

## Code:

```
#include <ESP8266WiFi.h>
#include <Firebase_ESP_Client.h>
#include "addons/TokenHelper.h"
#include "addons/RTDBHelper.h"
#include <U8g2lib.h>

U8G2_SSD1306_128X64_NONAME_F_HW_I2C    u8g2(U8G2_R0,    /*    reset=*/
U8X8_PIN_NONE);

const int voltagePin = A0; // Analog pin for voltage sensor
const int currentPin = D1; // Analog pin for current sensor
const float voltageDividerRatio = 2.0; // Adjust this ratio based on your voltage divider
circuit
const float currentSensorSensitivity = 185.0; // Sensitivity of the ACS712 current sensor

#define WIFI_SSID "123456789"
#define WIFI_PASSWORD "123456789"
#define API_KEY "AIzaSyC0gPSHesz3RxIsbFM48OkKK_zCBhftmc"
#define DATABASE_URL "https://test-26075-default-rtdb.firebaseio.com/"

FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;
unsigned long sendDataPrevMillis = 0;
bool signupOK = false;
const int buzzer = D3;
const int relayPin = D5; // GPIO pin where the relay is connected

void setup() {
  Serial.begin(115200);
  u8g2.begin();
```

```
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting to Wi-Fi");
while (WiFi.status() != WL_CONNECTED) {
  Serial.print(".");
  delay(300);
}
Serial.println();
Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();
```

```
config.api_key = API_KEY;
config.database_url = DATABASE_URL;
```

```
if (Firebase.signUp(&config, &auth, "", "")) {
  Serial.println("Firebase sign-up successful");
  signupOK = true;
} else {
  Serial.printf("%s\n", config.signer.signupError.message.c_str());
}
```

```
config.token_status_callback = tokenStatusCallback; // see addons/TokenHelper.h
Firebase.begin(&config, &auth);
Firebase.reconnectWiFi(true);
}
```

```
void loop() {
  int voltageSensorValue = analogRead(voltagePin);
  int currentSensorValue = analogRead(currentPin);

  // Convert the analog readings to actual values
  float voltage = voltageSensorValue * (3.3 / 1023.0) * voltageDividerRatio;
  float current = (currentSensorValue - 512) / currentSensorSensitivity;
```

```
Serial.print("Voltage Sensor Value: ");
Serial.print(voltageSensorValue);
Serial.print(", Voltage: ");
Serial.print(voltage);
Serial.println("V");
```

```
Serial.print("Current Sensor Value: ");
Serial.print(currentSensorValue);
Serial.print(", Current: ");
Serial.print(current);
Serial.println("mA");
```

```
displayDataOnOLED(voltage, current);
```

```
    if (Firebase.ready() && signupOK && (millis() - sendDataPrevMillis > 1000 ||
sendDataPrevMillis == 0)) {
    sendDataPrevMillis = millis();
```

```
    if (Firebase.RTDB.setInt(&fbdo, "main/CurrentValue", current)) {
        Serial.println("PATH: " + fbdo.dataPath());
        Serial.println("TYPE: " + fbdo.dataType());
    } else {
        Serial.println("Failed REASON: " + fbdo.errorReason());
    }
}
```

```
    if (Firebase.RTDB.setInt(&fbdo, "main/Voltage", voltage)) {
        Serial.println("PATH: " + fbdo.dataPath());
        Serial.println("TYPE: " + fbdo.dataType());
    } else {
        Serial.println("Failed REASON: " + fbdo.errorReason());
    }
}
```

```
delay(1000); // Add delay as needed
}
```

```
void displayDataOnOLED(float voltage, float current) {  
    u8g2.firstPage();  
    do {  
        u8g2.setFont(u8g2_font_ncenB08_tr); // choose a suitable font  
        u8g2.setCursor(0, 15);  
        u8g2.print("Voltage: ");  
        u8g2.setCursor(70, 15);  
        u8g2.print(voltage, 2); // Display voltage with 2 decimal places  
        u8g2.setCursor(100, 15);  
        u8g2.print("V");  
  
        u8g2.setCursor(0, 30);  
        u8g2.print("Current: ");  
        u8g2.setCursor(70, 30);  
        u8g2.print(current, 2); // Display current with 2 decimal places  
        u8g2.setCursor(100, 30);  
        u8g2.print("mA");  
    } while (u8g2.nextPage());  
}
```