Smart Energy Monitoring System - ESP32 Code

```cpp
#include <WiFi.h>
#include <HTTPClient.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET    -1
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

const char* ssid = "Your_WiFi_Name";
const char* password = "Your_WiFi_Password";

String apiKey = "Your_ThingSpeak_API_Key";
const char* server = "http://api.thingspeak.com/update";

#define VOLTAGE_SENSOR_PIN 34
#define CURRENT_SENSOR_PIN 35

float voltageCalibration = 0.312;
float currentCalibration = 0.100;

void setup() {
    Serial.begin(115200);
    if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
        Serial.println("SSD1306 allocation failed");
        for (;;);
    }
    display.clearDisplay();
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("Connected to WiFi!");
}

void loop() {
    float voltage = readVoltage();
    float current = readCurrent();
    float power = voltage * current;

    Serial.print("Voltage: "); Serial.print(voltage); Serial.println(" V");
    Serial.print("Current: "); Serial.print(current); Serial.println(" A");
    Serial.print("Power: "); Serial.print(power); Serial.println(" W");

    displayReadings(voltage, current, power);

    if (WiFi.status() == WL_CONNECTED) {
```

```cpp
        HTTPClient http;
        String url = String(server) + "?api_key=" + apiKey + "&field1=" + String(voltage) +
                        "&field2=" + String(current) + "&field3=" + String(power);
        http.begin(url);
        http.GET();
        http.end();
    }

    if (power > 500.0) {
        Serial.println("Overload Alert! Power consumption too high!");
    }

    delay(5000);
}

float readVoltage() {
    int sensorValue = analogRead(VOLTAGE_SENSOR_PIN);
    float voltage = (sensorValue / 4095.0) * 3.3 * (220.0 / 2.0);
    return voltage * voltageCalibration;
}

float readCurrent() {
    int sensorValue = analogRead(CURRENT_SENSOR_PIN);
    float current = ((sensorValue - 2048) / 2048.0) * 5.0;
    return current * currentCalibration;
}

void displayReadings(float voltage, float current, float power) {
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.setCursor(0, 0);
    display.print("Voltage: "); display.print(voltage); display.println(" V");
    display.print("Current: "); display.print(current); display.println(" A");
    display.print("Power: "); display.print(power); display.println(" W");
    display.display();
}
```