2020 ENGINEER'S WEEK

INTERNET OF THINGS

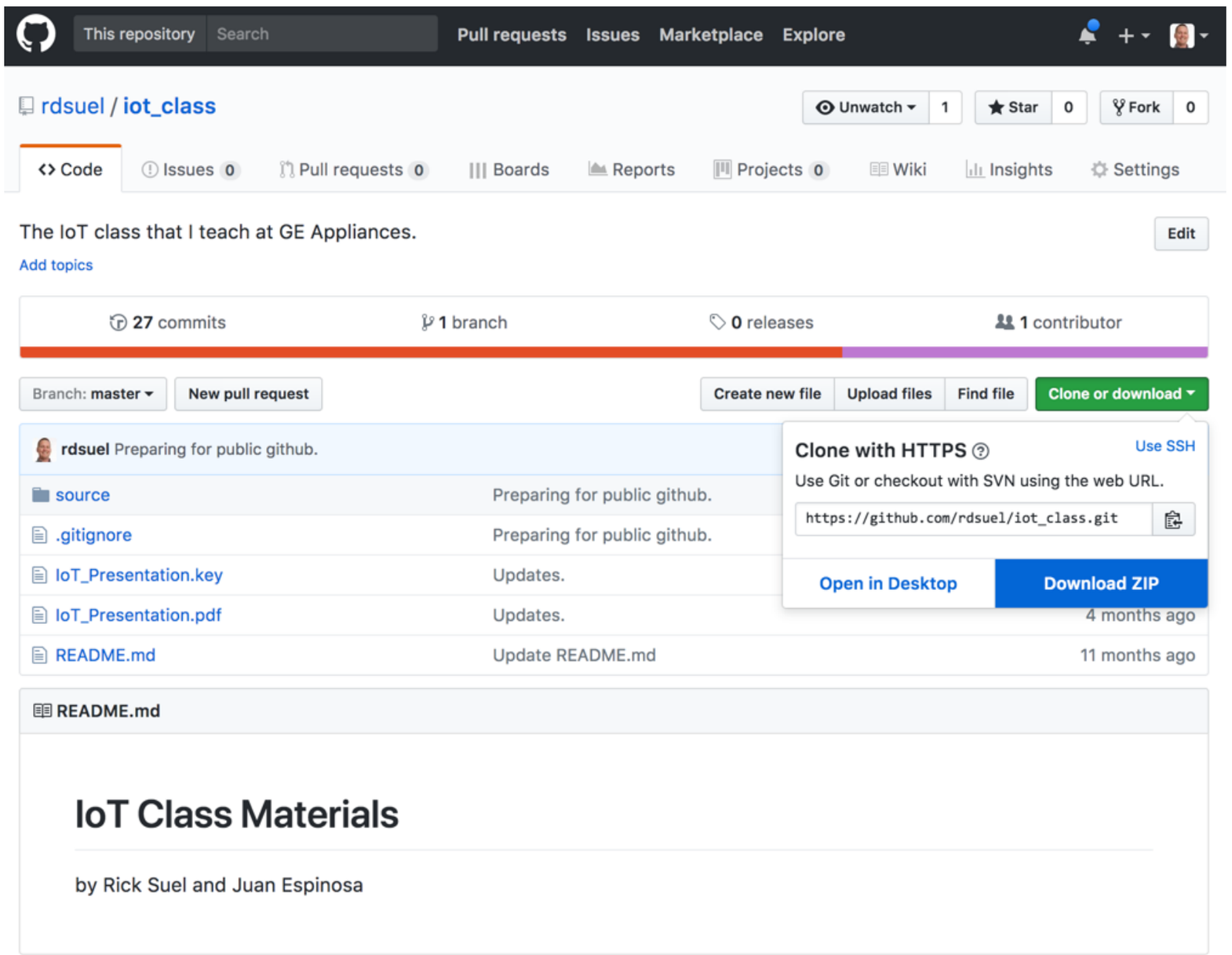
Rick Suel
Paul Goodjohn

CLASS PREREQUISITES

1. Some programming experience (i.e. Arduinos class, etc).
2. Download *Particle* mobile phone app:
   - iPhone: https://itunes.apple.com/us/app/particle-build-photon-electron/id991459054?ls=1&mt=8
   - Android: https://play.google.com/store/apps/details?id=io.particle.android.app
3. Create an account on https://www.particle.io
4. Create an account on https://ifttt.com
5. Create an amazon account if you don't already have one: https://www.amazon.com
6. Log into https://echosim.io with your amazon account.
7. Install the VS code editor https://code.visualstudio.com/ (do not use notepad for this class!)
8. Have the chrome browser installed and ready to go. I have not tested Internet Explorer and will not do so :).

CLASS MATERIALS

https://github.com/rdsuel/iot_class



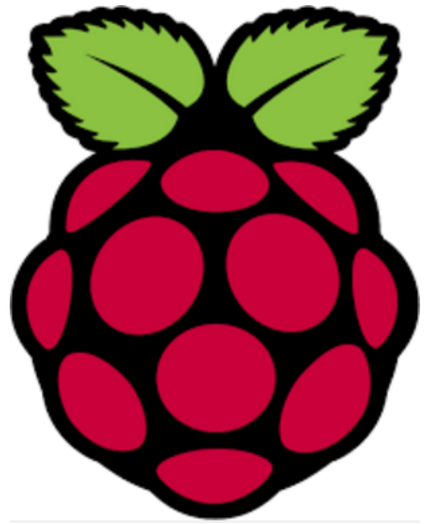
WHAT IS THE "INTERNET OF THINGS" (IOT)?

The Internet of Things includes "connected" embedded devices, sensors, etc that collect and share data, typically via a cloud service, where the data can be shared, analyzed etc. Some applications include:

- Smart home (nest, automation, etc)
- Industrial IoT (supply chain, sensors, GE predix, etc)
- Wearables (smart watches, fitness trackers, etc)
- Energy (smart meters)
- Healthcare
- And many, many more …

SOME DEVICE MANUFACTURERS

- Almost every embedded device manufacturer has an IoT offering. These are just a few such manufacturers:
- Texas Instruments, Intel, NXP, Raspberry Pi, Arduino, Onion, Particle, Electric Imp, and many, many more …



SOME CLOUD SERVICE PROVIDERS

- Similar to device manufacturers, many familiar software companies are now offering IoT cloud solutions:
- Microsoft Azure, Amazon Web Services, Salesforce.com, Google Cloud Platform, IBM Watson, etc …

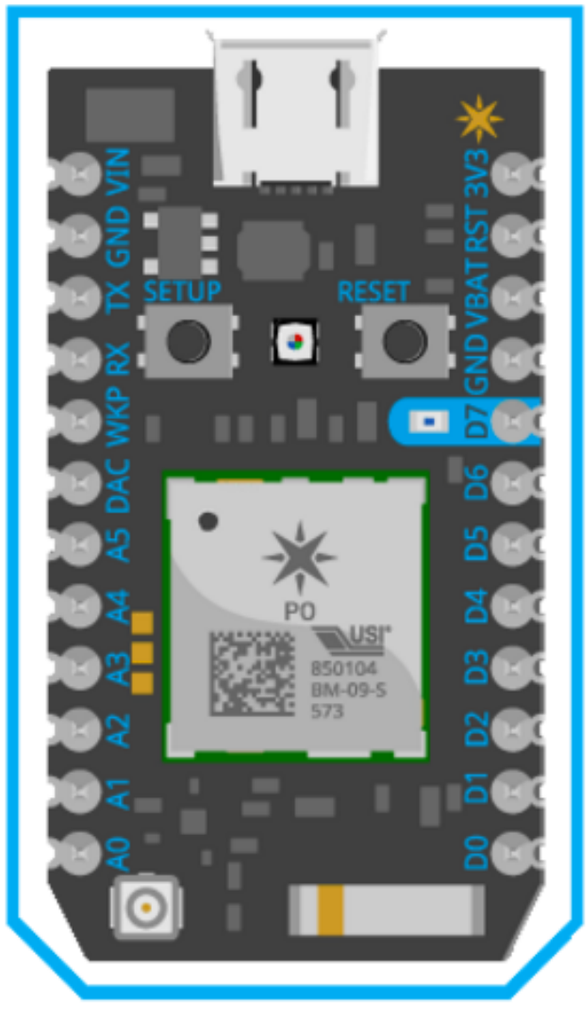
Google Cloud Platform

Watson Internet of Things

PARTICLE PHOTON ($19)

- We will be using the www.particle.io photon device and particle cloud for the duration of this class.
- This device is very simple to get started with, and even uses “Arduino-style” coding right in your web browser.
- Start here: https://docs.particle.io/guide/getting-started/intro/photon/
- Photon Features:
  - 802.11b/g/n Wifi
  - 120MHz Arm Cortex M3 micro
  - 1MB Flash, 128KB RAM
  - 18 Mixed-signal GPIO pins and advanced peripherals (ADC, DAC, SPI, I2C, CAN, USB, PWM).

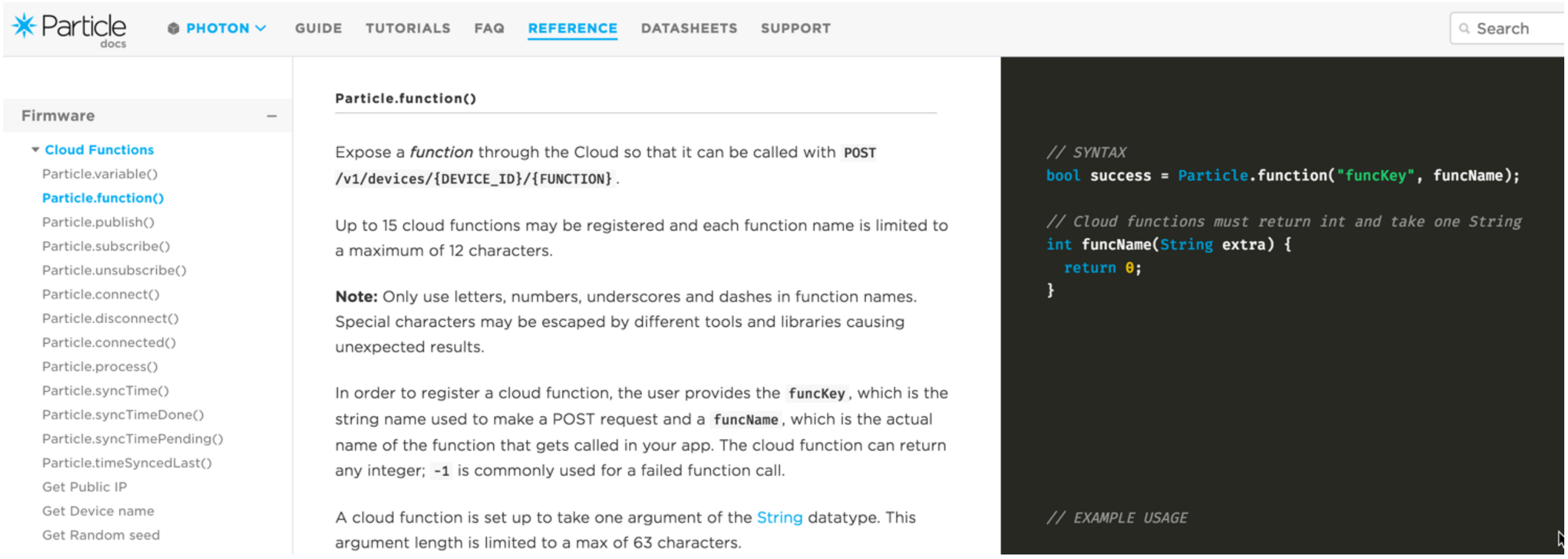
https://docs.particle.io/datasheets/photon-datasheet/



PARTICLE CLOUD API

Where to find Particle API documentation:

https://docs.particle.io/reference/firmware/photon/



SETUP AND "TINKER"

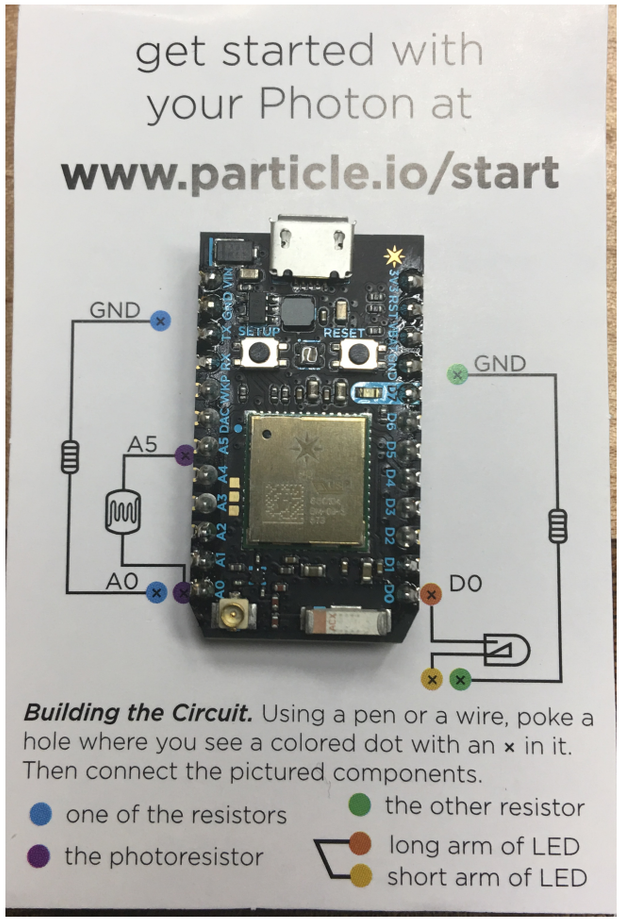
Goals:

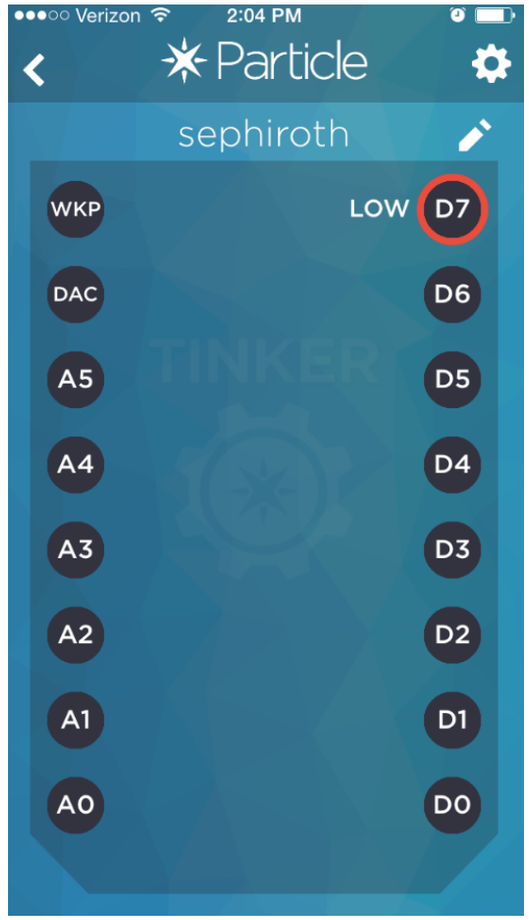
- Connect your photon to wifi.
- Interact with the device over the internet using your smart phone.

Steps:

1. Wire the circuit as shown in the kit.
2. Connect your photon to the "iot_class" wifi network using the Particle mobile app. The password is "password".
3. Use the "Tinker" utility built into the Particle app to drive the LED and read the light sensor in your circuit.

Alternate setup method: https://setup.particle.io/

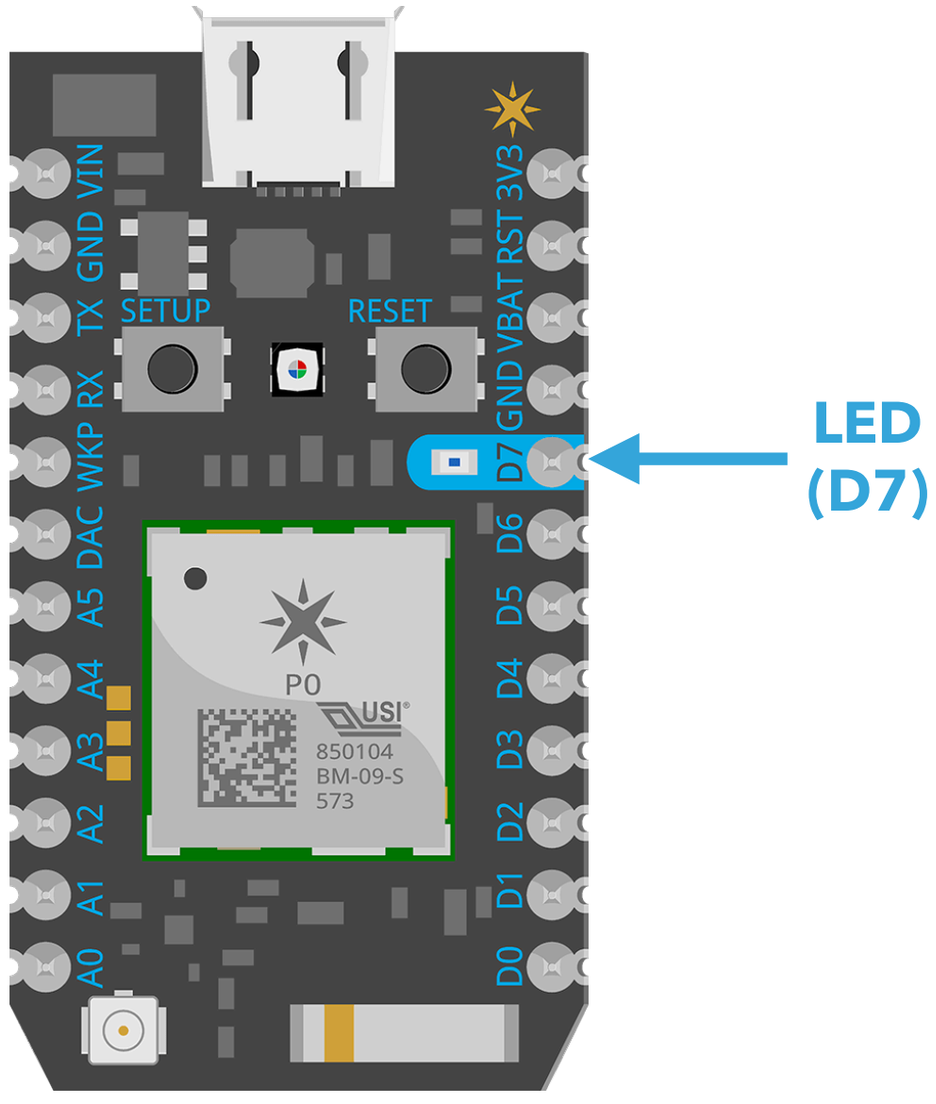




BLINKY

Goals:

- Blink the on-board LED (D7) once per second (500ms on 500ms off).
- Learn about software timers!



Steps:

1. Use a software Timer to blink the on-board LED.
2. Alternatively, you could do this in the loop function, but this is less precise.

Alternate setup method: https://setup.particle.io/

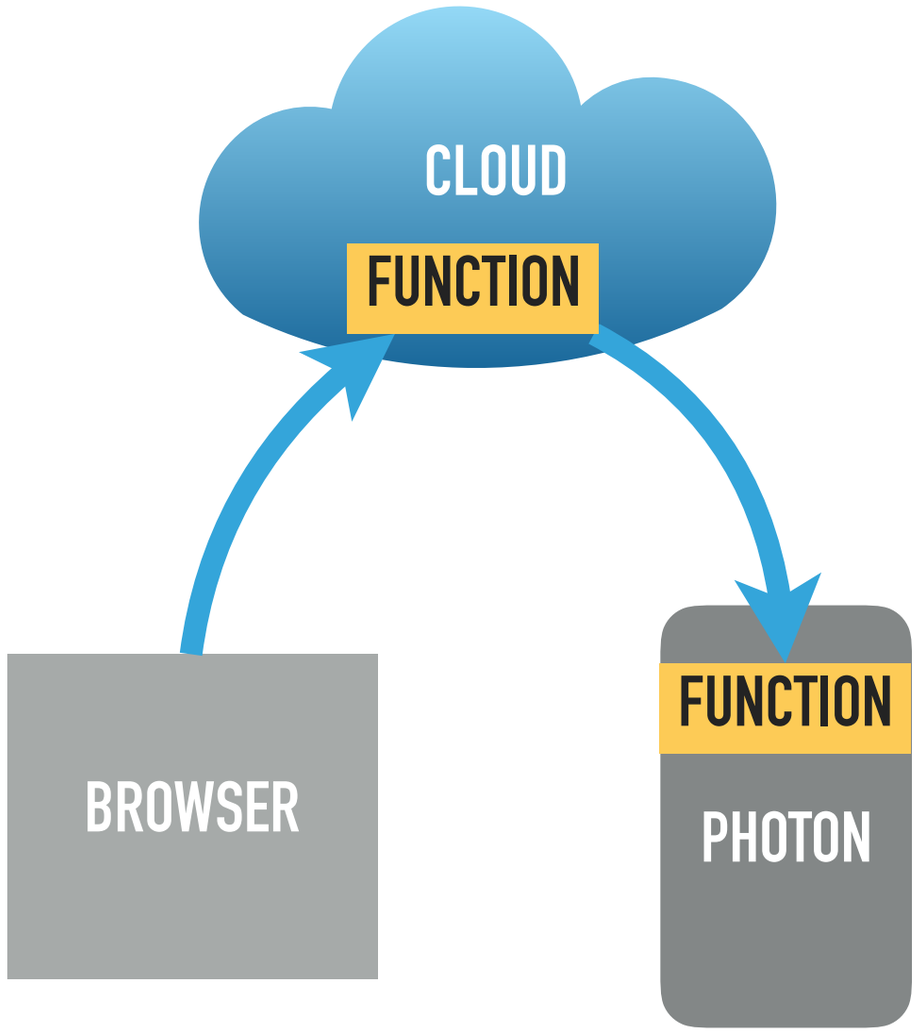
CONTROLLING AN LED WITH YOUR BROWSER

Goals

- Publish a "Web Function" from your photon.
- Call that function using a web browser to blink an LED on the photon.

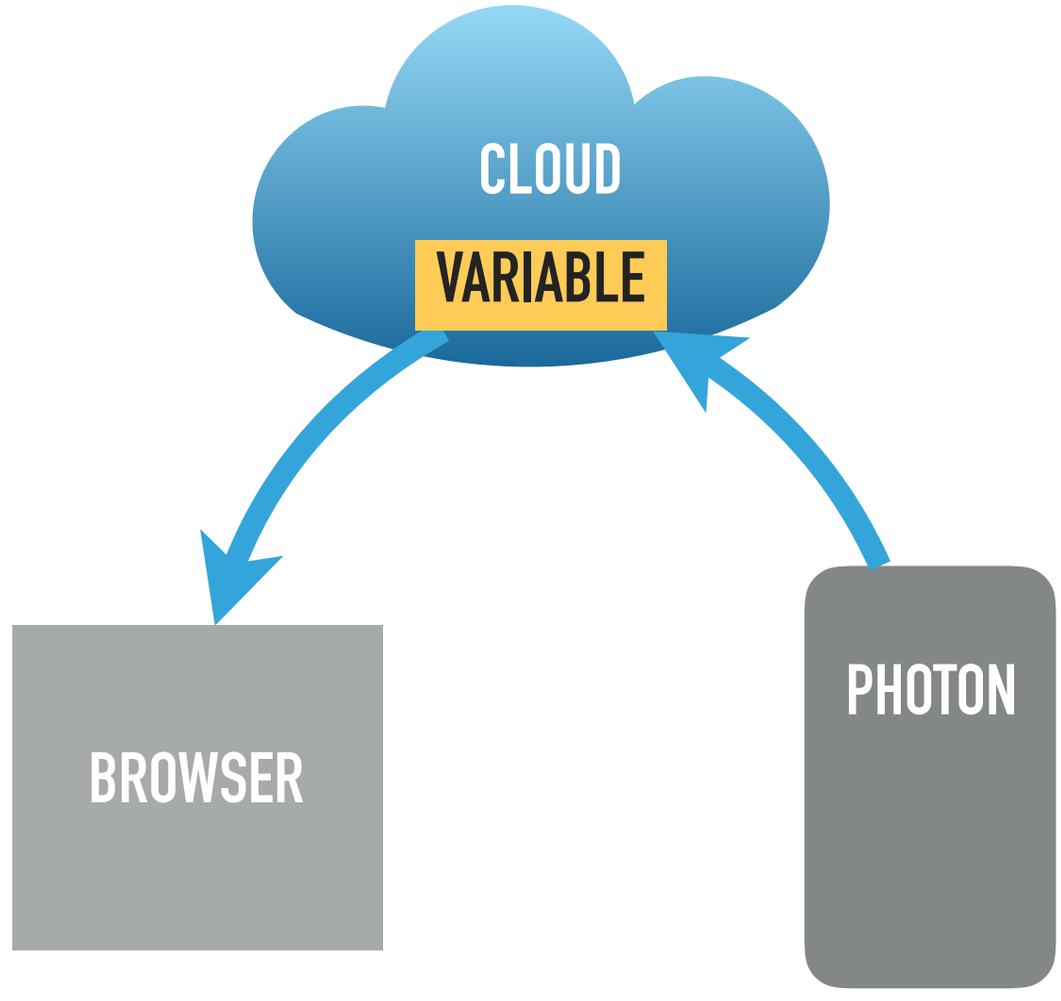
Steps:

1. Write the firmware to add a "web function" to your particle called "**SetLed**". This function will be used to control the LED over the web by writing "**ON**" and "**OFF**" to the web function.
2. Test your web function using the console (console.particle.io).
3. Add your device ID and access key to the HTML template.
4. Turn your LED on and off through the web browser.



Firmware: https://docs.particle.io/reference/firmware/photon/#particle-function
HTML API: https://docs.particle.io/reference/api/#call-a-function

READ A SENSOR WITH YOUR BROWSER



Goals:

- Publish a "Variable" to the cloud that holds the light sensor value.
- Read the "Variable" from the cloud with your web browser.

Steps:

1. Write the firmware to add a "web variable" called "**SensorValue**" to your particle. Update the variable with the photo sensor value periodically in the loop() function.
2. Verify the web variable using the console (console.particle.io).
3. Update the HTML template to add your device ID and access key (your photon's "address" in the cloud).
4. Update the HTML to periodically read the photo sensor value from the cloud and displays it in the browser window.

Firmware: https://docs.particle.io/reference/firmware/photon/#particle-variable

HTML API: https://docs.particle.io/reference/api/#get-a-variable-value

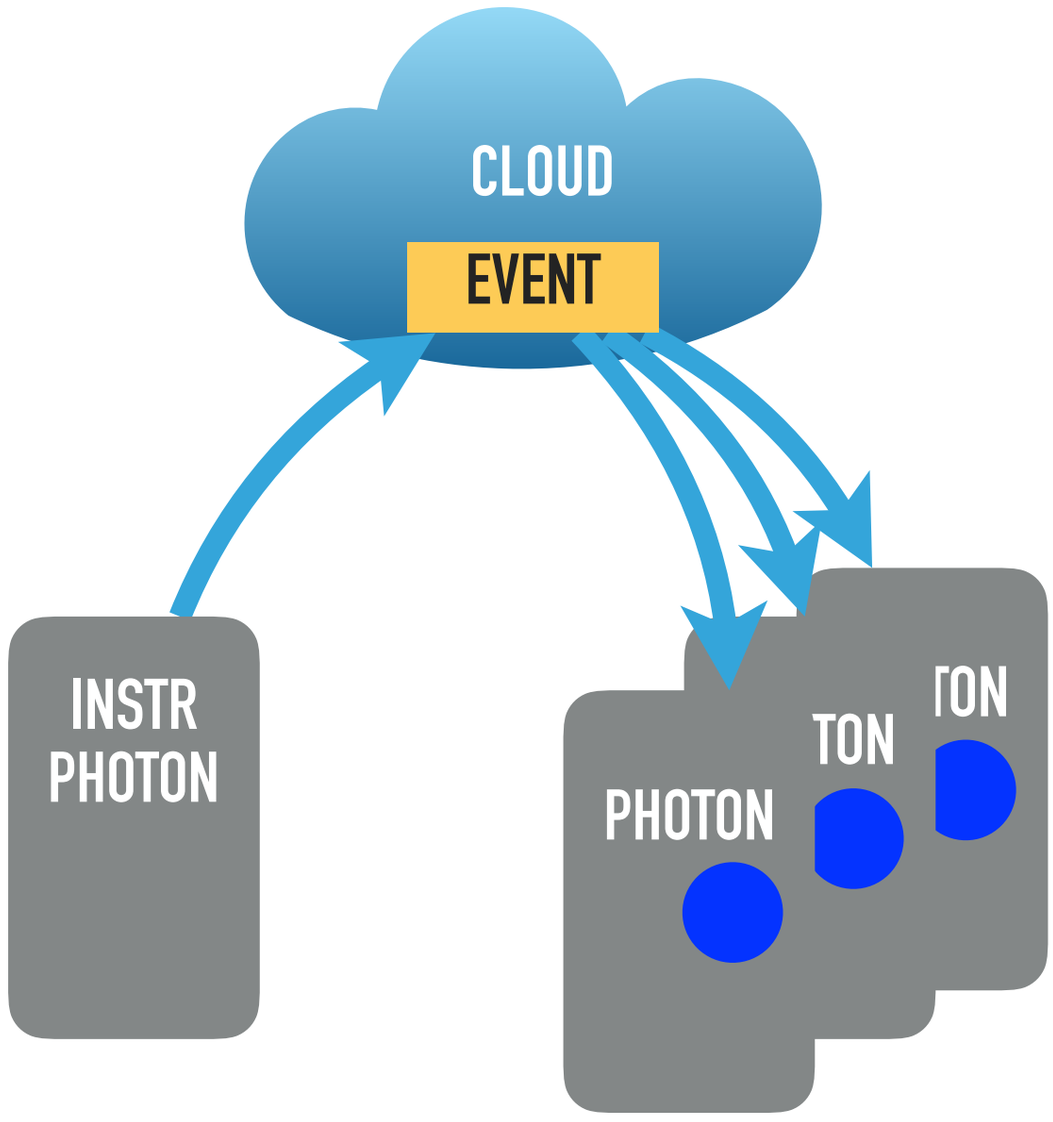
DEVICE-TO-DEVICE COMMUNICATION

Goals:

- Subscribe to the web event “night” and control the LEDs based on the event details.

Steps:

1. Add a Particle.subscribe() handler for the “**NightEvent**” event.
2. When the event is published with the value “**ON**”, turn both LEDs on (D0 and D7) and stop blinking D7.
3. When the event is published with the value “**OFF**”, turn both LEDs off (D0 and D7) and resume blinking D7.
4. The instructors can then control all photon LEDs in the class by publishing the “night” event with the value “ON” or “OFF”.



Subscribe: https://docs.particle.io/reference/firmware/photon/#particle-subscribe
Publish: https://docs.particle.io/reference/firmware/photon/#particle-publish

CONTROLLING AN LED WITH ALEXA AND IFTTT

If You say "Alexa trigger light on", then call a function

Goals:

- Create an IFTTT recipe that turns on a photon LED when an Alexa command is issued.

Steps:

1. Log into ifttt.com and create a new applet by clicking on your profile icon in the upper right corner and selecting “Create”
2. Select “Amazon Alexa” for *this* and choose the “Say a specific phrase” Trigger.
3. Set the phrase to “light on” and press “create trigger” button.
4. Select “Particle” for *that* and choose “Call a function” for the Action.
5. Select your “SetLed” function on your photon device.
6. Type “ON” in the “with input (Function Input)” field.
7. Press the “Create Action” button.
8. On the final screen hit “Finish”.
9. Go to echosim.io, log in, and say” Alexa trigger light on”.

HELPFUL LINKS

- Particle documentation: https://docs.particle.io/guide/getting-started/intro/photon/
- Particle web IDE: https://build.particle.io/build
- Particle console: https://console.particle.io/devices
- Particle community: https://community.particle.io/
- Project ideas: https://www.hackster.io/particle
- Great site for finding/understanding web APIs: http://www.programmableweb.com/
- Losant is powerful cloud platform. Here is how to connect your photon to it: https://docs.losant.com/getting-started/losant-iot-dev-kits/builder-kit-particle/
- plot.ly cheat sheet: https://images.plot.ly/plotly-documentation/images/plotly_js_cheat_sheet.pdf