## NMJ21704 MICROPROCESSOR SYSTEM

## MINI PROJECT PRESENTATION

## SYSTEM DESIGN: VOTING SYSTEM

#### BY:

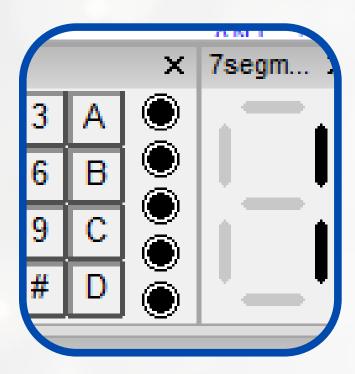
- ADAM HAKIMI BIN AHMAD GHOZALI (231021306)
  - MUHAMMAD FITRI BIN HASSAN (231021368)
- MUHAMAD EKMAL HAKIM BIN OSMAN (231021352)

#### INTRODUCTION

In an era where electronic systems play a vital role in ensuring transparency and speed, digital voting mechanisms have become increasingly important. This project presents a simulation of a digital voting system using the MylSim85 platform, an emulator for the Intel 8085 microprocessor. Rather than relying on physical hardware, all components such as the 4×4 keypad, LEDs, and 7-segment display are simulated in software using MylSim85, making it ideal for educational and proof-of-concept purposes







```
CALL DELAY
MVI A, 10101010B
OUT PA
CALL DELAY
; Show 'A'
MVI A, OCH
CALL LOOK
OUT PB
CALL DELAYL
                 ;LONGER DELAY
LDA VOTEA
DCR A
CALL LOOK
OUT PB
CALL DELAYL
CAND B VOTE----
```

```
EQU 83H
             ORG 0000H
             JMP MAIN
             ORG 002CH
10
             JMP SHOWVOTE
11
    MAIN:
12
13
             LXI SP, 3FFFH
             MVI A, 00001110B
14
15
             SIM
16
             EΙ
17
             ; Configure 8255: Port A &
18
19
             MVI A, 89H
             OUT CTRL
20
21
             MVI A, 00H
22
             STA VOTEA
23
24
             STA VOTEB
25
             STA VOTEC
26
             OUT PA
27
             OUT PB
28
             OUT PC
    START:
29
30
             ; JMP CHKCODE
31
    CHKKP:
             MVI A, 00H ; SHOW 0 7SEG
32
33
             OUT PB
```

The project simulates a simple voting machine for three candidates (Candidate A, B, and C). It provides real-time feedback through blinking LEDs and a 7-segment display. Additionally, an interrupt (RST 5.5) is utilized to display vote summaries without interfering with the main voting loop.

#### PROJECT OBJECTIVES

This project demonstrates the application of microprocessor system in developing a voting system. Users interact with the system via a keypad to select their preferred candidate. Each vote is stored in memory and displayed on 7-segment displays. System feedback is provided using onboard LEDs. An interrupt-based Admin Mode allows secure access to display or reset results.







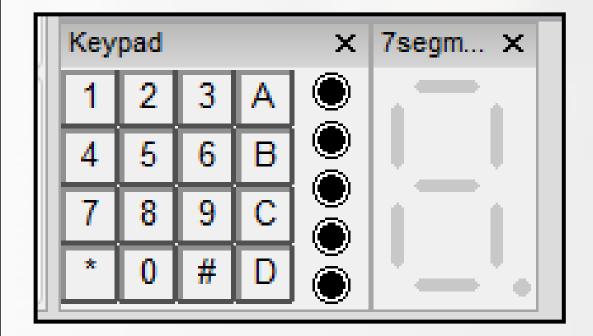
To design and implement
 a digital voting system
 entirely within the
 MylSim85 simulator and
 utilize the keypad
 scanning, vote counting,
 and real-time output of
 8085 microprocessor's
 assembly language.

 The system simulate a secure and responsive voting system using My1Sim85 and display selected candidate and corresponding vote count on a 7-segment display. • It use the RST 5.5 interrupt for admin-triggered vote summary and confirm successful voting via LED blinking patterns.

(1) The system needs LED,7-segment display and Keypad

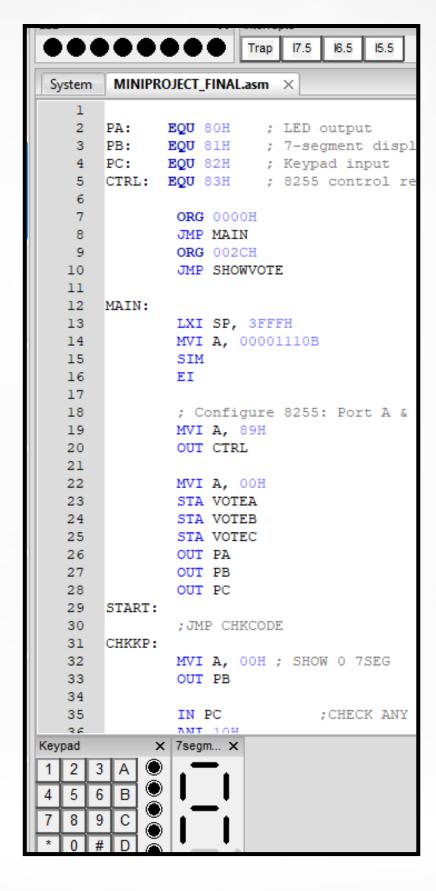
(2) Users can choose 1, 2, or 3 to vote for the candidate that they choose

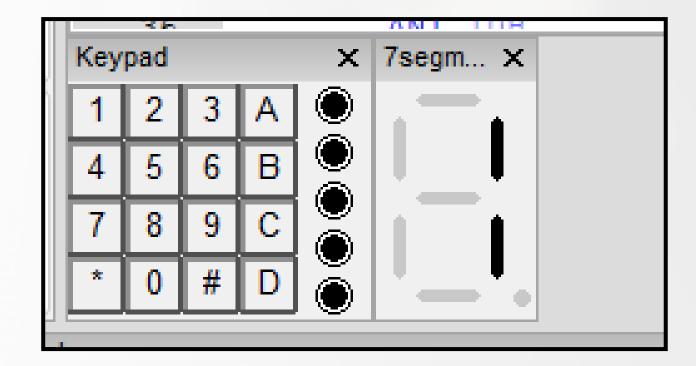
```
Trap 17.5 16.5 15.5
System MINIPROJECT_FINAL.asm
                       ; 7-segment display
             EQU 81H
             EQU 82H
                      ; Keypad input
            EQU 83H
                       ; 8255 control register
              ORG 0000H
              JMP MAIN
  9
              ORG 002CH
              JMP SHOWVOTE
 10
 11
 12 MAIN:
 13
              LXI SP, 3FFFH
              MVI A, 00001110B
 14
              SIM
 15
 16
              ΕI
 17
              ; Configure 8255: Port A & B = output, C = input
 18
 19
              MVI A, 89H
              OUT CTRL
 20
 21
 22
              MVI A, OOH
 23
              STA VOTEA
              STA VOTEB
 24
 25
              STA VOTEC
 26
              OUT PA
 27
              OUT PB
 28
              OUT PC
 29
 30
              ; JMP CHKCODE
 31
     CHKKP:
 32
              MVI A, 00H ; SHOW 0 7SEG
              OUT PB
 33
 34
 35
              IN PC
                              ; CHECK ANY KEYPRESS
           x 7segm... x
```



2

- (3) For example, a user press 1 to vote for candidate A, LED will blink 3 times as a confirmation that it's choosing candidate A. 7SEGMENT will display a chosen candidate.
- (4) 7SEGMENT will display a current vote for the chosen candidate.





4

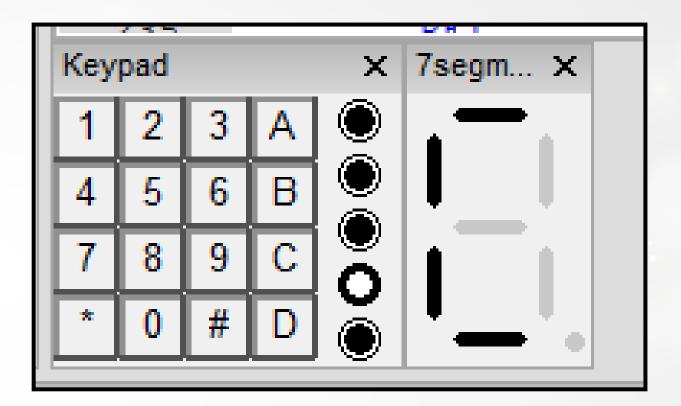
If the interrupt button are triggered, LED will keep blinking in a certain pattern



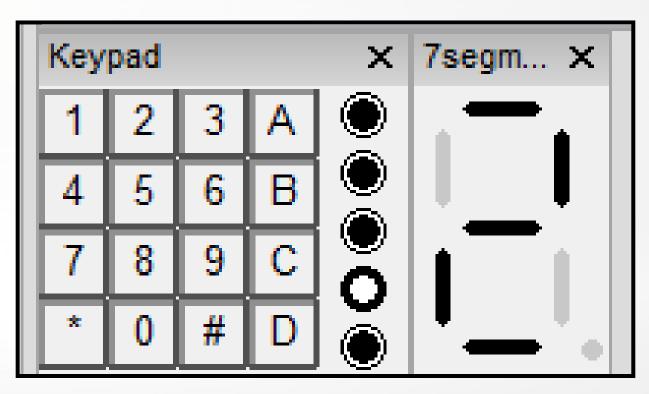




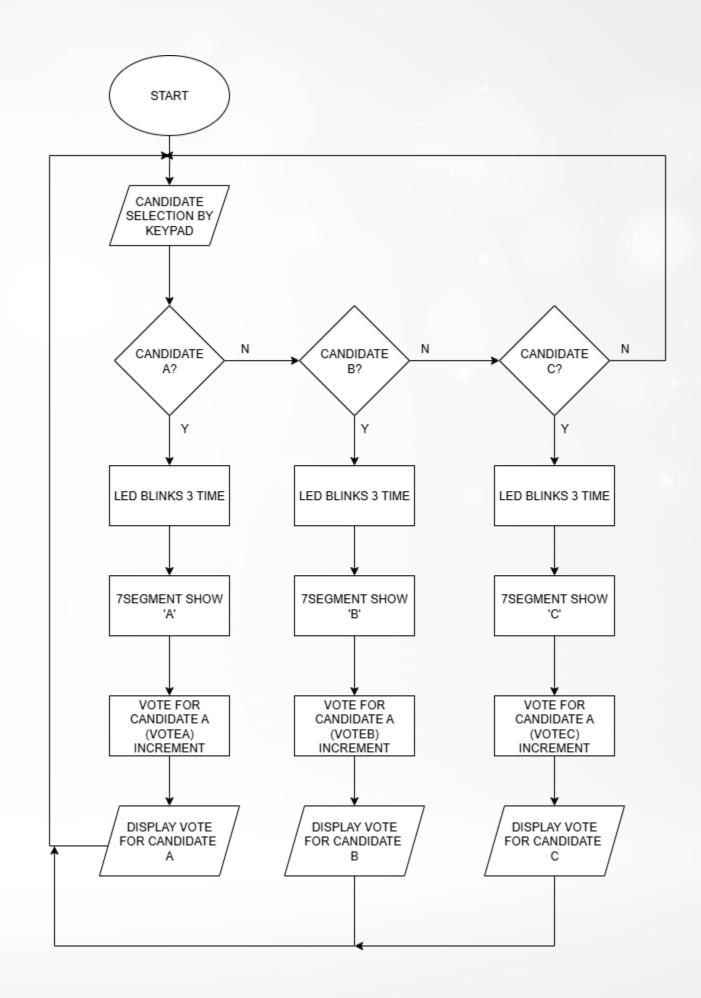
Then, 7SEGMENT will proceed into showing all votes from all candidates. 7SEGMENT will shows a candidate then the total vote that he/she got.



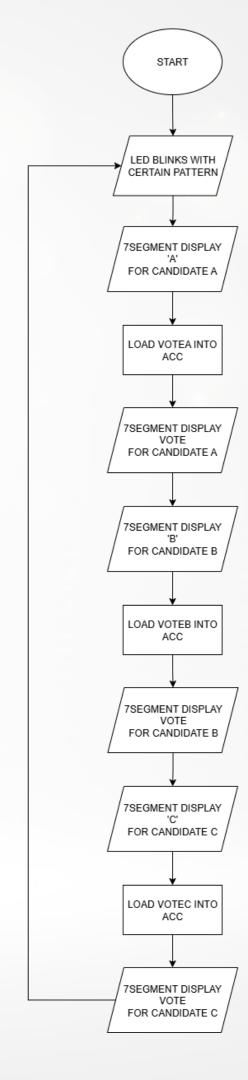




**FLOWCHART: MAIN** 



FLOWCHART: INTERRUPT TRIGGERED



### CODE

We initialize a port A, port B and port C and also initialize HEX origin with interrupt memory location

1

At the main section, we initialize control word for masking the interrupt

2

Lastly, all the candidates vote counts have been initialized so that it will be 0

3

```
EQU 80H
                 ; LED output
PA:
PB:
       EQU 81H
                 ; 7-segment display
PC:
       EQU 82H
                 ; Keypad input
                 ; 8255 control register
CTRL:
      EQU 83H
        ORG 0000H
        JMP MAIN
        ORG 002CH
        JMP SHOWVOTE
MAIN:
       LXI SP, 3FFFH
       MVI A, 00001110B
        SIM
        EΙ
        ; Configure 8255: Port A & B = output, C = input
       MVI A, 89H
        OUT CTRL
       MVI A, 00H
        STA VOTEA
        STA VOTEB
        STA VOTEC
       OUT PA
        OUT PB
        OUT PC
```

```
CHKKP:
       MVI A, 00H ; SHOW 0 7SEG
       OUT PB
       IN PC
                      ; CHECK ANY KEYPRESS
       ANI 10H
       JZ CHKKP
                      ;LOOP IF NO KEYPRESS
       IN PC
       ;CPI 60H
                             ;KEY 1 PRESSED
       CPI 00010000B
                              ;KEY 1 PRESSED
       JZ SELECT1
       ;CPI 61H
                              ;KEY 2
       CPI 00010001B
                              ;KEY 2
       JZ SELECT2
       ;CPI 62H
                              ;KEY 3
       CPI 00010010B
                              ;KEY 3
       JZ SELECT3
       JMP CHKKP
                       ; IGNORE OTHER KEY
       HLT
```

```
SELECT1:
       ; Increment vote
       LDA VOTEA
       INR A
       STA VOTEA
       ; Show 'A'
       MVI A, OCH
                       ; Index of 'A' in your table (77H)
       CALL LOOK
       OUT PB
       CALL CONFIRMLED ; LED TO CONFIRM VOTE REGISTERED
       CALL DELAY
       ; Show vote count
       LDA VOTEA
       DCR A
                       ;TO GET 7SEG INDEX
       CALL LOOK
       OUT PB
       CALL DELAY
       JMP START
```

For checking on keypad input, if users press 1, it will jump to select 1 and if the user press 2, it will jump to select 2 and so on.

Label for candidate A vote. It will show 'A' on 7SEGMENT then follows by the current vote for candidate A

```
SELECT3:
        ; Increment vote
       LDA VOTEC
       INR A
       STA VOTEC
       ; Show 'C'
       MVI A, OEH ; Index of 'C' in your table
       CALL LOOK
       OUT PB
       CALL CONFIRMLED
       CALL DELAY
       ; Show vote count
       LDA VOTEC
       DCR A
       CALL LOOK
       OUT PB
       CALL DELAY
       JMP START
```

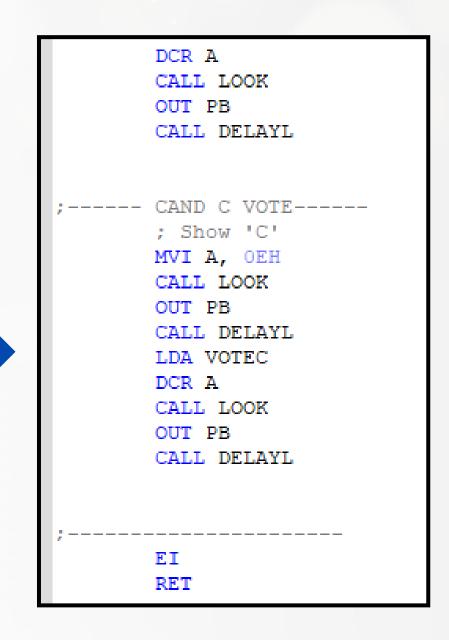
3

It's the same thing happen with the other candidate. The same logic is applied.

## CODE INTERRUPT SITUATION

If the interrupt has been triggered, LED will keep blinking in a certain pattern and the process will go through every candidate along with their total vote.

```
SHOWVOTE:
       LED TO CONFIRM INTRPT-----
       MVI A, 10101010B
       OUT PA
       CALL DELAY
       MVI A, 01010101B
       OUT PA
       CALL DELAY
       MVI A, 10101010B
       OUT PA
       CALL DELAY
       CAND A VOTE----
       ; Show 'A'
       MVI A, OCH
       CALL LOOK
       OUT PB
       CALL DELAYL
                         ;LONGER DELAY
       LDA VOTEA
       DCR A
       CALL LOOK
       OUT PB
       CALL DELAYL
       CAND B VOTE----
       ; Show 'B'
       MVI A, ODH
       CALL LOOK
       OUT PB
       CALL DELAYL
       LDA VOTEB
       DCR A
```



# CODE FOR LED BLINK IF THE USER PRESS A KEYPAD TO START VOTING FOR THEIR CANDIDATE AS A CONFIRMATION FOR THEIR VOTES

```
CONFIRMLED:
        : Blink LED 3 times
        MVI B, 03H
LEDLOOP:
        MVI A, OFFH
        OUT PA
        CALL DELAY
        MVI A, 00H
        OUT PA
        CALL DELAY
        DCR B
        JNZ LEDLOOP
        RET
```

# THIS AREA ARE FOR FINDING DATA IN THE TABLE ("DATA7SEG"), DELAY ROUTINE AND INITIALIZE VOTE COUNT

```
--- Convert number to 7-segment ---
LOOK:
        LXI H, DATA7SEG
        ADD L
        MOV L, A
        MOV A, M
        RET
; --- Delay Routine ---
DELAYL:
        MVI C, 50H
        JMP DL LOOP
DELAY:
        MVI C, 20H
DL LOOP:
        DCR C
        JNZ DL LOOP
        RET
; --- Var & Tables ---
        ORG 2000H
DATA7SEG:
        ;DB 3FH, 06H, 5BH, 4FH, 66H, 6DH, 7DH, 07H
        ;DB 7FH, 6FH, 77H, 7CH, 39H, 5EH, 79H, 71H
                06H, 5BH, 4FH, 77H, 66H, 6DH, 7DH, 7CH, 07H, 7FH, 6FH, 39H, 62H, 3FH, 76H, 5EH
                06H, 5BH, 4FH, 66H, 6DH, 7DH, 07H, 7FH, 6FH, 62H, 3FH, 76H, 77H, 7CH, 39H, 5EH
VOTEA:
                        ; Vote count for candidate A
VOTEB:
                         ; Vote count for candidate B
VOTEC:
                         ; Vote count for candidate C
```

#### CONCLUSION

The Digital Voting System project successfully demonstrates the capabilities of the Intel 8085 microprocessor within a simulated environment using the My1Sim85 platform. By integrating keypad input, LED indicators, a 7-segment display, and interrupt handling, the system replicates a simple yet functional voting mechanism for three candidates.

The project highlights essential microprocessor concepts such as I/O port interfacing, memory-based data manipulation, scanning, and the use of software interrupts (RST 5.5) to implement administrative features. The LED feedback and vote confirmation logic ensure a user-friendly experience, while the 7-segment display offers clear, real-time output of votes and system status.

By avoiding the use of physical hardware and relying solely on simulation, this project remains accessible, cost-effective, and highly suited for educational purposes. It provides a hands-on learning platform for students and educators to understand the practical application of assembly programming and embedded system design principles.

Overall, the project achieves its objectives and lays the groundwork for future enhancements such as multi-digit vote counts, authentication features, and persistent memory storage, making it a valuable exercise in digital systems development.

