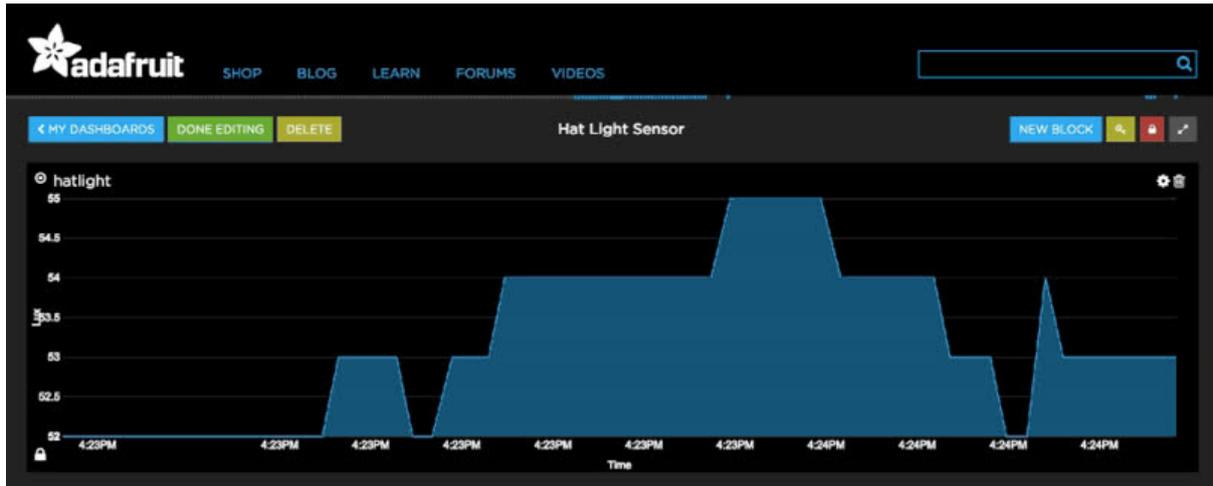




Datalogging Hat with FLORA BLE

Created by Becky Stern



<https://learn.adafruit.com/datalogging-hat-with-flora-ble>

Last updated on 2024-06-03 01:48:05 PM EDT

Table of Contents

Overview	3
Circuit and Arduino code	3
Bluefruit LE Connect settings	5
Adafruit IO Dashboard display	11

Overview

Log your wearables data to the cloud! Using the Bluefruit LE Connect app and Adafruit IO, you can display the changing values of a FLORA light sensor in realtime on a webpage.

This is a great upgrade to make to your [Sunscreen Reminder Hat \(https://adafru.it/dNc\)](https://adafru.it/dNc) or [Sparkle Skirt. \(https://adafru.it/dN1\)](https://adafru.it/dN1)

This guide focuses on getting your data from the Bluefruit LE Connect app to the Adafruit IO servers, so if you're a beginner with FLORA, try a few basic projects first! Here are some prerequisite guides to check out:

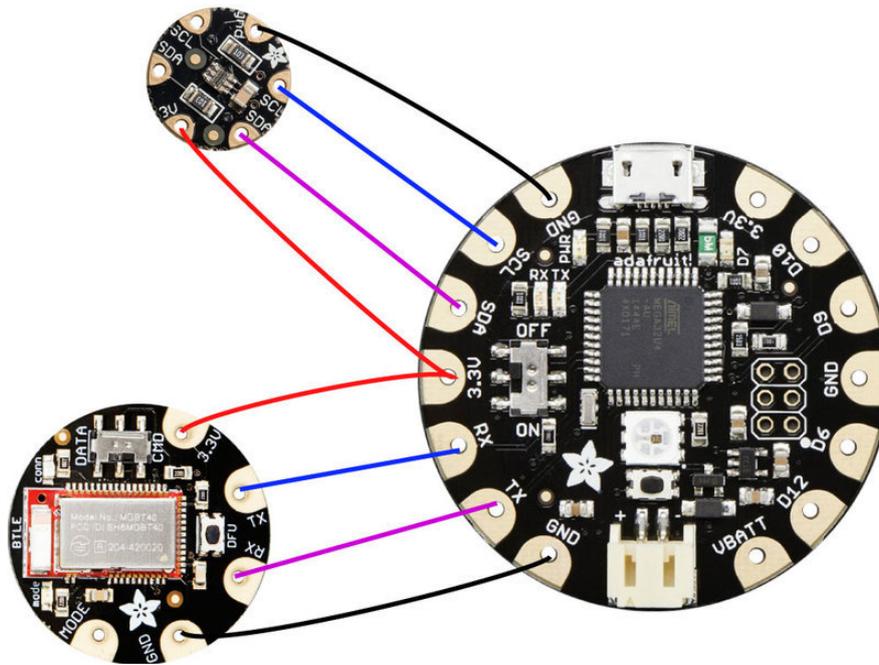
- [Getting Started with FLORA \(https://adafru.it/dwi\)](https://adafru.it/dwi)
- [FLORA Bluefruit LE module \(https://adafru.it/ioe\)](https://adafru.it/ioe)
- [FLORA Lux Sensor \(https://adafru.it/dNa\)](https://adafru.it/dNa) (or [FLORA Sensors \(https://adafru.it/iof\)](https://adafru.it/iof) if you want to swap to a different one)
- [Adafruit IO Basics: Feeds \(https://adafru.it/ioA\)](https://adafru.it/ioA)
- [Adafruit IO Basics: Dashboards \(https://adafru.it/f5m\)](https://adafru.it/f5m)

For this example, you will need:

- [FLORA microcontroller \(http://adafru.it/659\)](http://adafru.it/659)
- [FLORA Bluefruit LE Module \(http://adafru.it/2487\)](http://adafru.it/2487)
- [FLORA Lux Sensor \(http://adafru.it/1246\)](http://adafru.it/1246)

Circuit and Arduino code

The FLORA, FLORA BLE module, and FLORA light sensor are wired up as pictured:



Required Arduino libraries:

- [Adafruit Unified Sensor Library \(https://adafru.it/aZm\)](https://adafru.it/aZm)
- [Adafruit TSL2561 Light Sensor Library \(https://adafru.it/aZ9\)](https://adafru.it/aZ9)
- [Adafruit Bluefruit LE nRF51 Library \(https://adafru.it/f4V\)](https://adafru.it/f4V)

Download and unzip the following file, then open up light-logging-hat.ino in your Arduino IDE:

[light-logging-hat.zip](https://adafru.it/ioC)

<https://adafru.it/ioC>

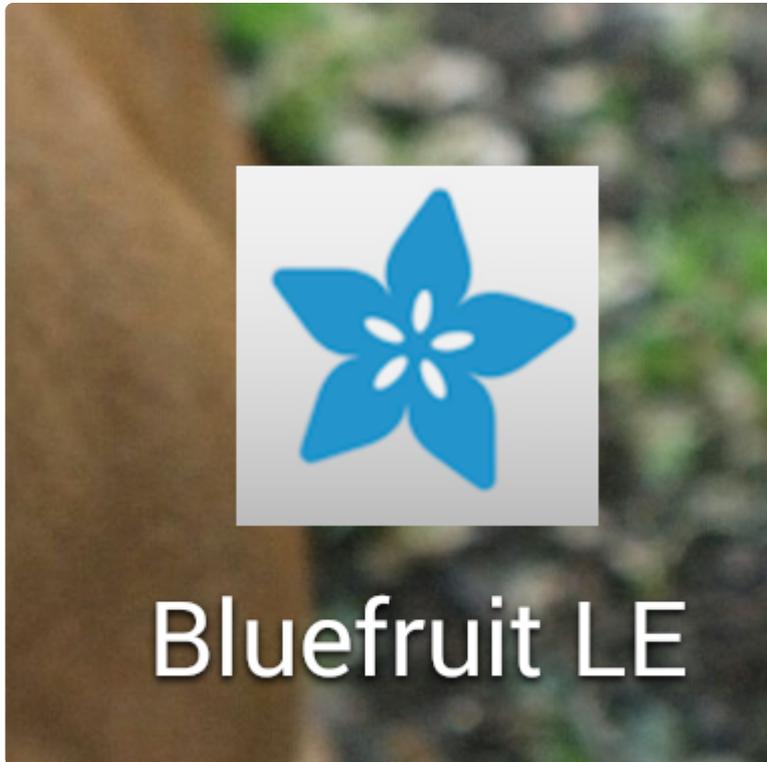
The main loop of the code reads the light sensor and sends the value out over the bluetooth connection:

```
void loop(void)
{
  /* Get a new sensor event */
  sensors_event_t event;
  tsl.getEvent(&event);

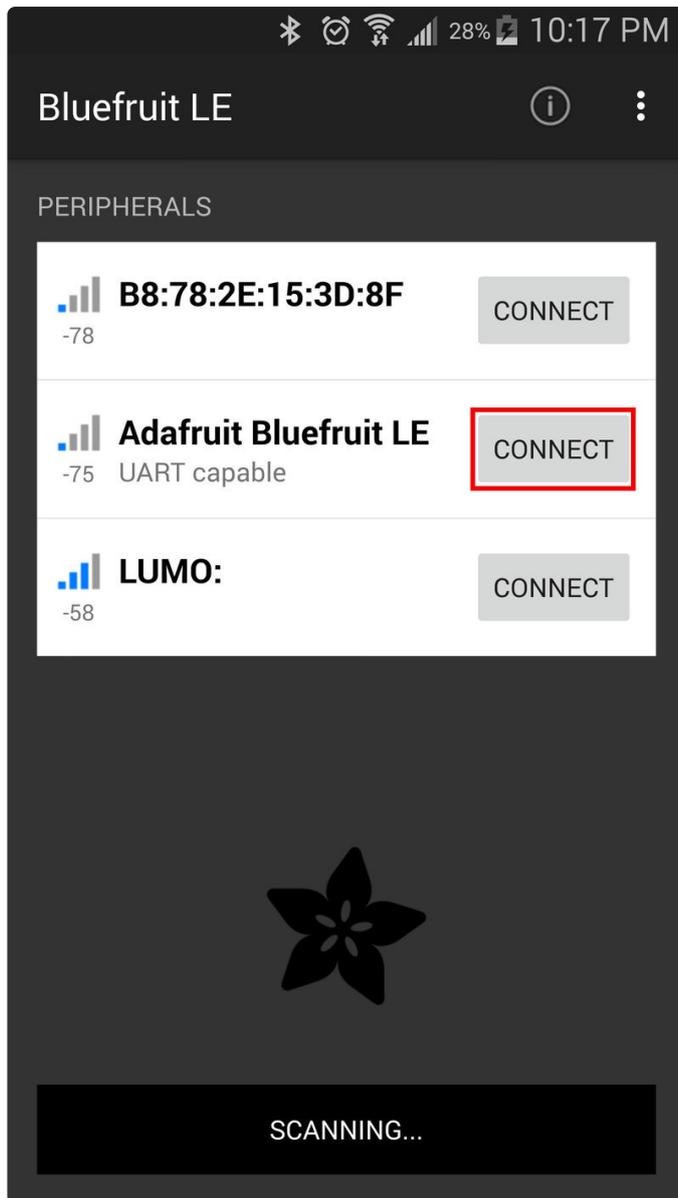
  /* Display the results (light is measured in lux) */
  if (event.light)
  {
    ble.println(event.light);
  }
  else
  {
    /* If event.light = 0 lux the sensor is probably saturated
       and no reliable data could be generated! */
    Serial.println("Sensor overload");
  }
}
```

```
delay(1000);  
}
```

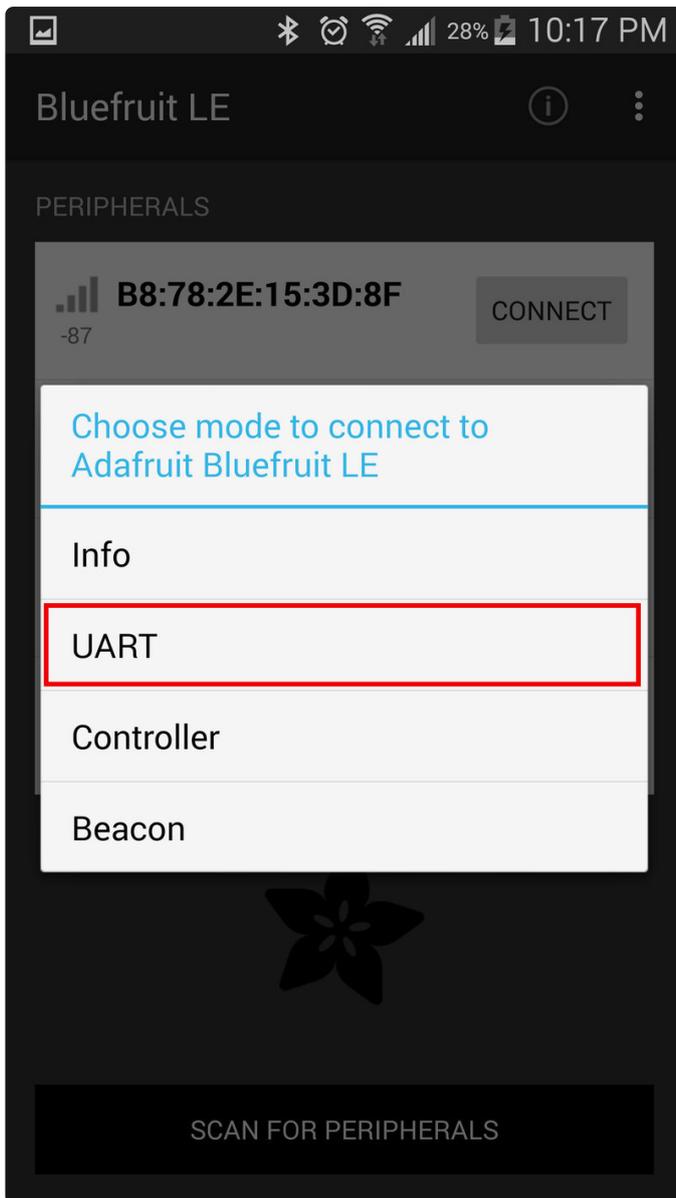
Bluefruit LE Connect settings



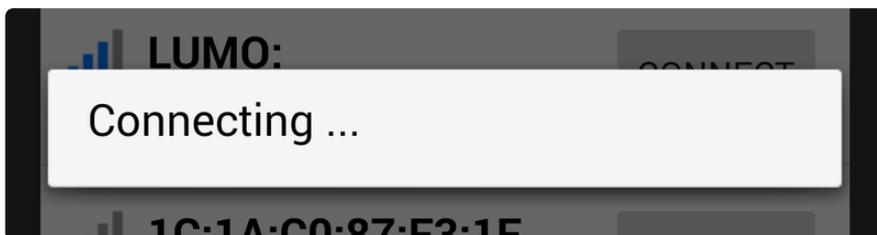
Launch the Bluefruit LE Connect app on your Android or iOS device. Below screenshots are from an Android phone, iOS app may vary slightly.

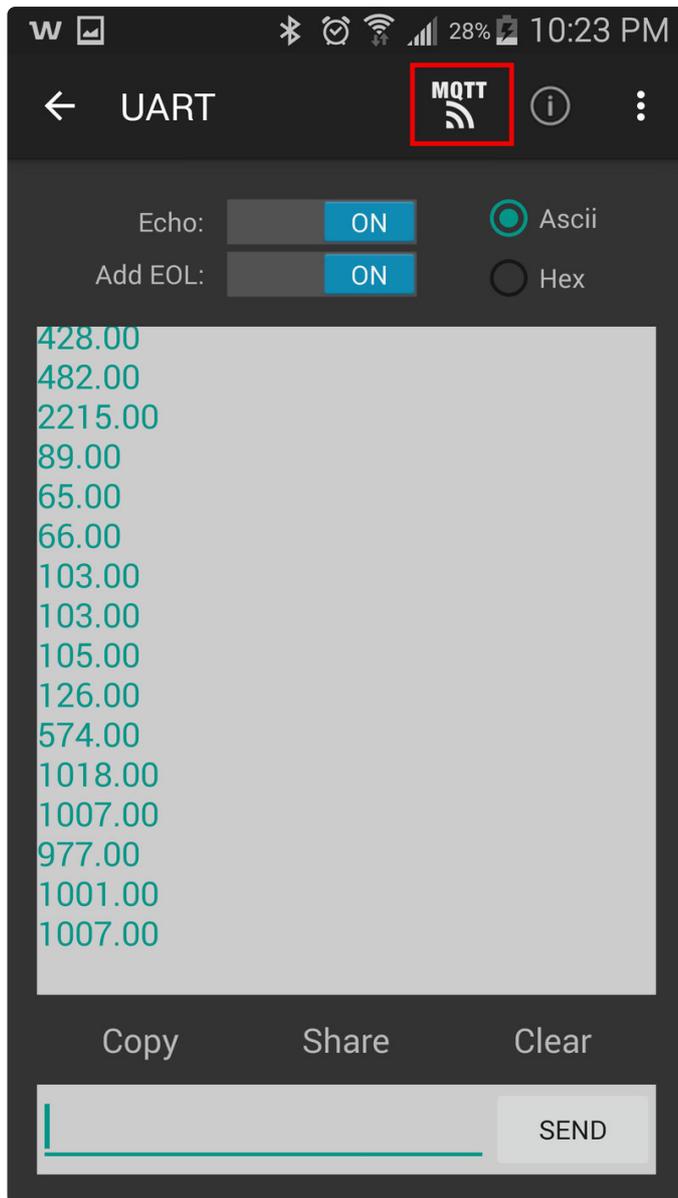


A list of available bluetooth devices will appear. Select the one labeled **Adafruit Bluefruit LE**.

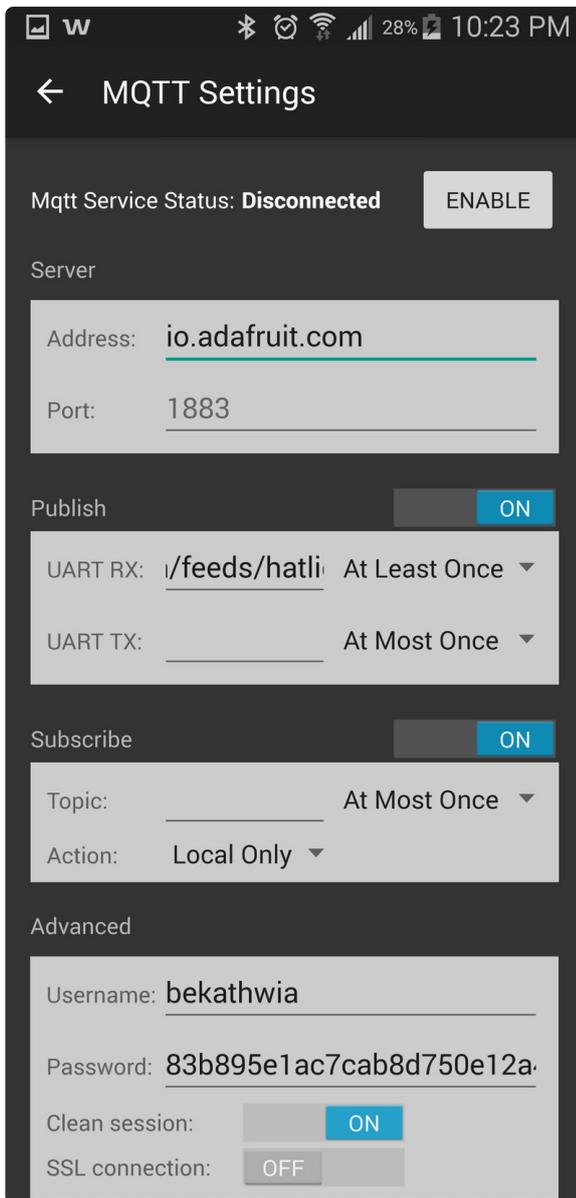


A prompt appears asking you to choose a mode to connect-- select **UART** mode.





If your connection is successful, you should see sensor values update on the screen. If your connection failed, try restarting both the FLORA circuit and the Bluefruit LE app and attempt to connect again.



Configure your first two boxes as follows:

Server

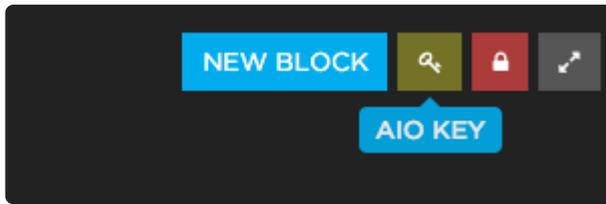
Address: io.adafruit.com

Port: 1833

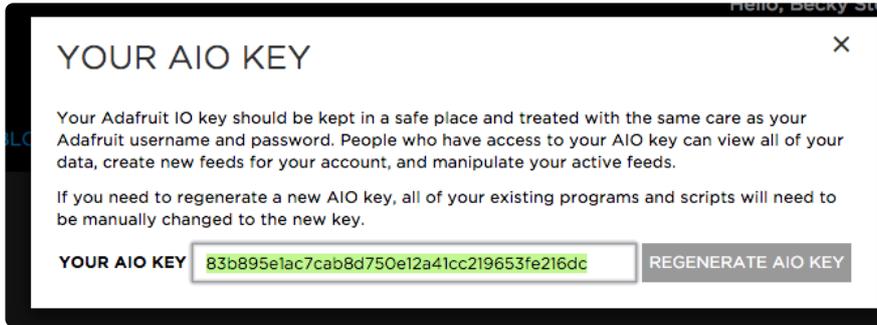
Publish

UART RX: **username/feeds/feedname**

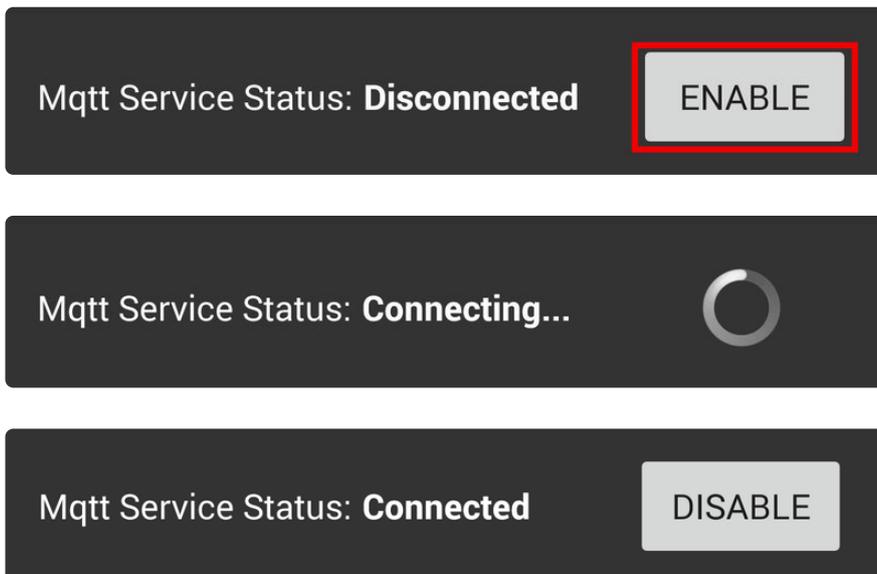
Under "Advanced," enter your username and AIO key, which can be found by clicking the key button in the upper right of any of your Dashboards.



Copy your AIO key into an email, file on dropbox, or other easy way to paste it into your device-- copying it from the screen is no fun (unless you love memorizing)!

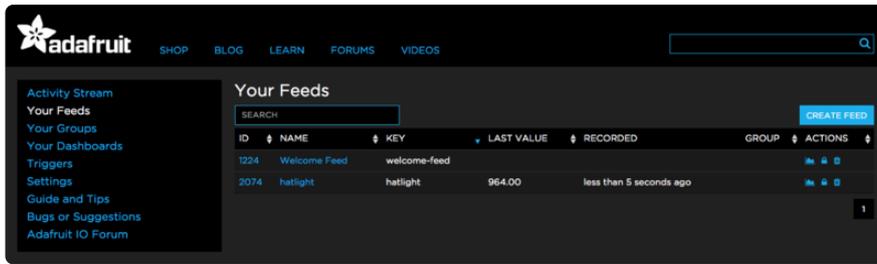


After you've configured your settings, go ahead and click ENABLE at the top of the page.



If the connection isn't successful, double check your settings including your username, UART/RX path, and AIO key.

If you're connected, you can now refresh your Feeds page on io.adafruit.com to see your new data stream. Hurray! Now let's learn how to display that incoming data.

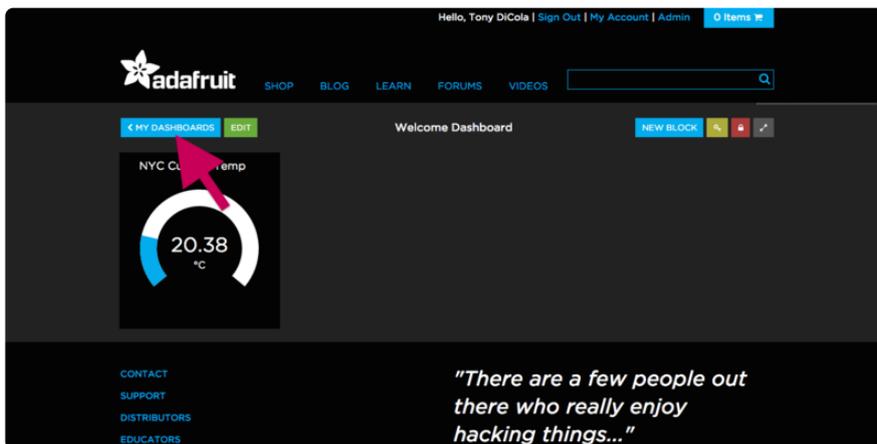


Adafruit IO Dashboard display

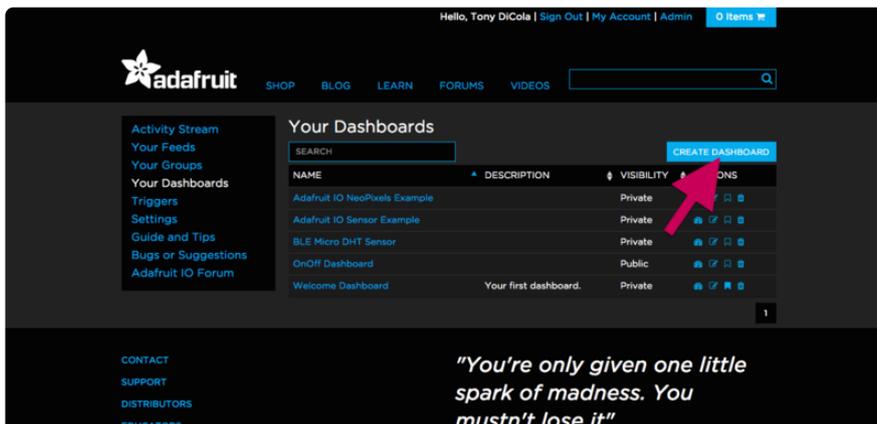
Once the Bluefruit LE Connect application is sending data to Adafruit IO then you're ready to build a dashboard to display the results.

If you're new to using Adafruit IO be sure to first read the [Adafruit IO Basics: Dashboards](https://adafru.it/ioE) (<https://adafru.it/ioE>) guide for an overview of creating a dashboard.

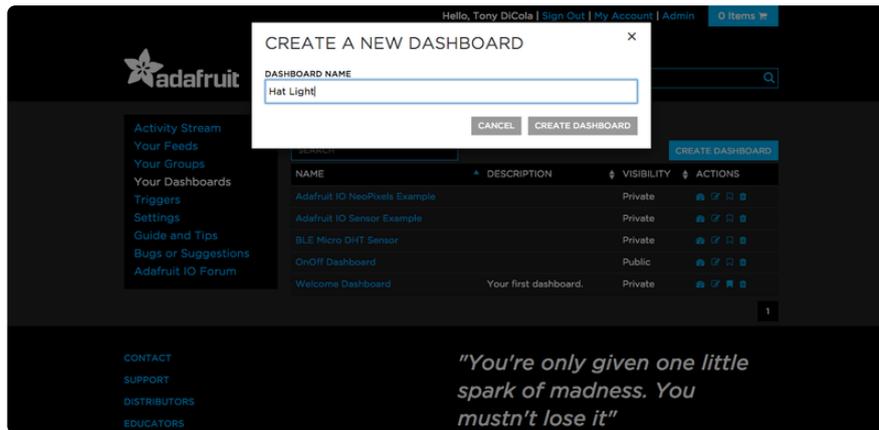
To create the dashboard for the light sensor feed first log in to [Adafruit IO](https://adafruit.io) (<https://adafruit.io>) (<https://io.adafruit.com> (<https://adafru.it/eZ8>)) and click the **My Dashboards** button in the upper left:



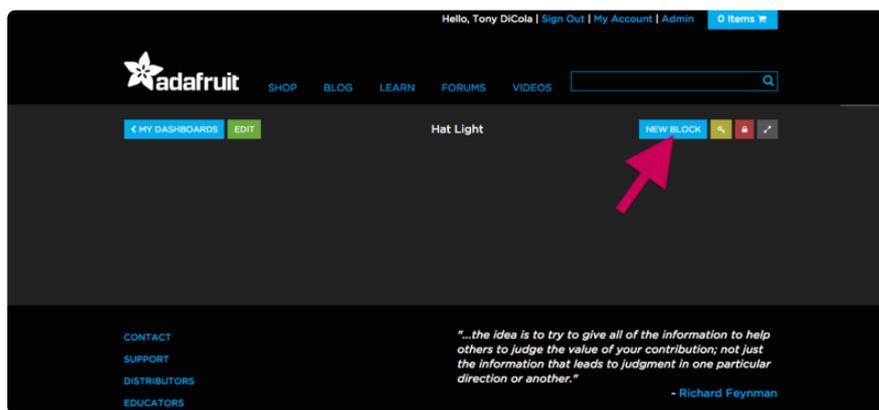
Once on the **Your Dashboards** page click the **Create Dashboard** button in the upper right to build a new dashboard:



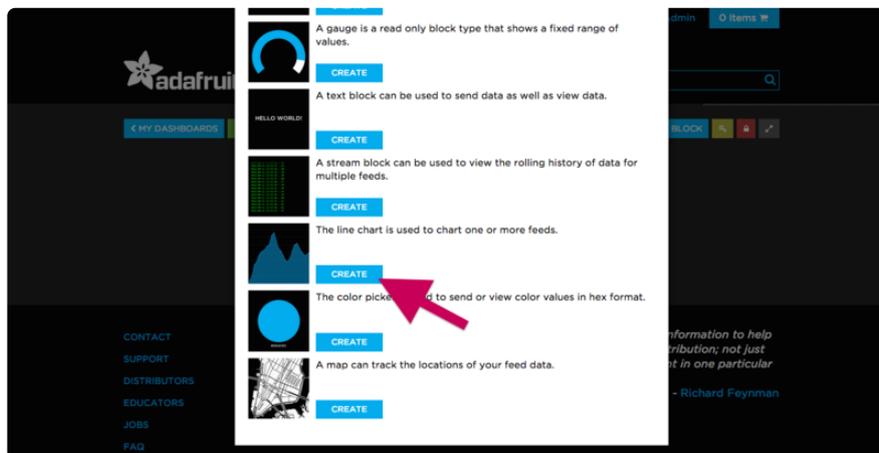
In the dialog box that pops up give the dashboard a good name, like Hat Light, then click **Create Dashboard**:



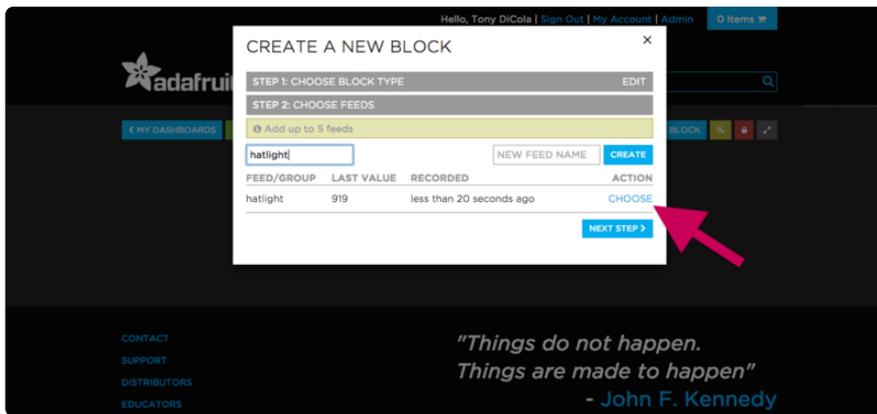
Now the new Hat Light dashboard is displayed, but it doesn't have any blocks that show sensor data. You can add a line chart block to see the light sensor value over time by clicking the **New Block** button in the upper right corner:



Scroll down to the line chart component and click **Create**:



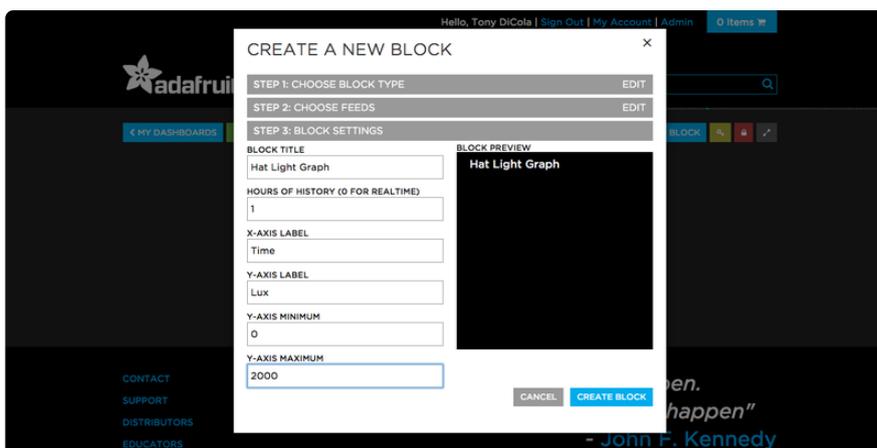
Now find the hatlight feed that was created by the Bluefruit LE Connect application. You can scroll through the feed list or type in the name to the search box. Once you find the feed click the **Choose** button next to it, then click **Next Step**:



Finally make the block easy to understand by filling in details about how it should be displayed:

- **Block Title:** Put in a descriptive title for the block, like **Hat Light Graph**.
- **Hours of History:** Stick with the default of 1 hour, or change to a larger/smaller range.
- **X-Axis Label:** Stick with the default **Time** since the X axis (left to right axis of the graph) is based on the time of each sensor reading.
- **Y-Axis Label:** Label this axis (top to bottom axis) **Lux** since it's the axis for the light sensor reading.
- **Y-Axis Minimum:** Use a value like **0** as a good minimum range for the graph values (since you know there are no negative lux values).
- **Y-Axis Maximum:** Use a large value like **2000** as a starting point, you can always edit the block and change this later to 'zoom in' on a range of data.

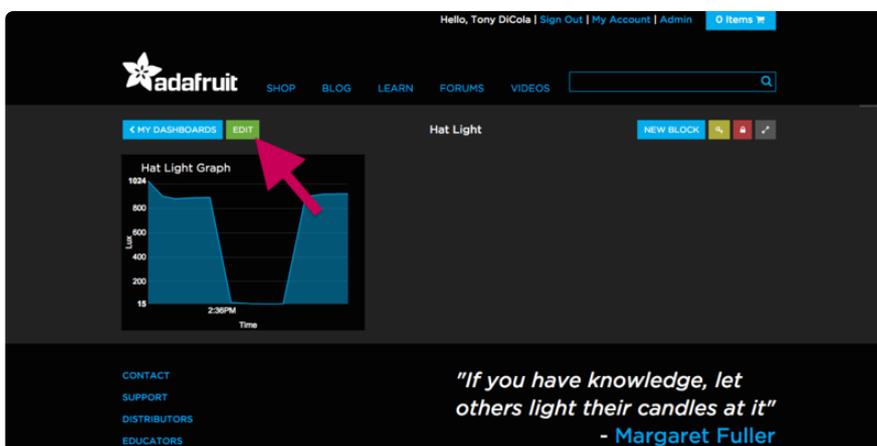
Once you've filled in the details click **Create Block**:



Awesome, the chart block should now be displayed!

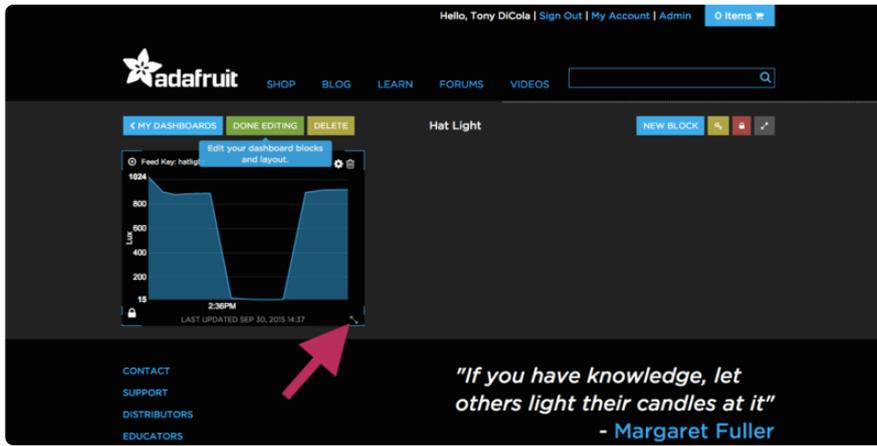


You can change the size, location, and details of the chart block by clicking the **Edit** button in the top left:



Once you're in edit mode you can move the mouse over a block to make changes to it:

- To change the block's settings click the **gear icon** in the upper right of the block.
- To delete the block click the **trash can icon** in the upper right of the block.
- To move the block click and drag it anywhere on the dashboard.
- To resize the block click and hold the **resize arrows** in the lower right of the block as shown below:



Drag the mouse while holding the resize arrows to change the size of the chart block.

Once you're done editing blocks be sure to click **Done Editing** to save the changes!

Let your creativity flow as you design cool dashboards to display sensor data! For example here's what you can do if you change the size of the chart and add a gauge block next to it (remember click **New Block** to add a new block, then follow the steps for customizing it):

