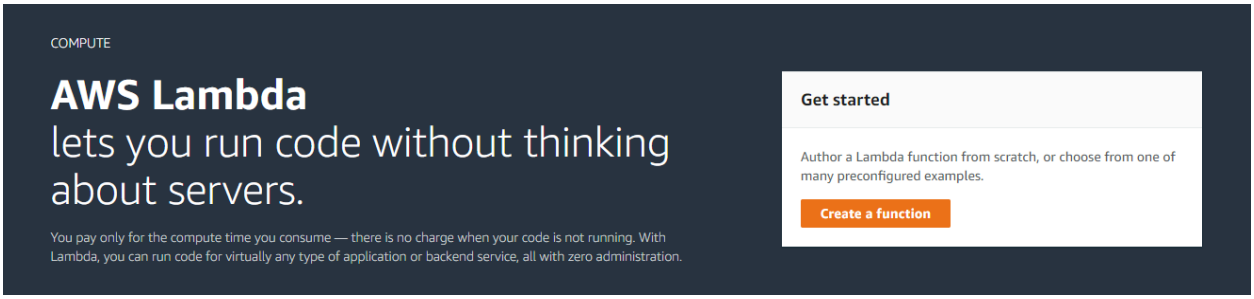
Create a "Hello World" Lambda Function:

AWS Greengrass cores can run Lambda functions in response to messages sent by your devices (or other Lambda functions). You will now create a Lambda function that you will add to your AWS Greengrass group and then deploy it to your AWS Greengrass core.

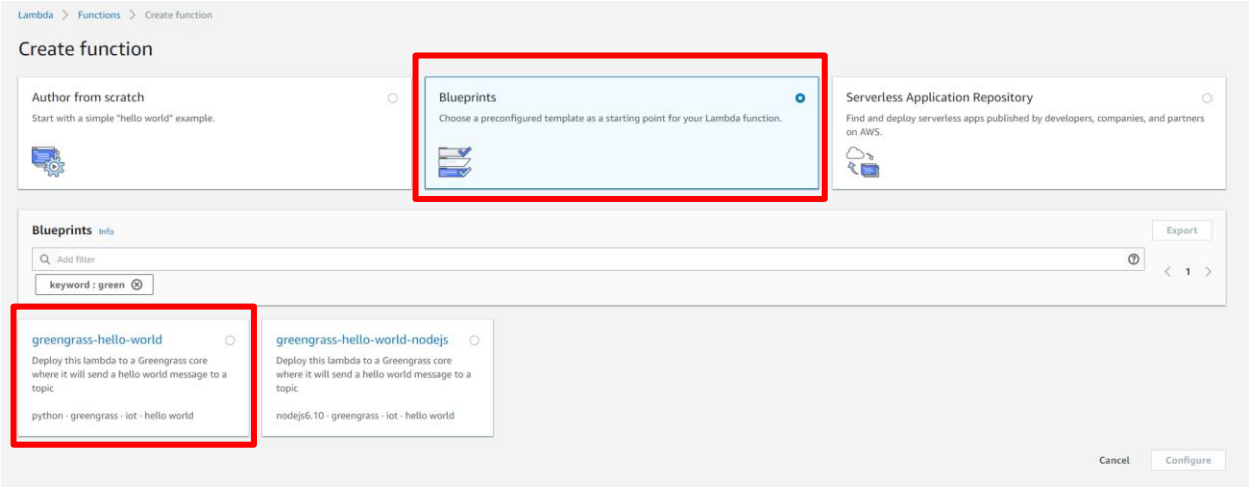
1. From the **Service** drop-down menu, navigate to the AWS Lambda console.



2. In the Lambda console, choose **Create a Lambda function.**

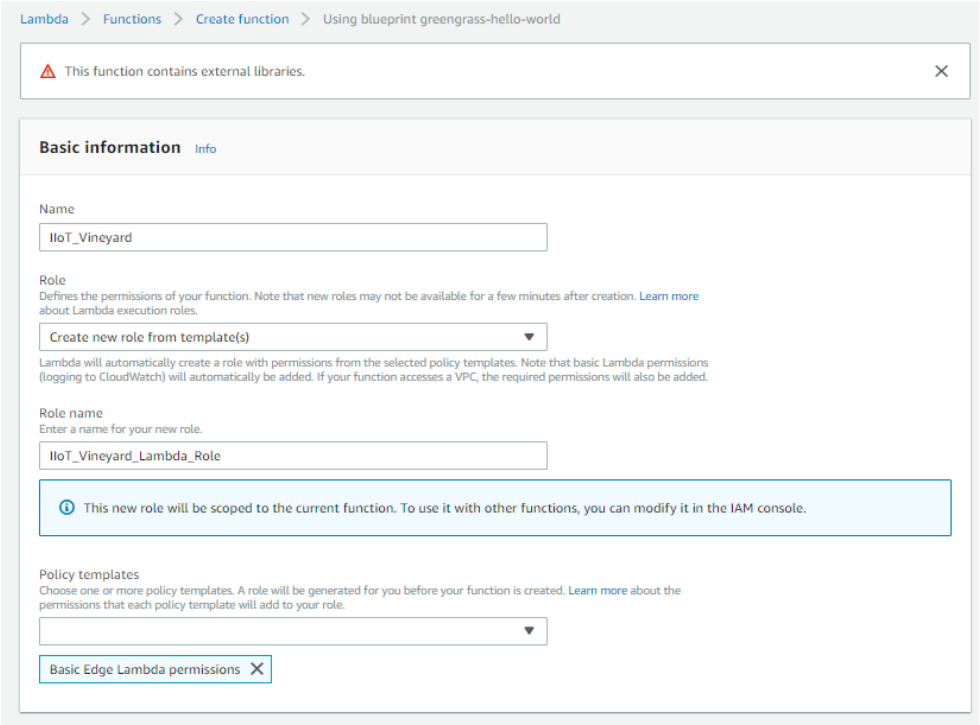


3. In the **Filter** text box, type Greengrass, and then choose the **greengrass-hello-world** Lambda function blueprint for python.

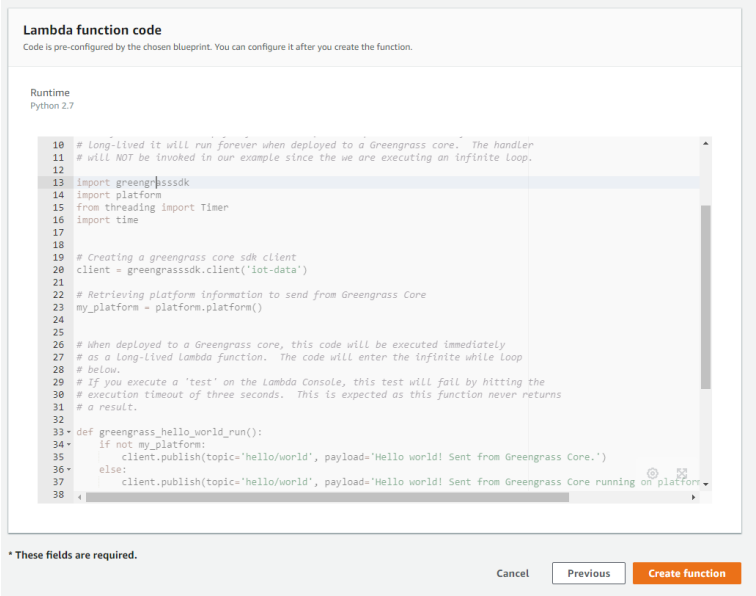


4. Choose **Configure**.

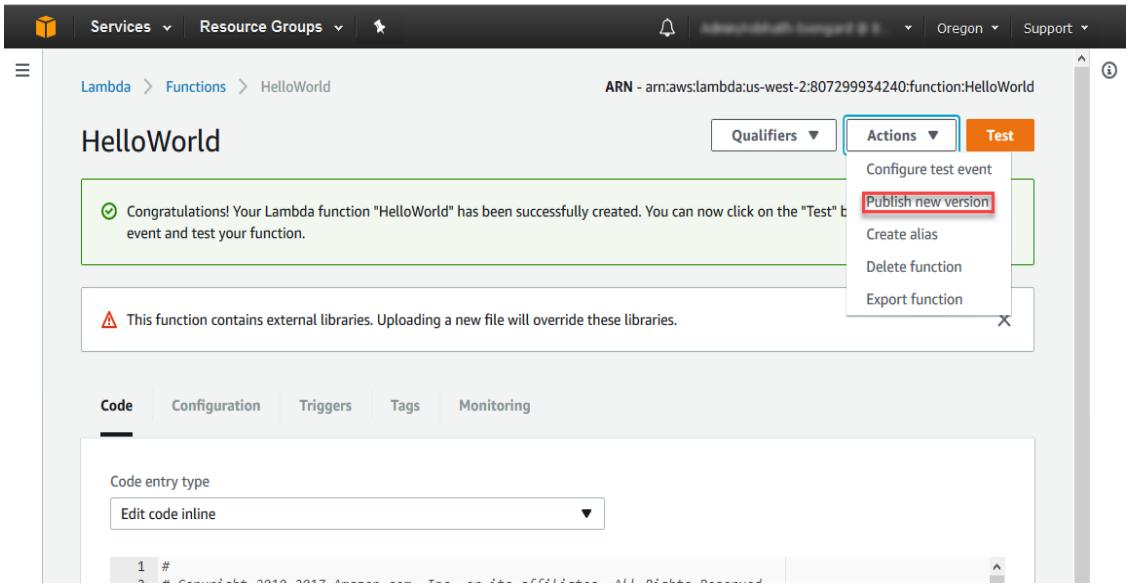
5. Scroll down until you see **Lambda function handler and role**. For **Role**, select **Choose an existing role**. For **Existing role**, select any role. If you don't have a role, select **Create a new role from template(s)**, and then choose any template from the **Policy templates** drop-down list. Choose **Next**.



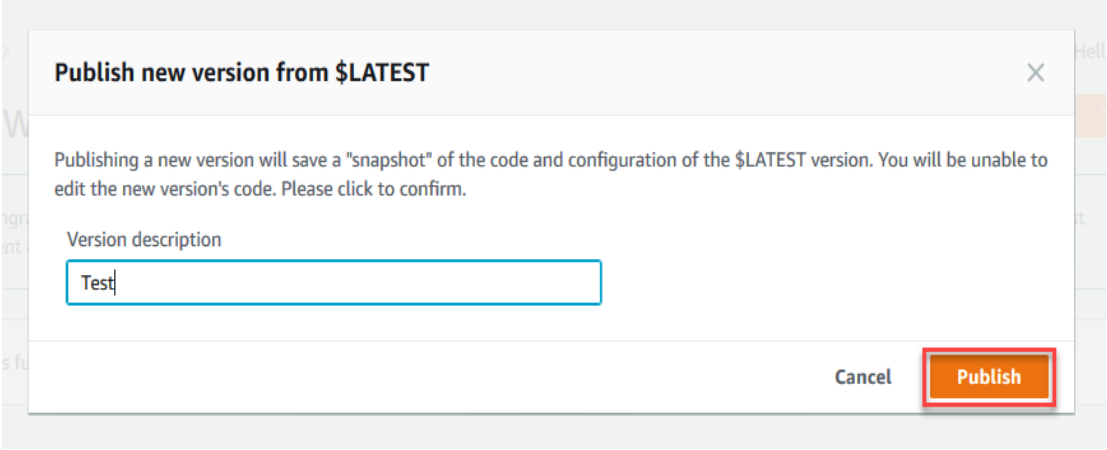
6. On the **Review** page, choose **Create function**.



7. Now you copy the lambda function code from software package and need to publish a new version of the Hello World (or your function name) Lambda function and configure the handler (iiot_event_handler) function name in the handler text box. From the **Actions** menu, choose **Publish new version**.



8. In Version description, type Test, and then choose Publish.

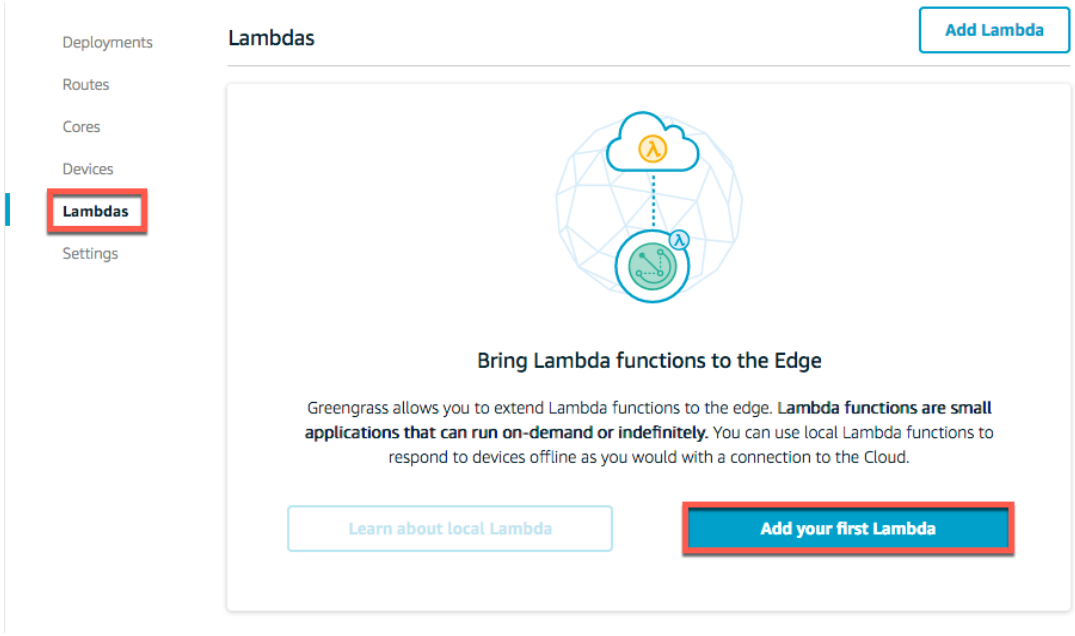


Add the Lambda Function to Your Group Definition:

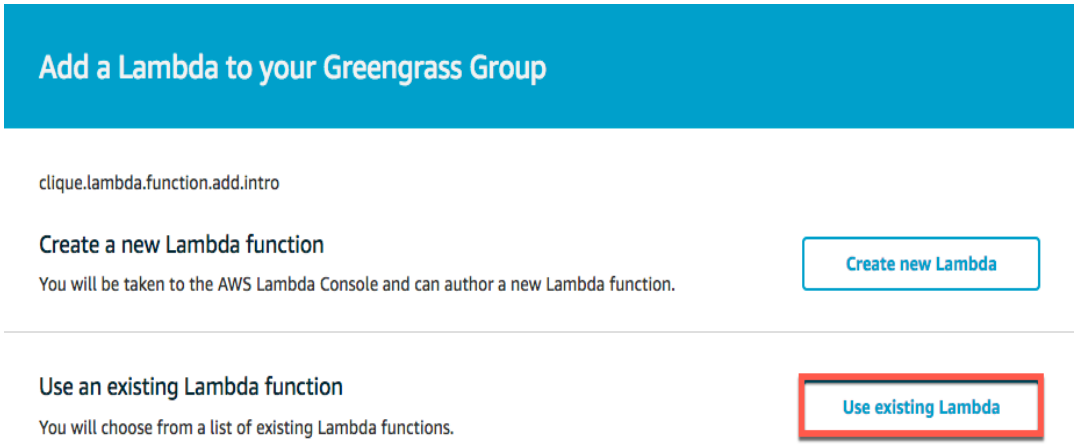
Next, you are going to add the Hello World Lambda function to your group definition. After it is deployed locally, that function sends data back to the AWS IoT platform and shows you have a deployed a functioning core.

1. In the AWS IoT console, choose **Greengrass**, and then choose **Groups**.
2. Select the title for your group.

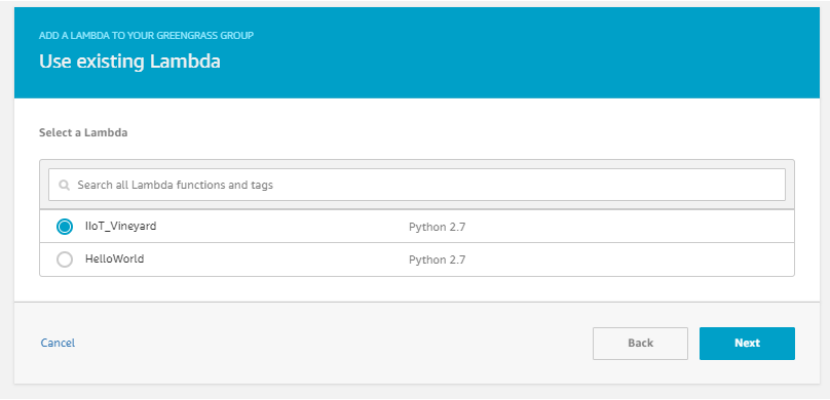
3. In the navigation pane, choose **Lambdas**, and then choose **Add your first Lambda**.



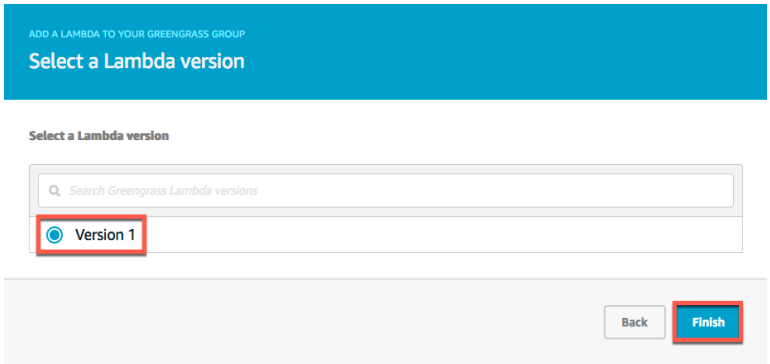
4. Choose **Use existing Lambda**.



5. Select the Lambda function you created earlier, and then choose **Next**. If you create many Lambda functions for Greengrass, you can organize them, and make searching for them easier, by using the Lambda function tagging feature. AWS Greengrass supports searching for Lambda functions by their tags in this console pane.

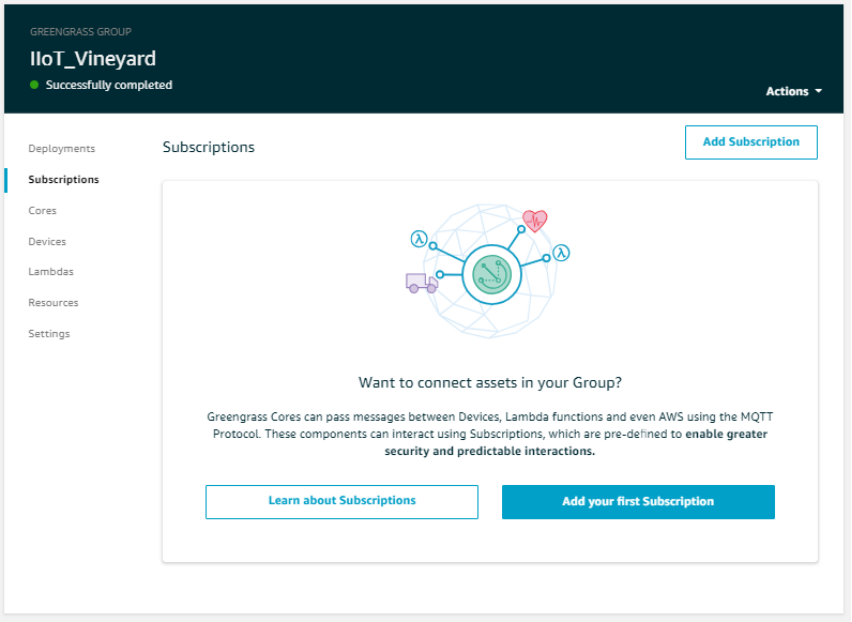


6. Select the version of the Lambda function to use, and then choose **Finish**.

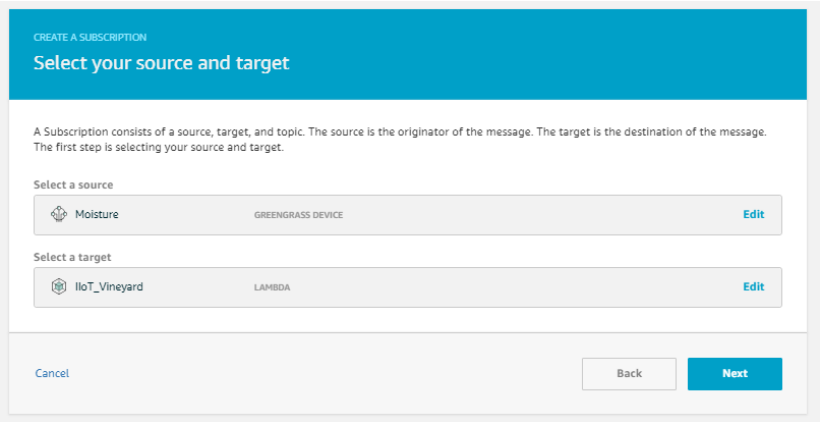


Add Subscription to Your Group Definition:

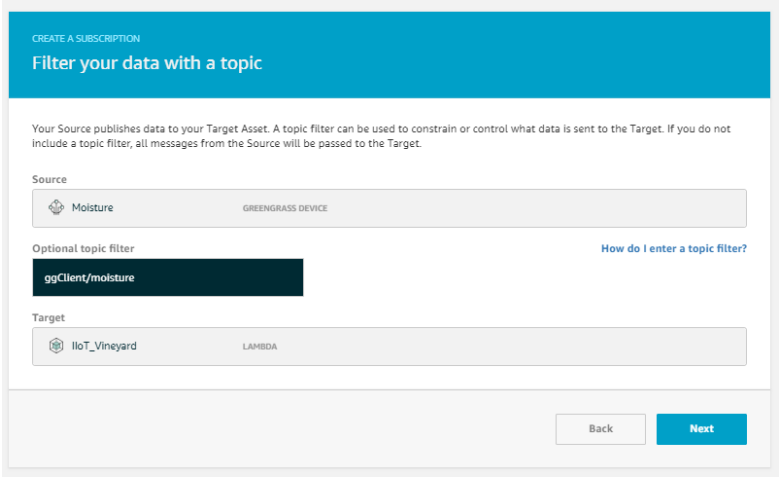
1. In the AWS Greengrass console, find your group, and then select it.
2. On the group details page, choose **Subscriptions**.
3. Choose **Add your first Subscription**.



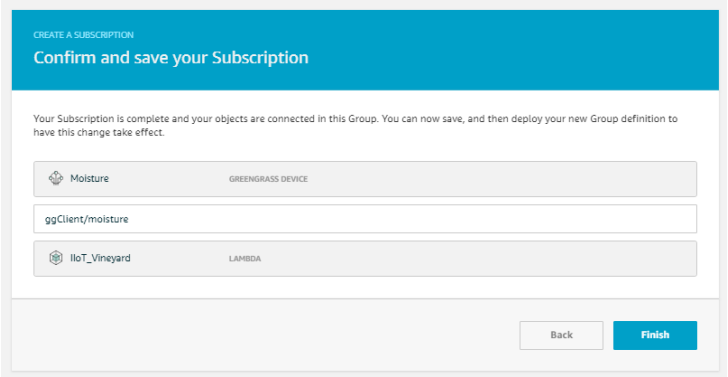
4. Under **Select a source**, choose your device name(**Moisture**).
5. Under **Select a target**, choose your created lambda function.
6. Choose **Next**.



7. In **Optional topic filter**, type **ggClient/moisture**, and choose **Next**.



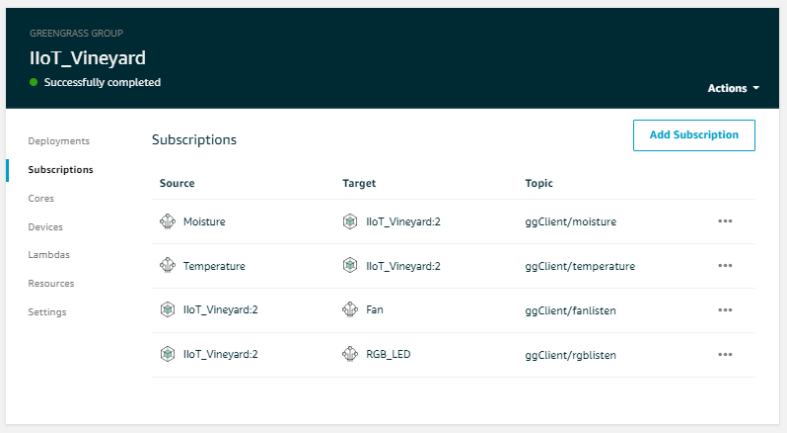
8. Choose **Finish** to confirm and save your subscription.



9. Repeat steps from 3 to 8 to add subscriptions for Fan, RGB_LED and Temperature, Based on the below configuration.

Source	Target	topic-filter
Moisture	your lambda-function	ggClient/moisture
your lambda-function	RGB_LED	ggClient/rgblisten
Temperature	your lambda-function	ggClient/temperature
your lambda-function	Fan	ggClient/fanlisten

If you add all, the following figure shows look as given below:



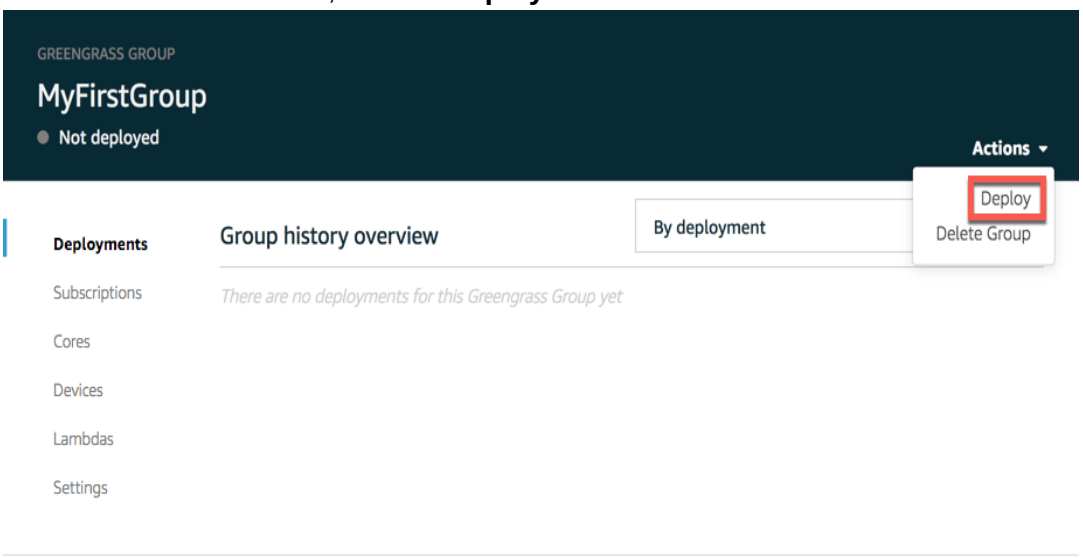
Deploy your Group:

You have created a group and a core definition, but this information exists only in the cloud. Deploying a group takes this configuration information and copies it onto your AWS Greengrass core device.

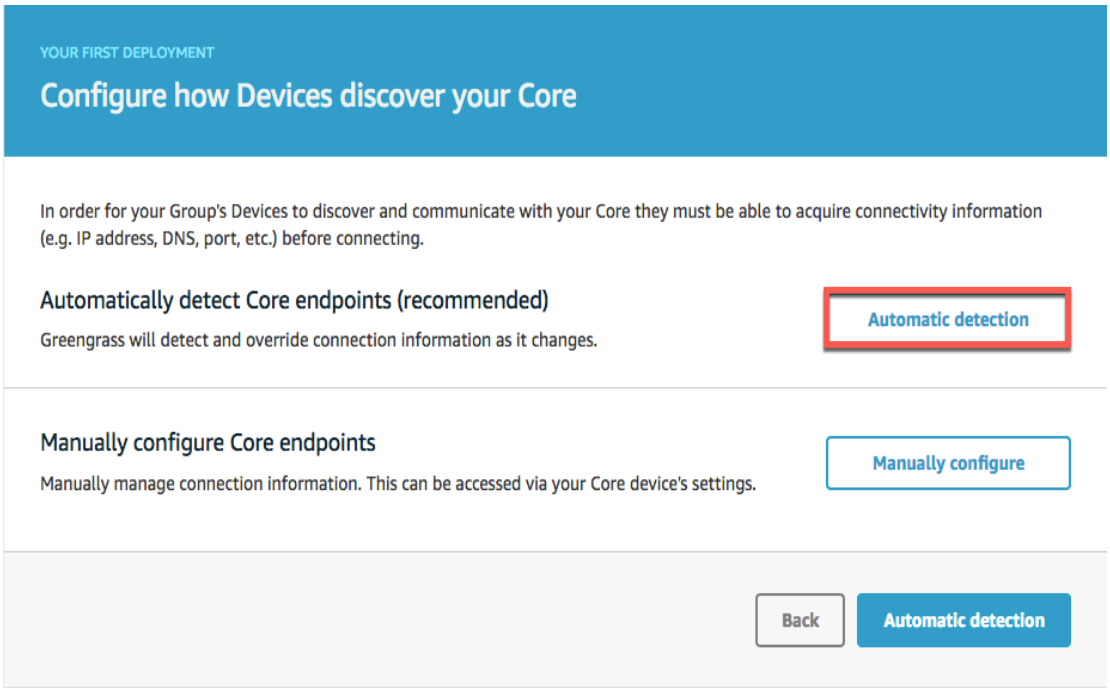
1. In the AWS Greengrass console, choose **Groups**, choose your group, and then choose **Deployments**.
   NOTE: Before Deployment, open a terminal application from your core device (Dragonboard 410c), Go to /greengrass directory and start your greengrass to run.

```
$ cd /greengrass
$ sudo ./greengrassed start
```

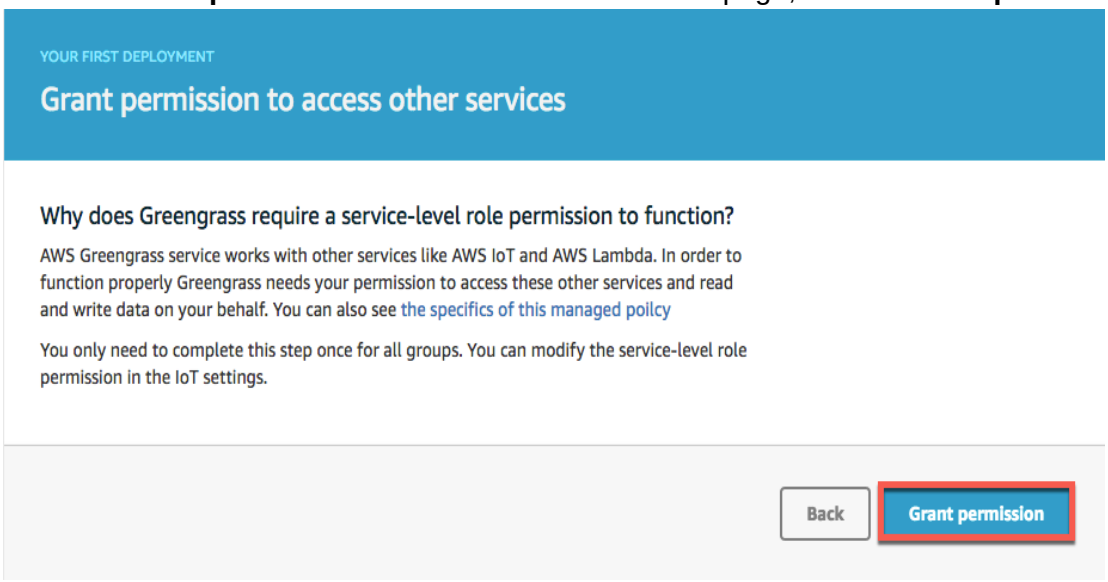
2. From the **Actions** menu, choose **Deploy**.



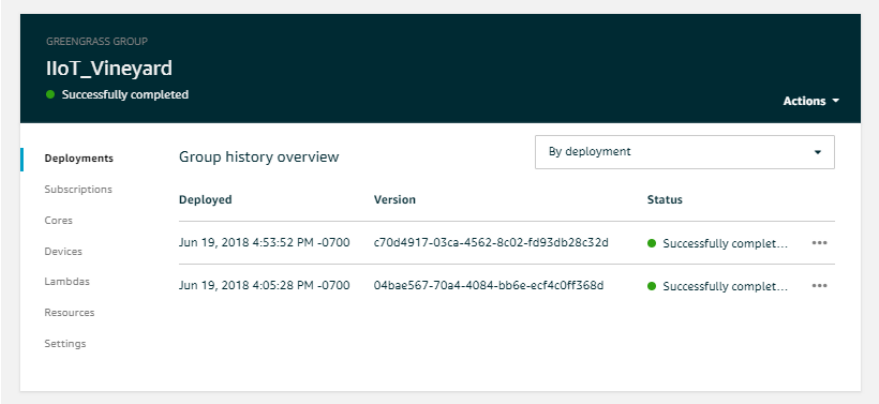
3. On the **Configure how Devices discover your Core** page, choose **Automatic detection**.



4. On the **Grant permission to access other services** page, choose **Grant permission**.



Your deployment might take a couple of minutes. You will know the deployment was successful when a Deployment successfully completed message is displayed in the group details page:



Installation of necessary packages

Installation of node.js6x:

From your terminal in DragonBoard 410c type below command to install in root user,

```
$ sudo su -
$ curl -sL https://deb.nodesource.com/setup_6.x | bash -
$ apt-get install -y nodejs
```

Installation of aws-iot-device-sdk for python:

From your terminal in DragonBoard 410c type below command to install,

```
$ git clone https://github.com/aws/aws-iot-device-sdk-python.git
$ cd aws-iot-device-sdk-python
$ python setup.py install
```

Installation of smbus:

From your terminal in DragonBoard 410c type below command to install,

```
$ sudo apt-get install python-smbus
```

NOTE: For configuring and starting the sensor and web application please, refer to the GitHub repository:

https://github.com/globaledgesoft/smartvine-greengrass