#include <Wire.h>

#include <LiquidCrystal\_I2C.h> // Include I2C LCD library

#include <NewPing.h> // Include NewPing library

// Pin definitions

const int alcoholPin = A0;

const int eyeBlinkPin = 2;

const int trigPin = 12;

const int echoPin = 13;

const int motor1Pin1 = 3;

const int motor1Pin2 = 4;

const int motor2Pin1 = 5;

const int motor2Pin2 = 6;

NewPing sonar(trigPin, echoPin, 200); // Initialize ultrasonic sensor with max distance 200cm

LiquidCrystal\_I2C lcd(0x27, 16, 2); // Set LCD address (usually 0x27 or 0x3F) and dimensions

void stopMotors() {

digitalWrite(motor1Pin1, LOW);

digitalWrite(motor1Pin2, LOW);

digitalWrite(motor2Pin1, LOW);

digitalWrite(motor2Pin2, LOW);

}

void moveMotors() {

digitalWrite(motor1Pin1, HIGH);

digitalWrite(motor1Pin2, LOW);

digitalWrite(motor2Pin1, HIGH);

digitalWrite(motor2Pin2, LOW);

}

void setup() {

// Initialize motor pins

pinMode(motor1Pin1, OUTPUT);

pinMode(motor1Pin2, OUTPUT);

pinMode(motor2Pin1, OUTPUT);

pinMode(motor2Pin2, OUTPUT);

pinMode(alcoholPin, INPUT);

pinMode(eyeBlinkPin, INPUT);

Serial.begin(9600);

// Initialize LCD

lcd.init(); // Initialize the I2C LCD

lcd.backlight(); // Turn on the backlight

lcd.clear(); // Clear the display

lcd.print("System Init");

delay(2000); // Display initialization message

lcd.clear();

}

void loop() {

// Read alcohol sensor

int alcoholValue = analogRead(alcoholPin);

if (alcoholValue > 200) {

lcd.clear();

lcd.print("Alcohol detected!");

Serial.println("Alcohol detected! Stopping motors.");

stopMotors();

return;

}

// Check eye blink detection

bool eyeBlinkDetected = digitalRead(eyeBlinkPin);

if (eyeBlinkDetected == HIGH) {

lcd.clear();

lcd.print("Eye Blink Detected!");

Serial.println("Eye blink detected! Stopping motors.");

stopMotors();

return;

}

// Check ultrasonic sensor for obstacles

int distance = sonar.ping\_cm();

if (distance > 0 && distance < 20) {

lcd.clear();

lcd.print("Obstacle detected!");

Serial.println("Obstacle detected! Stopping motors.");

stopMotors();

return;

}

// If all systems are clear

lcd.clear();

lcd.print("systems clear...");

lcd.setCursor(0, 1);

lcd.print("Moving motors...");

Serial.println("All systems clear. Moving motors.");

moveMotors();

delay(1000);

}