PROJECT REPORT

ON

"EVM machine (Arduino uno)"

THIS PROJECT & RESEARCH REPORT HAS BEEN INDEPENDENTLY DEVELOPED BY ME AND IS NOT SUBMITTED TO ANY COLLEGE OR UNIVERSITY FOR ACADEMIC EVALUATION

Developed by

Name:-Suzen Kumar Mohanty (**Reg.No**.2421331063)



SKMP PROJECT MANAGEMENT

DEPARTMENT OF IOT

POLISHLINE.BALASORE, ODISHA, INDIA

MAY-2026

“Dedicated to my beloved parents ,and teachers who have worked hard throughout my education”

SKMP PROJECT MANAGEMENT

CERTIFICATE

This is to certify that the project entitled" **EVM machine (Arduino uno)***"*

Has been independently developed and completed by Suzen Kumar Mohanty for the purpose of enhancing technical knowledge, skills, and practical understanding.

This project is a self-initiated work carried out with person al interestand research. It has not been submitted to any college, university, or institution for the award of any degree or diploma.

The work presented in this project is genuine and has been completed by the developer himself.



Suzen Kumar Mohanty

(Project Developer)

Date: 26/05/2026

Place:-Balasore,Odisha,India

SELF ATTESTATION

I here by declare that I have personally worked on the project entitled " **EVM machine (Arduino uno)**".

All the data, analysis, and implementation presented in this project are the result of my own efforts and genuine work .Any external sources of information, if used, have been properly acknowledged.

This project is created solely for learning , skill development ,and professional improvement, and has not been submitted to any institution for academic purposes.



SKMP PROJECT MANAGEMENT

Suzen Kumar Mohanty

(Project Developer)

Date: 26/05/2026

Place:Balasore,Odisha,India

DECLARATION

I hereby declare that the project titled "**EVM machine (Arduino uno)"i**s an original work carried out by me based on my own idea, research, and implementation.

This project that has not been submitted previously ,in part or in full, to any college, university, or institution for the award of any degree, diploma, or certification.

I confirm that all the information presented in this project is true to the best of my knowledge. Any references or sources used have been properly acknowledged.

I take full responsibility for the authenticity and originality of this work.

SKMP PROJECT MANAGEMENT

Name: Mr. Suzen Kumar Mohanty

Branch: Computer Science And Engineering

Institution:[Srinix Collage Of Engineering ,Independent]

Date:[26/05/2026]

My PortPolio Link:-https://suzen-kumar-mohanty-8fdv.vercel.app/

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to everyone who contributed, directly or indirectly, to the development of this project, "**EVM machine (Arduino uno)**."

This project is the result of my personal interest, dedication, continuous learning, and exploration of electronics, automation, sensor technology, and control systems.

I am thankful to the developers of Arduino, the creators of open-source hardware and software resources, and the technical communities whose documentation, tutorials, and knowledge-sharing platforms helped me understand and implement the concepts used in this project.

I would also like to acknowledge the importance of self-learning, practical experimentation, and problem-solving throughout the development process. These experiences helped me strengthen my technical knowledge and improve my understanding of real-world automation systems.

Finally, I am grateful to everyone who supports innovation, DIY engineering, and independent technical projects. This work reflects my enthusiasm for learning, creating, and exploring new possibilities in technology.

Developed By:

Suzen Kumar Mohanty

ABOUT MYSELF



I am Mr. Suzen Kumar Mohanty , a passionate and dedicated Web Developer and SOC Analyst (Level 1) from the field of Computer Science and Engineering, with strong expertise in HTML, CSS, JavaScript, and I specialize in building scalable, responsive, and high-performance web applications that deliver seamless user experiences across all devices and screen sizes.

With a keen eye for detail, I focus on writing clean, maintainable, and well-structured code. My development approach is centered around Creating intuitive user interfaces, ensuring both functionality and aesthetic quality. I follow modern front-end development practices such as component-based architecture, responsive design principles, and performance optimization techniques to build efficient and user-friendly applications.

Beyond web development, I actively work on IoT-based solutions, where I integrate hardware and software to create smart, real-world systems. I have hands-on experience with Arduino Uno and various sensors, enabling me to develop projects such as environmental monitoring systems, smart automation solutions, and real-time data tracking applications.

One of my key strengths lies in connecting IoT devices with web technologies. I build interactive dashboards that allow real-time data visualization, giving users the ability to monitor, analyze, and control devices remotely. My work focuses on bridging the gap between the physical and digital world by transforming raw sensor data into meaningful insights.

I am deeply interested in developing innovative, data-driven solutions that combine web technologies with embedded systems. My goal is to create impactful applications that solve real-world problems while improving efficiency, automation, and user experience.

With a strong passion for continuous learning and innovation, I consistently explore new tools, frameworks, and technologies to enhance my skills and deliver high-quality, future-ready solutions.

Name: Mr. Suzen Kumar Mohanty

Branch: Computer Science And Engineering

MyPortPolioLink:-https://suzen-kumar-mohanty-8fdv.vercel.app/

ABSTRACT

The project aims to design an Arduino Uno-based Electronic Voting Machine prototype that allows an authorised operator to enable voting, allows a voter to select one candidate, records the vote in non-volatile memory, displays candidate-wise totals, and provides a simple voter-verifiable receipt mechanism. The system uses push buttons as candidate-selection inputs, LEDs for visual feedback, a buzzer for confirmation, an LCD display for status and results, and EEPROM for retaining vote counts during power failure.

Real election EVMs are secure, certified, standalone devices with a Control Unit, Ballot Unit, and Voter Verifiable Paper Audit Trail (VVPAT). This Arduino project reproduces only the conceptual voting workflow and must not be treated as an election-grade voting system.

SKMP PROJECT MANAGEMENT
SKMP PROJECT MANAGEMENT

LIST OF FIGURES

Figure No.	Figure Title	Page No.
01	Arduino Evm image	01
02	Arduino Uno Board	08
03	Ultrasonic Sensor	09
04	Resistor 220 ohm	10
05	Bread board	11
06	Servo Motor(Sg90)	12
07	Jumper Wire	13
08	Led	14
09	Usb Cable	15
10	Power Supply 9v Battery with Connector	16
11	Circuit Diagram	17
12	Block Diagram	18
13	Evm Final Result	24

SKMP PROJECT MANAGEMENT
SKMP PROJECT MANAGEMENT

INDEX

Sl. No.	CONTENTS / PARTICULARS	PAGE NO.
1	Introduction	1
2	Objectives	2
3	Problem Statement	2–3
4	Literature Review	4
4.1	Real EVM Architecture	4
4.2	VVPAT	4
4.3	Arduino Uno	4–5
5	Proposed System	5
5.1	System Overview	5
5.2	Working Principle	5–6
6	System Requirements	6
6.1	Hardware Requirements	6
6.2	Software Requirements	7
7	Hardware Requirements	7
7.1	Arduino UNO Board	7–8
7.2	HC-SR04 Ultrasonic Sensor	8–9
7.3	Resistor (220 Ohm)	9
7.4	Breadboard	10
7.5	Servo Motor (SG90)	11
7.6	Jumper Wires	12
7.7	LED	13

7.8	USB Cable	14
7.9	Power Supply (9V Battery with Connector)	15–16
8	Circuit Design	17
8.1	Block Diagram	17
8.2	Circuit Diagram	18
8.3	Pin Connection Plan	18–19
8.4	Circuit Description	19
9	Software Design	19
9.1	Arduino Program Logic	20–23
10	Working Explanation	23–24
11	Features	24–25
12	Advantages	25
13	Applications	25
14	Conclusion	26
15	Future Scope	27–33
15.1	Voter Identification System	27
15.2	VVPAT Printing System	27–28
15.3	Secure Data Storage	28
15.4	Touch-Screen Interface	29
15.5	Multi-Booth Networking	29–30
15.6	Mobile Application Monitoring	30
15.7	Battery Backup and Power Management	30–31
15.8	Tamper Detection	31
15.9	Accessibility Features	31–32
15.10	Artificial Intelligence Integration	32
15.11	Blockchain-Based Audit Trail	32–33

15.12	Advanced Microcontroller Upgrade	33
16	References	34–35
17	Project Links	35
18	Project Department Details	38

SKMP PROJECT MANAGEMENT
SKMP PROJECT MANAGEMENT

1. Introduction

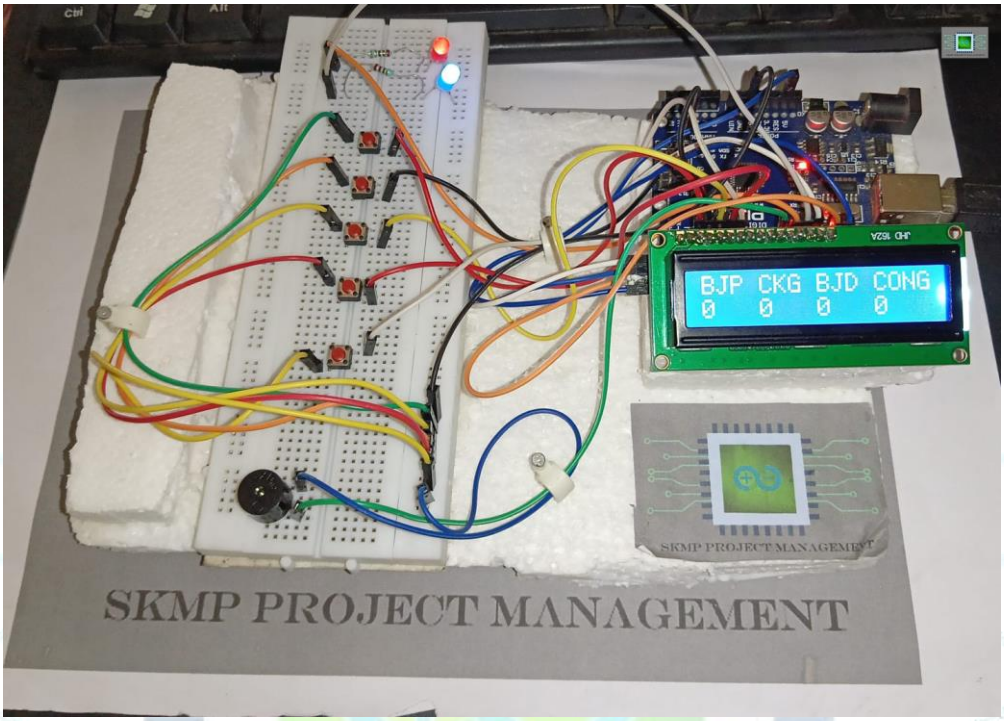


Fig-01

An Electronic Voting Machine is an electronic device used to record votes electronically instead of using paper ballots. In India, the Election Commission of India (ECI) uses EVMs along with VVPAT to allow voters to verify that their vote was recorded as intended.

A conventional paper-ballot system requires printing, distribution, manual marking, secure storage, and manual counting. An EVM reduces invalid votes caused by unclear markings, speeds up counting, and stores candidate-wise vote totals electronically. However, electronic voting systems must also address security, secrecy, reliability, audit ability, accessibility, and public trust.

SKMP PROJECT MANAGEMENT

This project uses Arduino Uno because it is inexpensive, widely available, easy to program, and suitable for demonstrating digital input/output, memory storage, display interfacing, and embedded-system control.

2. Objectives

The main objectives of this project are:

- To design a working prototype EVM using Arduino Uno.
- To allow an administrator or polling officer to enable one vote at a time.
- To allow a voter to select one candidate using push buttons.
- To prevent multiple votes from being cast until the officer enables the next voter.
- To store vote counts in EEPROM so data remains available after a power interruption.
- To display voting status, candidate selection feedback, and final results on an LCD.
- To provide audio and visual confirmation after a vote is recorded.
- To simulate a voter-verifiable paper trail using an LED indicator, buzzer, serial monitor, or optional mini printer.
- To study the limitations of Arduino-based voting systems and the importance of secure election hardware.

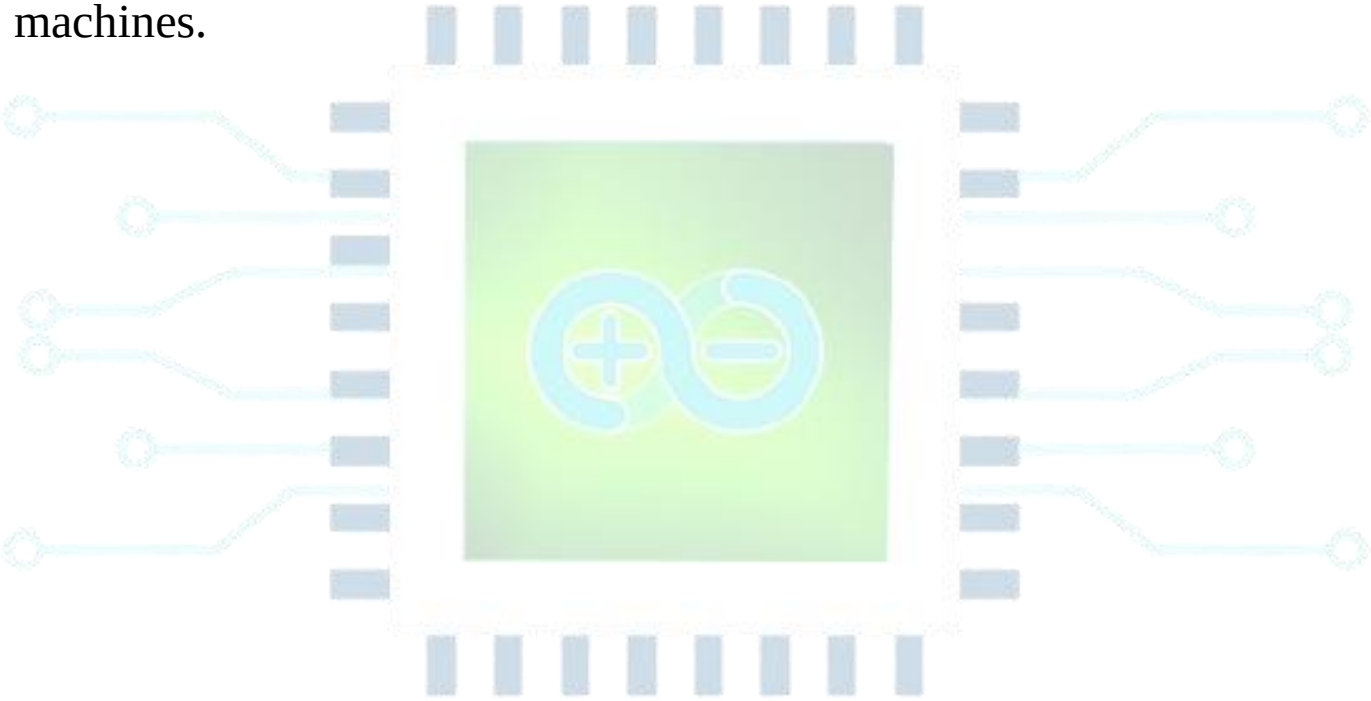
SKMP PROJECT MANAGEMENT

3. Problem Statement

Manual voting systems can face issues such as invalid votes, slow counting, human error during counting, and difficulty maintaining accurate records. A basic electronic voting prototype can demonstrate how votes may be recorded, counted, and displayed automatically.

The problem addressed in this project is to create a simple, affordable, and understandable EVM prototype that:

- Accepts only one vote per authorized voting session.
- Records votes reliably in non-volatile memory.
- Prevents accidental double voting.
- Displays live voting status and final results.
- Provides confirmation through LED, buzzer, and display feedback.
- Demonstrates the basic principle behind real-world electronic voting machines.



SKMP PROJECT MANAGEMENT

4. Literature Review

4.1 Real EVM Architecture

According to the Election Commission of India, an EVM consists of a **Control Unit (CU)** and a **Balloting Unit (BU)** connected by a cable. The Control Unit remains with the presiding or polling officer, while the Balloting Unit is placed inside the voting compartment.

The polling officer presses the **Ballot Button** on the Control Unit to enable the Balloting Unit. The voter then presses the blue button beside the chosen candidate. A red LED glows beside the selected candidate, and a beep confirms that the vote has been recorded.

4.2 VVPAT

VVPAT stands for **Voter Verifiable Paper Audit Trail**. After a vote is cast, the VVPAT prints a slip containing the candidate serial number, name, and symbol. The slip is visible for about seven seconds and then drops into a sealed box.

This paper trail provides an additional verification mechanism and supports audit procedures. In the Indian EVM system, electronic vote counts can be statistically correlated with VVPAT slips from randomly selected polling stations.

4.3 Arduino Uno

Arduino Uno is a microcontroller development board based on the ATmega328P. It provides 14 digital input/output pins, 6 analog input pins, 32 KB flash memory, 2 KB SRAM, 1 KB EEPROM, and a 16 MHz clock.store-usa.arduino.

The Arduino Uno is suitable for this prototype because it can read push-button inputs, control LEDs and buzzers, drive an LCD display, and store small vote-count data in EEPROM.

5. Proposed System

5.1 System Overview

The proposed system has two functional sections:

Section	Purpose	Main Components
Control Unit	Enables voting, controls session flow, stores and displays results	Arduino Uno, LCD, buttons, EEPROM
Balloting Unit	Allows voter to choose a candidate	Candidate push buttons, LEDs, buzzer
Verification Unit	Simulates voter verification	LED, serial monitor, optional printer

In this prototype, the Arduino Uno acts as the central controller. The officer first presses the “Start/Enable Vote” button. The system then waits for the voter to press one candidate button. After the vote is recorded, the system locks itself until the officer enables the next vote.

5.2 Working Principle

1. The system starts in **Standby Mode**.
2. The polling officer presses the **Ballot/Enable** button.

3. The LCD displays: **"VOTING ENABLED – PLEASE CAST YOUR VOTE."**
4. The voter presses one candidate button.
5. The corresponding LED glows.
6. The Arduino increments the vote count for that candidate.
7. The updated count is written to EEPROM.
8. The buzzer produces a confirmation beep.
9. The LCD displays a confirmation message.
10. The system returns to **Locked Mode** until the officer enables the next vote.
11. At the end of voting, the officer presses the **Result** button.
12. The LCD displays candidate-wise vote totals.

This follows the basic real-world EVM sequence: officer enables voting, voter selects a candidate, vote is recorded, and confirmation is provided.

6. System Requirements

- Arduino UNO
- LCD Display with I2C Module
- 5 Push Buttons
- 2 LEDs (Voting Confirmation & Winner Indicator)
- 2 Resistors (220Ω)
- Jumper Wires
- Breadboard
- USB Cable

- Arduino UNO
- LCD Display with I2C Module
- 5 Push Buttons
- 2 LEDs (Voting Confirmation & Winner Indicator)
- 2 Resistors (220Ω)
- Jumper Wires

SIMP PROJECT MANAGEMENT

The Arduino Uno operates at 5 V and its digital pins are rated for up to 40 mA per pin, although 20 mA is the recommended operating condition.

6.2 Software Requirements

- Arduino IDE
- C/C++ programming language
- LiquidCrystal_I2C library for I2C LCD
- EEPROM library for permanent storage
- Serial Monitor for testing and optional audit-log output

7. Hardware Requirements

- ❖Arduino.
- ❖Ultrasonic Sensor(Hc-Sr04).
- ❖Lcd Display With I2c
- ❖Resistor 220ohm.
- ❖Led.
- ❖Breadboard.
- ❖Jumper Wire.

●Arduino UNO Board

➢ The Arduino UNO is an open-source microcontroller development board based on the ATmega328P microcontroller. It is widely used in electronics, robotics, automation, and embedded system projects. It allows users to connect various electronic components, such as sensors, LEDs, motors, displays, and other devices, to develop interactive electronic applications.

- The Arduino UNO board has 14 digital input/output pins, of which 6 can be used as PWM outputs, 6 analog input pins, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button.
- It can be programmed using the Arduino IDE with a simple programming language based on C and C++. The board receives instructions from the program and controls connected devices according to the given commands.

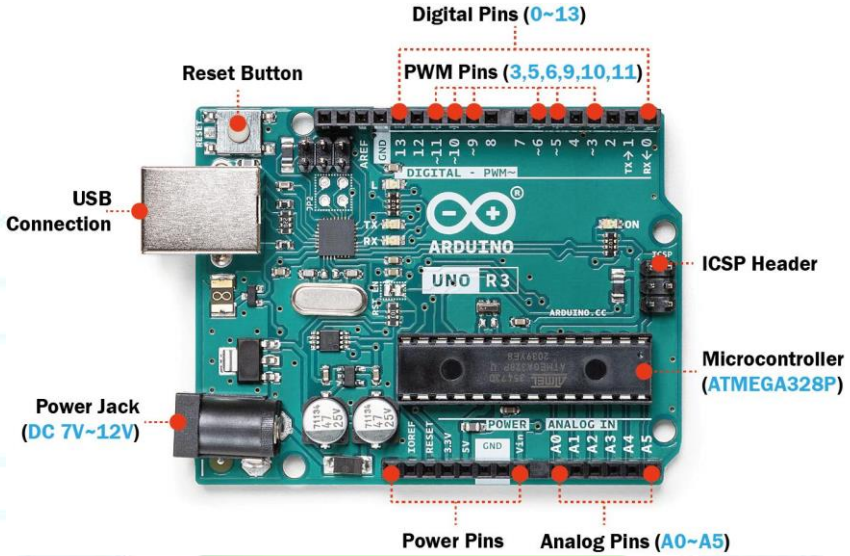


Fig -02

• *HC-SR04 Ultrasonic Sensor*

- The HC-SR04 Ultrasonic Sensor is an electronic distance-measuring sensor that uses ultrasonic sound waves to detect objects and measure the distance between the sensor and an object. It operates by transmitting high-frequency ultrasonic waves (approximately 40 kHz) and receiving the reflected waves (echo) from an object.
- The sensor consists of two main ultrasonic transducers: one acts as a transmitter (Trig), which sends ultrasonic waves, and the other acts as a receiver (Echo), which detects the reflected waves. The distance is

SKVP PROJECT MANAGEMENT

calculated by measuring the time taken for the ultrasonic waves to travel to the object and return to the sensor.

- The HC-SR04 sensor is commonly used with microcontrollers such as Arduino UNO for distance measurement, obstacle detection, robotics, and automatic control systems.



Fig -03

. Resistor 220ohm

- A 220-ohm (220Ω) Resistor is a passive electronic component used to limit or control the flow of electric current in an electronic circuit. It provides a specific resistance of 220 ohms and helps protect sensitive electronic components from excessive current that may cause damage.
- In Arduino-based projects, a 220Ω resistor is commonly used to limit current flowing through LEDs and other electronic components. It helps maintain safe operating conditions and improves the reliability of the electronic circuit.
- The resistor works according to Ohm's Law, which describes the relationship between voltage, current, and resistance.

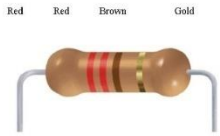


FIG-4

• Breadboard

- A Breadboard is a reusable electronic prototyping board used to build, test, and develop electronic circuits without soldering. It provides a convenient platform for connecting different electronic components using jumper wires.
- A breadboard contains several small holes arranged in rows and columns, which are internally connected through conductive metal clips. These connections allow electronic components such as resistors, LEDs, sensors, and microcontrollers to be connected easily.
- In Arduino-based projects, a breadboard is commonly used to connect components such as the Arduino UNO, HC-SR04 ultrasonic sensor, SG90 servo motor, and resistors. It helps users assemble and modify circuits quickly during the testing and development process.

SKMP PROJECT MANAGEMENT

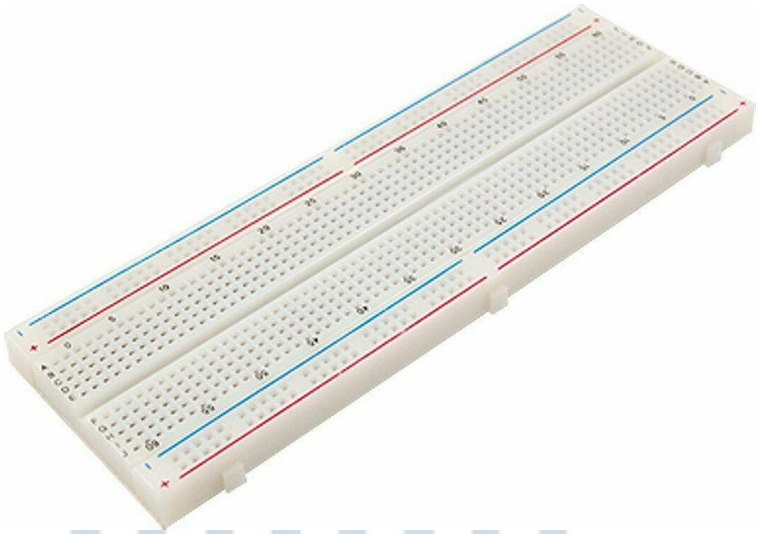


Fig-5

• Servo Motor (SG90)

- The SG90 Servo Motor is a small, lightweight, and compact rotary actuator used to control angular position with high precision. It is commonly used in robotics, automation, and Arduino-based electronic projects. The SG90 is a micro servo motor that can rotate approximately 180 degrees, allowing accurate control of its shaft position.
- The SG90 servo motor consists of a DC motor, a gear mechanism, a position feedback system, and an electronic control circuit. It operates based on Pulse Width Modulation (PWM) signals received from a microcontroller such as the Arduino UNO. By varying the width of the control pulse, the motor shaft can be positioned at a desired angle.
- The SG90 servo motor is widely used in projects requiring controlled movement, such as robotic arms, automatic doors, obstacle-avoiding robots, and digital measurement systems.

SKOLR PROJECT MANAGEMENT



Fig -06

• *Jumper Wires*

- Jumper Wires are electrical connecting wires used to establish temporary connections between different electronic components, sensors, and microcontroller boards. They are commonly used in electronics prototyping, Arduino projects, and embedded system development.
- Jumper wires are used to connect components such as Arduino UNO, ultrasonic sensors, servo motors, breadboards, LEDs, and other electronic modules without requiring soldering. They provide an easy and flexible method of transferring electrical power and signals between different components.
- Jumper wires are generally available in three types: Male-to-Male, Male-to-Female, and Female-to-Female. They are commonly insulated with different colours to help identify power supply, ground, and signal connections.

SKMP PROJECT MANAGEMENT



Fig -07

• *LED*

- An LED (Light Emitting Diode) is a semiconductor electronic component that produces light when an electric current flows through it in the forward direction. It is a type of diode that converts electrical energy into light energy through a process known as electroluminescence.
- LEDs are widely used in electronic circuits as visual indicators to display the operating status of a system. They are available in different colours, such as red, green, blue, yellow, and white.
- In Arduino-based projects, LEDs are commonly connected to digital output pins to indicate different operating conditions. The Arduino UNO controls the LED by switching it ON or OFF according to the programmed instructions.

SIMPLE PROJECT MANAGEMENT

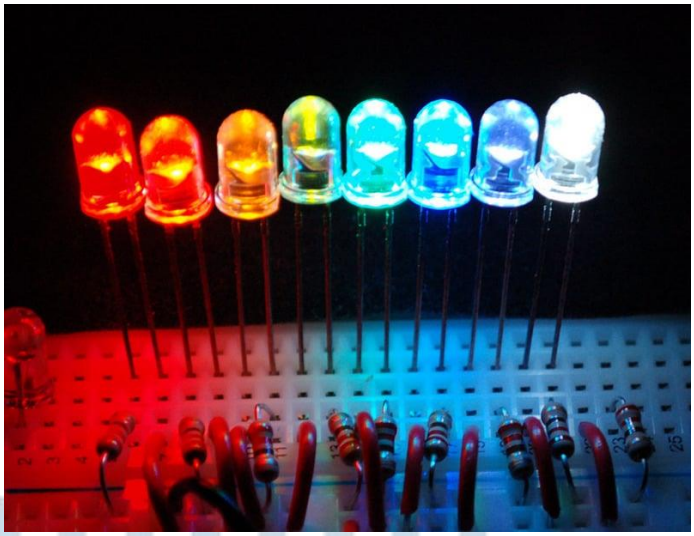


Fig-08

• *USB CABLE*

- A USB (Universal Serial Bus) Cable is an electrical communication cable used to establish a connection between electronic devices for data transmission and power supply. In Arduino-based projects, a USB cable is commonly used to connect the Arduino UNO board to a computer or laptop for programming, uploading code, and supplying electrical power.
- The USB cable allows communication between the computer and the Arduino UNO microcontroller. It transfers program instructions from the Arduino IDE to the board and also supports serial communication for monitoring sensor readings and debugging the project.
- The Arduino UNO R3 commonly uses a USB Type-B cable for connecting to a computer. One end of the cable connects to the Arduino board, while the other end connects to the computer's USB port.



Fig -09

• *Power Supply (9V Battery with connector)*

- A 9V Battery with Connector is a portable DC power source used to provide electrical energy to electronic circuits and microcontroller-based projects. It consists of a 9V battery and a battery connector that allows easy connection to the power input terminals of an electronic circuit or Arduino UNO board.
- In Arduino-based projects, the 9V battery is commonly connected to the Arduino UNO through its DC barrel jack using a 9V battery snap connector with a suitable DC plug. It supplies electrical power to the microcontroller and connected electronic components when a computer or external power adapter is not available.

- The Arduino UNO has an onboard voltage regulator that converts the input voltage into a suitable operating voltage for the board. The 9V battery provides portability and makes the project suitable for standalone operation.

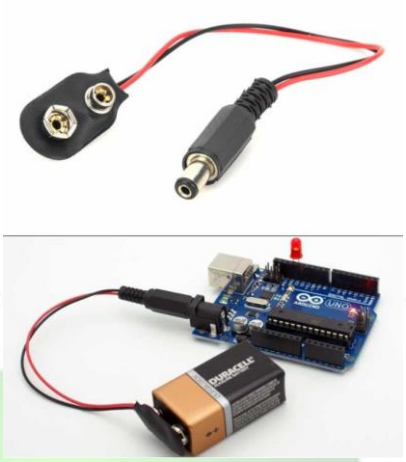


Fig -`10

SKMP PROJECT MANAGEMENT

8. Circuit Design

8.1 Block Diagram

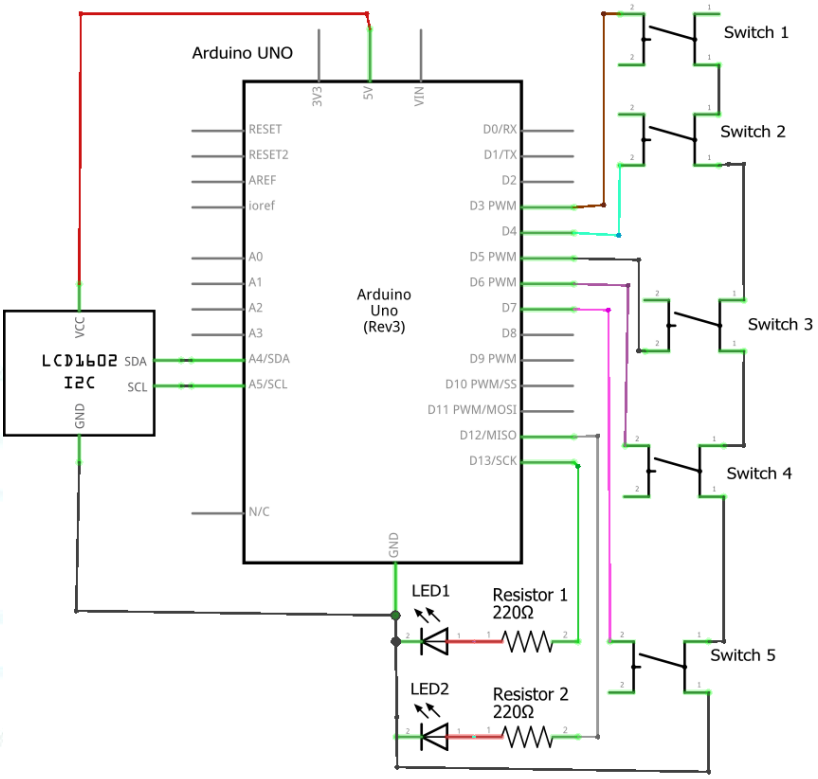


Fig-11

SKMP PROJECT MANAGEMENT

8.2 Circuit Diagram

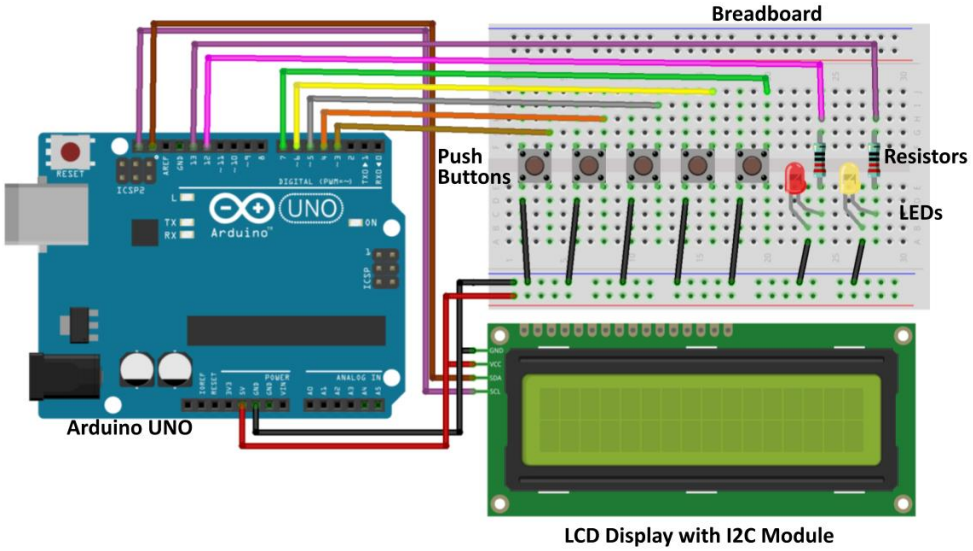


Fig-12

8.3 Pin Connection Plan

Arduino Pin	Connected Component	Function
D2	Candidate 1 button	Select Candidate 1
D3	Candidate 2 button	Select Candidate 2
D4	Candidate 3 button	Select Candidate 3
D5	Candidate 4 button	Select Candidate 4
D6	Officer Enable button	Enable one voting session
D7	Result button	Display final result
D8	Reset Counts button	Clear stored votes
D9	Candidate 1 LED	Candidate 1 selection indicator
D10	Candidate 2 LED	Candidate 2 selection indicator

SKMP PROJECT MANAGEMENT

Arduino Pin	Connected Component	Function
D11	Candidate 3 LED	Candidate 3 selection indicator
D12	Candidate 4 LED	Candidate 4 selection indicator
D13	Status LED	System ready/busy indicator
A4 (SDA)	LCD SDA	I2C data line
A5 (SCL)	LCD SCL	I2C clock line
D5/D6 or PWM pin	Buzzer	Confirmation sound

Push buttons should use either internal pull-up resistors or external pull-down resistors to avoid floating inputs. LEDs must be connected through current-limiting resistors, typically 220 ohm to 330 ohm.

8.4 Circuit Description

The Arduino Uno continuously monitors the officer and candidate buttons. In the initial state, candidate buttons are disabled. When the officer presses the Enable button, the Arduino changes the system state to “Voting Enabled.”

When a candidate button is pressed, the Arduino checks whether voting is enabled. If enabled, it increments that candidate’s vote count, saves the updated count to EEPROM, activates the corresponding LED, sounds the buzzer, and displays a confirmation message. The system then returns to the locked state.

SKMP PROJECT MANAGEMENT

The Result button reads all stored vote counts from EEPROM and displays them on the LCD. The Reset button should be protected or require a long press, because accidental resetting can erase voting data.

09. Software Design

9.1 Arduino Program Logic

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

#define v1 3
#define v2 4
#define v3 5
#define v4 6
#define v5 7
int vote1=0;
int vote2=0;
int vote3=0;
int vote4=0;

void setup()
{

 pinMode(v1, INPUT);
 pinMode(v2,INPUT);
 pinMode(v3,INPUT);
 pinMode(v4,INPUT);
 pinMode(v5,INPUT);
 pinMode(13,OUTPUT);
 pinMode(12,OUTPUT);

  lcd.begin();
  lcd.backlight();
  lcd.clear();
  lcd.setCursor(3,0);
  lcd.print("Roboarmy    ");
  delay(2000);
  lcd.setCursor(0,0);
  lcd.print("Digital Voting ");
  lcd.setCursor(0,1);
  lcd.print("Machine");

  delay(2600);
  digitalWrite(v1, HIGH);
  digitalWrite(v2, HIGH);
  digitalWrite(v3, HIGH);
  digitalWrite(v4, HIGH);
  digitalWrite(v5, HIGH);
  lcd.clear();
  lcd.setCursor(1,0);
  lcd.print("A");
  lcd.setCursor(5,0);
  lcd.print("B");
  lcd.setCursor(9,0);
  lcd.print("C");
```

```
  lcd.setCursor(13,0);
  lcd.print("D");
}

void loop()
{
  lcd.setCursor(1,0);
  lcd.print("A");
  lcd.setCursor(1,1);
  lcd.print(vote1);
  lcd.setCursor(5,0);
  lcd.print("B");
  lcd.setCursor(5,1);
  lcd.print(vote2);
  lcd.setCursor(9,0);
  lcd.print("C");
  lcd.setCursor(9,1);
  lcd.print(vote3);
  lcd.setCursor(13,0);
  lcd.print("D");
  lcd.setCursor(13,1);
  lcd.print(vote4);

if(digitalRead(v1)==0)
{
  digitalWrite(12,HIGH);
  vote1++;
  while(digitalRead(v1)==0);
  digitalWrite(12,LOW);
  delay(100);
}

if(digitalRead(v2)==0)
{
  vote2++;
  digitalWrite(12,HIGH);
  while(digitalRead(v2)==0);
  digitalWrite(12,LOW);
  delay(100);
}

if(digitalRead(v3)==0)
{
  digitalWrite(12,HIGH);
  vote3++;
  while(digitalRead(v3)==0);
  digitalWrite(12,LOW);
  delay(100);
}

if(digitalRead(v4)==0)
{
digitalWrite(12,HIGH);
vote4++;
while(digitalRead(v4)==0);
digitalWrite(12,LOW);
delay(100 );
}
```

```
if(digitalRead(v5)==0)
{
digitalWrite(13,HIGH);
int vote=vote1+vote2+vote3+vote4;
if(vote)
{
if((vote1 > vote2 && vote1 > vote3 && vote1 > vote4))
{
lcd.clear();
lcd.print("A Wins");
delay(3000);
lcd.clear();
}
else if((vote2 > vote1 && vote2 > vote3 && vote2 > vote4))
{
lcd.clear();
lcd.print("B Wins");
delay(3000);
lcd.clear();
}
else if((vote3 > vote1 && vote3 > vote2 && vote3 > vote4))
{
lcd.clear();
lcd.print("C Wins");
delay(3000);
lcd.clear();
}
else if(vote4 > vote1 && vote4 > vote2 && vote4 > vote3)
{
lcd.setCursor(0,0);
lcd.clear();
lcd.print("D Wins");
delay(3000);
lcd.clear();
}

else
{
lcd.clear();
lcd.print(" Vote Tied ");
lcd.setCursor(0,1);
lcd.print("   ");
delay(3000);
lcd.clear();
}

}
else
{
  lcd.clear();
  lcd.print("No Vote Polled");
  delay(2000);
  lcd.clear();
}
vote1=0;
vote2=0;
vote3=0;
vote4=0;
vote=0;
```

```
lcd.clear();
digitalWrite(12,HIGH);
digitalWrite(13,LOW);
}
}
```

10. Working Explanation

When the system is powered on, the Arduino reads previous vote counts from EEPROM. This ensures that the vote data is not lost if power is disconnected. The Arduino Uno provides 1 KB of EEPROM, which is sufficient for a small prototype storing several candidate counts.store-usa.arduino+1

The polling officer presses the Enable button to begin one voting session. The LCD informs the voter that voting is active. When the voter presses a candidate button, the Arduino increments the corresponding counter and writes the updated value to EEPROM.

The LED provides immediate visual feedback, while the buzzer confirms successful recording. After one vote is recorded, the system becomes inactive until the officer enables the next voter. This prevents one voter from casting multiple votes in the prototype.

SKMP PROJECT MANAGEMENT

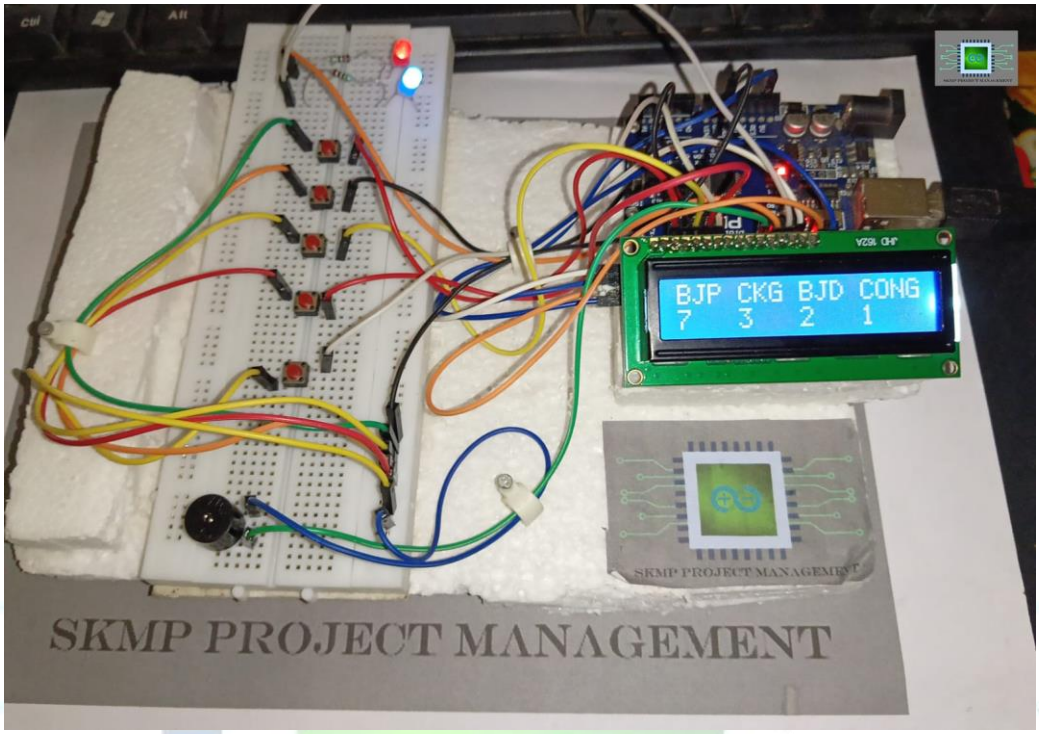


Fig-13

11. Features

- Arduino Uno-based low-cost EVM prototype.
- Supports four candidates; expandable to more candidates.
- Officer-controlled voting session.
- One-vote-per-session control.
- EEPROM-based non-volatile vote storage.
- LCD status and result display.
- LED indication for selected candidate.
- Buzzer confirmation after vote recording.
- Result-display facility.
- Vote-reset facility for testing.
- Optional VVPAT simulation using serial output or a mini printer.
- Simple, portable, and educational design.

SKMP PROJECT MANAGEMENT

12. Advantages

- **Low cost:** Arduino Uno and basic electronic components are affordable.
- **Easy to build:** Suitable for school, college, and hobby projects.
- **Fast demonstration:** Votes are recorded and counted automatically.
- **Non-volatile storage:** EEPROM retains vote counts after power failure.
- **User-friendly:** LCD messages and LED indicators make operation simple.
- **Educational value:** Demonstrates embedded systems, digital electronics, and programming.
- **Expandable:** More candidates, password protection, SD-card logging, and printer output can be added.

13. Applications

- Engineering and diploma project demonstration.
- Embedded-systems and microcontroller learning.
- Science exhibition and technical competition.
- Prototype for understanding voting-machine workflow.
- Demonstration of digital electronics and human-machine interaction.
- Educational model for explaining EVM and VVPAT concepts.
- Base platform for research on secure voting interfaces and audit trails.

Last Chapter

Conclusion

The Arduino Uno-based Electronic Voting Machine prototype successfully demonstrates the fundamental working principle of an EVM: officer-controlled voting, candidate selection, vote recording, confirmation feedback, non-volatile storage, and result display. The project provides a practical and affordable platform for learning embedded systems, digital electronics, programming, and human-machine interaction.

While the prototype is useful for education and demonstration, it is not suitable for actual elections. Real EVM systems require certified hardware, secure firmware, tamper protection, strict operational procedures, VVPAT-based verification, and independent auditing.

SKMP PROJECT MANAGEMENT

Future scope

The Arduino Uno-based EVM prototype can be expanded from a basic voting demonstration into a more advanced, secure, and realistic electronic voting system. Future improvements can focus on voter identification, data security, auditability, accessibility, and remote result monitoring.

1. Voter Identification System

A future version can include voter verification before allowing a vote. Possible methods include:

- **RFID card reader:** Each voter receives an RFID card; the system checks whether the card has already voted.
- **Fingerprint sensor:** Biometric verification can prevent duplicate voting in a demonstration environment.
- **QR-code scanner:** A voter ID or QR code can be scanned before voting.
- **Keypad-based voter ID:** A voter enters a unique ID before the ballot unit is enabled.
- **Face-recognition module:** A camera module can be used for experimental identity verification.

SKMP PROJECT MANAGEMENT

After verification, the system can mark the voter as “voted” so the same person cannot vote again.

2. VVPAT Printing System

A thermal printer can be connected to the Arduino to simulate a **Voter Verifiable Paper Audit Trail (VVPAT)**. After a voter selects a candidate, the printer can produce a small slip showing:

- Candidate serial number
- Candidate name
- Party symbol
- Date and time
- Booth ID
- A confirmation message

The printed slip can be displayed for a few seconds and then stored in a sealed transparent box. This improves transparency and gives voters a way to verify that their selection was recorded correctly. Real Indian EVMs use VVPAT for this purpose.

3. Secure Data Storage

Instead of relying only on Arduino EEPROM, the project can use external memory modules:

- **SD card module:** Stores encrypted vote logs, timestamps, booth information, and audit records.
- **External EEPROM:** Provides additional storage for vote counts and system logs.
- **Flash memory module:** Stores firmware configuration and candidate details.
- **Encrypted data storage:** Vote data can be encrypted before saving to prevent easy tampering.

A more advanced design can store a hash or digital signature of each vote record, making unauthorised modification easier to detect.

4. Touch-Screen Interface

A 16x2 LCD can be replaced with a TFT touch-screen display. This would allow:

- Candidate names and symbols to be shown clearly.
- Multi-language voting instructions.
- On-screen confirmation before final vote submission.
- Better accessibility for elderly and visually impaired voters.
- A more realistic and user-friendly voting experience.

The touch screen can also display candidate photographs, party symbols, and voting instructions.

5. Multi-Booth Networking

Multiple EVM prototypes can be connected to a central monitoring system using:

- **ESP32 Wi-Fi module**
- **Ethernet module**
- **RS-485 communication**
- **LoRa module for long-range communication**
- **GSM/GPRS module for remote locations**

SKMP PROJECT MANAGEMENT

The central server can collect booth-wise vote counts, monitor machine status, detect faults, and display live results. However, internet connectivity should not be used for direct vote transmission in a real election system unless it is designed with strong end-to-end security and audit controls.

6. Mobile Application Monitoring

A mobile application can be developed for authorised election officials. It can provide:

- Live voting status
- Booth-wise vote count
- Machine battery and connectivity status
- Error alerts
- Candidate-wise result summary
- Secure login for administrators
- Exportable result reports

The mobile app should only display authorised information and must not compromise voter secrecy.

7. Battery Backup and Power Management

A real voting machine must continue working during power interruptions. The project can be improved using:

- Rechargeable lithium-ion battery
- Battery-management system
- Voltage regulator
- Power-failure detection circuit
- Automatic switching between mains power and battery
- Low-power sleep mode when the machine is idle

This will make the prototype more reliable and suitable for outdoor demonstrations.

8. Tamper Detection

SKILL PROJECT MANAGEMENT

Future versions can include security features such as:

- Tamper-proof enclosure
- Seal-detection switch
- Intrusion alarm
- Buzzer alert when the cover is opened
- EEPROM write protection
- Password-protected administrator mode
- Firmware-signature verification
- Event logging for unauthorised access attempts

These features help demonstrate how physical and software security can protect an electronic voting system.

9. Accessibility Features

The system can be made more inclusive by adding:

- Braille markings near candidate buttons
- Audio instructions through a speaker
- Large-button interface for voters with limited mobility
- Adjustable display brightness
- Multi-language support
- Headphone jack for private audio guidance
- Wheelchair-accessible enclosure design

Accessibility is important because every eligible voter should be able to vote independently and confidently.

10. Artificial Intelligence Integration

SKMP PROJECT MANAGEMENT

Although AI is not necessary for basic vote counting, it can support election administration:

- Detect abnormal voting patterns.
- Identify device faults using sensor data.
- Analyse turnout trends.
- Detect duplicate or suspicious access attempts.
- Provide voice-based assistance for voters.
- Support predictive maintenance of voting machines.

Any AI-based feature must protect voter privacy and should not link voter identity with voting choice.

11. Blockchain-Based Audit Trail

A blockchain-inspired system can be used to create a tamper-evident audit trail. Instead of storing votes directly on a public blockchain, the system can store encrypted vote records or hashes on a private, permissioned blockchain.

Possible benefits include:

- Tamper detection
- Transparent audit history
- Decentralised verification
- Improved public trust
- Secure record keeping

SKOLR PROJECT MANAGEMENT

However, blockchain does not automatically solve voter secrecy, device security, or identity verification. It should be used only as one part of a complete secure election architecture.

12. Advanced Microcontroller Upgrade

Arduino Uno is suitable for learning, but future versions can use more powerful controllers:

Controller	Benefits
ESP32	Wi-Fi, Bluetooth, more memory, faster processing
Raspberry Pi Pico	Low cost, dual-core processor, more GPIO options
Raspberry Pi	Full Linux system, touchscreen support, camera and printer support
STM32	Industrial reliability, better performance, advanced peripherals

For a portable EVM prototype, ESP32 or Raspberry Pi Pico would be better choices because they offer more memory, processing power, and communication options.

SKMP PROJECT MANAGEMENT

REFERENCES

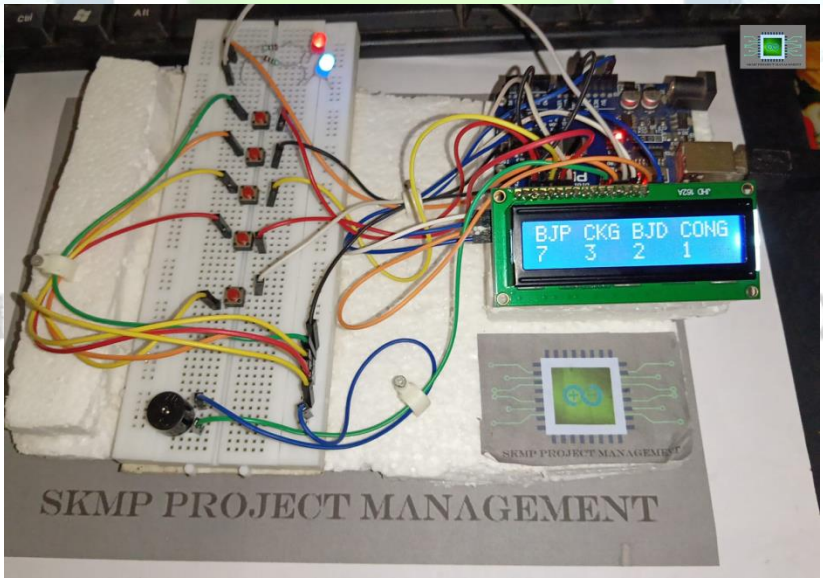
- Election Commission of India. *EVM/VVPAT*. Available at: https://www.eci.gov.in/evm-vvpat
- Election Commission of India. *Electronic Voting Machine — General Q/A*. Available at: https://www.eci.gov.in/evm-faq
- Election Commission of India. *Manual on Electronic Voting Machine and VVPAT*, Edition 8, August 2023. Available at: https://www.eci.gov.in/eci-backend/public/uploads/monthly_2023_08/EVMManualAugust2023_pdf.1f8976b609ce6fefe9b0fe69d3f848ff
- Election Commission of India. *How to Vote*. Available at: https://www.eci.gov.in/faq/en/how-to-vote/
- Election Commission of India. *History of EVM*. Available at: https://old.eci.gov.in/voter/history-of-evm/
- Arduino. *Arduino Uno Rev3 — Tech Specs*. Available at: https://store-usa.arduino.cc/collections/lab-tools/products/arduino-uno-rev3
- Arduino Documentation. *Overview of the Arduino UNO Components*. Available at: https://docs.arduino.cc/tutorials/uno-rev3/intro-to-board
- Election Commission of India. *Handbook for Candidates*. Available at: https://www.eci.gov.in/eci-backend/public/api/download?url=LMAhAK6sOPBp/NFF0iRfXbEB1EVSLT41NNRLq9bWZlY2F0aW9uIEVsZWN0cmlvbiBDb21taXNzaW9uIG9mIEluZGlh
- Chief Electoral Officer, Meghalaya. *Electronic Voting Machine (EVM): Brochure for Candidates and Political Parties*. Available at: https://ceomeghalaya.nic.in/pdfs/evm_brochure_for_candidates_political%20parties.pdf
- Unique Identification Authority of India. *Electronic Voting Machine (EVM) and Voter Verifiable Paper Audit Trail (VVPAT)*. Available at: https://uaoa.gov.in/sites/default/files/2025-08/Election%20Electronic%20Voting%20Machine%20and%20Voter%20Verifiable%20Paper%20Audit%20Trail.pdf

PROJECTLINK

Website lnk:-https://skmp-project-management.vercel.app/

Google Drive Link:-
https://drive.google.com/drive/folders/1EJDMBkBUKcmnmmdxj8793uWqp6GdjD0I?usp=sharing

Video Link:- https://youtu.be/eOheJ0iZPyQ?si=sJWPdmpYFRsxeMbg





SKMP PROJECT MANAGEMENT

SKMP PROJECT MANAGEMENT

==========

SKMP PROJECT MANAGEMENT

DEPARTMENT OF IOT