

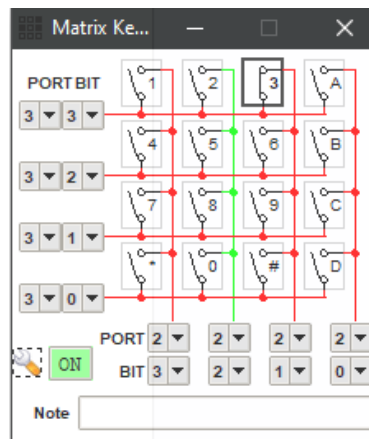
# MULTIPLEXED FOUR 7-SEGMENT WITH KEYPAD

EEE184.1 M67 - Microprocessor, Microcontroller Systems and Design Laboratory

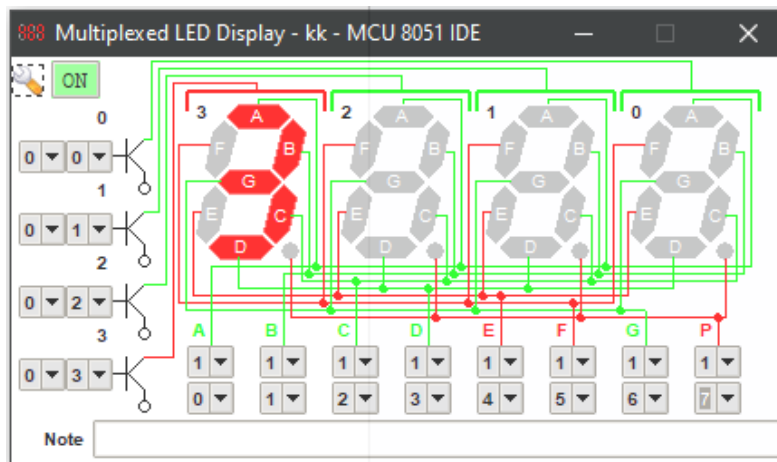
Ryan Christopher V. Edquila  
Department of Electrical Engineering and Technology  
Mindanao State University - Iligan Institute of Technology  
Iligan City, Philippines  
[ryanchristopoher.edquila@g.msuiit.edu.ph](mailto:ryanchristopoher.edquila@g.msuiit.edu.ph)

The provided code below represents a laboratory activity aimed at demonstrating the functionality of a 4x4 matrix keypad and a 4 seven-segment display and multiplexing it to shift to each every seven-segment display. The goal is to read input from the keypad, process the input values, manipulate specific pins based on the input, and generate corresponding output on other pins to control the display.

## 4X4 Matrix Keypad



## Four 7-Segment Display



## Code

In the code, the output pins P1.0 to P1.7 are used to control the seven-segment display. However, only four pins (P1.0 to P1.3) are required to control the individual segments of each seven-segment display.

To utilize these four pins effectively, multiplexing is employed. The code rapidly switches between the four seven-segment displays and displays the appropriate segment pattern for each digit.

```
;Edquila, Ryan Christopher V.  
;184.1 - M67  
;Lab 7 - Project
```

```
ORG 0000H
```

```
;Input
```

```
SETB P0.0  
SETB P0.1  
SETB P0.2  
SETB P0.3  
SETB P0.4  
SETB P0.5  
SETB P0.6  
SETB P0.7
```

```
SETB P1.0  
SETB P1.1  
SETB P1.2  
SETB P1.3  
SETB P1.4  
SETB P1.5  
SETB P1.6  
SETB P1.7
```

```
MAIN:
```

```
;FIRST INPUT  
ACALL INPUT  
CLR P0.3  
SETB P0.2  
SETB P0.1  
SETB P0.0  
CPL A  
MOV P1, A  
ACALL DELAY
```

```
;SECOND INPUT  
ACALL INPUT  
SETB P0.3
```

```
CLR P0.2  
SETB P0.1  
SETB P0.0  
CPL A  
MOV P1, A  
ACALL DELAY
```

```
;THIRD INPUT  
ACALL INPUT  
SETB P0.3  
SETB P0.2  
CLR P0.1  
SETB P0.0  
CPL A  
MOV P1, A  
ACALL DELAY
```

```
;FOURTH INPUT  
ACALL INPUT  
CLR P0.3  
SETB P0.2  
SETB P0.1  
CLR P0.0  
CPL A  
MOV P1, A  
ACALL DELAY  
  
SJMP MAIN
```

```
INPUT:
```

```
MOV P2, #1011b ;Keypad 0  
MOV C, P3.0  
CPL C  
JC ZERO
```

```
MOV P2, #0111b ;Keypad 1  
MOV C, P3.3
```

CPL C		RET
ACALL CHECKONE		
MOV P2, #1011b	;Keypad 2	CHECKTWO: JC TWO
MOV C, P3.3		RET
CPL C		TWO:
ACALL CHECKTWO		MOV A, #5BH
		RET
MOV P2, #1101b	;Keypad 3	CHECKTHREE: JC THREE
MOV C, P3.3		RET
CPL C		THREE:
ACALL CHECKTHREE		MOV A, #4FH
		RET
MOV P2, #0111b	;Keypad 4	CHECKFOUR: JC FOUR
MOV C, P3.2		RET
CPL C		FOUR:
ACALL CHECKFOUR		MOV A, #66H
		RET
MOV P2, #1011b	;Keypad 5	CHECKFIVE: JC FIVE
MOV C, P3.2		RET
CPL C		
ACALL CHECKFIVE		
MOV P2, #1101b	;Keypad 6	FIVE:
MOV C, P3.2		MOV A, #6DH
CPL C		RET
ACALL CHECKSIX		
		CHECKSIX: JC SIX
MOV P2, #0111b	;Keypad 7	RET
MOV C, P3.1		
CPL C		SIX:
ACALL CHECKSEVEN		MOV A, #7DH
		RET
MOV P2, #1011b	;Keypad 8	CHECKSEVEN: JC SEVEN
MOV C, P3.1		RET
CPL C		
ACALL CHECKEIGHT		SEVEN:
		MOV A, #07H
MOV P2, #1101b	;Keypad 9	RET
MOV C, P3.1		
CPL C		CHECKEIGHT: JC EIGHT
ACALL CHECKNINE		RET
RET		EIGHT:
		MOV A, #7FH
ZERO:		RET
MOV A, #3FH		
RET		CHECKNINE: JC NINE
		RET
CHECKONE: JC ONE		
RET		NINE:
ONE:		MOV A, #6FH
MOV A, #06H		

RET

DELAY:

MOV R0, #0FH

LOOP:

DJNZ R0, LOOP

RET

END

## Demo

