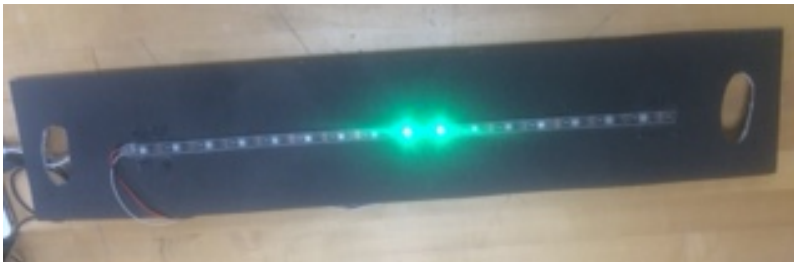
Assignment 2

The idea is to make an interactive game using a sensor that can measure the change in the orientation of a board that has a LED strip with one flashing LED fixed on it. This change in the board orientation will be reflected on the direction of motion of the flashing LED. The game is made for two players each one carries the board from one side (Left and Right). Each player will try to change the orientation if the board when the flashing LED moves towards his/her end. If the Flashing LED reaches to one of the ends it means the player who is on the opposite direction won the game.



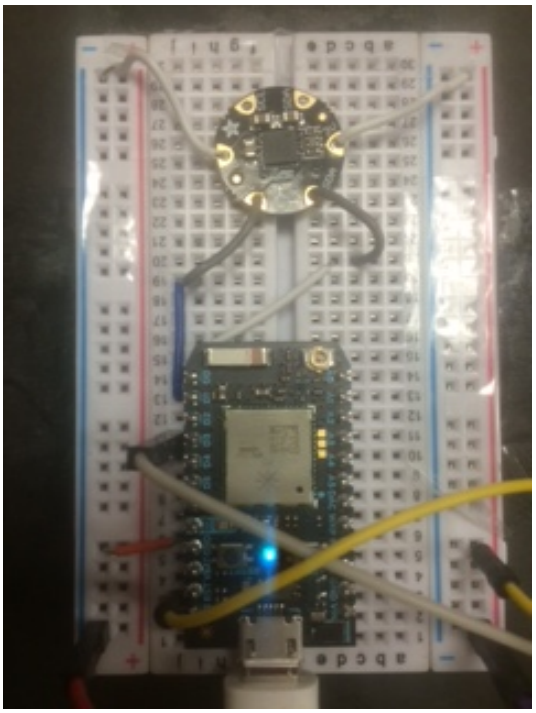
Hardware

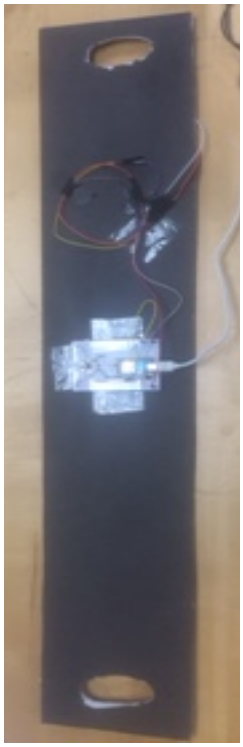
Components used in the project:

- **Output: Neopixel LED strip**
  In this project the LED strip includes 16 LEDS, which can be controlled independently. It needs a 3.3v input voltage.
- **Input: Flora 9DoF IMU**
  It has motion (accelerometer) , direction (magnetometer) and orientation (gyroscope) sensors that take the measurements on the 3axis (x, y, z). The calculation needed in this project involved the measurement taken from the accelerometer in x y and z.
- **Photon**
  Reprogrammable Wi-Fi board used to control and connect the above components to the internet.
- **USB cable, breadboards and couple of jumper wires.**

Connections:

- Neopixel LED Strip: it needs three connections 3.3v, the control signal and the GND.
- Flora 9DoF IMU: the sensor and the photon will communicate over I2C therefore the following connections are needed. SDA, SCL and GND and 3V.
- The photon needs around 5V to power up, it is taken from the computer through the USB port.





Software

The program was written and compiled using Particle Build WebIDE. The following libraries were added:

- **ADAFRUIT_LSM9DS0**

  Functions used:

- getSensor accel , mag ,gyro, temp
- setupAccel , setupMag , setupGyro

- **NEOPIXEL**

  Functions used:

- setPixelColor
- Show

Input sensor measurement and calculation:

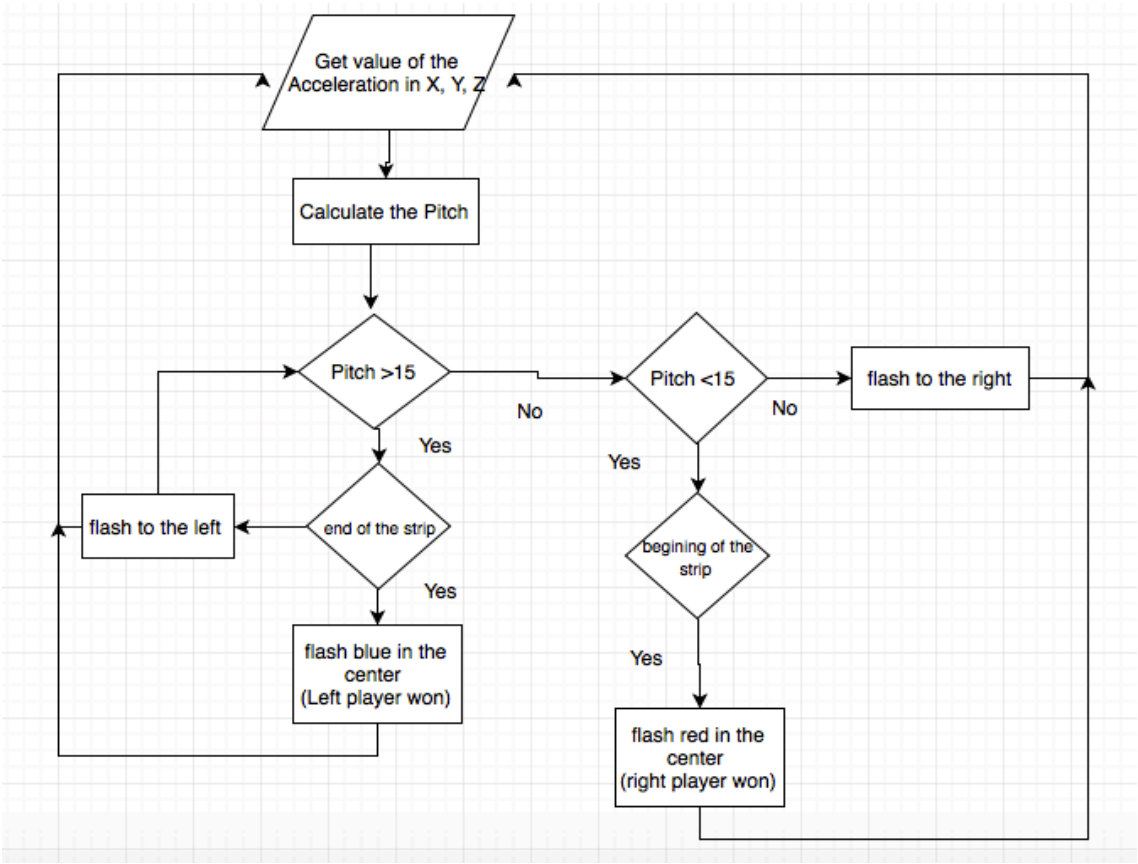
- The scale of the reading of Acceleration sensor set so be ±16 G

- All the calculations needed to get the Acceleration value from the raw readings are taken care of by using the getSensor function.

- The following Equation was used to get the pitch:

  Pitch = arctan (Gy /sqrt((Gx)^2 +(Gy)^2)

Flow chart of the game logic:



Input Visualization part

A Circle will change its color to blue when the pitch >15 or red when pitch <15. When the value is in between it will show different colors.
This was done in a JavaScript code that contains two parts:
Getting the data from particle cloud:

- In the code used to program the photon, the following cloud function was called Particle. publish. It will publish an event called Side in the particle cloud. The "side" event is the value of the pitch. This is called Server Sent Events
- In order to receive those events, the particle cloud is publishing, a new EventSource object was created and the URL of the source that generates the event was specified (particle cloud).
- The event message object we get contains other variable which we don't care about at this point so we need to pull the object of data type which contains a string that represent the value we need (pitch).

Visualize the data:
Paper.js library was used for this part.

References

- Examples from the used libraries.
- https://learn.adafruit.com/multi-tasking-the-arduino-part-3/using-neopatterns

- https://learn.adafruit.com/ahrs-for-adafruits-9-dof-10-dof-breakout

- http://paperjs.org/tutorials/animation/creating-animations/