

PULSE SENSOR & OLED INTEGRATION WITH ESP8266

[12TH JUNE CLASS WORK]

```
#include <Wire.h>

#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

Adafruit_SSD1306 display(128, 64, &Wire, -1);

#define SENSOR_PIN A0
#define THRESHOLD 540
#define SAMPLE_INTERVAL 2000

unsigned long lastBeatTime = 0;
int beatCounter = 0;
int BPM = 0;

void setup() {
  Serial.begin(9600);

  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println(F("SSD1306 allocation failed"));
    while (true); // Stop here if SSD1306 initialization failed
  }

  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setTextSize(2);
  display.setCursor(0, 0);
```

```

display.println("BPM: ");
display.display();
}

void loop() {
    int sensorValue = analogRead(SENSOR_PIN);
    Serial.println(sensorValue);

    drawGraph(sensorValue);

    if (sensorValue > THRESHOLD) {
        if (millis() - lastBeatTime >= SAMPLE_INTERVAL) {
            beatCounter++;
            lastBeatTime = millis();
        }
    } else {
        if (beatCounter > 0 && millis() - lastBeatTime >= SAMPLE_INTERVAL) {
            BPM = beatCounter * (60000 / SAMPLE_INTERVAL);
            beatCounter = 0;
            updateBPM(BPM);
        }
    }
}

void drawGraph(int value) {
    static int prevX = 0;
    static int prevY = 0;
    int y = map(value, 0, 1023, 0, 64);
    int x = (millis() / 1000) % 128; // Display for 128 seconds before resetting

    display.drawLine(prevX, prevY, x, 64 - y, WHITE);

```

```
    prevX = x;  
    prevY = 64 - y;  
    display.display();  
}
```

```
void updateBPM(int value) {  
    display.setTextSize(2);  
    display.setCursor(0, 0);  
    display.fillRect(0, 0, 128, 16, BLACK);  
    display.setTextColor(WHITE);  
    display.print("BPM: ");  
    display.print(value);  
    display.display();  
}
```

