ELECTRONIC VOTING MACHINE USING 8051 MICROCONTROLLER

BECE301L MicroProcessor and MicroController

By

Maadesh A 22BEC1348

Hariharan M 22BEC1126

Udit Krishna B 22BEC1080

Danus D 22BEC1160

Pradesh Kumar M 22BEC1478

Senthil Kumar M 21BEC1228

Submitted to

Dr. SOURABH PAUL

SCHOOL OF ELECTRONICS ENGINEERING

VELLORE INSTITUTE OF TECHNOLOGY

CHENNAI - 600127

April 2024

Certificate

This is to certify that the Project work titled "**Electronic Voting Machine using 8051 Microcontroller**" is being submitted by **Maadesh A - 22BEC1348, Hariharan M - 22BEC1126, Udit Krishna B - 22BEC1080, Danus D - 22BEC1160, Pradesh Kumar M - 22BEC1478, Senthil Kumar M - 21BEC1228** for the course B**ECE301L MicroProcessor and MicroController** is a record of bonafide work done under my guidance. The contents of this project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University.

Dr. SOURABH PAUL

ABSTRACT

The project states the development of an Electronic Voting Machine (EVM) using the 8051 microcontroller offering a modern solution to the challenges faced by traditional paper-based voting systems. Traditional paper-based voting systems often suffer from challenges such as time-consuming manual counting, errors in tallying, and susceptibility to fraud. The implementation of an EVM based on the 8051 microcontroller addresses these issues by automating the entire voting process.In this project, the 8051 microcontroller serves as the central processing unit, managing the input from voters, storing the votes securely, and generating accurate results. The system consists of a user-friendly interface for voters to cast their ballots electronically, ensuring simplicity and accessibility for all participants. Key features of the proposed EVM include real-time vote counting, encryption mechanisms to maintain the integrity and confidentiality of votes, and robust security measures to prevent unauthorized access and tampering. Additionally, the use of electronic storage eliminates the need for physical ballot papers, reducing costs and environmental impact.

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide**,** **Dr. Sourabh Paul,** School of Electronics Engineering, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to **Dr. Susan Elias,** Dean of School of Electronics Engineering, VIT Chennai, for extending the facilities of the School towards our project and for his unstinting support.

We express our thanks to our Head of the Department **Dr. K. Mohanaprasad** for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

Maadesh A **Hariharan M** **Udit Krishna B**

Danus D **Pradesh Kumar M** **Senthil Kumar M**

Table of contents

Chapter No	Page No
Abstract	3
Acknowledgement	4
1.Introduction	6
1.1 Purpose	6
1.2 Scope	6
2.Implementation	7-32
2.1 Components Required	7
2.2 Block Diagram	7
2.3 Algorithm and Process Invoved	8
2.4 Code	9-32
3.Results	33-38
3.1 Output	33-35
3.2 Future Scope	36
3.3 Conclusion and Discussion	37
4.Refernces	38
5.Biodata	39

1.Introduction

1.1 Purpose

The primary purpose of this project is to design and develop an EVM based on the 8051 microcontroller, with the goal of modernizing the voting process and overcoming the limitations of traditional paper-based systems. By leveraging microcontroller technology, the project aims to create a voting system that is efficient, secure, and user-friendly. Additionally, the project seeks to explore the feasibility of implementing such a system in real-world electoral contexts, with a focus on improving transparency, accuracy, and accessibility in the voting process.

1.2 Scope

The scope of the project encompasses the design, development, and testing of the EVM using the 8051 microcontroller platform. Key aspects to be addressed include the hardware and software components of the EVM, user interface design, data encryption and security measures, as well as integration with existing electoral infrastructure. Furthermore, the project will evaluate the performance of the EVM in simulated voting scenarios and consider potential challenges and opportunities for deployment in diverse electoral settings. Overall, the scope of the project extends to both technical implementation and practical considerations related to the adoption and deployment of EVM technology in electoral systems.

2.Implementation

2.1 Components Required

- **8051 Microcontroller Kit**
- **Power source**
- **Keypad**
- **LCD**
- **Wires**
- **TTL connector**

2.2 Block Diagram

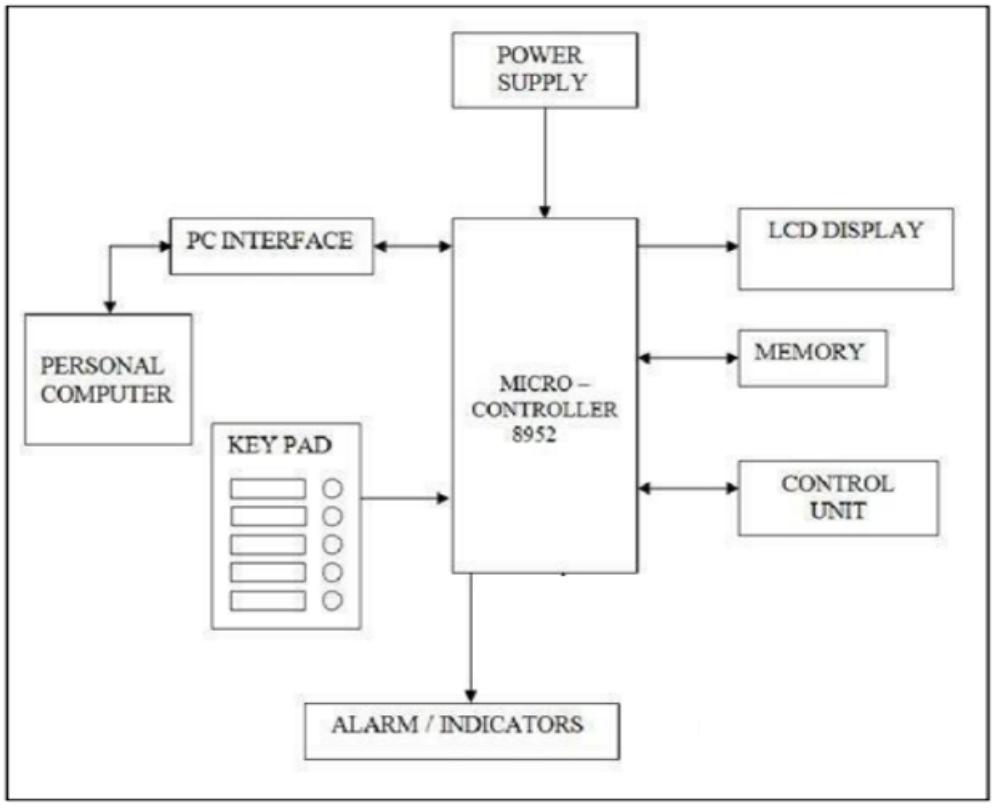

Fig2.2.1: Block Diagram of the EVM

2.3 Algorithm and Process Involved

To ensure efficient management of unique voter IDs and their corresponding votes within the specified range of 1000 to 1500, a mapping algorithm is employed. This algorithm facilitates the setting of specific bits to indicate whether a particular ID has cast a vote or not. Here's a detailed explanation of the mapping algorithm:

1. **Initialization**:
    - The database is stored from address 40H to 7E.4.
    - Registers from address 40H to 7E.4 are initialized to zero. This ensures a clean slate for recording votes.
2. **Voter ID Range Constraint**:
    - The range of unique IDs is constrained from 1000 to 1500, ensuring that only valid IDs are considered for voting.
3. **Bit Setting Mechanism**:
    - When a person casts a vote, a specific bit corresponding to their ID is set to one.
    - If an individual attempts to vote again, detection is straightforward because the corresponding bit is already set hence displays “INVALID ID”.
4. **Handling Non-Bit Addressable Range**:
    - Address range 40H to 7EH is not bit-addressable. Thus, to set a particular bit to 1, the following steps are undertaken:
        - Copy the entire byte containing the relevant ID from its original address to a bit-addressable region.
        - Set the specific bit corresponding to the voter ID to 1 within the copied byte.
        - Copy the modified byte back to its original address in the database.
5. **Efficiency Considerations**:
    - Despite the need for additional steps due to the non-bit addressable nature of the database range, the algorithm ensures efficient utilization of memory and streamlined access to voting records.
    - By restricting the range of unique IDs and employing a systematic bit-setting mechanism, the algorithm minimizes the likelihood of errors and ensures the integrity of the voting process.

6. **Implementation Considerations**:
    - The algorithm is implemented within the firmware of the EVM using the 8051 microcontroller platform.
    - Careful attention is paid to optimizing memory usage and execution speed to maintain the responsiveness and reliability of the voting system.

2.4 Code

```
ORG 0000H
RS EQU P1.0
RW EQU P1.1
E EQU P1.2
ROW EQU P0
COL EQU P3

STARTELECTION:ACALL INITIALIZEDB
MOV R0,#36H
L7:MOV @R0,#00H
INC R0
CJNE R0,#3EH,L7

START:ACALL INITIALIZELCD
MOV DPTR,#ENTERID
MOV B,#8
ACALL DISPLAY
MOV P2,#0C0H
ACALL COMMAND
ACALL GETID
MOV R3,30H
MOV R2,31H
MOV R1,32H
MOV R0,33H
```

```
ACALL GETADDRESS
MOV 34H,A
MOV 35H,B
ACALL CHECKVALIDITY
MOV A,20H
CJNE A,#0BH,VALID1
INVALID1:MOV P2,#01H
ACALL COMMAND
MOV DPTR,#INVALIDID
MOV B,#11
ACALL DISPLAY
ACALL GETNUM
ACALL DELAY1S
SJMP START
VALID1:
MOV P2,#01H
ACALL COMMAND
MOV DPTR,#CASTVOTE
MOV B,#14
ACALL DISPLAY
ACALL GETNUM
MOV P2,#0C0H
ACALL COMMAND
MOV P2,A
ACALL DATA1
ACALL DELAY1S
ACALL INCVOTES
MOV A,34H
MOV B,35H
ACALL SETADDRESS
MOV P2,#01H
ACALL COMMAND
MOV DPTR,#VOTECOUNTED1
MOV B,#12
```

```
ACALL DISPLAY
MOV P2,#0C4H
ACALL COMMAND
MOV DPTR,#VOTECOUNTED2
MOV B,#7
ACALL DISPLAY
ACALL GETNUM
ACALL DELAY1S
SJMP START

FUNCTIONS: LJMP FINISH

GETCOLUMN:   ;;STORES THE COL NO. OF PRESSED KEY IN R5
MOV R5,#00H
NEXT:RRC A
JNC FOUNDCOLUMN
INC R5
SJMP NEXT
FOUNDCOLUMN:RET

GETROW:  ;STORES THE ROW NO. OF PRESSED KEY IN B
MOV B,#00H
MOV ROW,#00001110B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW


INC B
MOV ROW,#00001101B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW
```

```
INC B
MOV ROW,#00001011B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW

INC B
MOV ROW,#00000111B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW

FOUNDROW:RET


DELAY20MS:
MOV R1,#36
HERE1:MOV R0,#255
HERE:DJNZ R0,HERE
DJNZ R1,HERE1
RET

DELAY1S:
MOV R2,#100; OLD VAL=157
L3:MOV R1,#255
L2:MOV R0,#10
L1:DJNZ R0,L1
DJNZ R1,L2
DJNZ R2,L3
RET

INITIALIZELCD:
MOV P2,#01H
ACALL COMMAND
```

```
MOV P2,#38H
ACALL COMMAND
MOV P2,#0EH
ACALL COMMAND
RET

DELAYFORLCD:
MOV R3,#2
HERE3:MOV R4,#255
HERE4:DJNZ R4,HERE4
DJNZ R3,HERE3
RET

COMMAND:
CLR RW
CLR RS
SETB E
ACALL DELAYFORLCD
CLR E
ACALL DELAYFORLCD
RET

DATA1:
CLR RW
SETB RS
SETB E
ACALL DELAYFORLCD
CLR E
ACALL DELAYFORLCD
RET

DISPLAY:   ;DISPLAYS A WORD STORED IN ROM, WHOSE LENGTH IS STORED IN
B AND ADDRESS OF FIRST CHARACHTER IS STORED IN DPTR
NOP
```

```
L6:
CLR A
MOVC A,@A+DPTR
MOV P2,A
ACALL DATA1
INC DPTR
DJNZ B,L6
RET

GETADDRESS: ;ID=R3R2R1R0
MOV R4,#40H
MOV B,#0CH          ;ADDRESS=R2(0CH)+R1+40H +QUOTIENT OF
{R2(04H)+R0+R1(02H)}/8
MOV A,R2            ;ADDRESS WILL BE STORED IN A
MUL AB                  ;BIT NO. =  REMAINDER OF
{R2(04H)+R0+R1(02H)}/8
ADD A,R1            ;BIT NO. WILL B ESTORED IN B
ADD A,R4
MOV R4,A

MOV A,R2
MOV B,#04H
MUL AB
ADD A,R0
MOV R5,A
MOV A,R1
MOV B,#02H
MUL AB
ADD A,R5
MOV B,#08H
DIV AB

ADD A,R4
```

```
RET

GETNUM:                         ;STORES ASCII VALUE OF PRESSED KEY IN A
MOV COL,#0FFH
NOTPRESSED:MOV ROW,#00H
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,PRESSED
SJMP NOTPRESSED

PRESSED:
ACALL DELAY20MS

MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,GETKEY
SJMP NOTPRESSED
GETKEY:
ACALL GETROW
ACALL GETCOLUMN

MOV A,#04
MUL AB
ADD A,R5
;ADD A,#0C0H
MOV DPH,#03H
MOV DPL,A
CLR A
MOVC A,@A+DPTR
RET
```

```
GETID:                    ;STORES ID OF VOTER IN 30H,31H,32H,33H.e.g IF
ID IS 1348 THEN 1 WILL BE STORED IN 30H 2 IN 31H AND SO ON..
MOV R7,#30H
MOV R6,#04H

NEXTNUM:ACALL GETNUM
CJNE A,#2AH,NOTSTAR
AJMP START
NOTSTAR:CJNE A,#23H,NOTSTARANDNOTHASH
ACALL SHOWRESULTS
AJMP FINISH
NOTSTARANDNOTHASH:MOV P2,A
ACALL DATA1
SUBB A,#48
MOV B,R7
MOV R0,B
MOV @R0,A
INC R7
ACALL DELAY1S
DJNZ R6,NEXTNUM
RET

INITIALIZEDB:
MOV R0,#40H
ITER:MOV @R0,#00H
INC R0
CJNE R0,#07FH,ITER
RET

GETMASK:
MOV A,#00000001B
ROT:RL A
DJNZ B,ROT
```

```
RET

CHECKVALIDITY:          ;A CONTAINS ADDRESS AND B CONTAINS BIT NO.
MOV R0,A
ACALL GETMASK
ANL A,@R0
CJNE A,#00H,INVALID
VALID:
MOV 20H,#0AH
SJMP OVER1
INVALID:MOV 20H,#0BH
OVER1:RET

SETADDRESS:             ;A CONTAINS ADDRESS AND B CONTAINS BIT NO.
MOV R0,A
ACALL GETMASK
ORL A,@R0
MOV @R0,A
RET

INCVOTES:               ;A CONTAINS THE ASCII VALUE OF CANDIDATE EXAMPLE
#42H FOR B AFTER CALLING INCVOTES VOTE COUNT OF B
MOV B,#2H               ;WILL BE INCREASED BY 1,THE VOTE COUNT IS STORED IN
36H,37H FOR A ,38H AND 39H FOR B 3AH AND 3BH FOR C
SUBB A,#41H             ;3CH AND 3DH FOR D.VOTE COUNT IS IN DECIMAL
FORMAT.e.g: IF VOTE COUNT OF B IS 199( 01 IN 38H AND 99 IN 39H)
MUL AB                  ;AFTER CALLING INCVOTES THE VOTE COUNT OF B
WILL BE 200(02 IN 38H AND 00 IN 39H).
ADD A,#37H
MOV R0,A

CJNE @R0,#99H,NOT99
MOV @R0,#00H
DEC R0
```

```
INC @R0
SJMP OVER2

NOT99:MOV B,#10H
MOV A,@R0
DIV AB
MOV R1,B
CJNE R1,#09H,NOT9ANDNOT99
MOV A,#7H
ADD A,@R0
MOV @R0,A
SJMP OVER2

NOT9ANDNOT99:INC @R0
OVER2:RET


PRINTNUMBER:        ;A CONTAINS ADDRESS OF REGISTER WHERE THE HIGHER BYTE OF NUMBER IS STORED
MOV R0,A
MOV A,@R0
ADD A,#48
MOV P2,A
ACALL DATA1

INC R0
MOV A,@R0
MOV B,#10H
DIV AB
ADD A,#48
MOV P2,A
ACALL DATA1
MOV A,B
ADD A,#48
```

```
MOV P2,A
ACALL DATA1

RET

SHOWRESULTS:
MOV P2,#01H
ACALL COMMAND

MOV B,#2H
MOV DPTR,#LA
ACALL DISPLAY
MOV A,#36H
ACALL PRINTNUMBER

MOV DPTR,#SPACE
MOV B,#5H
ACALL DISPLAY

MOV B,#2H
MOV DPTR,#LB
ACALL DISPLAY
MOV A,#38H
ACALL PRINTNUMBER

MOV P2,#0C0H
ACALL COMMAND

MOV B,#2H
MOV DPTR,#LC
ACALL DISPLAY
MOV A,#3AH
ACALL PRINTNUMBER
```

```
MOV DPTR,#SPACE
MOV B,#5H
ACALL DISPLAY

MOV B,#2H
MOV DPTR,#LD
ACALL DISPLAY
MOV A,#3CH
ACALL PRINTNUMBER
ACALL DELAY1S
ACALL GETNUM
ACALL DELAY1S
AJMP STARTELECTION
RET
FINISH:NOP

ORG 0300H
DB "A321B654C987D#0*"
ENTERID:DB "ENTER ID:"
LA:DB "A:"
LB:DB "B:"
LC:DB "C:"
LD:DB "D:"
SPACE:DB "     "
INVALIDID:DB "INAVALID ID"
CASTVOTE:DB "CAST YOUR VOTE"
VOTECOUNTED1:DB "YOUR VOTE IS"
VOTECOUNTED2:DB "COUNTED"
ENDORG 0000H
RS EQU P3.7
RW EQU P3.6
E EQU P3.5
ROW EQU P0
COL EQU P1
```

```
STARTELECTION:ACALL INITIALIZEDB
MOV R0,#36H
L7:MOV @R0,#00H
INC R0
CJNE R0,#3EH,L7

START:ACALL INITIALIZELCD
MOV DPTR,#ENTERID
MOV B,#8
ACALL DISPLAY
MOV P2,#0C0H
ACALL COMMAND
ACALL GETID
MOV R3,30H
MOV R2,31H
MOV R1,32H
MOV R0,33H
ACALL GETADDRESS
MOV 34H,A
MOV 35H,B
ACALL CHECKVALIDITY
MOV A,20H
CJNE A,#0BH,VALID1
INVALID1:MOV P2,#01H
ACALL COMMAND
MOV DPTR,#INVALIDID
MOV B,#10
ACALL DISPLAY
ACALL GETNUM
ACALL DELAY1S
SJMP START
VALID1:
MOV P2,#01H
```

```
ACALL COMMAND
MOV DPTR,#CASTVOTE
MOV B,#14
ACALL DISPLAY
ACALL GETNUM
MOV P2,#0C0H
ACALL COMMAND
MOV P2,A
ACALL DATA1
ACALL DELAY1S
ACALL INCVOTES
MOV A,34H
MOV B,35H
ACALL SETADDRESS
MOV P2,#01H
ACALL COMMAND
MOV DPTR,#VOTECOUNTED1
MOV B,#12
ACALL DISPLAY
MOV P2,#0C4H
ACALL COMMAND
MOV DPTR,#VOTECOUNTED2
MOV B,#7
ACALL DISPLAY
ACALL GETNUM
ACALL DELAY1S
SJMP START


FUNCTIONS: LJMP FINISH

GETCOLUMN:   ;;STORES THE COL NO. OF PRESSED KEY IN R5
MOV R5,#00H
NEXT:RRC A
```

```
JNC FOUNDCOLUMN
INC R5
SJMP NEXT
FOUNDCOLUMN:RET

GETROW:  ;STORES THE ROW NO. OF PRESSED KEY IN B
MOV B,#00H
MOV ROW,#00001110B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW


INC B
MOV ROW,#00001101B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW

INC B
MOV ROW,#00001011B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW

INC B
MOV ROW,#00000111B
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,FOUNDROW

FOUNDROW:RET
```

```
DELAY20MS:
MOV R1,#36
HERE1:MOV R0,#255
HERE:DJNZ R0,HERE
DJNZ R1,HERE1
RET

DELAY1S:
MOV R2,#100; OLD VAL=157
L3:MOV R1,#255
L2:MOV R0,#10
L1:DJNZ R0,L1
DJNZ R1,L2
DJNZ R2,L3
RET

INITIALIZELCD:
MOV P2,#01H
ACALL COMMAND
MOV P2,#38H
ACALL COMMAND
MOV P2,#0EH
ACALL COMMAND
RET

DELAYFORLCD:
MOV R3,#2
HERE3:MOV R4,#255
HERE4:DJNZ R4,HERE4
DJNZ R3,HERE3
RET

COMMAND:
CLR RW
```

```
CLR RS
SETB E
ACALL DELAYFORLCD
CLR E
ACALL DELAYFORLCD
RET

DATA1:
CLR RW
SETB RS
SETB E
ACALL DELAYFORLCD
CLR E
ACALL DELAYFORLCD
RET

DISPLAY:   ;DISPLAYS A WORD STORED IN ROM, WHOSE LENGTH IS STORED IN B AND ADDRESS OF FIRST CHARACHTER IS STORED IN DPTR
NOP
L6:
CLR A
MOVC A,@A+DPTR
MOV P2,A
ACALL DATA1
INC DPTR
DJNZ B,L6
RET

GETADDRESS: ;ID=R3R2R1R0
MOV R4,#40H
MOV B,#0CH              ;ADDRESS=R2(0CH)+R1+40H +QUOTIENT OF {R2(04H)+R0+R1(02H)}/8
MOV A,R2                ;ADDRESS WILL BE STORED IN A
```

```
MUL AB                    ;BIT NO. =  REMAINDER  OF
{R2(04H)+R0+R1(02H)}/8
ADD A,R1                  ;BIT NO. WILL B ESTORED IN B
ADD A,R4
MOV R4,A

MOV A,R2
MOV B,#04H
MUL AB
ADD A,R0
MOV R5,A
MOV A,R1
MOV B,#02H
MUL AB
ADD A,R5
MOV B,#08H
DIV AB

ADD A,R4

RET


GETNUM:                   ;STORES ASCII VALUE OF PRESSED KEY IN A
MOV COL,#0FFH
NOTPRESSED:MOV ROW,#00H
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,PRESSED
SJMP NOTPRESSED

PRESSED:
ACALL DELAY20MS
```

```
MOV A,COL
ANL A,#00001111B
CJNE A,#00001111B,GETKEY
SJMP NOTPRESSED
GETKEY:
ACALL GETROW
ACALL GETCOLUMN

MOV A,#04
MUL AB
ADD A,R5
;ADD A,#0C0H
MOV DPH,#03H
MOV DPL,A
CLR A
MOVC A,@A+DPTR
RET



GETID:                    ;STORES ID OF VOTER IN 30H,31H,32H,33H.e.g IF ID IS 1348 THEN 1 WILL BE STORED IN 30H 2 IN 31H AND SO ON..
MOV R7,#30H
MOV R6,#04H

NEXTNUM:ACALL GETNUM
CJNE A,#2AH,NOTSTAR
AJMP START
NOTSTAR:CJNE A,#23H,NOTSTARANDNOTHASH
ACALL SHOWRESULTS
AJMP FINISH
NOTSTARANDNOTHASH:MOV P2,A
```

```
ACALL DATA1
SUBB A,#48
MOV B,R7
MOV R0,B
MOV @R0,A
INC R7
ACALL DELAY1S
DJNZ R6,NEXTNUM

RET

INITIALIZEDB:
MOV R0,#40H
ITER:MOV @R0,#00H
INC R0
CJNE R0,#07FH,ITER
RET

GETMASK:
MOV A,#00000001B
ROT:RL A
DJNZ B,ROT
RET

CHECKVALIDITY:          ;A CONTAINS ADDRESS AND B CONTAINS BIT NO.
MOV R0,A
ACALL GETMASK
ANL A,@R0
CJNE A,#00H,INVALID
VALID:
MOV 20H,#0AH
SJMP OVER1
INVALID:MOV 20H,#0BH
OVER1:RET
```

```
SETADDRESS:          ;A CONTAINS ADDRESS AND B CONTAINS BIT NO.
MOV R0,A
ACALL GETMASK
ORL A,@R0
MOV @R0,A
RET

INCVOTES:            ;A CONTAINS THE ASCII VALUE OF CANDIDATE EXAMPLE #42H FOR B AFTER CALLING INCVOTES VOTE COUNT OF B
MOV B,#2H            ;WILL BE INCREASED BY 1,THE VOTE COUNT IS STORED IN 36H,37H FOR A ,38H AND 39H FOR B 3AH AND 3BH FOR C
SUBB A,#41H          ;3CH AND 3DH FOR D.VOTE COUNT IS IN DECIMAL FORMAT.e.g: IF VOTE COUNT OF B IS 199( 01 IN 38H AND 99 IN 39H)
MUL AB               ;AFTER CALLING INCVOTES THE VOTE COUNT OF B WILL BE 200(02 IN 38H AND 00 IN 39H).
ADD A,#37H
MOV R0,A



CJNE @R0,#99H,NOT99
MOV @R0,#00H
DEC R0
INC @R0
SJMP OVER2

NOT99:MOV B,#10H
MOV A,@R0
DIV AB
MOV R1,B
CJNE R1,#09H,NOT9ANDNOT99
MOV A,#7H
ADD A,@R0
```

```
MOV @R0,A
SJMP OVER2

NOT9ANDNOT99:INC @R0
OVER2:RET


PRINTNUMBER:        ;A CONTAINS ADDRESS OF REGISTER WHERE THE HIGHER BYTE OF NUMBER IS STORED
MOV R0,A
MOV A,@R0
ADD A,#48
MOV P2,A
ACALL DATA1

INC R0
MOV A,@R0
MOV B,#10H
DIV AB
ADD A,#48
MOV P2,A
ACALL DATA1
MOV A,B
ADD A,#48
MOV P2,A
ACALL DATA1

RET

SHOWRESULTS:
MOV P2,#01H
ACALL COMMAND

MOV B,#2H
```

```
MOV DPTR,#LA
ACALL DISPLAY
MOV A,#36H
ACALL PRINTNUMBER

MOV DPTR,#SPACE
MOV B,#5H
ACALL DISPLAY

MOV B,#2H
MOV DPTR,#LB
ACALL DISPLAY
MOV A,#38H
ACALL PRINTNUMBER

MOV P2,#0C0H
ACALL COMMAND

MOV B,#2H
MOV DPTR,#LC
ACALL DISPLAY
MOV A,#3AH
ACALL PRINTNUMBER

MOV DPTR,#SPACE
MOV B,#5H
ACALL DISPLAY

MOV B,#2H
MOV DPTR,#LD
ACALL DISPLAY
MOV A,#3CH
ACALL PRINTNUMBER
ACALL DELAY1S
```

```
ACALL GETNUM
ACALL DELAY1S
AJMP STARTELECTION
RET

FINISH:NOP

ORG 0300H
DB "A321B654C987D#0*"
ENTERID:DB "ENTER ID:"
LA:DB "A:"
LB:DB "B:"
LC:DB "C:"
LD:DB "D:"
SPACE:DB "     "
INVALIDID:DB "INVALID ID"
CASTVOTE:DB "CAST YOUR VOTE"
VOTECOUNTED1:DB "YOUR VOTE IS"
VOTECOUNTED2:DB "COUNTED"
END
```

3.Results

3.1 Output

1.Enter the ID between 1000-1500.

2.Cast your vote to the candidate you wish by entering in the Keypad. The candidates competing are given by A, B, C and D

3.Once the vote is casted it displays “YOUR VOTE IS COUNTED”

5.To check the number of votes casted to each candidate Press #*# in Keypad

Special Case:

If the voter tries to cast his vote again it displays an “INVALID ID” message

3.2 Future Scope

The future scope of Electronic Voting Machines (EVMs) comprises several potential advancements to enhance their functionality, security, and accessibility. Integration of biometric authentication methods like fingerprint sensing, iris scanning, or facial recognition can bolster the security and integrity of the voting process by preventing impersonation and ensuring that only eligible voters participate. Blockchain technology holds promise for providing transparent and immutable record-keeping, offering decentralized and tamper-resistant platforms for vote counting and verification. The development of remote voting options through secure online platforms or mobile applications can increase voter accessibility, necessitating robust authentication mechanisms and encryption protocols to protect the integrity and confidentiality of remote votes. Incorporating accessibility features such as audio interfaces, braille keypads, and tactile feedback ensures inclusivity for individuals with disabilities, while continuous improvement in user interface design and usability testing optimizes the user experience for voters of all backgrounds. Enhanced security measures, including advanced encryption algorithms and intrusion detection systems, are crucial to mitigate cyber threats and safeguard electoral data. Innovations in voter verification solutions, such as blockchain-based digital identities or secure smart cards, can streamline the authentication process and minimize fraud risks. Additionally, real-time monitoring and reporting capabilities enable electoral authorities to track voting activities, identify anomalies, and make informed decisions during elections, supported by data analytics tools for insights into voter behaviour and preferences. Implementing these advancements can foster transparent, inclusive, and democratic electoral processes, ensuring the integrity and reliability of EVMs in future elections.

3.3 Conclusion and Discussion

In summary, the creation of an Electronic Voting Machine (EVM) through the use of an 8051 microcontroller has proven to be a great success and represents a major advancement in the modernization of electoral processes. By means of careful preparation, careful execution, and thorough testing, we have proven that using microcontroller technology to expedite the voting process is both feasible and beneficial.

Looking ahead, the implications of our project extend beyond its immediate implementation. The successful development of the EVM opens doors to further innovations and enhancements in electoral technology. Future iterations of the EVM could incorporate advanced features such as biometric authentication, blockchain integration, and remote voting options, thereby continuing to improve the integrity and inclusivity of elections. Additionally, our project serves as a foundation for ongoing research and development efforts aimed at addressing the evolving challenges and opportunities in electoral systems worldwide.

References

1. https://ieeexplore.ieee.org/abstract/document/6828142
2. https://www.ieeeprojectsmadurai.com/IEEE%202019%20IOT%20BASE PAPERS/4_VOTING%20MACHINE.pdf
3. http://rameshmanza.in/Publication/05-10-2013/Architecture%20of%20Electronic%20Voting%20Machine%20for%20Preferential%20Voting.pdf

BIO DATA

Name: Maadesh A

Email: maadesh.a2022@vitstudent.ac.in

Mobile number: 9940835432

Name: Hariharan M

Email: 24hariharan@gmail.com

Mobile number: 6381317177

Name: Udit Krishna B

Email: uditkrishna.b2022@vitstudent.ac.in

Mobile number: 9944856730

Name: Danus D

Email: danus.d2022@vitstudent.ac.in

Mobile number: 9360549439

Name: Pradesh Kumar M

Email: pradeshkumar.m2022@vitstudent.ac.in

Mobile number: 9962609887

Name: Senthil Kumar M

Email: senthilkumar.m2021@vitstudent.ac.in

Mobile number: 9345661106