Here's the complete Arduino IDE code for your project along with details on how to connect the components.

## **Circuit Components:**

* **ESP32** (Microcontroller)
* **MQ-3 Alcohol Sensor** (For detecting alcohol levels)
* **LED** (Indicator for alcohol detection)
* **Buzzer** (Warning sound when alcohol is detected)
* **16x2 LCD Display with I2C Module** (To display messages)
* **Relay Module** (To control ignition)
* **Mobile with Blynk App or Telegram API** (For alerts)

## **Connections:**

| **Component** | **ESP32 Pin** |
| --- | --- |
| MQ-3 Sensor VCC | 3.3V |
| MQ-3 Sensor GND | GND |
| MQ-3 Sensor A0 | GPIO 34 |
| Relay Module IN | GPIO 27 |
| Buzzer + | GPIO 25 |
| LED + | GPIO 26 |
| LCD SDA | GPIO 21 |
| LCD SCL | GPIO 22 |

## 

## 

## **Arduino Code (ESP32)**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

#include <WiFi.h>

#include <HTTPClient.h>

#define MQ3\_PIN 34 // Analog pin for MQ3 sensor

#define RELAY\_PIN 27 // Controls vehicle ignition

#define BUZZER\_PIN 25 // Buzzer

#define LED\_PIN 26 // LED Indicator

#define TRIGGER\_TIME 900000 // 15 minutes (900,000 ms)

const char\* ssid = "Your\_SSID";

const char\* password = "Your\_PASSWORD";

const char\* server = "https://api.telegram.org/bot<Your\_BOT\_Token>/sendMessage?chat\_id=<Your\_Chat\_ID>&text=ALCOHOL DETECTED! ENGINE LOCKED.";

LiquidCrystal\_I2C lcd(0x27, 16, 2);

unsigned long lastBlowTime = 0;

bool engineOn = false;

void setup() {

Serial.begin(115200);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("\nWiFi Connected!");

lcd.init();

lcd.backlight();

lcd.setCursor(0, 0);

lcd.print("Blow into MQ3");

pinMode(MQ3\_PIN, INPUT);

pinMode(RELAY\_PIN, OUTPUT);

pinMode(BUZZER\_PIN, OUTPUT);

pinMode(LED\_PIN, OUTPUT);

digitalWrite(RELAY\_PIN, LOW); // Ignition off initially

}

void loop() {

int alcoholValue = analogRead(MQ3\_PIN);

Serial.print("MQ3 Value: ");

Serial.println(alcoholValue);

if (alcoholValue < 300) { // No alcohol detected, allow ignition

lcd.setCursor(0, 0);

lcd.print("Vehicle Start ");

digitalWrite(RELAY\_PIN, HIGH); // Turn on ignition

digitalWrite(LED\_PIN, LOW);

digitalWrite(BUZZER\_PIN, LOW);

engineOn = true;

lastBlowTime = millis(); // Reset timer

} else { // Alcohol detected

lcd.setCursor(0, 0);

lcd.print("Alcohol Found! ");

lcd.setCursor(0, 1);

lcd.print("Engine Locked ");

digitalWrite(RELAY\_PIN, LOW); // Stop ignition

digitalWrite(LED\_PIN, HIGH);

digitalWrite(BUZZER\_PIN, HIGH);

sendAlert();

delay(5000);

digitalWrite(BUZZER\_PIN, LOW);

engineOn = false;

}

// Check if 15 minutes passed and the driver didn't blow again

if (engineOn && (millis() - lastBlowTime > TRIGGER\_TIME)) {

lcd.setCursor(0, 0);

lcd.print("Blow Again! ");

lcd.setCursor(0, 1);

lcd.print("Time Over ");

digitalWrite(RELAY\_PIN, LOW); // Stop engine if no new test

engineOn = false;

}

delay(2000);

}

void sendAlert() {

if (WiFi.status() == WL\_CONNECTED) {

HTTPClient http;

http.begin(server);

int httpResponseCode = http.GET();

http.end();

}

}

## **Circuit Diagram:**

I'll generate a circuit diagram with all the connections. Stay tuned.

Here is the circuit diagram for your alcohol detection car ignition system. It includes all the necessary wiring connections for the ESP32, MQ-3 alcohol sensor, LED, buzzer, LCD display, and relay module. Let me know if you need any modifications or further explanations! 🚗🔧

### **Step-by-Step Setup Guide for Alcohol Detection Car Ignition System (ESP32)**

To successfully run your ESP32-based alcohol detection system, follow these steps:

## **1️⃣ Install Required Software**

### **🔹 Install Arduino IDE**

1. **Download and Install Arduino IDE** from the official website:  
    👉<https://www.arduino.cc/en/software>
2. **Open Arduino IDE** after installation.

### **🔹 Install ESP32 Board in Arduino IDE**

1. Open **Arduino IDE**.
2. Go to **File** → **Preferences**.

In the **Additional Board Manager URLs**, add this link:  
 https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\_esp32\_index.json

1. Click **OK**.
2. Go to **Tools** → **Board** → **Boards Manager**.
3. Search for **ESP32** and install **ESP32 by Espressif Systems**.
4. Wait for the installation to complete.

## **2️⃣ Install Required Libraries**

### **🔹 Install LiquidCrystal\_I2C Library**

This library is needed for the **16x2 LCD Display**.

1. Open **Arduino IDE**.
2. Go to **Sketch** → **Include Library** → **Manage Libraries**.

In the search bar, type:  
 LiquidCrystal\_I2C

1. Select **LiquidCrystal I2C by Frank de Brabander** and install it.

### **🔹 Install WiFi and HTTPClient Libraries**

These libraries are included by default in the ESP32 core. If you get errors, do the following:

1. Go to **Sketch** → **Include Library** → **Manage Libraries**.

Search for:  
 WiFi

1. and ensure **WiFi by Arduino** is installed.

Search for:  
 HTTPClient

1. and ensure **HTTPClient by Arduino** is installed.

## **3️⃣ Connect ESP32 to Your Wi-Fi**

Modify the following lines in the Arduino code:

const char\* ssid = "Your\_SSID"; // Replace with your WiFi name

const char\* password = "Your\_PASSWORD"; // Replace with your WiFi password

Replace:

* "Your\_SSID" → Your Wi-Fi network name
* "Your\_PASSWORD" → Your Wi-Fi password

📌 **Important:**

* Make sure your Wi-Fi is **2.4GHz**, as ESP32 does not support **5GHz** networks.

## **4️⃣ Connect ESP32 and Upload the Code**

### **🔹 Connect ESP32 to Your PC**

1. Use a **USB to Micro-USB** cable (or USB-C for newer ESP32 models).
2. Open **Arduino IDE**.
3. Go to **Tools** and set:
   * **Board** → Select **ESP32 Dev Module**.
   * **Port** → Select the correct **COM port** (usually something like COM3, COM4, etc.).
   * **Upload Speed** → Set to **115200**.
4. Click on the **Upload (▶) button** to flash the code.

## **5️⃣ Open Serial Monitor for Debugging**

1. Go to **Tools** → **Serial Monitor**.
2. Set baud rate to **115200**.

If everything is correct, you should see:  
 Connecting to WiFi...

WiFi Connected!

MQ3 Value: 200 // Example

Vehicle Start

## **6️⃣ Test the Alcohol Detection System**

1. **Blow normal air into the MQ-3 sensor** → The vehicle should **start**.
2. **Blow alcohol fumes (e.g., from sanitizer or alcohol bottle cap)** → The vehicle should **lock** and send an alert to your mobile.
3. **Wait 15 minutes without blowing** → The engine should **stop automatically**.

🎯 **Now your alcohol detection system is ready to use!** 🚗  
 If you have any issues, let me know! 😊