```
#include <WiFi.h>
#include <ThingSpeak.h>
#include <MAX30100_PulseOximeter.h>

// WiFi credentials
const char* ssid = "Your_SSID";
const char* password = "Your_PASSWORD";

// ThingSpeak
const char* apiKey = "YOUR_THINGSPEAK_API_KEY";
const long channelID = YOUR_CHANNEL_ID;
WiFiClient client;

// Sensor Pins
#define trigPin 12
#define echoPin 14
#define currentSensorPin 34
#define mq135Pin 35

PulseOximeter pox;
uint32_t tsLastReport = 0;

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

```
  // Setup pins
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);

  // WiFi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWiFi connected");

  ThingSpeak.begin(client);

  // MAX30100 init
  if (!pox.begin()) {
    Serial.println("MAX30100 not found");
  } else {
    Serial.println("MAX30100 ready");
  }
}

void loop() {
  // Ultrasonic Sensor
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
```

```
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
long duration = pulseIn(echoPin, HIGH);
int distance = duration * 0.034 / 2;
int carDetected = (distance < 20) ? 1 : 0;

// Power Monitoring (ACS712)
int rawValue = analogRead(currentSensorPin);
float voltage = rawValue * (3.3 / 4095.0);
float current = (voltage - 2.5) / 0.066;  // Adjust depending on your ACS712 version
float power = current * 230.0;

// Air Quality (MQ135)
int airQuality = analogRead(mq135Pin);

// Heart Rate & SpO2
pox.update();
float bpm = pox.getHeartRate();
float spo2 = pox.getSpO2();

// Send to ThingSpeak
ThingSpeak.setField(1, carDetected);
ThingSpeak.setField(2, power);
ThingSpeak.setField(3, bpm);
ThingSpeak.setField(4, spo2);
```

```
  ThingSpeak.setField(5, airQuality);

  int status = ThingSpeak.writeFields(channelID, apiKey);
  if (status == 200) {
    Serial.println("Data sent to ThingSpeak.");
  } else {
    Serial.println("ThingSpeak update failed: " + String(status));
  }

  delay(15000); // ThingSpeak allows updates every 15 seconds
}
```