

## Create IFTTT Trigger

1. Describe how your system works in writing and clearly outline the following: the schematic of the circuit board (breadboard), the overall infrastructure of the various parts of the system, the IFTTT trigger mechanism, and the notification mechanism.

- Pin configuration
  - **Photoresistor - Particle Argon**
  - GND - GND
  - VCC - VUSB
  - OUT - A0

- Code

```
1  const int thresh = 2500; // Threshold to trigger the event
2  int lightIntensity = 0; // Light intensity variable
3  int photoResistorPin = A0; // Analog pin that the sensor is connected to
4  bool isSunUp = false; // Stores previous event
5
6
7  // setup() runs once, when the device is first turned on.
8  void setup() {
9      Serial.begin(9600); // Serial for debugging
10 }
11
12 // loop() runs over and over again, as quickly as it can execute.
13 void loop() {
14     lightIntensity = analogRead(photoResistorPin); // Reading sensor value
15
16     Serial.println(lightIntensity); // printing for debugging
17
18     if (lightIntensity <= thresh && !isSunUp) {
19         // if intensity is below the threshold and current event is not already triggered
20         isSunUp = !isSunUp;
21         Particle.publish("sun", "down"); // publish event
22         Serial.println("Sun down event triggered");
23     }
24
25     if (lightIntensity > thresh && isSunUp) {
26         // if intensity is above the threshold and current event is not already triggered
27         isSunUp = !isSunUp;
28         Particle.publish("sun", "up"); // publish event
29         Serial.println("Sun up event triggered");
30     }
31
32     delay(10s);
33 }
```

- IFTTT
  - Create an account and link your particle account to it
  - Create a new applet
  - As "IF-THIS" section add the particle event

4:34

Complete trigger fields

New event published

This Trigger fires when an interesting event comes from a particular device. Send events using Particle.publish.

Particle Account

dathalage@deakin.edu.au

Pro+ Add more accounts

If (Event Name)

sun

Fill in your published event name; ex: monitoring a washing machine?  
Event Name = Wash\_Status

is (Event Contents)

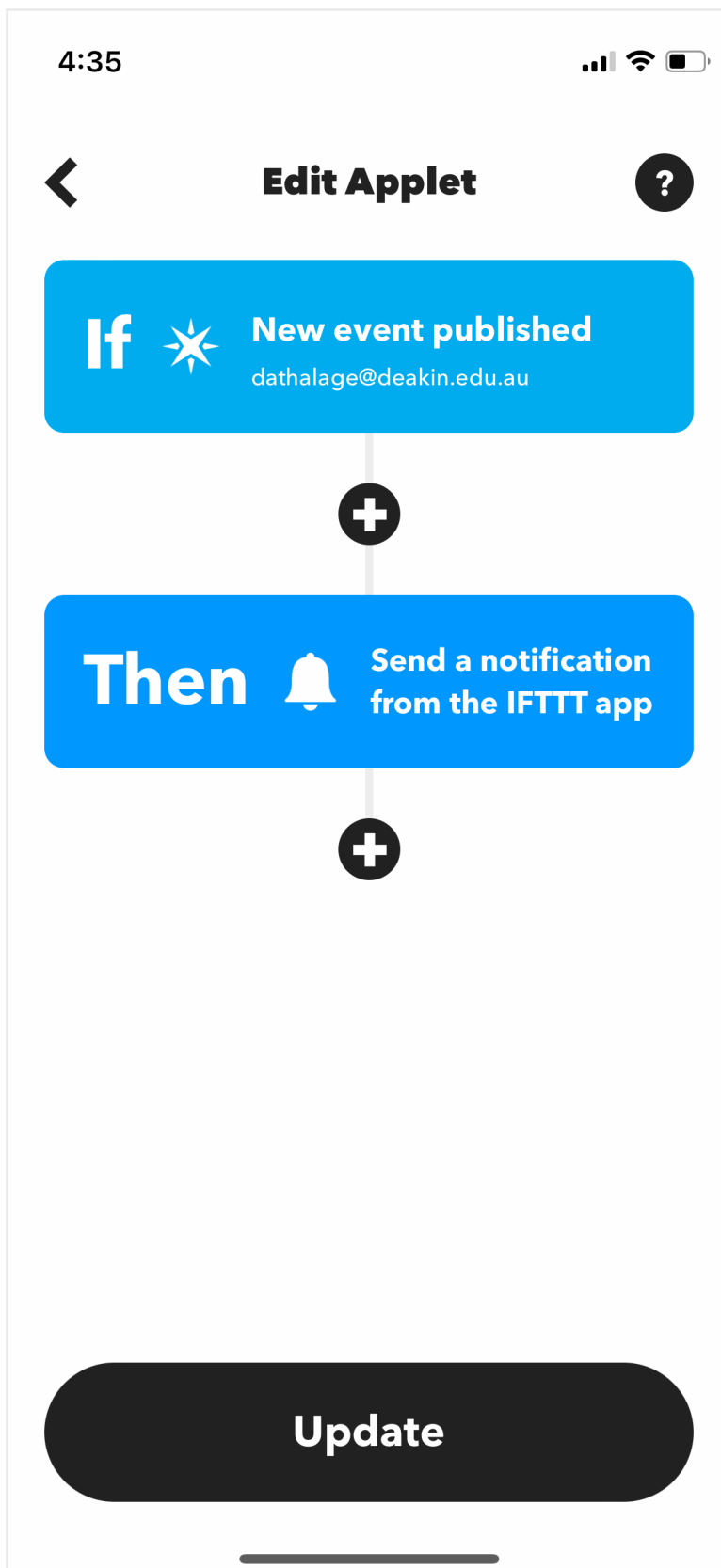
The contents of the published event, "Data"; ex: monitoring a washing machine? Event Contents = Done

Device Name or ID

my-particle

An optional id for a particular device

- as THEN-THAT functionality add app notification
- the applet should look like this



2. Create a repository named SIT210-Task3.2C-ParticleIFTTT on Github. Upload your code to the repository. Include the link to your repository there.

- <https://github.com/Kushan-Nilanga/SIT210/tree/master/ifttt>

3. Produce a video demonstrating your solution. Provide the link in your submission.

- <https://youtu.be/KnJ7Clz1ulQ>

4. In less than two paragraphs, describe how you would test the system you have built?

- First events can be debugged at the microcontroller level by using `Serial.println()`
- Then we can check the published events using particle console
- Next step would be to check the IFTTT integration of particle and check if you are using correct device names and event names
- Finally we can time when an event is triggered from the Particle argon and IFTTT and compare if the outputs have a constant delay. (about 30 seconds for this example)
- We can check the event propagation from the Argon to IFTTT from Serial console, Event log (particle console), Event log(IFTTT)