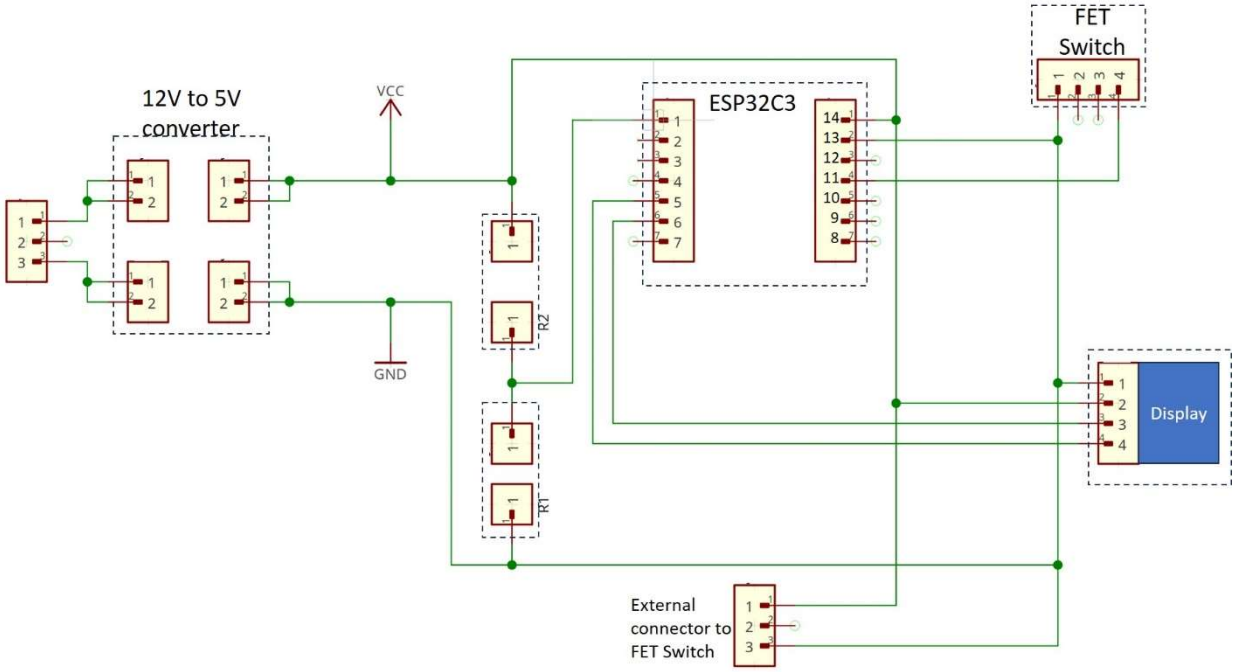
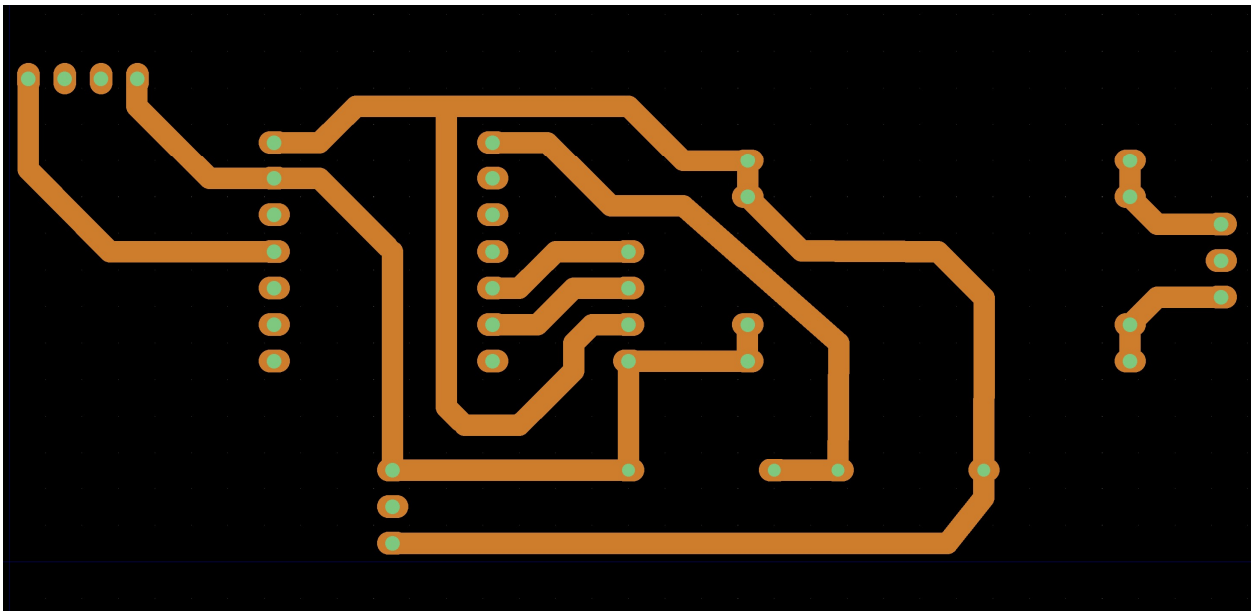
Schematic



PCB Layout



Parts List

FET Switch	https://www.amazon.com/gp/product/B08CXB4WCQ/
Voltage Converter	https://www.amazon.com/gp/product/B098JF17LY/
SSD1306 Display	https://www.amazon.com/gp/product/B09C5K91H7/
ESP32C3	https://www.seeedstudio.com/Seeed-XIAO-ESP32C3-p-5431.html
R1, R2	20K, ¼ watt
Bare PCB Board	https://www.amazon.com/gp/product/B01MCVLDDZ/

Software code

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET    4
#define SCREEN_WIDTH  128
#define SCREEN_HEIGHT 64

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

const long TWO_MINUTES = 120000; // Two minutes in milliseconds
const long FIFTEEN_MINUTES = 900000; // Fifteen minutes in milliseconds
const int THRESHOLD_VOLTAGE = 2375; // 4750 millivolts (assuming a 50% divider
network)
const int DEBUG = 1;

enum Stage { BELOW_THRESHOLD, OFF, ON };

void showStatus(float volts, Stage stage, unsigned long stageDuration);

void setup() {
  pinMode(D2, OUTPUT);  // Maintain compatibility with prototypes
  pinMode(D10, OUTPUT);
  pinMode(A0, INPUT);
  Serial.begin(9600);
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  showStatus(analogReadMilliVolts(A0), BELOW_THRESHOLD, 0);
}

void loop() {
  unsigned long stageEntry = millis();
  unsigned long stageDuration;
  float volts;

  while (analogReadMilliVolts(A0) < THRESHOLD_VOLTAGE) {
    digitalWrite(D2, LOW);
    digitalWrite(D10, LOW);
    if (DEBUG) {
      Serial.println("Voltage below threshold");
    }
    stageDuration = millis() - stageEntry;
```

```
    volts = analogReadMilliVolts(A0);
    showStatus(volts, BELOW_THRESHOLD, stageDuration);
    delay(100);
  }


  Stage currentStage = OFF;
  stageEntry = millis();

  do {
    delay(2000);
    volts = analogReadMilliVolts(A0);
    stageDuration = millis() - stageEntry;
    showStatus(volts, currentStage, stageDuration);
    currentStage = OFF;
  } while (stageDuration < TWO_MINUTES);

  digitalWrite(D2, HIGH);
  digitalWrite(D10, HIGH);
  int powerBadCount = 0;
  stageEntry = millis();
  do {
    volts = analogReadMilliVolts(A0);
    if (volts < THRESHOLD_VOLTAGE) {
      powerBadCount++;
    } else {
      powerBadCount = 0;
    }
    if (DEBUG) {
      Serial.printf("Status: %d, Voltage: %f V\n", powerBadCount, volts /
1000.0);
    }
    stageDuration = millis() - stageEntry;
    showStatus(volts, ON, stageDuration);
    delay(1000);
  } while (powerBadCount < 6);
}

void showStatus(float volts, Stage stage, unsigned long stageDuration) {
  display.clearDisplay();
  display.setTextSize(1,2);
  display.setTextColor(WHITE);
  display.setCursor(0, 0);
  display.print("V.Monitor:  ");
```

```
  switch (stage) {
    case BELOW_THRESHOLD:
      display.println("Low V");
      break;
    case OFF:
      display.println("Off");
      break;
    case ON:
      display.println("On");
      break;
  }

  display.println();
  display.print("Volts: ");
  display.println(volts / 500.0); // Convert mV to V for display
  display.print("Time: ");
  unsigned long seconds = (stageDuration / 1000) % 60;
  unsigned long minutes = (stageDuration / 60000) % 60;
  unsigned long hours = (stageDuration / 3600000) % 24;
  unsigned long days = (stageDuration/86400000);
  display.print(days);
  display.print(':');
  if (hours < 10) display.print('0');
  display.print(hours);
  display.print(':');
  if (minutes < 10) display.print('0');
  display.print(minutes);
  display.print(':');
  if (seconds < 10) display.print('0');
  display.println(seconds);

  display.display();
}
```