**University of Central Punjab**

**Basic Electronic (Lab)**

****

**Group Project  
  
Presented By**

**Muhammad Muzammil L1F24BSDS0049**

**Areebah Moeed L1F24BSDS0073**

**Areeba Faheem L1F24BSDS0092**

**Sharique Fayyaz Malik L1F24BSDS0103**

**Ayesha Zaman L1F24BSDS0111**

**Submitted To**

**Khadija Moazzam**

**Faculty of Information and Technology(FOIT)**

**Table Of Content:**

1. Introduction
2. Key Features
3. Importance and Benefits of EVM
4. Components and Functions
5. Code Implementation
6. Explanation of Code
7. How Machine Works

**Electronics Voting Machine**

### ****Introduction****

An **Electronic Voting Machine (EVM)** is an efficient and reliable device for conducting elections. It simplifies the voting process by allowing voters to select their candidates electronically and ensures accurate vote counting. This project demonstrates a small-scale EVM using an Arduino microcontroller, a liquid crystal display (LCD), buttons, and LEDs.

**Key Features:**

 **User-Friendly Interface:**

The system is designed to be simple and intuitive. Voters interact with the EVM using physical buttons labeled for each candidate, and the results are displayed on a clear, backlit LCD screen. This simplicity ensures that the machine can be used by individuals with minimal technical knowledge.

 **Real-Time Counting:**

As soon as a vote is cast by pressing a button, the corresponding candidate's vote count is instantly updated and displayed on the LCD. This eliminates the delay involved in manual vote counting and ensures transparency during the voting process.

 **Result Calculation:**

Once all votes have been cast, the "Result" button triggers the calculation process. The machine determines the winner based on the highest number of votes and displays the result on the LCD. In the case of a tie, the machine displays an appropriate message.

 **LED Indicators:**

To enhance user interaction, the system uses LEDs to provide visual feedback. The green LED lights up to confirm a successful vote, while the red LED is used during the result announcement process, ensuring voters are aware of the system’s status at all times.

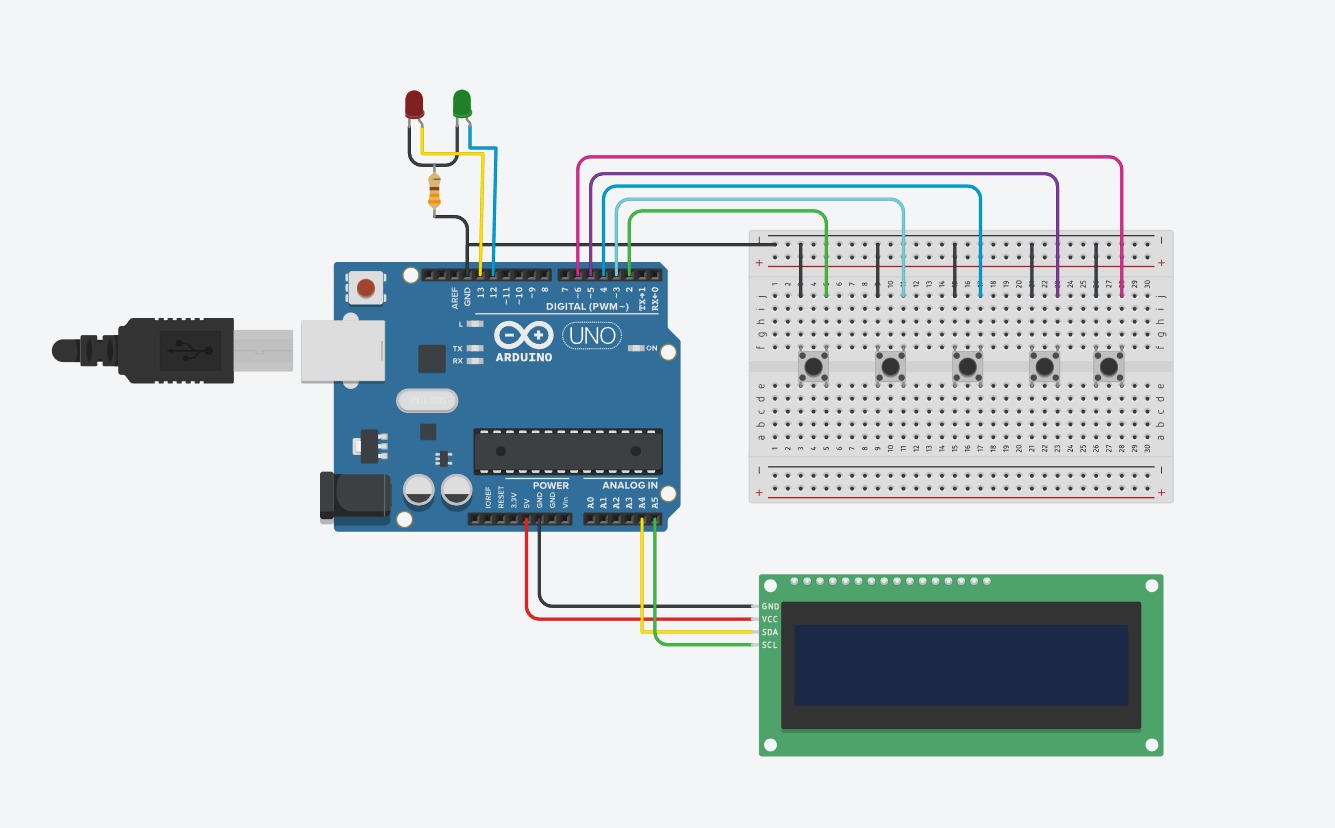
 **Reset After Results:**

After displaying the election results, the system automatically resets all vote counts to zero. This ensures the machine is ready for the next round of elections without any manual intervention

**Importance and Benefits of EVM**

* **Accuracy:**  
  Electronic Voting Machines eliminate the chances of human errors during vote counting. Each button press corresponds to a single, accurately registered vote, reducing the likelihood of disputes over vote tallies.
* **Efficiency:**  
  Traditional election processes involving paper ballots can be time-consuming, especially during counting and result declaration. EVMs simplify and speed up these processes, allowing results to be announced almost immediately after voting concludes.
* **Security:**  
  By using physical buttons and a secure microcontroller system, the EVM minimizes the risk of tampering or fraudulent activity. The use of LEDs and real-time display updates ensures transparency throughout the process.
* **Environmentally Friendly:**  
  Paper ballots lead to significant paper waste, especially in large-scale elections. EVMs reduce environmental impact by eliminating the need for paper altogether.
* **Scalability:**  
  This small-scale EVM can be easily adapted for larger elections by increasing the number of candidates, integrating biometric authentication for voters, or networking multiple machines together for centralized result aggregation.

.

****

**Components and Functions:**

 **Arduino UNO:**  
The Arduino UNO is the central microcontroller that processes inputs from the buttons and controls the outputs (LCD and LEDs). It runs the program that handles vote registration, result calculation, and display operations. Its versatility makes it an ideal choice for beginners and advanced developers alike.

 **LCD Display (16x2):**  
The 16x2 LCD is used to show information such as candidate names, vote counts, and election results. The Adafruit LiquidCrystal library simplifies communication with the LCD, allowing it to display text in real-time.

 **Buttons:**  
Five buttons are connected to the Arduino to handle voting and result operations:

* **4 buttons for voting**: Each button corresponds to a candidate.
* **1 button for results**: This button calculates the results and displays the winner.

 **LEDs (Red and Green):**

* **Green LED:**

Provides instant feedback when a vote is successfully registered. This assures voters that their vote has been counted.

* **Red LED:**

Lights up when the result is displayed, signaling the end of the voting process.

 **Resistors:**

Resistors protect the buttons and LEDs from excess current. They ensure that the components operate safely within their limits.

 **Breadboard:**  
The breadboard provides a platform for connecting all the components in a temporary and organized manner, making it easier to modify or troubleshoot the circuit.

1. **Code Implementation:**

**#include <Adafruit\_LiquidCrystal.h>**

**Adafruit\_LiquidCrystal lcd\_1(0);**

**#define sw1 2 // Button 1**

**#define sw2 3 // Button 2**

**#define sw3 4 // Button 3**

**#define sw4 5 // Button 4**

**#define sw5 6 // Button 5 for result**

**int vote1=0; // PTI**

**int vote2=0; // MQM**

**int vote3=0; // PPP**

**int vote4=0; // PMLN**

**void setup()**

**{**

**pinMode(sw1, INPUT);**

**pinMode(sw2,INPUT);**

**pinMode(sw3,INPUT);**

**pinMode(sw4,INPUT);**

**pinMode(sw5,INPUT);**

**pinMode(13,OUTPUT);// Red LED**

**pinMode(12,OUTPUT);// Green LED**

**lcd\_1.begin(16, 2);**

**lcd\_1.setCursor(0,0);**

**lcd\_1.print(" EVM ");**

**lcd\_1.setCursor(0,1);**

**lcd\_1.print("Circuit design ");**

**delay(1000);**

**digitalWrite(sw1, HIGH);**

**digitalWrite(sw2, HIGH);**

**digitalWrite(sw3, HIGH);**

**digitalWrite(sw4, HIGH);**

**digitalWrite(sw5, HIGH);**

**lcd\_1.clear();**

**lcd\_1.setCursor(0,0);**

**lcd\_1.print("PTI");**

**lcd\_1.setCursor(4,0);**

**lcd\_1.print("MQM");**

**lcd\_1.setCursor(8,0);**

**lcd\_1.print("PPP");**

**lcd\_1.setCursor(12,0);**

**lcd\_1.print("PMLN");**

**}**

**void loop()**

**{**

**lcd\_1.setCursor(0,0);**

**lcd\_1.print("PTI");**

**lcd\_1.setCursor(1,1);**

**lcd\_1.print(vote1);**

**lcd\_1.setCursor(4,0);**

**lcd\_1.print("MQM");**

**lcd\_1.setCursor(5,1);**

**lcd\_1.print(vote2);**

**lcd\_1.setCursor(8,0);**

**lcd\_1.print("PPP");**

**lcd\_1.setCursor(9,1);**

**lcd\_1.print(vote3);**

**lcd\_1.setCursor(12,0);**

**lcd\_1.print("PMLN");**

**lcd\_1.setCursor(13,1);**

**lcd\_1.print(vote4);**

**if(digitalRead(sw1)==0)**

**{**

**vote1++;**

**digitalWrite(12,HIGH);**

**delay(500);**

**while(digitalRead(sw1)==0);**

**digitalWrite(12,LOW);**

**delay(1000);**

**}**

**if(digitalRead(sw2)==0)**

**{**

**vote2++;**

**digitalWrite(12,HIGH);**

**delay(500);**

**while(digitalRead(sw2)==0);**

**digitalWrite(12,LOW);**

**delay(1000);**

**}**

**if(digitalRead(sw3)==0)**

**{**

**vote3++;**

**digitalWrite(12,HIGH);**

**delay(500);**

**while(digitalRead(sw3)==0);**

**digitalWrite(12,LOW);**

**delay(1000);**

**}**

**if(digitalRead(sw4)==0)**

**{**

**vote4++;**

**digitalWrite(12,HIGH);**

**delay(500);**

**while(digitalRead(sw4)==0);**

**digitalWrite(12,LOW);**

**delay(1000);**

**}**

**if(digitalRead(sw5)==0)**

**{**

**digitalWrite(13,HIGH);**

**int vote=vote1+vote2+vote3+vote4;**

**if(vote)**

**{**

**if((vote1 > vote2 && vote1 > vote3 && vote1 > vote4))**

**{**

**lcd\_1.clear();**

**lcd\_1.print("PTI Wins");**

**delay(5000);**

**lcd\_1.clear();**

**}**

**else if((vote2 > vote1 && vote2 > vote3 && vote2 > vote4))**

**{**

**lcd\_1.clear();**

**lcd\_1.print("MQM Wins");**

**delay(5000);**

**lcd\_1.clear();**

**}**

**else if((vote3 > vote1 && vote3 > vote2 && vote3 > vote4))**

**{**

**lcd\_1.clear();**

**lcd\_1.print("PPP Wins");**

**delay(5000);**

**lcd\_1.clear();**

**}**

**else if(vote4 > vote1 && vote4 > vote2 && vote4 > vote3)**

**{**

**lcd\_1.clear();**

**lcd\_1.print("PMLN Wins");**

**delay(5000);**

**lcd\_1.clear();**

**}**

**else**

**{**

**lcd\_1.clear();**

**lcd\_1.print(" Tie Up Or ");**

**lcd\_1.setCursor(0,1);**

**lcd\_1.print(" No Result ");**

**delay(5000);**

**lcd\_1.clear();**

**}**

**}**

**else**

**{**

**lcd\_1.clear();**

**lcd\_1.setCursor(0,0);**

**lcd\_1.print(" No Voting... ");**

**delay(5000);**

**lcd\_1.clear();**

**}**

**vote1=0;vote2=0;vote3=0;vote4=0;**

**lcd\_1.clear();**

**digitalWrite(12,LOW);**

**digitalWrite(13,LOW);**

**}**

**}**

**Explanation of Code:**

1. **Libraries and Initialization:**
   * The **Adafruit\_LiquidCrystal library** is used to communicate with the 16x2 LCD.
   * Buttons are mapped to digital pins 2–6 on the Arduino, and LEDs are connected to pins 12 and 13.
2. **Setup Function:**
   * The setup() function initializes all pins as input or output as required.
   * It displays an introductory message on the LCD and prepares the system for voting by displaying candidate names on the screen.
3. **Main Loop:**
   * The loop() function continuously checks the state of each button.
   * If a button is pressed, the corresponding candidate's vote count is incremented, and the green LED lights up briefly as feedback.
   * The updated vote counts are displayed on the LCD in real-time.
4. **Result Calculation:**
   * When the "Result" button is pressed, the total votes are calculated.
   * The candidate with the highest vote count is declared the winner, and their name is displayed on the LCD.
   * In case of a tie or no votes, a message is displayed indicating the result.
5. **Vote Reset:**
   * After displaying the results, all vote counts are reset to zero, ensuring the system is ready for the next round of elections.

**How Machine Works**

1. **Voting Process:**  
   Each voter approaches the EVM and presses the button corresponding to their chosen candidate. The system detects the button press using digitalRead() and increments the candidate's vote count. The green LED lights up momentarily to confirm the vote, and the updated vote count is displayed on the LCD.
2. **Result Declaration:**  
   Once voting is complete, the election official presses the "Result" button. The machine calculates the winner by comparing vote counts for all candidates. The name of the winning candidate is displayed on the LCD, accompanied by the red LED lighting up to indicate the announcement of results.
3. **Resetting the System:**  
   After the results are displayed, the machine automatically resets all vote counts to zero. This ensures that the system is ready for subsequent elections without requiring manual intervention.

