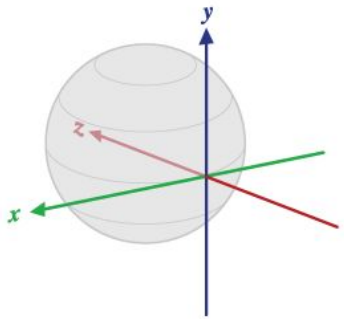
ACCELEROMETERS

PROJECT 1

LUCA SCARCI

DEMI PEPPAS



ACCELERATION IN TECHNOLOGY

WHAT ARE THEY?

Accelerometers measure the amount of G force occurring when moving at a specific speed. It can measure the rate at which something accelerates, as well as the rate at which is slows down.

We decided to primarily use 2 types of accelerometers, 1 which we consider the starter pack for it, and the other for a more complex use.

We discovered external complexities with the Sparkfun Accelerometers, where there were issues connecting them to the arduino and receiving data with them. The provided documentation was not always reliable, especially from SparkFun.

INSPIRATION / VISUAL RESEARCH

FIREWALKER GEMMA SNEAKERS

INSPIRATION / VISUAL RESEARCH

REDBULL SKYLIGHTERS

INSPIRATION / VISUAL RESEARCH

MOTION ACTIVATED SPEED-SUIT

CODE BREAKDOWN

```
#include <Wire.h>
#include <Adafruit_MMA8451.h>
#include <Adafruit_Sensor.h>
//calling adafruit libraries

Adafruit_MMA8451 mma = Adafruit_MMA8451();
//call on the accelerometer object form the Adafruit library

int n =  11; //set variable for "North" LED - Pitch Down
int s =  9; //set variable for "South" LED - Pitch Up
int e =  7; //set variable for "East" LED - Roll Right
int w =  13; //set variable for "West" LED - Roll Left

int accelY; //set variable for "Y" (Forward" Acceleration + Reverse Acceleration)

int stasis = 2; //stasis is the state in which indicates that the acceleration detection is active.
Stands for "0" acceleration.

int s1a = 3; //first stage of acceleration - if statement determines the range it is accelerating at
int s2a = 4; //second stage of acceleration - ...
int s3a = 5; //third stage of acceleration - ...
int s4a = 6; //fourth stage of acceleration - ...
//if arduino is setup with the correct set of LEDs and are attached to the correct pins (in order)
then the acceleration will be visualized by using a "level" diagram, where acceleration reaches 5
different levels

int gyroX; //The axis in which the accelerometer is changing - X AXIS
int gyroY; //The axis in which the accelerometer is changing - Y AXIS
int gyroZ; //The axis in which the accelerometer is changing - Z AXIS


void setup(void) {
  Serial.begin(9600);
  Serial.println("Adafruit MMA8451 test!");
  //setting up the serial port to print to the monitor

  if (! mma.begin()) {
    Serial.println("Couldnt start");
    //if the accelerometer has trouble starting or connecting - this error will be displayed - repeats
once
    while (1);
  }
  Serial.println("MMA8451 found!");
  // if the accelerometer is able to successfully connect

  mma.setRange(MMA8451_RANGE_8_G);
  //setting the range which the accelerometer reads its acceleration - the level of sensitivity. 2, 4, 8, 16

  Serial.print("Range = "); Serial.print(2 << mma.getRange());
  Serial.println("G");
  //printing the range into the console

  pinMode(n, OUTPUT); //determining the North LED pinOut for the gyroscope
  pinMode(e, OUTPUT); //determining the East LED pinOut for the gyroscope
  pinMode(w, OUTPUT); //determining the West LED pinOut for the gyroscope
  pinMode(s, OUTPUT); //determining the South LED pinOut for the gyroscope

  pinMode(stasis, OUTPUT); //determining the Stasis LED pinOut for the acceleration
  pinMode(s1a, OUTPUT); //determining the Level 1 LED pinOut for the acceleration
  pinMode(s2a, OUTPUT); //determining the Level 2 LED pinOut for the acceleration
  pinMode(s3a, OUTPUT); //determining the Level 3 LED pinOut for the acceleration
  pinMode(s4a, OUTPUT); //determining the Level 4 LED pinOut for the acceleration
  //these pins are ordered on the breadboard in accordance to how they are set up in the code,
putting them out of order or in different arrangements will alter the results

}
```

CODE BREAKDOWN

```
void loop() {

  // Read the 'raw' data in 14-bit counts
  mma.read();

  gyroX = mma.x;
  gyroY = mma.y;
  gyroZ = mma.z;
  //adding data to given gyroscope axis


// if statement beginning for GYROSCOPIC AXIS
// All units run between a range of 1 to 1000


if (gyroX > 150 ) {
  Serial.println("ROLL RIGHT");
  //print to the serial monitor - ROLL RIGHT
  digitalWrite(e, HIGH);
  //if the X axis is greater than 150, turn on "Roll Right" LED
} else {
  digitalWrite(e, LOW);
  //if it is anything else, turn it off
};

if (gyroX < -150 ) {
  Serial.println("ROLL LEFT");
  digitalWrite(w, HIGH);
    //if the X axis is greater than 150, turn on "Roll Left" LED

} else {
  digitalWrite(w, LOW);
  //if it is anything else, turn it off

};
```

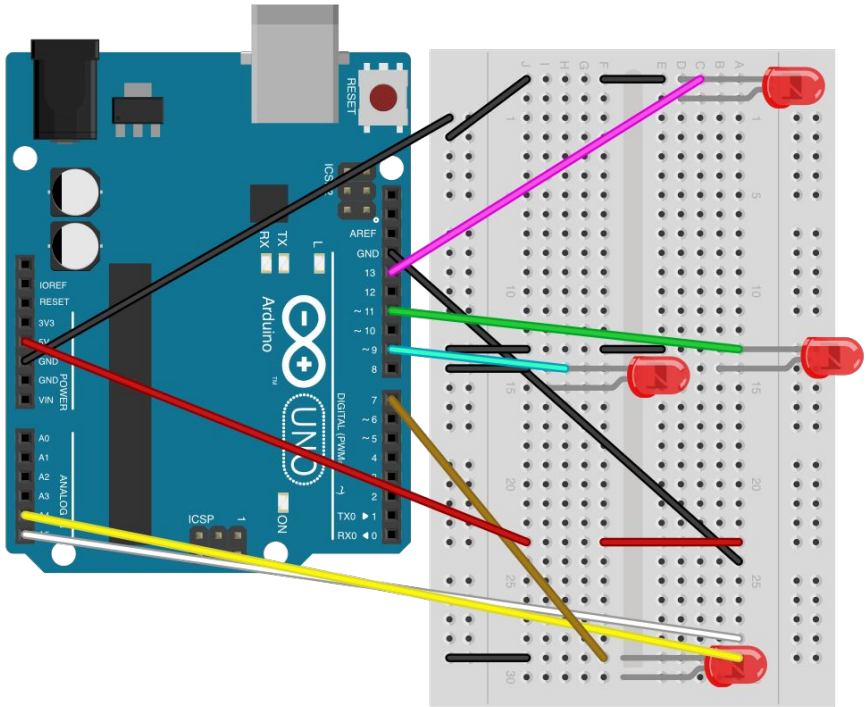
```
if (gyroY > 150 ) {
  Serial.println("PITCH DOWN");
  digitalWrite(n, HIGH);
   //if the Y axis is greater than 150, turn on "Pitch Down" LED

} else {
    digitalWrite(n, LOW);
  //if it is anything else, turn it off

};

if (gyroY < -150 ) {
  Serial.println("PITCH UP");
  digitalWrite(s, HIGH);
    //if the Y axis is greater than -150, turn on "Pitch Up" LED

} else {
    digitalWrite(s, LOW);
  //if it is anything else, turn it off

};


  sensors_event_t event;
  mma.getEvent(&event);
  //using the object "&event" in the

accelY = event.acceleration.y;
//assigning the acceleration object to the variable

Serial.println(event.acceleration.y);
```

CODE BREAKDOWN

```
// if statement beginning for ACCELERATION LEVELS
   digitalWrite(stasis, HIGH);

  if (accelY > 5) {
    digitalWrite(s1a, HIGH);
  } else {
    digitalWrite(s1a, LOW);
  };

  if (accelY > 10 ) {
    digitalWrite(s2a, HIGH);
  } else {
    digitalWrite(s2a, LOW);
  };

  if (accelY > 15) {
    digitalWrite(s3a, HIGH);
  } else {
    digitalWrite(s3a, LOW);
  };

  if (accelY > 20) {
    digitalWrite(s4a, HIGH);
  } else {
    digitalWrite(s4a, LOW);
  };




  delay(50);
  //delays the print to the serial monitor by 0.05s

}
```

FRITZING DIAGRAM

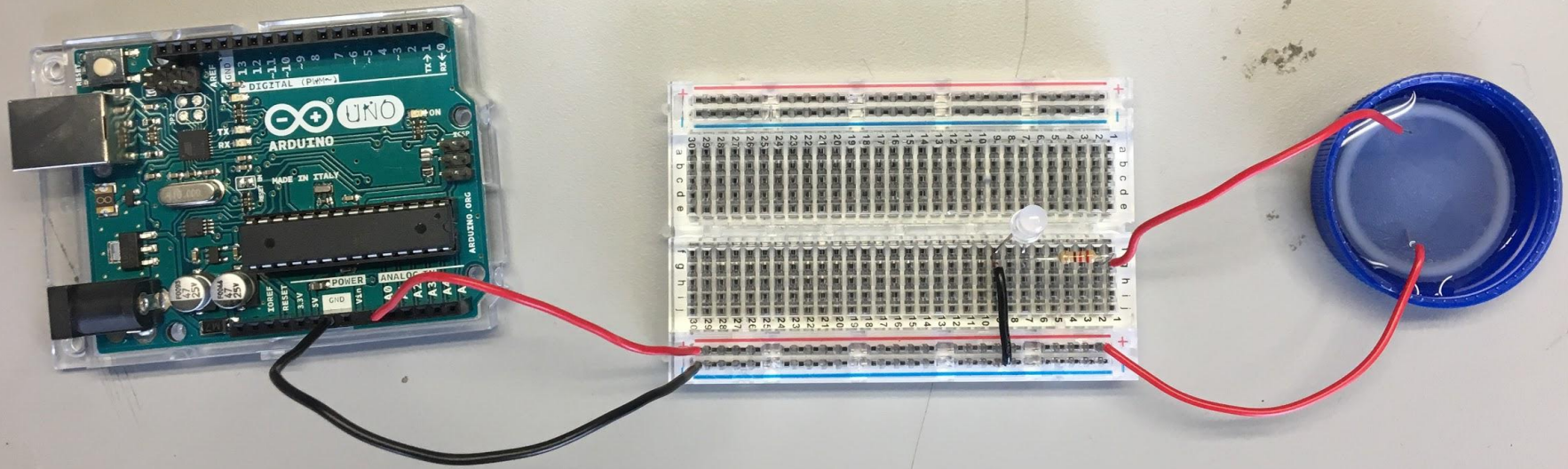
GYROSCOPE SETUP



VIDEO

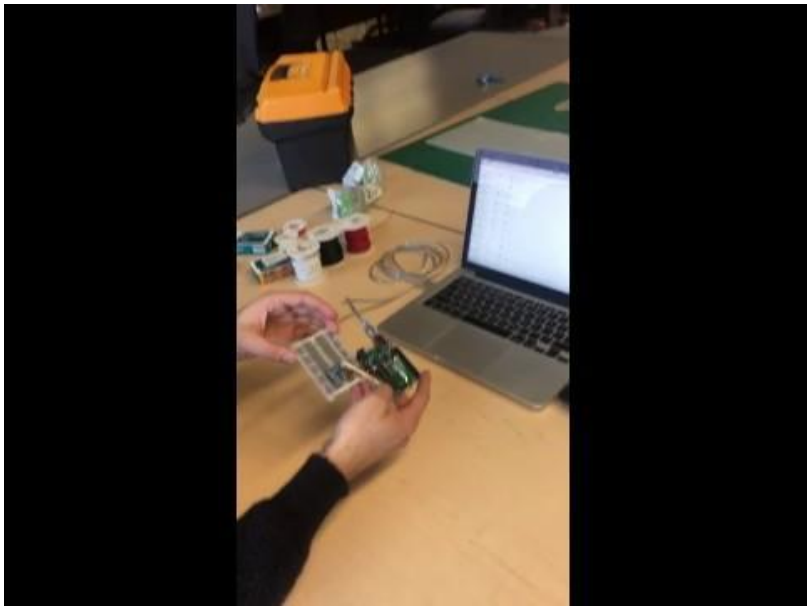
Video Link

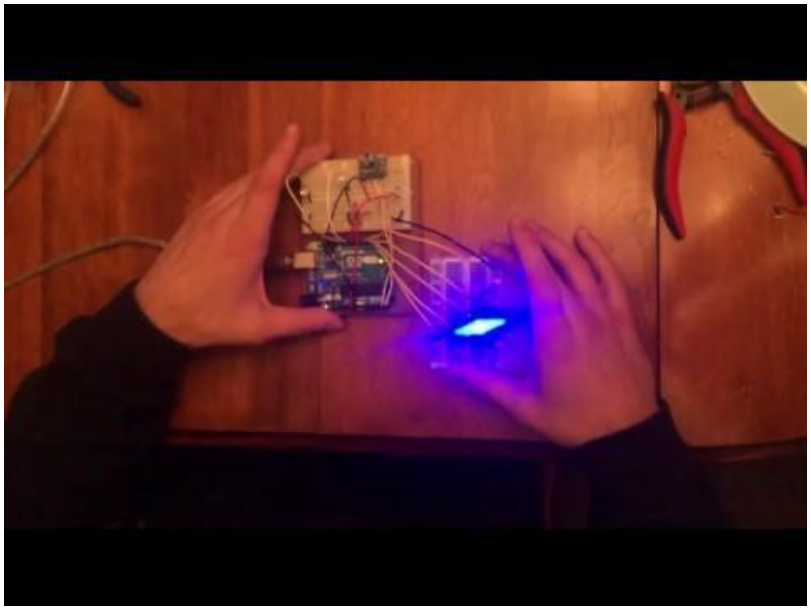
OTHER VARIATIONS & EXPLORATION



OTHER VARIATIONS & EXPLORATION

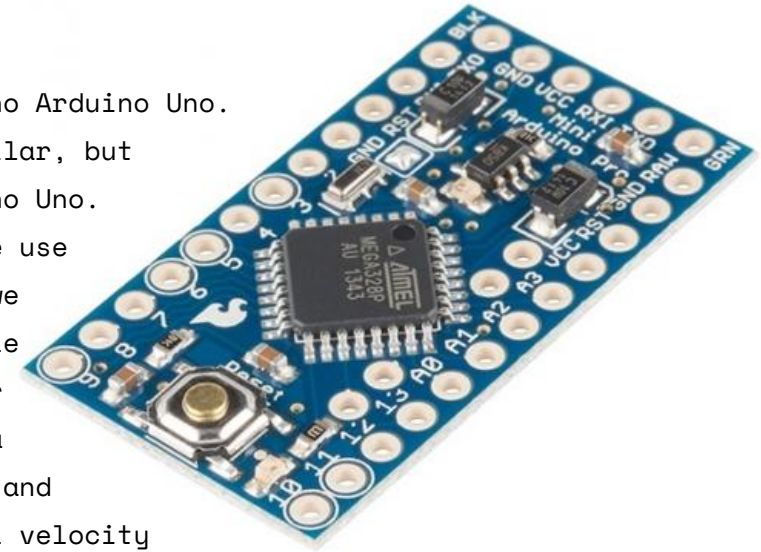
HOMEMADE ACCELEROMETER





ARDUINO PRO MINI

This little guy – well, we can tell you it is no Arduino Uno. It is a TOTALLY DIFFERENT ANIMAL. Pins are similar, but does not use the same serial port as the Arduino Uno. It uses a legacy port that does not support the use of the serial monitor, as well as the Arduino we were using. It is unfortunate that it was unable To work, since we were going to use a USB power breakout pin and Bluetooth NFC to transmit data wirelessly, so we could place it inside a ball and throw it to measure the acceleration and travel velocity



FIN