ELECTRONIC VOTING MACHINE USING ARDUINO UNO

Project report

Submitted by

Lakshmanan S - 2022504310

In partial fulfilment of the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS ENGINEERING,

MADRAS INSTITUTE OF TECHNOLOGY, ANNA UNIVERSITY,

CHENNAI-600044

(AUG-NOV 2024)

BONAFIDE CERTIFICATE

This is to certify that the project title ” **Electronic Voting Machine using Arduino uno**” is the Bonafide work of

Lakshmanan S - 2022504310

who carried out the summer project work (EC5512) under my supervision.

I attest that the work presented in this project report is original and has not been submitted elsewhere for any academic or professional purpose. **Lakshmanan S**, have demonstrated commendable skills and dedication in the completion of the project.

SIGNATURE

Dr.K.MARIAMMAL

Associate Professor,

Department of Electronics Engineering,

Madras Institute of Technology,

Anna University, Chennai-600044.

ACKNOWLEDGEMENT

We consider it as our privilege and our primary duty to express our gratitude and respect to all those who guided and inspired us in the successful completion of the project.

We would like to express our profound gratitude **to Dr. RAVICHANDRAN**, Dean, Madras Institute of Technology for providing us with this opportunity and a good environment.

We would also like to extend our sincere thanks to **Dr.D.MEGANATHAN,** Head of the Department, Department of Electronics Engineering, Madras Institute of Technology for providing us with the opportunity to carry out a thes is of our choice.

We would also like to express our great appreciation **to Dr. K. MARIAMMAL**, Associate Professor, Department of Electronics Engineering for guiding us sparing time and efforts throughout the completion of the project, which helped us enhance our skills and knowledge deeper in the domain.

We are eternally grateful to you.

We also thank all the teaching and non-teaching staff members of the Department of Electronics Engineering for their support in all the aspect.

NAME	REGISTER NUMBER
LAKSHMANAN	2022504310

ABSTRACT

This project explores the design and implementation of an Electronic Voting Machine (EVM) using Arduino, aiming to provide a secure, efficient, and cost-effective solution for modern voting systems. The EVM is developed to address issues such as tampering, manual errors, and inefficiencies inherent in traditional voting methods. The system incorporates an Arduino microcontroller, push buttons for candidate selection, an LCD display for user interaction, and a secure memory module for storing votes. The proposed EVM operates with a user-friendly interface, allowing voters to select their candidates with ease. The system ensures data integrity through real-time vote recording and safeguards against multiple voting using a voter authentication mechanism.

EVM or Electronic Voting machine is a simple electronic device used to record votes instead of ballot paper and boxes. Ballot boxes and papers used paper and pen technique which can be manipulated easily. Using Electronic voting machines manipulation is eliminated and free and fair elections can be held. Use of these electronic voting machines makes sure our democracy isn't affected. Many schools and colleges have elections for different posts, we cannot use official EVM for schools and colleges as it is very costly to eliminate this problem Simple EVMs have been made. Due to EVM the following points become possible:

TABULATION FOR CONTENTS

INTRODUCTION:

1. 1. Working Mechanism

Electronic Voting Machines (EVMs) are designed to facilitate secure and efficient elections by automating the process of voting and counting votes. An EVM typically consists of two units: the Control Unit and the Ballot Unit, connected by a cable. Voters select their candidate using a button on the Ballot Unit, and the Control Unit records the vote. EVMs eliminate manual vote counting, significantly reducing human errors and time. Their simplicity and portability make them suitable for diverse electoral settings.

1. 2. Security Challenges

While EVMs offer advantages in speed and accuracy, they are also susceptible to various security challenges. Potential vulnerabilities include tampering, software manipulation, and unauthorized access during storage or transport. Additionally, concerns over the lack of a verifiable paper trail in some systems have raised questions about transparency and trust. Addressing these issues requires robust encryption, regular security audits, and the implementation of Voter-Verifiable Paper Audit Trails (VVPAT) to enhance accountability.

1.3. Benefits and Limitations

EVMs bring multiple benefits, such as faster vote counting, reduced costs compared to paper ballots, and lower chances of invalid or fraudulent votes. However, they also have limitations, including dependency on power supply, potential technical malfunctions, and challenges in voter education. Accessibility for voters with disabilities and concerns about inclusivity in rural areas with limited infrastructure are also critical factors to consider.

OBJECTIVE

The primary objective of an electronic voting machine (EVM) is to enhance the efficiency, accuracy, and security of the electoral process. By automating voting and vote counting, EVMs significantly reduce the time required to declare results, minimizing delays and human errors. They are designed to ensure the integrity of elections by preventing fraudulent practices such as ballot stuffing and invalid votes.

Security is a critical aspect, with features implemented to protect against tampering, unauthorized access, and cyber threats. EVMs also aim to improve transparency and accountability through mechanisms like the Voter-Verifiable Paper Audit Trail (VVPAT), allowing voters to confirm their choices and enabling post-election audits. Accessibility and inclusivity are essential objectives, ensuring that the system is user-friendly and accommodates all voters, including those with disabilities. Furthermore, EVMs promote sustainability by reducing the reliance on paper ballots, making elections more environmentally friendly.

With scalable and adaptable designs, EVMs cater to diverse electoral needs, from local to national elections, while building public trust in the voting process. Over time, they also contribute to cost-effectiveness by lowering expenses related to manual processes, making them a vital tool for modern democracies

Problem Statement:

An Evaluation Module (EVM) machine plays a crucial role in the development and testing of embedded systems. It serves as a tool to validate the performance, compatibility, and functionality of hardware components and software interfaces in a controlled environment. However, the process of evaluating complex systems such as microcontrollers, sensors, communication modules, and signal processing circuits often encounters several key challenges that hinder effective testing. One of the primary issues faced is inconsistent data logging, where the system struggles to accurately record or process data due to limitations in memory management or improper interfacing. In many cases, response times may also be slow when interacting with multiple peripherals or sensors, which can significantly affect real-time system analysis and decision-making.

Another challenge faced by some EVMs is limited flexibility, where the module may not support a wide range of hardware configurations or may lack customizable functionality required for different testing scenarios. This limits its adaptability across various projects, which can become a significant bottleneck during development. Additionally, difficult user interfaces present a considerable barrier to efficient operation. Complex or unintuitive interfaces often lead to user errors, inefficient system setup, and longer debugging cycles

Moreover, many tradition.nal EVMs face poor error handling, resulting in systems that are not capable of detecting or managing faults properly. This can lead to inaccurate test results or system failures, affecting the reliability of the test outcomes. Without proper error detection mechanisms, users may overlook critical issues that could affect the performance of embedded systems in real-world conditions. These limitations create the need for an enhanced EVM that addresses these common problems to ensure that embedded system development is as efficient and effective as possible.

LITERATURE SURVEY

Since 1948, After every five times, India has conducted choices at a public position. The following styles are used to handpick an applicable seeker Paper ballot system Electronic Voting machine. PAPER BALLOT VOTING SYSTEM In India, before 1977 Paper Ballots were used to conduct choices at a public position. Votes captured in ballots can be stored for a veritably short period as the essay used in voting may discharge or ballot paper may lose its quality.

ELECTRONIC VOTING SYSTEM. The Idea of using an electronic voting machine was introduced by the principal election manager in 1997.The EVMs were designed by the Election Commission of India in collaboration with Bharat Electronics Limited, Bangalore and Electronics Corporation of India Limited(ECIL), Hyderabad.

Comparison of technologies used in EVMs.

2.1 Introduction and Implementation of EVMs in India

- **Kumar and Shukla (2010)** examined the phased implementation of EVMs in India, starting with pilot projects in 1982 and full-scale adoption by 2004. Their study highlighted how EVMs addressed issues such as ballot stuffing, invalid votes, and logistical challenges in managing paper ballots across large constituencies.
- **Election Commission of India (ECI, 2009)** published reports detailing the functionality, design, and advantages of Indian EVMs, emphasizing their simplicity, portability, and suitability for diverse electoral conditions.

2.2 Electoral Reforms and Policy Frameworks

- **Suri and Mahajan (2015)** analyzed electoral reforms in India, focusing on how EVMs reduced election-related malpractices, improved vote counting efficiency, and increased voter turnout.
- **Singh and Yadav (2017)** explored the role of EVMs in ensuring fair elections in politically sensitive states, noting a decline in booth capturing and voter intimidation post-EVM implementation.

2.3 Public Perception and Trust in EVMs

- **Pande and Sharma (2018)** conducted surveys across urban and rural India to assess public trust in EVMs. Their findings revealed that while urban voters trusted EVMs due to their speed and efficiency, rural voters had concerns about technology reliability and transparency.
- **Ghosh and Chakrabarti (2020)** studied the impact of VVPAT on voter confidence, finding that its introduction improved public perception of EVMs, especially in constituencies with a history of contested results.

2.4 Comparative Studies and Case Analyses

- **ECI and UNDP (2012)** presented a comparative study of India's EVMs with global electronic voting systems, showcasing the unique design features tailored for large-scale, multilingual, and diverse electorates.
- **Mehta and Saini (2020)** analyzed case studies of elections conducted using EVMs in states like Uttar Pradesh and Kerala, noting improvements in election management and reduced invalid votes
.

2.5 technological Innovations and Challenges

- **Mitra et al. (2019)** examined advancements in Indian EVM technology, such as tamper-resistant hardware and dual microcontrollers, which increased system security and reliability.
- **Rajan and Thomas (2021)** highlighted challenges related to EVM deployment in remote areas, including logistical difficulties and power supply issues, and proposed solar-powered solutions to ensure seamless operation.
-

Advantages and drawbacks of electronic systems.

Electronic systems offer numerous advantages, including high efficiency, precision, and automation, which make them essential in fields like healthcare, manufacturing, and communications. These systems enable faster processing, reduce human error, and allow for compact and integrated devices that combine multiple functions into a single unit, lowering costs while increasing capabilities. Additionally, electronic systems are highly reliable, require minimal maintenance, and are energy-efficient, contributing to long-term sustainability. Their programmability and flexibility make them versatile for a wide range of applications, and their ability to facilitate seamless communication has revolutionized industries and everyday life.

However, electronic systems also come with drawbacks. The initial costs for developing and implementing complex systems can be high, making them less accessible for some users. Their design and maintenance can be complicated, requiring specialized knowledge, and rapid technological advancements mean that devices quickly become obsolete.

Environmental factors like power surges or extreme conditions can cause failures, and many electronic systems are vulnerable to power outages. Moreover, electronic devices often rely on continuous power and contribute to electronic waste, raising sustainability concerns. As more systems become interconnected, security risks such as cyberattacks also become more prevalent. Despite these challenges, the benefits of electronic systems continue to outweigh their drawbacks in many applications.

SYSTEM DEVELOPMENT

3.1 TOOLS AND TECHNOLOGIES USED

The EVM project utilizes specific tools and technologies to achieve functionality, efficiency, and user interaction. Each component plays a vital role in the project. Below is a detailed explanation of the tools and technologies used and why they are needed.

3.1.1 HARDWARE USE:

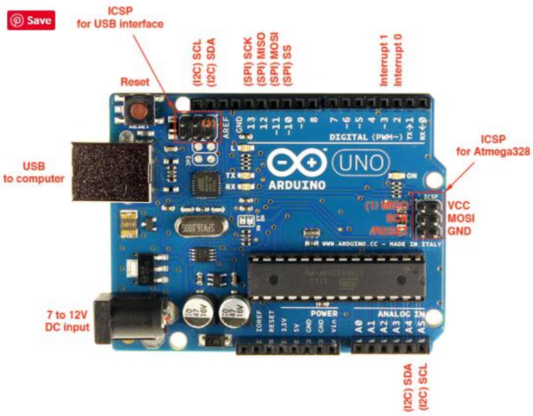


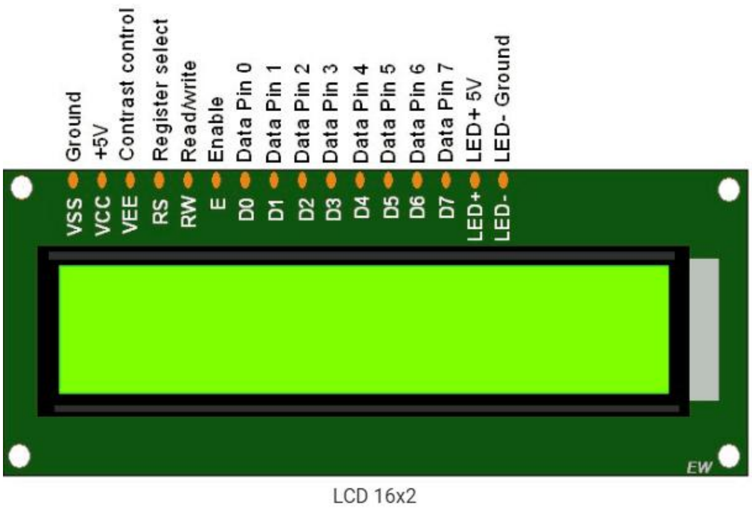
Fig. Arduino uno

The Arduino Uno is a widely used microcontroller board based on the ATmega328P, designed for beginners and professionals in electronics. It operates at 5V with a 16 MHz clock speed, providing reliable performance for various applications. The board features 14 digital I/O pins (6 with PWM capabilities) and 6 analog input pins, enabling it to interface with sensors, actuators, and other devices.

It is programmed using the Arduino IDE, supporting easy coding and uploading via a USB Type-B port. A voltage regulator and power jack allow operation through external power supplies (7-12V recommended). The ICSP header enables advanced programming, while onboard LEDs assist in debugging. Its robust design and open-source ecosystem make it ideal for prototyping, IoT, robotics, and automation projects.

3.1.2 LCD DISPLAY 16X2:

An LCD (Liquid Crystal Display) is an electronic display module that uses liquid crystals to control light for presenting text, symbols, or images. It operates by manipulating light through polarization and backlighting, making it energy-efficient. Commonly used in embedded systems, the most popular type is the 16x2 LCD, which displays 2 rows of 16 characters each. It is controlled by an HD44780 driver, allowing alphanumeric and custom characters. The display requires a 5V or 3.3V power supply and is interfaced with microcontrollers using parallel or I2C communication. A potentiometer adjusts the contrast, while the backlight ensures readability in low light. LCDs are widely used in Arduino projects for user interfaces.



Fig, LCD DISPLAY 16x2

3.1.3 SWITCH

In an electronic voting machine (EVM), buttons play a critical role in enabling voters to select their preferred candidate and confirm their choice. Each button corresponds to a specific candidate or option and is designed for durability, reliability, and ease of use. These buttons are typically momentary push buttons, which register an input only when pressed.

Fig. push buttion

3.2.4.Potentiometer

The 10k potentiometer in the electronic voting machine adjusts display brightness, volume levels, and sensor input sensitivity. It optimizes the visibility of the interface, enhances user experience with audio feedback, and ensures accurate operation of touch-sensitive buttons. Its inclusion improves usability and functionality, catering to diverse user preferences

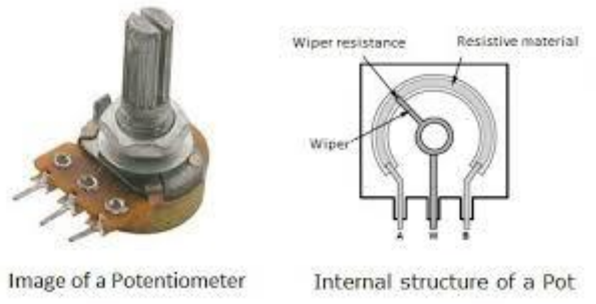


Fig Potentiometer

3.1.5: Resistor:

Resistors play crucial roles in various circuitry. They are used for current limiting to protect components from excessive current flow, voltage division to adjust signal levels, and pull-up or pulldown configurations for digital inputs. In voting machines, resistors help regulate voltage levels, control currents, and ensure the proper functioning of electronic components, contributing to the reliability and stability of the system. Additionally, they aid in setting thresholds for sensors, enabling accurate detection of inputs such as button presses or card insertions.

3.1.6 LED

Arduino is an open-source electronics platform that simplifies the creation of interactive projects. It consists of both hardware—such as microcontroller boards like Arduino Uno, Nano, or Mega—and software, primarily the Arduino Integrated Development Environment (IDE). The platform enables users to write, upload, and run code on the boards to control a variety of components, such as LEDs, sensors, motors, and displays.

Its versatility makes it popular among beginners and professionals for projects ranging from simple blinking LEDs to complex automation and Internet of Things (IoT) systems. Arduino's extensive online community provides abundant resources, tutorials, and libraries, making it accessible and adaptable for countless applications.

3.1.7 .PROJECT DAMO:

The Electronic Voting Machine (EVM) project is a simplified, Arduino-based system designed to replicate the functionality of a real-world voting machine. It allows users to cast votes for predefined candidates, view live vote counts, and display the winner once the voting process is completed. The EVM consists of buttons for each candidate, an LCD screen to display information, and LEDs to provide visual feedback.

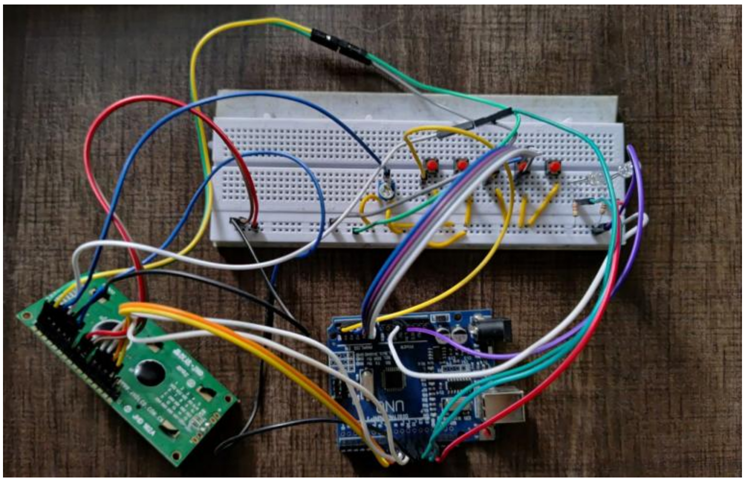


Fig ,project demo

3.2.1 SOFTWARE USED:

ARDUINO IDE The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment.

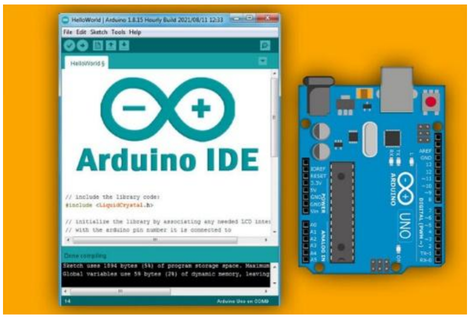


Fig. Arduino IDE

The program or code written in the Arduino IDE is often called as sketching. Connect the board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '. into

Features

The Arduino IDE comes with a variety of essential features for embedded programming, including a text editor for writing code, a message area for error feedback, and a toolbar with buttons for common actions like verifying, compiling, and uploading code to the microcontroller. Additionally, the IDE supports a range of libraries that simplify working with different sensors, displays, and communication protocols, which speeds up the development process and reduces complexity

Fig,select port

The Arduino IDE 2 is a versatile editor with many features. You can install libraries directly, sync your sketches with Arduino Cloud, debug your sketches and much more. In this section, some of the core features are listed, along with a link to a more detailed article

Advantage:

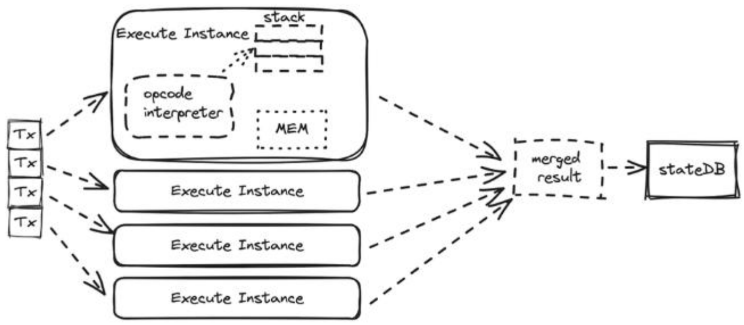
- It is simple and easy to use. ...
- Arduino Uno can be used to build many projects, including LED blinkers, Robots etc.
- It consists of various pins, which makes it more compatible and can be used to connect different electronic components.
- Arduino Uno boards are cheap in comparison to other microcontroller boards.

Disadvantage

- Limited Memory and Processing Power. ...
- Limited Support for Some Communication Protocols. ...
- Limited Real-Time Performance. ...
- Limited Security Features.

3.3 PROTOCOL:

The operation of Electronic Voting Machines (EVMs) in India follows a secure and transparent procedure to ensure fair elections. Here's a detailed explanation of the protocol.



Schematic diagram of parallel EVM

Fig. protocol

3.3.1 Voter Verification

When a voter arrives at the polling station, they present their identification to the polling officials. The voter's details are checked against the electoral roll to confirm their eligibility to vote. Once verified, the polling officer marks their finger with indelible ink to prevent multiple votes.

3.3. 2. Activation of the EVM

The polling officer uses the **Control Unit** of the EVM to activate the **Ballot Unit**, signaling that the voter is now permitted to cast their vote. This ensures that the machine is ready only for one voter at a time.

3.3. 3. Casting the Vote

The voter proceeds to the Ballot Unit, a simple device with a list of candidates and their respective symbols, each linked to a button. The voter presses the button corresponding to their chosen candidate. Upon pressing, the machine emits a "beep" sound to confirm that the vote has been successfully recorded.

3.3.4 Voter Verifiable Paper Audit Trail (VVPAT)

If the EVM is equipped with a **Voter Verifiable Paper Audit Trail (VVPAT)**, the voter sees a printed slip displayed in a transparent window. This slip contains the candidate's name and symbol, providing the voter a chance to verify their selection. The slip remains visible for a few seconds before being securely stored in the VVPAT box.

3.3.5 Securing the Vote

Once the vote is cast, the machine resets itself and is ready for the next voter. The Control Unit securely stores all recorded votes and can only be accessed by authorized personnel after the polling concludes.

USER INPUT HANDLING

OPERATION OF EVM:

The operation of an Evaluation Module (EVM) involves several key steps. First, it receives an external signal, such as an RF or digital signal, which is then conditioned through amplifiers, filters, or converters to match the required format. If the signal is analog, it is converted to digital via an ADC for further processing. The digital signal is processed by a microcontroller, DSP, or FPGA to extract or modify the necessary information.

The EVM typically includes a control interface, such as USB or Ethernet, for configuration and parameter adjustments. Processed data may be temporarily stored in memory or logged for analysis. The system then outputs the processed signal in the desired format (analog, digital, or RF). A power management system regulates and distributes power to the components, ensuring proper operation. Additionally, the EVM provides feedback through LEDs, displays, or software interfaces, allowing users to monitor performance. Finally, integrated measurement tools enable detailed analysis of signal characteristics, making the EVM a critical tool for testing and development.

4.1 Initialization:

In this step, an instance of the Liquid Crystal class is initialized with the pin numbers for the RS, EN, D4, D5, D6 and D7 of LCD (connected to pins 11, 10, 9, 8, 7 and 6 of Arduino) and, all the variables used are declared and assigned initial values. The pin modes for the push buttons and LED outputs are set (A0, A1, A2, A5, A4 of Arduino set as input and, 13 and 12 set as output).

4.2. Checking for the pressed button:

To check for a pressed button using an Arduino in an EVM setup, you can use the digitalRead() function to monitor the state of a button connected to a digital input pin. Below is an example of how you can achieve this.

Wiring:

- Connect one terminal of the button to a digital pin (e.g., pin 2) on the Arduino.
- Connect the other terminal of the button to the ground (GND).
- Enable the **internal pull-up resistor** on the pin in the code to avoid external resistors.

Key Detection Process:

Port Configuration: The push buttons (switches) of the ballot unit are connected to the pins A0, A1, A2, A5, and A4 of the Arduino. To configure the port for an Arduino, start by connecting the Arduino board to your computer using a USB cable.

Ensure that you have the Arduino IDE installed, which can be downloaded from the official Arduino website. Once the board is connected, open the Arduino IDE and navigate to the **Tools** menu. Under the **Port** option, you'll see a list of available ports. Select the port corresponding to your Arduino board, typically labeled as COM3, COM4 on Windows, If you're unsure of the correct port, disconnect the board and observe which port disappears from the list, then reconnect it to verify. Additionally, ensure that the correct board type is selected under **Tools > Board** to match your specific Arduino model. After completing these steps, the Arduino is ready to upload code and communicate with the IDE.

Detection Algorithm:

The controller continuously reads data from the pins mentioned above and checks whether one of them has released a low signal. This method allows for the efficient detection of key presses on the ballot unit, ensuring accurate recording of voter selections in the Electronic Voting Machine (EVM).

A switch can be used with an Arduino to control specific actions, such as toggling an LED, activating a motor, or triggering a function. To set it up, connect one terminal of the switch to a digital input pin on the Arduino, such as pin 7, and the other terminal to the ground (GND). Enable the internal pull-up resistor in the Arduino by configuring the pin as INPUT_PULLUP in the code. This ensures the pin reads HIGH when the switch is not pressed and LOW when the switch is pressed, as the circuit completes and pulls the voltage to ground. By reading the state of the pin using digital Read(), the Arduino can execute commands based on whether the switch is pressed or released. This simple configuration is reliable for projects that require user input through a physical button or switch.

Process of Incrementing, Output Display and Resetting:

The process of incrementing, output display, and resetting in an Evaluation Module (EVM) involves managing inputs, processing data, and providing feedback to the user. The incrementing begins when a user interacts with a button or sensor, which is detected as an input signal. To ensure reliability, debounce handling is implemented to filter out signal noise. Each valid input triggers an increment operation, updating a counter or data value stored in the EVM's memory.

The updated value is then displayed on an output interface, such as an LCD, LED, or other visual module, providing real-time feedback. If a reset action is required, triggered by another button press or a specific condition, the system clears the counter or data, setting it back to its initial state and updating the display accordingly. This cycle of incrementing, output display, and resetting ensures efficient interaction and monitoring in various applications.

4.3 METHODOLOGY

Designing an Electronic Voting Machine (EVM) using Arduino Uno involves several steps, from conceptualization to implementation. The project begins with identifying the need for a digital voting system to enhance efficiency, accuracy, and user-friendliness. The system comprises hardware components such as an Arduino Uno microcontroller, push buttons or a keypad for vote selection, an LCD display for instructions and candidate names, LED indicators or a buzzer for feedback, and an EEPROM for storing vote counts securely. Power can be supplied through a USB or battery.

The design includes connecting these components based on a predefined circuit schematic. The Arduino acts as the central controller, managing inputs from buttons or a keypad and displaying results on the LCD. The software is developed using the Arduino IDE, with logic to handle vote registration, confirmation, and secure storage. Libraries like Liquid Crystal. facilitate LCD control, while ensures data persistence.

The EVM works by displaying candidate options on the LCD, allowing users to select their choice using buttons or a keypad. Each vote increments the respective candidate's count, which is stored in the . The system provides real-time feedback through LEDs or a buzzer and includes an administrative feature to display or reset vote counts securely. Testing involves verifying individual components and simulating voting sessions to ensure the system's reliability.

Security measures, such as voter authentication using RFID or PIN codes, prevent unauthorized or duplicate voting. Finally, the system is enclosed in a tamper-proof casing and accompanied by detailed documentation, including circuit diagrams, source code, and a user manual. This methodology ensures a robust and user-friendly EVM, capable of handling secure and efficient voting processes.

SOURCE CODE:

```
#include<LiquidCrystal.h>
LiquidCrystal lcd(11,10,9,8,7,6);
#define sw1 A0 // Button 1
#define sw2 A1 // Button 2
#define sw3 A2 // Button 3
#define sw4 A3 // Button 4
#define sw5 A4 // Button 5 for result
int vote1=0;
int vote2=0;
int vote3=0;
int vote4=0;
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.begin(16, 2);
lcd.setCursor(0,0);
lcd.print("VOTING MACHINE ");
lcd.setCursor(0,1);
```

```
lcd.print("Circuit desigh ");
delay(3000);
digitalWrite(sw1, HIGH);
digitalWrite(sw2, HIGH);
digitalWrite(sw3, HIGH);
digitalWrite(sw4, HIGH);
digitalWrite(sw5, HIGH);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("BJP");
lcd.setCursor(4,0);
lcd.print("INC");
lcd.setCursor(8,0);
lcd.print("AAP");
lcd.setCursor(12,0);
lcd.print("OTH");
}
void loop()
{
lcd.setCursor(0,0);
lcd.print("A");
lcd.setCursor(1,1);
lcd.print(vote1);
lcd.setCursor(4,0);
lcd.print("B");
lcd.setCursor(5,1);
```

```
lcd.print(vote2);
lcd.setCursor(8,0);
lcd.print("C");
lcd.setCursor(9,1);
lcd.print(vote3);
lcd.setCursor(12,0);
lcd.print("D");
lcd.setCursor(13,1);
lcd.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);
```

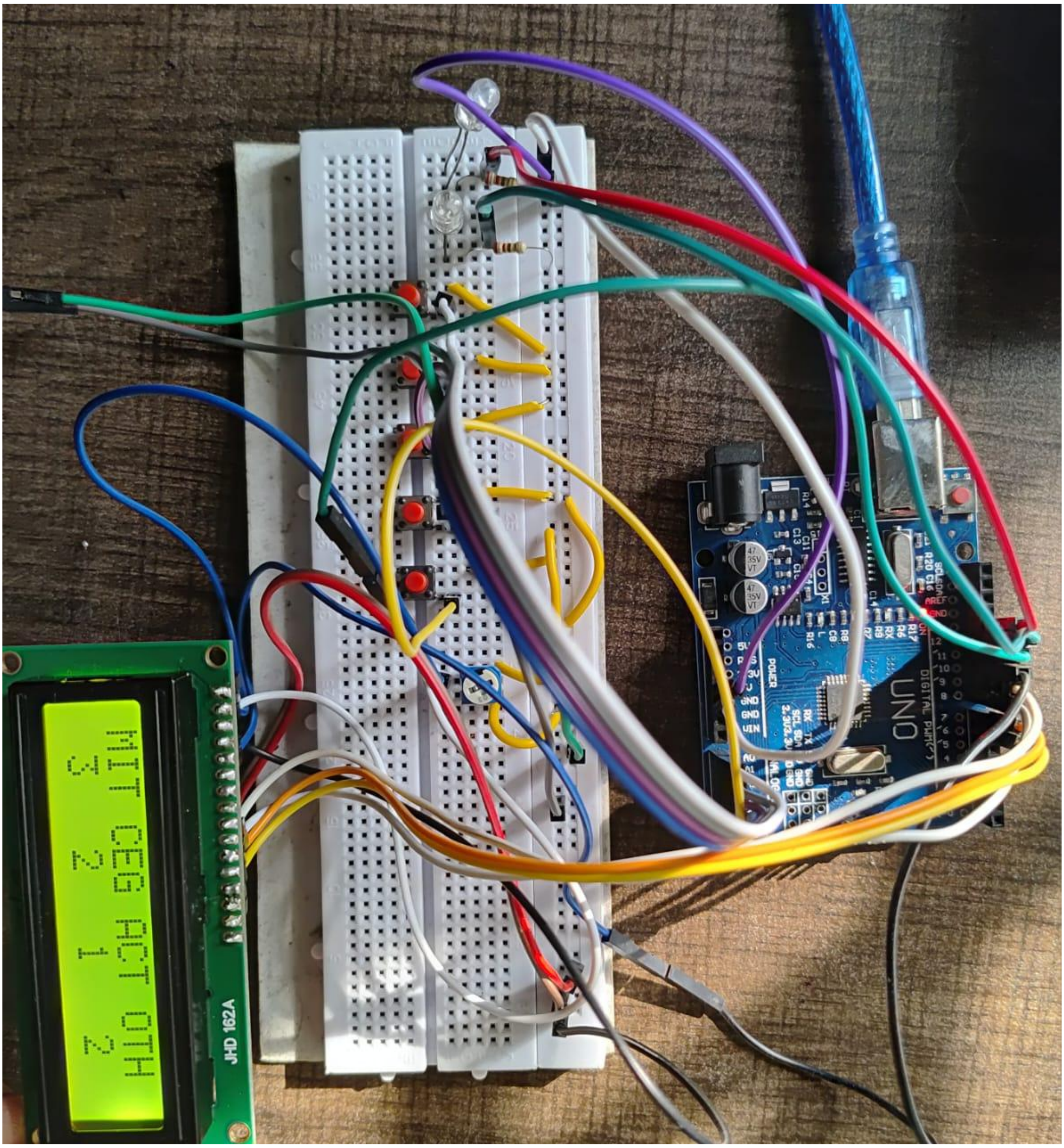
```
  delay(500);
while(digitalRead(sw2)==0);
digitalWrite(12,LOW);
delay(1000);

}

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);
```

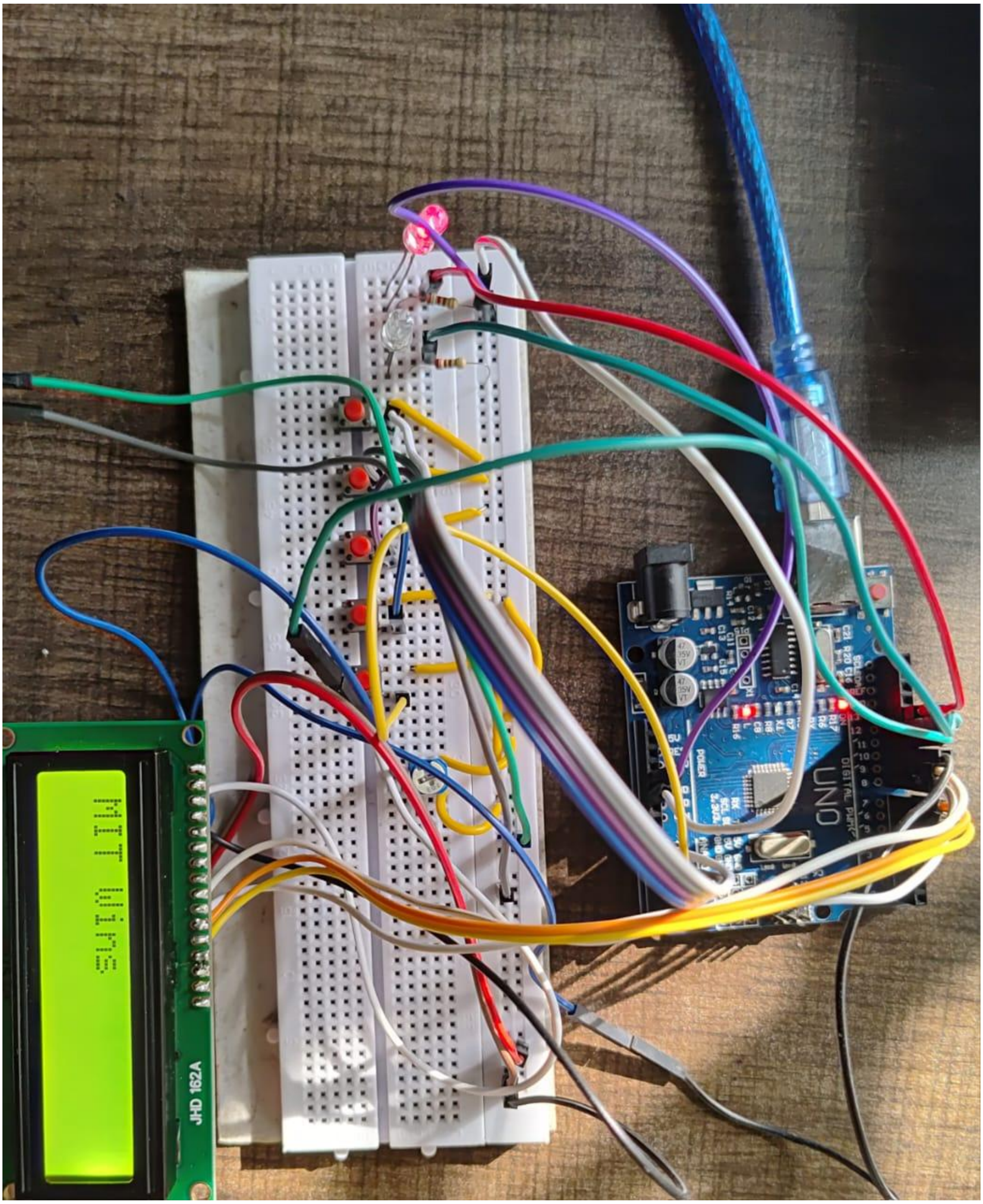
```
delay(1000 );
}

if(digitalRead(sw5)==0)
{
digitalWrite(13,HIGH);
int vote=vote1+vote2+vote3+vote4;
if(vote)
{
if((vote1 > vote2 && vote1 > vote3 && vote1 > vote4))
{
lcd.clear();
lcd.print("BJP Wins");
delay(5000);
lcd.clear();
}
else if((vote2 > vote1 && vote2 > vote3 && vote2 > vote4))
{
lcd.clear();
lcd.print("INC Wins");
delay(5000);
lcd.clear();
}
else if((vote3 > vote1 && vote3 > vote2 && vote3 > vote4))
{
lcd.clear();
```

```
lcd.print("AAP Wins");
delay(5000);
lcd.clear();
}
else if(vote4 > vote1 && vote4 > vote2 && vote4 > vote3)
{
lcd.setCursor(0,0);
lcd.clear();
lcd.print("OTH Wins");
delay(5000);
lcd.clear();
}

else if(vote4 > vote1 && vote4 > vote2 && vote4 > vote3)
{
lcd.setCursor(0,0);
lcd.clear();
lcd.print("OTH Wins");
delay(2000);
lcd.clear();
}

else
{
lcd.clear();
lcd.print(" Tie Up Or ");
```

```
lcd.setCursor(0,1);
lcd.print(" No Result ");
delay(5000);
lcd.clear();
}


}
else
{
lcd.clear();
lcd.setCursor(0,0);
lcd.print("  No Votingâ€¦.         ");
delay(5000);
lcd.clear();
}
vote1=0;vote2=0;vote3=0;vote4=0,vote=0;
lcd.clear();
digitalWrite(12,LOW);
digitalWrite(13,LOW);
}


}
```

RESULTS :




JHD 162A
UNO

CODE EXPLANITION:

1. Purpose of the Code

The goal of the code is to create an automated system where:

- Users can vote for predefined candidates by pressing buttons.
- Votes are counted and displayed in real time on an LCD.
- The results are calculated and displayed when requested, along with feedback through LEDs.

2. Initialization and Setup

Hardware Initialization

- The system uses:
  - Buttons as inputs to cast votes and display results.
  - LEDs for feedback to indicate successful actions.
  - An LCD screen to display real-time vote counts and results.

Setup Logic

- The setup() function initializes all the components:
  - Buttons are set as inputs using pinMode().
  - LEDs are set as outputs.
  - The LCD is initialized to display messages and prepare for voting.

Pre-Voting Display

- After setup, the LCD shows the voting options ("BJP", "INC", "AAP", "OTH") along with their respective vote counters initialized to zero.

3. Main Code Logic

The loop() function continuously checks for button presses and handles the voting process, live vote count updates, and result calculation.

Real-Time Voting and Feedback

- Each button corresponds to a candidate (e.g., sw1 for BJP, sw2 for INC).
- When a button is pressed:
  - The associated vote counter is incremented.
  - The LCD immediately updates the vote count in real time.
  - The green LED lights up briefly to confirm a successful vote.
  - A debounce mechanism ensures only one vote is registered per button press.

Result Calculation and Display

- When the result button (sw5) is pressed, the system calculates and displays the voting results:
  - Case 1: No Votes Cast:
    - If all vote counters are zero, the LCD displays "No Voting".
  - Case 2: Winner Determination:
    - The system compares the vote counts of all candidates.
    - The candidate with the highest vote count is declared the winner, and their name is displayed on the LCD.
  - Case 3: Tie Condition:
    - If two or more candidates have the same highest vote count, the system displays "Tie Up Or No Result".
  - During result calculation, the red LED lights up for visual feedback.

4. Reset Mechanism

- After displaying the result for 5 seconds, the system:
  - Resets all vote counters to zero.
  - Clears the LCD.
  - Turns off the LEDs.
  - Prepares the system for a new voting session.

5. Core Functionalities

Debouncing for Accurate Input

- When a button is pressed, the code waits (while(digitalRead() == 0)) until the button is released.
- This ensures that only one vote is registered per press.

Dynamic Display Updates

- The LCD dynamically updates the vote counts in real time, providing transparency to the voting process.

Result Processing

- The code uses conditional statements (if-else) to determine the winner or handle tie conditions.

LED Feedback

- The green LED confirms a successful vote, enhancing user interaction.
- The red LED signals the result calculation phase, indicating that the system is processing data.

Applications and Limitations

Applications

- Demonstrates the concept of electronic voting systems.
- Can be expanded for real-world use by adding security features and scalability.

Limitations

- This code handles only basic functionality (e.g., ties are not resolved beyond detection).
- No data storage or security features are implemented.

SPECIFICATION OF EVM:

An electronic voting machine (EVM) based on Arduino is a compact and efficient system designed for small to medium-scale elections. The core of the system is an Arduino Uno or Mega microcontroller, powered by a 5V DC supply, which acts as the primary control unit. The voting interface consists of push buttons or capacitive touch sensors, allowing users to cast votes easily.

A 16x2 or 20x4 LCD screen provides clear instructions, displays candidate names, and shows results. The system supports up to five candidates, with the potential for expansion using additional I/O modules. For feedback, LED indicators confirm button presses, and an optional buzzer provides audio signals for valid or invalid actions.

To ensure data integrity, votes are securely stored in an external EEPROM or on an SD card module, with data encryption and timestamping enabled via a real-time clock (RTC) module. Security is further enhanced with a password-protected admin mode and authentication mechanisms like RFID modules or fingerprint scanners.

A numeric keypad can also be integrated for voter identification. Anti-tampering measures include a lockable enclosure and tamper-detection sensors. The system enforces a strict one-vote-per-session rule using debounce logic and lockout timers to prevent accidental or fraudulent votes.

Connectivity options, such as Wi-Fi or Bluetooth modules (e.g., ESP8266), enable remote monitoring of the polling process. A reset button is provided for administrators to reinitialize the machine securely. The system is highly reliable, with error detection features for invalid inputs or hardware malfunctions.

It also includes a small rechargeable battery to ensure continuous operation during power outages. Votes are automatically counted and displayed securely to authorized personnel, ensuring transparency. Developed using the Arduino IDE with C/C++ programming, the system leverages libraries such as Liquid Crystal, SD, and EEPROM to simplify implementation.

This EVM is portable, cost-effective, and scalable, with durable components for repeated use. Its modular design allows the addition of new features or extended capacity for more candidates, making it ideal for elections in schools, organizations, or small communities.

Applications of Arduino-Based Electronic Voting Machine

1. Educational Institutions: Used for conducting student council elections in schools and universities.

2. Community Organizations: Suitable for decision-making processes and elections in clubs, societies, and local bodies.

3. Small-Scale Elections: Ideal for local elections in villages, neighborhoods, or small municipalities.

4. Corporate Use: Enables secure voting for corporate boards, employee polls, or internal decision-making.

5. Non-Profit Organizations: Conducting fair and transparent elections for leadership roles.

6. Survey and Feedback Collection: Simplified mechanism for collecting opinions in a structured way.

7. Prototype for Large-Scale Voting: Acts as a prototype for developing advanced, secure voting systems.

8. Workshops and Education: Demonstrates the functionality and technical aspects of electronic voting in technology workshops.

Advantages of Arduino-Based Electronic Voting Machine

1. Cost-Effective: Uses affordable and widely available components, making it suitable for low-budget implementations.

2. Compact and Portable: Lightweight and small, making it easy to transport

3. Ease of Use: Simple interface with buttons, LEDs, and a display ensures accessibility for all users.

4. Customizable: Can be tailored to specific requirements, such as the number of candidates or security features.

5. Faster Results: Automatic vote counting eliminates manual tallying, reducing time and effort.

6. Improved Accuracy: Reduces human errors in vote counting and recording.

7. Secure: Data encryption and tamper-proof mechanisms protect against manipulation or fraud.

8. Environmentally Friendly: Eliminates the need for paper ballots, contributing to eco-friendly practices.

9. Scalable: Modular design allows for additional features, such as connectivity or more candidate options.

10. Educational Value: Provides practical learning for students in electronics and programming.

Disadvantages of Arduino-Based Electronic Voting Machine

1. Limited Scale: Suitable only for small to medium-scale elections due to hardware constraints.

2. Data Security Risks: While secure, it may be vulnerable to advanced hacking if robust encryption isn't implemented.

3. Dependence on Power Supply: Requires consistent power; a failure can disrupt the voting process.

4. Component Failure: Issues like button malfunction or display errors can affect reliability.

5. Limited Voter Authentication: Basic systems may lack advanced biometric or multi-factor authentication.

6. Skill Requirement: Requires technical expertise to design, maintain, and troubleshoot the system.

7. Lack of Standardization: Variability in design may lead to inconsistencies in functionality or security.

8. Initial Setup Time: Requires time for programming, wiring, and testing before deployment.

9. Manual Intervention: In case of system failure, manual voting or intervention may be necessary, impacting trust.

10. Hardware Costs for Add-ons: Advanced features like fingerprint scanners or connectivity modules can increase costs.