#include <Wire.h>

#include <Adafruit\_LiquidCrystal.h>

// LCD setup using Adafruit\_LiquidCrystal

Adafruit\_LiquidCrystal lcd(0); // 0 means use I2C address 0x27

// Pin definitions

const int potTempPin = A0; // Simulated temperature sensor

const int potHumidPin = A1; // Simulated humidity sensor

const int photoResistorPin = A2; // Light sensor

const int pirPin = 3; // PIR motion sensor output

const int ledLightPin = 4; // White LED - light control

const int ledHeaterPin = 5; // Yellow LED - heater

const int ledFanPin = 6; // Blue LED - fan

// Sensor readings

int tempValue = 0;

int humidValue = 0;

int lightValue = 0;

bool motionDetected = false;

// LED states

bool lightOn = false;

bool heaterOn = false;

bool fanOn = false;

void setup() {

Serial.begin(9600);

// Initialize pins

pinMode(pirPin, INPUT);

pinMode(ledLightPin, OUTPUT);

pinMode(ledHeaterPin, OUTPUT);

pinMode(ledFanPin, OUTPUT);

// Start LCD

lcd.begin(16, 2);

lcd.setBacklight(1);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Smart Home Init");

delay(1500);

lcd.clear();

Serial.println("Smart Home Controller Ready");

Serial.println("Commands:");

Serial.println("L ON/OFF - Light control");

Serial.println("H ON/OFF - Heater control");

Serial.println("F ON/OFF - Fan control");

Serial.println("STATUS - Show sensor readings");

Serial.println();

}

void loop() {

// Read sensor values

tempValue = analogRead(potTempPin);

humidValue = analogRead(potHumidPin);

lightValue = analogRead(photoResistorPin);

motionDetected = digitalRead(pirPin);

// Apply current device states to pins

digitalWrite(ledLightPin, lightOn ? HIGH : LOW);

digitalWrite(ledHeaterPin, heaterOn ? HIGH : LOW);

digitalWrite(ledFanPin, fanOn ? HIGH : LOW);

// Update LCD with live values

displayLCD();

// Handle serial commands

if (Serial.available()) {

String command = Serial.readStringUntil('\n');

command.trim();

command.toUpperCase();

if (command.startsWith("L ")) {

if (command.endsWith("ON")) {

lightOn = true;

Serial.println("Light turned ON");

} else if (command.endsWith("OFF")) {

lightOn = false;

Serial.println("Light turned OFF");

} else {

Serial.println("Invalid Light command");

}

}

else if (command.startsWith("H ")) {

if (command.endsWith("ON")) {

heaterOn = true;

Serial.println("Heater turned ON");

} else if (command.endsWith("OFF")) {

heaterOn = false;

Serial.println("Heater turned OFF");

} else {

Serial.println("Invalid Heater command");

}

}

else if (command.startsWith("F ")) {

if (command.endsWith("ON")) {

fanOn = true;

Serial.println("Fan turned ON");

} else if (command.endsWith("OFF")) {

fanOn = false;

Serial.println("Fan turned OFF");

} else {

Serial.println("Invalid Fan command");

}

}

else if (command == "STATUS") {

printStatus();

}

else {

Serial.println("Unknown command");

}

}

delay(200); // Slight delay to prevent overwhelming loop

}

void displayLCD() {

lcd.setCursor(0, 0);

lcd.print("T:");

lcd.print(map(tempValue, 0, 1023, 0, 50)); // Simulated temp in °C

lcd.print("C H:");

lcd.print(map(humidValue, 0, 1023, 0, 100)); // Simulated humidity %

lcd.setCursor(0, 1);

lcd.print("L:");

lcd.print(lightOn ? "ON " : "OFF");

lcd.print(" M:");

lcd.print(motionDetected ? "Yes" : "No ");

lcd.print(" Lt:");

lcd.print(map(lightValue, 0, 1023, 0, 100)); // Simulated light %

}

void printStatus() {

Serial.println("--- Sensor Status ---");

Serial.print("Temperature: ");

Serial.print(map(tempValue, 0, 1023, 0, 50));

Serial.println(" C");

Serial.print("Humidity: ");

Serial.print(map(humidValue, 0, 1023, 0, 100));

Serial.println(" %");

Serial.print("Light level: ");

Serial.print(map(lightValue, 0, 1023, 0, 100));

Serial.println(" %");

Serial.print("Motion detected: ");

Serial.println(motionDetected ? "YES" : "NO");

Serial.println("--- Device Status ---");

Serial.print("Light: ");

Serial.println(lightOn ? "ON" : "OFF");

Serial.print("Heater: ");

Serial.println(heaterOn ? "ON" : "OFF");

Serial.print("Fan: ");

Serial.println(fanOn ? "ON" : "OFF");

Serial.println();

}