FORMAN CHRISTIAN COLLEGE

(A CHARTERED UNIVERSITY)

CSCS 306 A

FA24

Lab 7 Report

Daim Bin Khalid – 251686775

Hafsah Shahbaz - 251684784

21st November 2024

Introduction

This lab aimed to simulate a control system for monitoring the gas volume in a cylinder using an Arduino-based system. The project involved utilizing a potentiometer to simulate volume levels, an LCD to display these values, and LEDs for real-time visual feedback. Task 1 focused on volume display using an LCD, while Task 2 extended this functionality to include a 4-LED setup to indicate volume percentages.

These tasks introduced essential embedded system concepts, including sensor data mapping, LED control, and serial communication for debugging purposes.

Functions

Task 1: LCD-Based Volume Display

The code for Task 1 was designed to:

- Read analog values from a potentiometer.
- Map these values to gas volume (0 to 100 cubic feet).
- Display the mapped volume on a 16x2 LCD in real time.

Task 2: LED-Based Volume Display

Task 2 added 4 LEDs to represent volume percentages:

- 1-25%: Turn ON the 1st LED.
- 26-50%: Turn ON the 1st and 2nd LEDs.
- 51-75%: Turn ON the 1st, 2nd, and 3rd LEDs.
- 76-100%: Turn ON all LEDs.

Each task integrated serial communication to display debugging information, including potentiometer readings and mapped percentages.

Algorithms and Logic

Task 1: LCD Volume Display

Setup Phase:

- **LCD Initialization**: Pins for the LCD were specified, and the display was initialized using lcd.begin(16, 2).

- **Potentiometer Connection**: The analog value from the potentiometer was read using analogRead(A0) and mapped to gas volume using the map() function.

Loop Phase:

1. Read the potentiometer value.
2. Map the value to gas volume (0 to 100 cubic feet).
3. Display the mapped value on the LCD.
4. Print the raw potentiometer value and the mapped volume to the Serial Monitor.

Task 2: LED Volume Display

Setup Phase:

- LEDs were configured as outputs using pinMode().

Loop Phase:

1. Read the potentiometer value and map it to a percentage.
2. Control LED states based on the mapped percentage:
   - Turn ON LEDs progressively as volume increases.
   - Turn OFF LEDs when the volume decreases.
3. Display real-time readings on the Serial Monitor for debugging.

Code Breakdown

Task 1: LCD-Based Volume Display

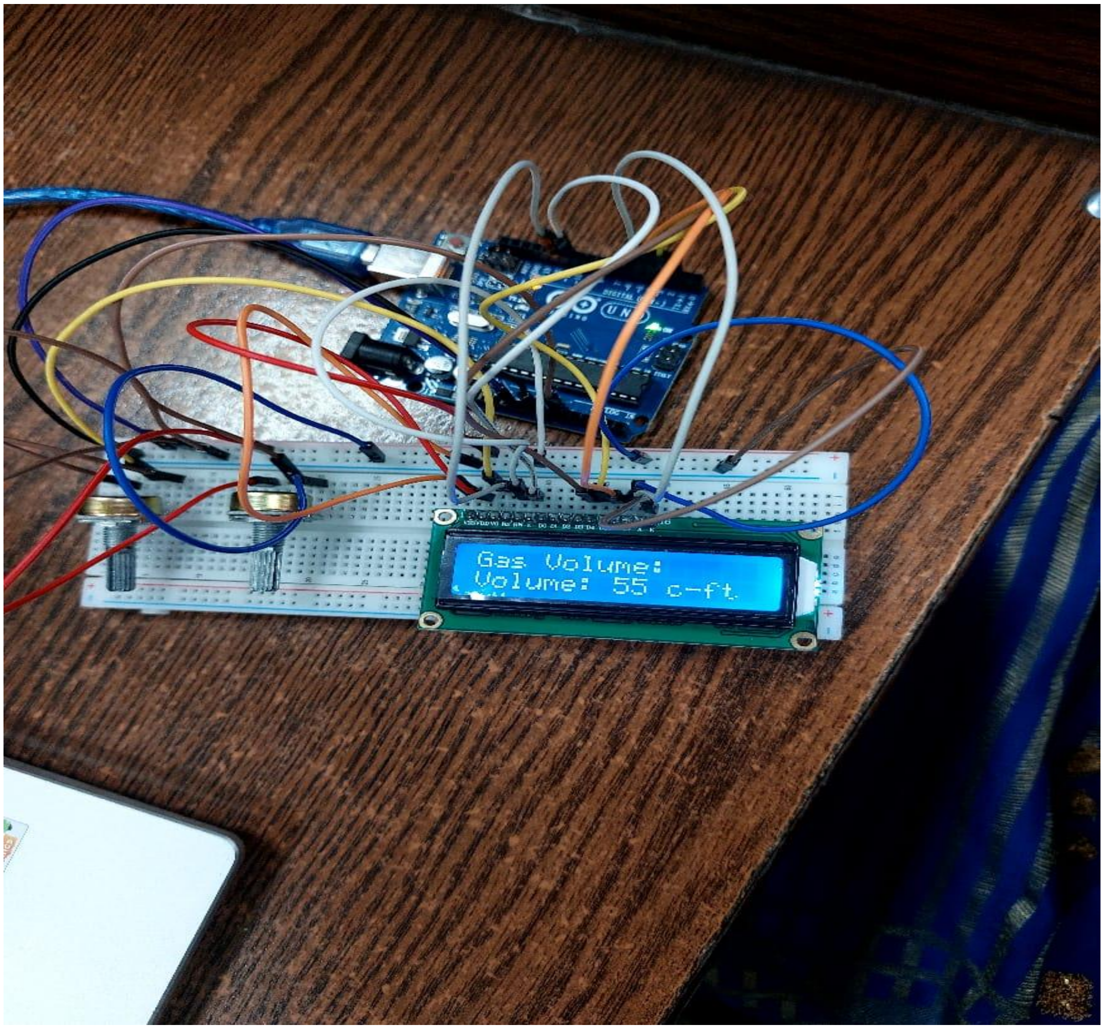
```
#include <LiquidCrystal.h>

// Initialize the LCD (RS, E, D4-D7 pins)
LiquidCrystal lcd(12, 11, 2, 3, 4, 5);

void setup() {
  lcd.begin(16, 2); // Initialize the LCD
  lcd.setCursor(0, 0);
  lcd.print("Gas Volume:"); // Print a test message on the first row


  Serial.begin(9600); // Initialize Serial Monitor
}


void loop() {
  int potValue = analogRead(A0); // Read potentiometer value
  int gasVolume = map(potValue, 0, 1023, 0, 100); // Map to 0-100 cubic feet


  lcd.setCursor(0, 1);
  lcd.print("Volume: ");
  lcd.print(gasVolume);
  lcd.print(" c-ft  "); // Add unit


  // Print debugging information
  Serial.print("Potentiometer Value: ");
  Serial.print(potValue);
  Serial.print(" | Gas Volume: ");
  Serial.println(gasVolume);


  delay(500); // Delay for stability
}
```

Task 2: LED-Based Volume Display

```
// Define LED pins
const int led1 = 2; // 1st LED (1-25% volume)
const int led2 = 3; // 2nd LED (26-50% volume)
const int led3 = 4; // 3rd LED (51-75% volume)
const int led4 = 5; // 4th LED (76-100% volume)

void setup() {
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);

  Serial.begin(9600); // Initialize Serial Monitor
  Serial.println("System Initialized");
}

void loop() {
  int potValue = analogRead(A0); // Read potentiometer value
  int gasPercentage = map(potValue, 0, 1023, 0, 100); // Map to
0-100%

  // Control LEDs based on volume percentage
  if (gasPercentage > 0 && gasPercentage <= 25) {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, LOW);
    digitalWrite(led3, LOW);
    digitalWrite(led4, LOW);
  } else if (gasPercentage <= 50) {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, HIGH);
    digitalWrite(led3, LOW);
    digitalWrite(led4, LOW);
  } else if (gasPercentage <= 75) {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, HIGH);
    digitalWrite(led3, HIGH);
    digitalWrite(led4, LOW);
  } else {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, HIGH);
    digitalWrite(led3, HIGH);
    digitalWrite(led4, HIGH);
  }

  // Debugging output
  Serial.print("Potentiometer Value: ");
  Serial.print(potValue);
  Serial.print(" | Gas Percentage: ");
  Serial.println(gasPercentage);

  delay(500); // Small delay for stability
}
```

Output

Task 1



Task 2

Serial Monitor Output

```
Potentiometer Value: 512 | Gas Percentage: 50%

Potentiometer Value: 768 | Gas Percentage: 75%

Potentiometer Value: 1023 | Gas Percentage: 100%
```

Images