NYP NANYANG POLYTECHNIC

_____

Course: EGDF20 Diploma in Electronic and Computer Engineering

Module: EGE356 IoT System Architecture & Technology
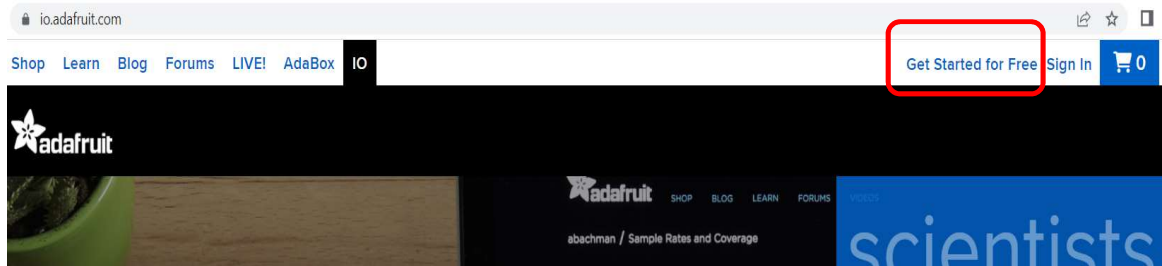
Lab 1:   Introduction to Cloud Platform Connectivity

Objectives:

1. Cloud Setup for Client Connection
2. Edge device and Cloud Connection Setup

Part 1: Cloud Setup for Client Connection

1. To setup access to Adafruit Cloud, go to https://io.adafruit.com and create a free account by following the steps below. Use the NYP student email address to create the account.

EMAIL

chng_jit_heng@nyp.edu.sg ✔

USERNAME

chng_jh ✔

Username is viewable to the public on the forums, Adafruit IO, and elsewhere.

PASSWORD

•••••••••••• ✔

CREATE ACCOUNT

> Click on "CREATE ACCOUNT" once the email, username and password is accepted.

2. Once the account creation is successful, the user will be redirected to the following page
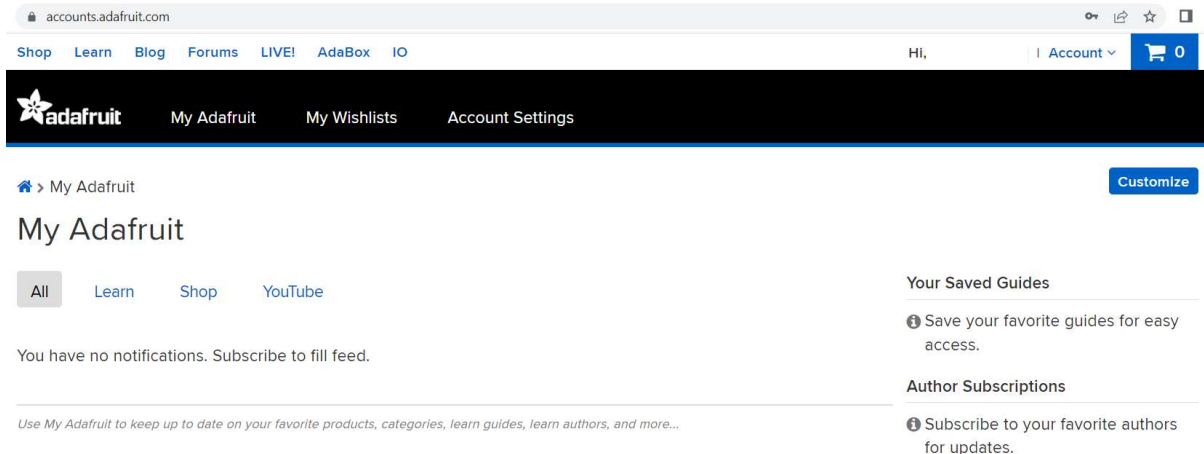
3. Note that this is a free account. There are limitations to a free account. To view the limitations, click on IO, and scroll down to view the free account access limitations. Take note of the limitations highlighted below for the free account.

---
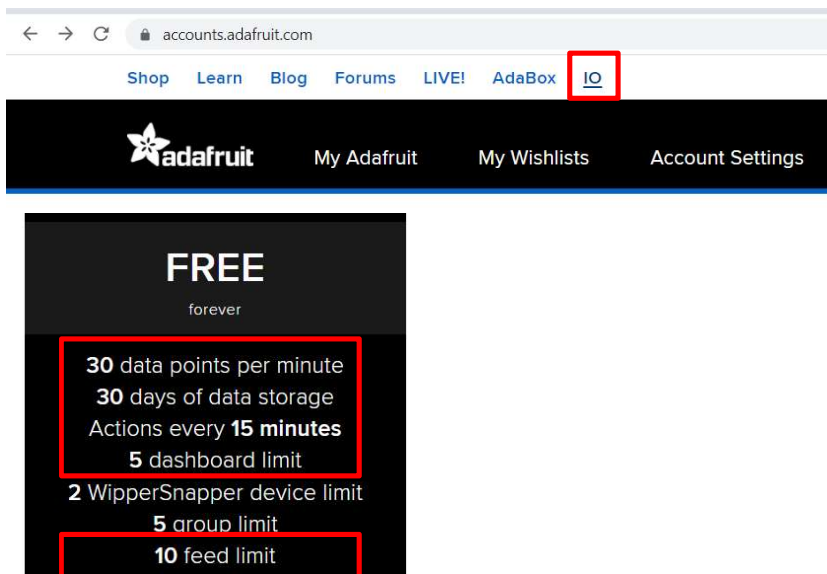
4. What does 30 datapoints per minute refer to and when writing an Arduino code, will delay be required in the code? If required, what delay should be implemented?
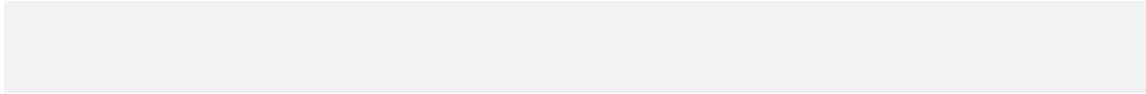
5. Upon login to Adafruit IO, the side should be automatically redirected to the Dashboard page. If not, click on the Dashboard tab in the navigator bar as shown below.
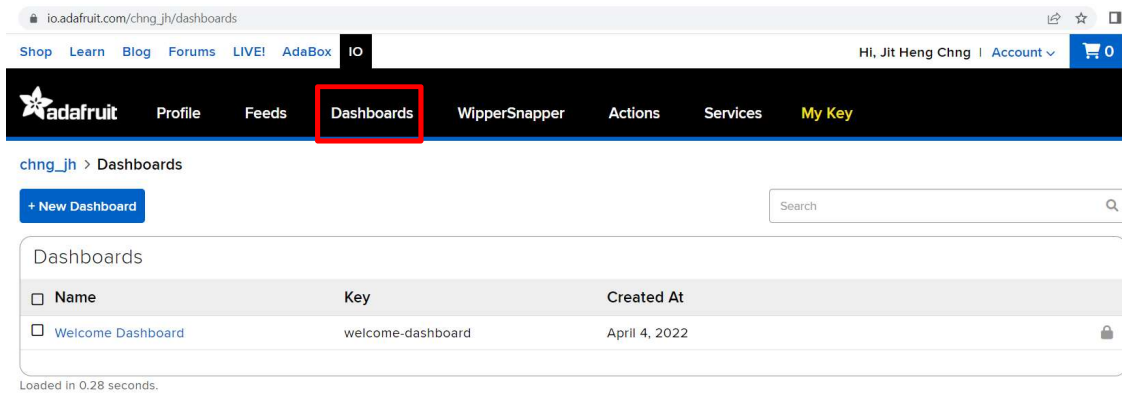
6. Click on the "Welcome Dashboard" link to view the Dashboard page.
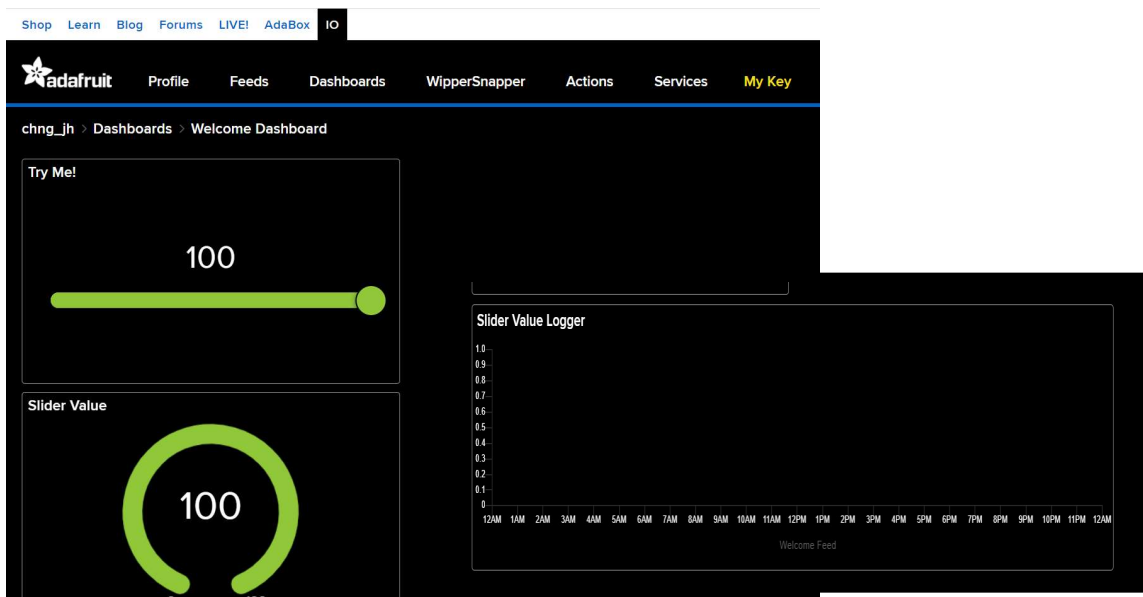
7. The first block is a slider block, the second block a gauge and the third a line chart.

8. Click on the [⚙^] icon and select "Edit Layout" as shown below.

9. Click on the icon ![icon] to edit the block.

10. Note that the block is connected to a Feed.

11. Click on "Next step" to view the block settings.

Block settings   ✕

In this final step, you can give your block a title and see a preview of how it will look.
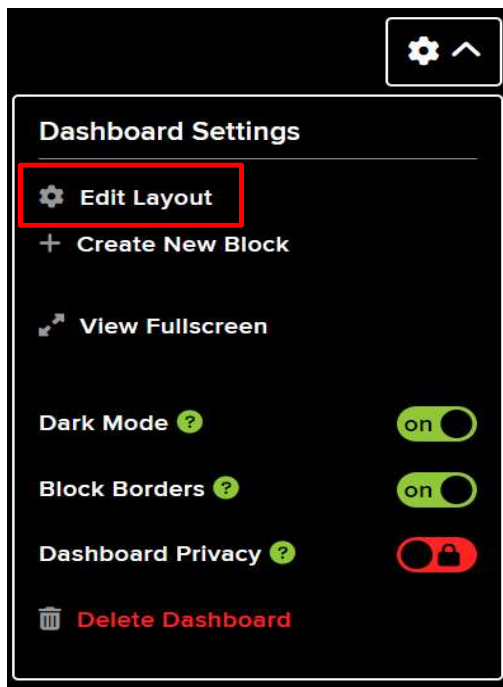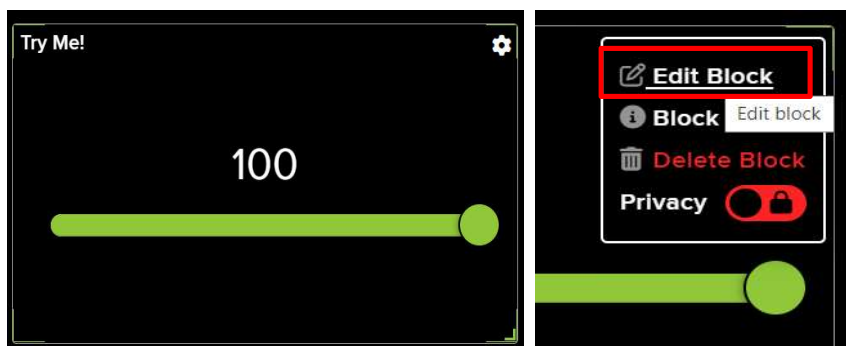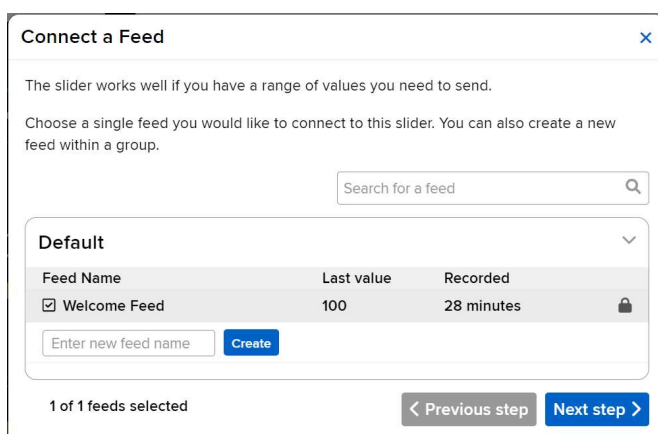Customize the look and feel of your block with the remaining settings. When you are ready,
click the "Create Block" button to send it to your dashboard.

Block Title (optional)   Block Preview

Try Me!   **Try Me!**

Slider Min Value   **49**

0

Slider Max Value

100

Slider Step Size   Slider The slider works well if you have a
                                          range of values you need to send.
1

Slider Label   **Test Value**

                                          49

Decimal Places   **Published Value**

4   49

Number of decimal places to display,   2 bytes
defaults to 4.

                                          ‹ Previous step   Update block

12. Note that Block Title, Slider values and Slider Step Size are customizable.

   * Take note of display label – 2 decimal places limit.

13. Close the block settings by clicking on [✕] at the top right corner.

14. Repeat steps 9 and 10 for the "Gauge" block and the Line Chart. Note that they all use the same feed.
   The "Welcome Feed".

 15. In this Dashboard setup, the Input is provided by Slider, and both the Gauge and Line Chart provide the
      output display. Changes in data from the Slider are provided thru the Feed and the values are
      displayed via the Gauge and Line Chart.

 16. What are the three steps in the create block form?

            1.   Select block type
            2.   Select feed(s)
            3.   Block settings

_____

**Part 2: Edge Device and Cloud Connection Setup**

1. In this part of the lab, a New Dashboard will be created with digital control button to remotely turn the M5StickC Plus RED LED ON and OFF.

2. Click on Dashboard in the navigator bar as shown below to go to the Dashboard page.
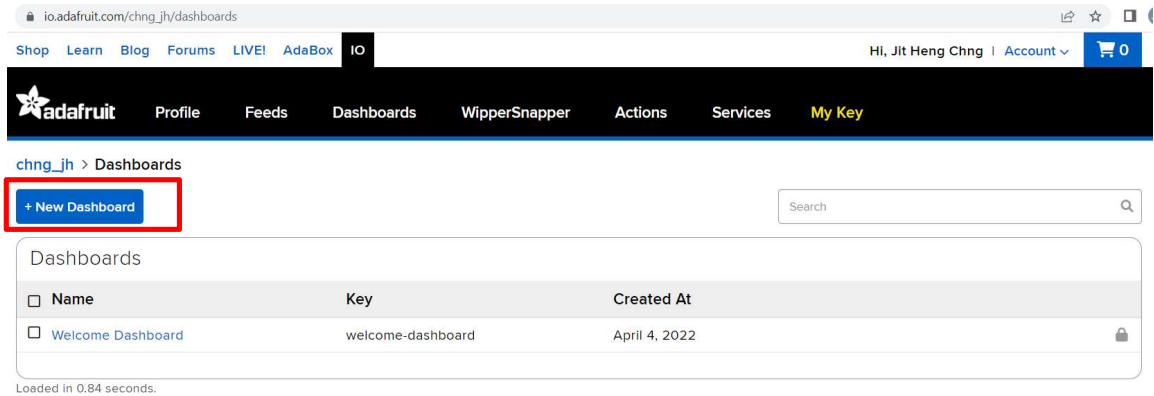
3. Click on "New Dashboard" to create a new dashboard as shown below.

4. Click on "Create" to complete the creation of the new Dashboard – Lab1
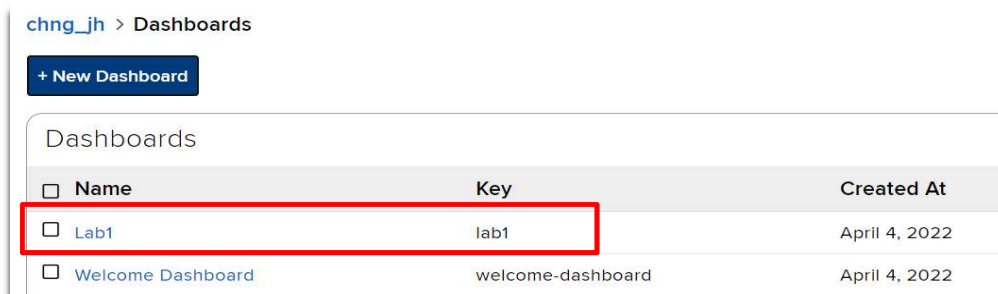
Note that within the Cloud, each Dashboard has an ID, and the name of the Dashboard is linked to the ID. Likewise each Key, although having a common name in this case "lab1", it has an ID generated programmatically.

5.   To add blocks to the Boards, click on . To switch off the DarkMode, toggle the "Dark Mode" button to 'OFF' to change the background color to white. If not, keep the DarkMode button 'ON' mode. Click on 'Create New Block'.
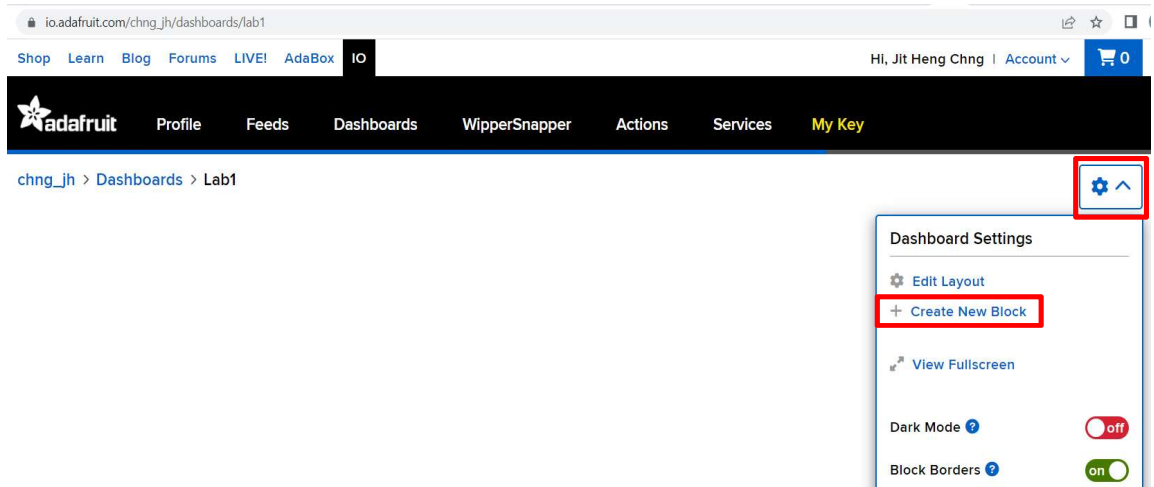
6.   Select the Toggle button as shown in the diagram below.
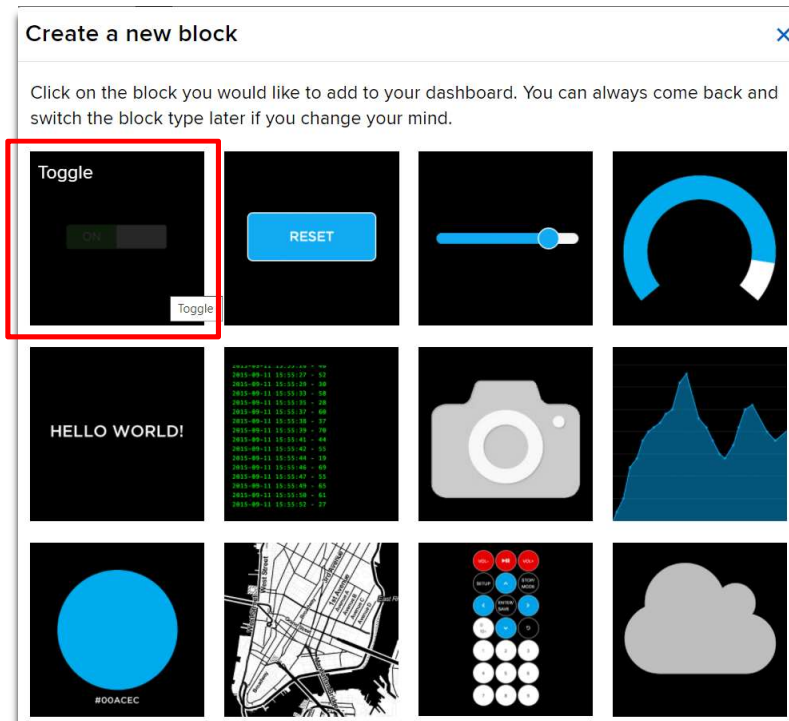
7. Click on the toggle button and follow the process below:

Key the Feed Name 'lab1_led' and click 'Create'

Select the feed to be use it with the Toggle Block as shown in the illustration below.

Connect a Feed

A toggle button is useful if you have an ON or OFF type of state. You can configure what values are sent on press and release.

Choose a single feed you would like to connect to this toggle. You can also create a new feed within a group.

Search for a feed

Default

Feed Name	Last value	Recorded
☑ lab1_led		36 minutes 🔒
☐ Welcome Feed	100	about 4 hours 🔒

Enter new feed name   **Create**

1 of 1 feeds selected   **‹ Previous step**   **Next step ›**

Once the feed has been selected, click on "Next Step".

Block settings

In this final step, you can give your block a title and see a preview of how it will look. Customize the look and feel of your block with the remaining settings. When you are ready, click the "Create Block" button to send it to your dashboard.

Block Title (optional)

Block Preview   LED ON/OFF

Button On Text

ON

Limit of 6 characters for the toggle text. Use the block title to be more descriptive.

Button On Value (uses On Text if blank)   1

Button Off Text

OFF   **Toggle** A toggle button is useful if you have an ON or OFF type of state. You can configure what values are sent on press and release.

Limit of 6 characters for the toggle text. Use the block title to be more descriptive.

Button Off Value (uses Off Text if blank)   **Test Value**   0

45

Published Value

0 bytes

‹ Previous step   **Create block**

To test the Toggle Button, mouse over the toggle button and click to ON and OFF and notice the change in the Test Value respectively.

Block Preview

Block Preview

Block Preview

OFF

ON

Toggle A toggle button is useful if you have an ON or OFF type of state. You can configure what values are sent on press and release.

Test Value

0

Published Value

0

`1 bytes`

Toggle A toggle button is useful if you have an ON or OFF type of state. You can configure what values are sent on press and release.

Test Value

1

Published Value

1

`1 bytes`

Toggle A toggle button is useful if you have an ON or OFF type of state. You can configure what values are sent on press and release.

Click Create Block to complete the creation of the Toggle Button.

Test Value

1

Published Value

1

`1 bytes`

‹ Previous step   **Create block**

› Dashboards › **Lab1**

LED ON/OFF

8.   Edit the layout via the steps as follows:

› Dashboards › **Lab1**

LED ON/OFF

Click and select 'Edit Layout'

Mouse over and drag the corners to extend the view

Click 'Save Layout'
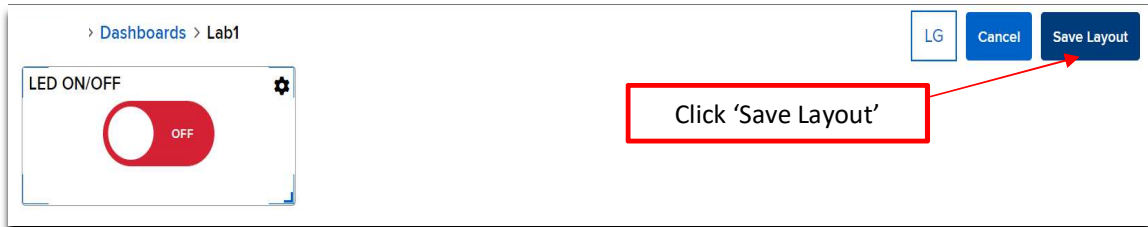
9. Launch Arduino from **arduino-1.8.16-windows** folder (if using an older version) or **arduino-1.8.19-windows** (if using the current updated version). *Ensure that the previous instructions to setup Arduino for Adafruit and M5StcikC Plus have already been completed. If not, do the setup now.*

Launch Arduino.exe program from the previously installed folder.
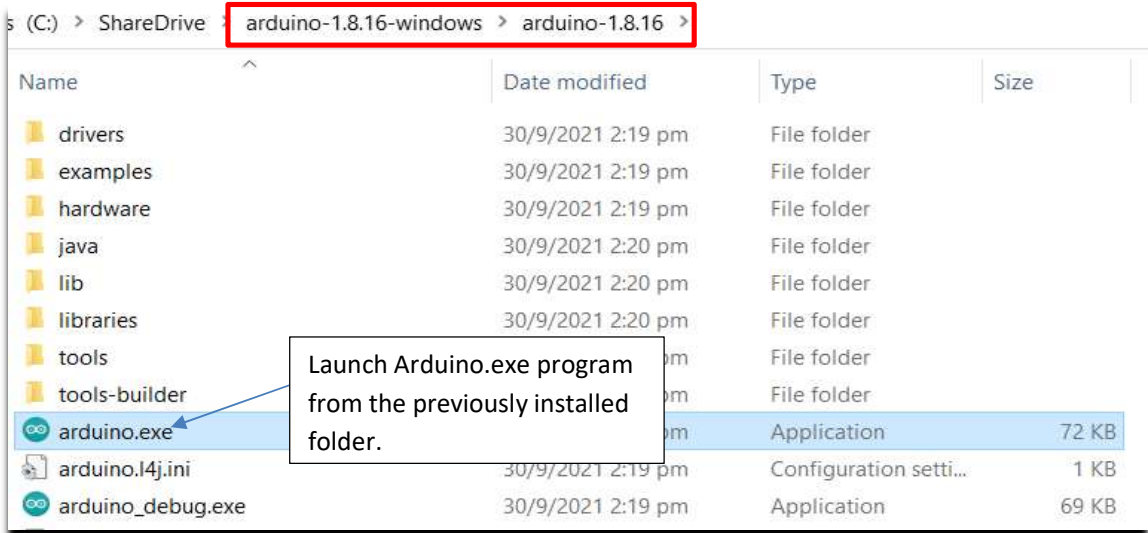
10. Upon launch of Arduino.exe, go to File → Examples →Adafruit IO Arduino → adafruitio_07_digital_out

```
// digital pin 5
#define LED_PIN 5
```

Replace code → #define LED_PIN 10

```
// set up the 'digital' feed
AdafruitIO_Feed *digital = io.feed("digital");

void setup() {

  pinMode(LED_PIN, OUTPUT);

  // start the serial connection
  Serial.begin(115200);

  // wait for serial monitor to open
  while(! Serial);

  // connect to io.adafruit.com
  Serial.print("Connecting to Adafruit IO");
  io.connect();
```

Modify to "lab1_led"

Modify to "digital_lab1"

Add the following libraries as shown just below the #include "config.h":

```
/*********************** Configuration ***********************/

// edit the config.h tab and enter your Adafruit IO credentials
// and any additional configuration needed for WiFi, cellular,
// or ethernet clients.
#include "config.h"



#include <M5StickCPlus.h>
/*********************** Example Starts Here ***********************/
```

Add the 2 libraries as shown to the Arduino code.

// set up the button access for M5StickCPlus
```
const int buttonA = 37;
const int buttonB = 39;
int cur_valueA = 0;
int cur_valueB = 0;

void setup() {
```

Add the following codes, just above **void setup()**

In **void setup(),** make the following code modifications:

```
void setup() {

  // set button pin as an input
  pinMode(BUTTON_PIN, INPUT);

  // start the serial connection
  Serial.begin(115200);

  // wait for serial monitor to open
  while(! Serial);

  // connect to io.adafruit.com
  Serial.print("Connecting to Adafruit IO");
  io.connect();
```

Replace code

```
pinMode(LED_PIN, OUTPUT);
digitalWrite(LED_PIN, HIGH);
M5.begin();
```

To turn LED PIN 10 off – because it is wired as ACTIVE-LOW

```
Serial.print("Connecting to Adafruit IO");
io.connect();

// set up a message handler for the 'digital' feed.
// the handleMessage function (defined below)
// will be called whenever a message is
// received from adafruit io.
digital->onMessage(handleMessage);

// wait for a connection
while(io.status() < AIO_CONNECTED) {
  Serial.print(".");
  delay(500);
}
```

Modify to "LED_ctrlmsg"

Modify to "digital_lab1"

```
void loop() {

  // io.run(); is required for all sketches.
  // it should always be present at the top of your loop
  // function. it keeps the client connected to
  // io.adafruit.com, and processes any incoming data.
  io.run();

}

// this function is called whenever an 'digital' feed message
// is received from Adafruit IO. it was attached to
// the 'digital' feed in the setup() function above.
void handleMessage(AdafruitIO_Data *data) {

  Serial.print("received <- ");

  if(data->toPinLevel() == HIGH)
    Serial.println("HIGH");
  else
    Serial.println("LOW");

  digitalWrite(LED_PIN, data->toPinLevel());
}
```

Modify to "LED_ctrlmsg"

Add '!' to data->toPinLevel() as shown:

  digitalWrite(LED_PIN, !(data->toPinLevel()));

11. Note that to connect to the Cloud, WiFi Connection is required when using the M5StickC Plus. This is enabled by default in config.h, as such there is a requirement to modify the WIFI_SSID and WIFI_PASS options in the config.h tab.

12. Note also that to access the Feeds in Adafruit IO, authentication is required. As such, username and API_Token is required to be configured in the config.h file.

13. To make changes to the config.h file, click on config.h tab in Arduino as shown below.

```
ege356_lab1 §    config.h §
/************************ Adafruit IO Config ******************************/

// visit io.adafruit.com if you need to create an account,
// or if you need your Adafruit IO key.
#define IO_USERNAME "your_username"
#define IO_KEY "your_key"
```

Click on My Key to obtain the API token and username required to access the Feed and add to the code here

Click to view API_Token

Username [   ]

Active Key [aio_]   REGENERATE KEY

Hide Code Samples

Data here has been blanked out, students should be able to see their own API_Tokens and Username when logged in

Arduino

```
#define IO_USERNAME  "        "
#define IO_KEY       "aio_fB                        "
```

14. Modify the SSID and password as shown below. Obtain the SSID and password used for the Lab from the instructor during class. When at home, use the Wifi SSID and password available at home.


```
ege356_lab1 - config.h | Arduino 1.8.13
File Edit Sketch Tools Help

ege356_lab1 §    config.h §
//    - HUZZAH ESP8266 Breakout -> https://www.adafruit.com/products/2471
//    - Feather HUZZAH ESP8266 -> https://www.adafruit.com/products/2821
//    - Feather HUZZAH ESP32 -> https://www.adafruit.com/product/3405
//    - Feather M0 WiFi -> https://www.adafruit.com/products/3010
//    - Feather WICED -> https://www.adafruit.com/products/3056
//    - Adafruit PyPortal -> https://www.adafruit.com/product/4116
//    - Adafruit Metro M4 Express AirLift Lite ->
//    https://www.adafruit.com/product/4000
//    - Adafruit AirLift Breakout -> https://www.adafruit.com/product/4201
//    - Adafruit AirLift Shield -> https://www.adafruit.com/product/4285
//    - Adafruit AirLift FeatherWing -> https://www.adafruit.com/product/4264

#define WIFI_SSID "your_ssid"
#define WIFI_PASS "your_pass"
```
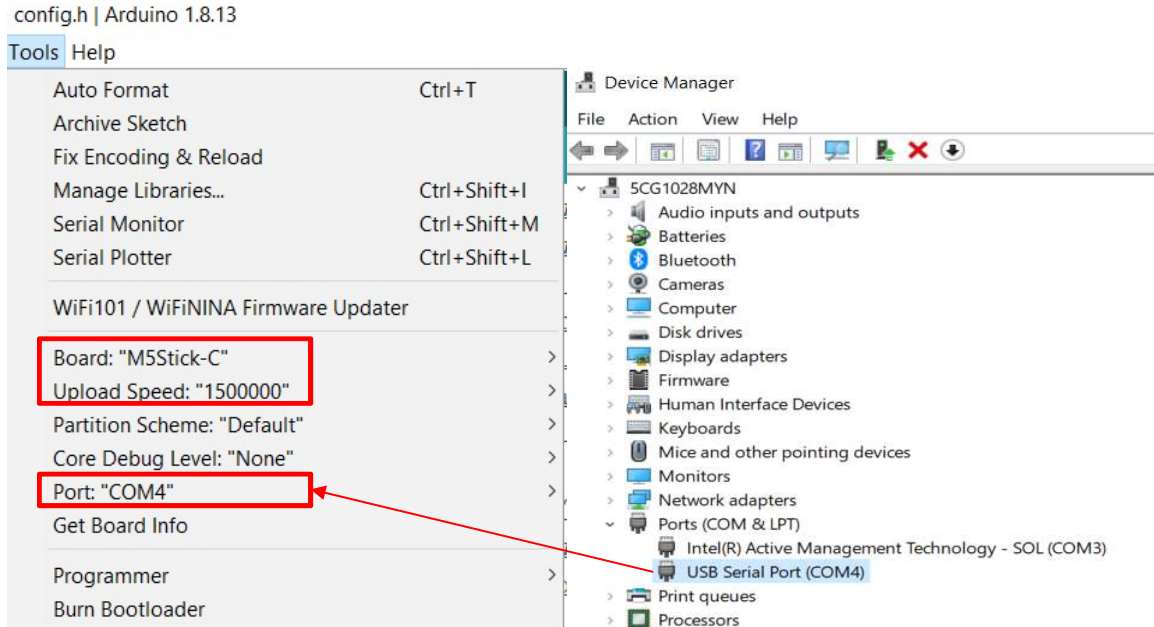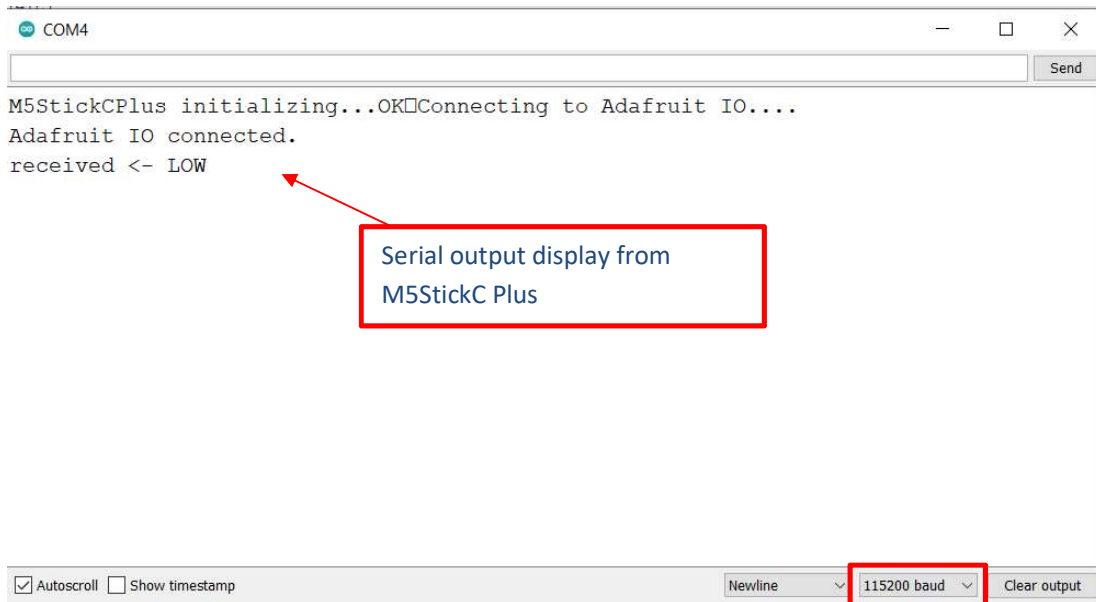
15. Connect up the M5StickC Plus to the Computer USB port and load the code.

16. Upon loading the code, click the serial monitor icon to view the serial output display. Ensure that the baud rate is 115200 as highlighted in red. Note the following output.

Note that RED LED is OFF, and device displays IP address.

17. When the Button is Toggled ON, the RED LED will light up and the Serial output message will be as shown.

Serial output display from M5StickC Plus, LED Toggle turns on RED LED

18. Summary: The codes in this lab enables the sending of data from cloud to edge device and data sent is in bytes form.