

Microcontrollers and Industrial Applications 1

KOM3722

Week 4

Asst. Prof. Bahadır Çatalbaş

Dip Switch Application (Correction)

```
#include <stdio.h>
#include <stdlib.h>
#include <pic18f45k22.h>
#include <htc.h>

#define _XTAL_FREQ 8000000

#pragma config FOSC = HSHP
#pragma config WDTCN = OFF

#define RB0 PORTBbits.RB0
#define RB1 PORTBbits.RB1
#define RB2 PORTBbits.RB2
#define RB3 PORTBbits.RB3
#define RB4 PORTBbits.RB4
#define RB5 PORTBbits.RB5
#define RB6 PORTBbits.RB6
#define RB7 PORTBbits.RB7

void main(void) {
    ANSELB = 0x00; //all is digital
    TRISB = 0xF0; //4 high bit is input
    PORTB = 0;
    INTCON2bits.RBPU = 0;
    while(1){
        RB0 = !RB4;
        RB1 = !RB5;
        RB2 = !RB6;
        RB3 = !RB7;
        // or
        //PORTB = (~PORTB & 0xF0)>>4;
    }
    return;
}
```

```
#include <stdio.h>
#include <stdlib.h>
#include <pic18f45k22.h>
#include <htc.h>

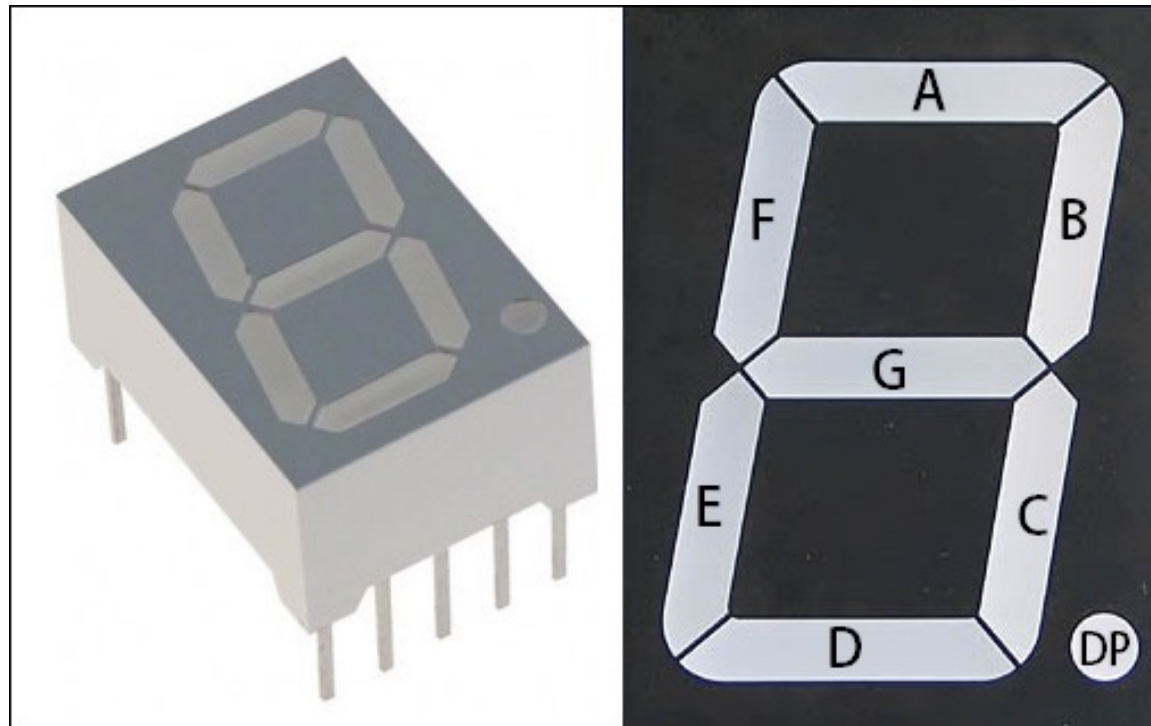
#define _XTAL_FREQ 8000000

#pragma config FOSC = HSHP
#pragma config WDTCN = OFF

#define RB0 PORTBbits.RB0
#define RB1 PORTBbits.RB1
#define RB2 PORTBbits.RB2
#define RB3 PORTBbits.RB3
#define RB4 PORTBbits.RB4
#define RB5 PORTBbits.RB5
#define RB6 PORTBbits.RB6
#define RB7 PORTBbits.RB7

void main(void) {
    ANSELB = 0x00;
    TRISB = 0xF0;
    PORTB = 0;
    INTCON2bits.RBPU = 0;
    while(1){
        //RB0 = !RB4;
        //RB1 = !RB5;
        //RB2 = !RB6;
        //RB3 = !RB7;
        PORTB = (~PORTB & 0xF0)>>4;
    }
    return;
}
```

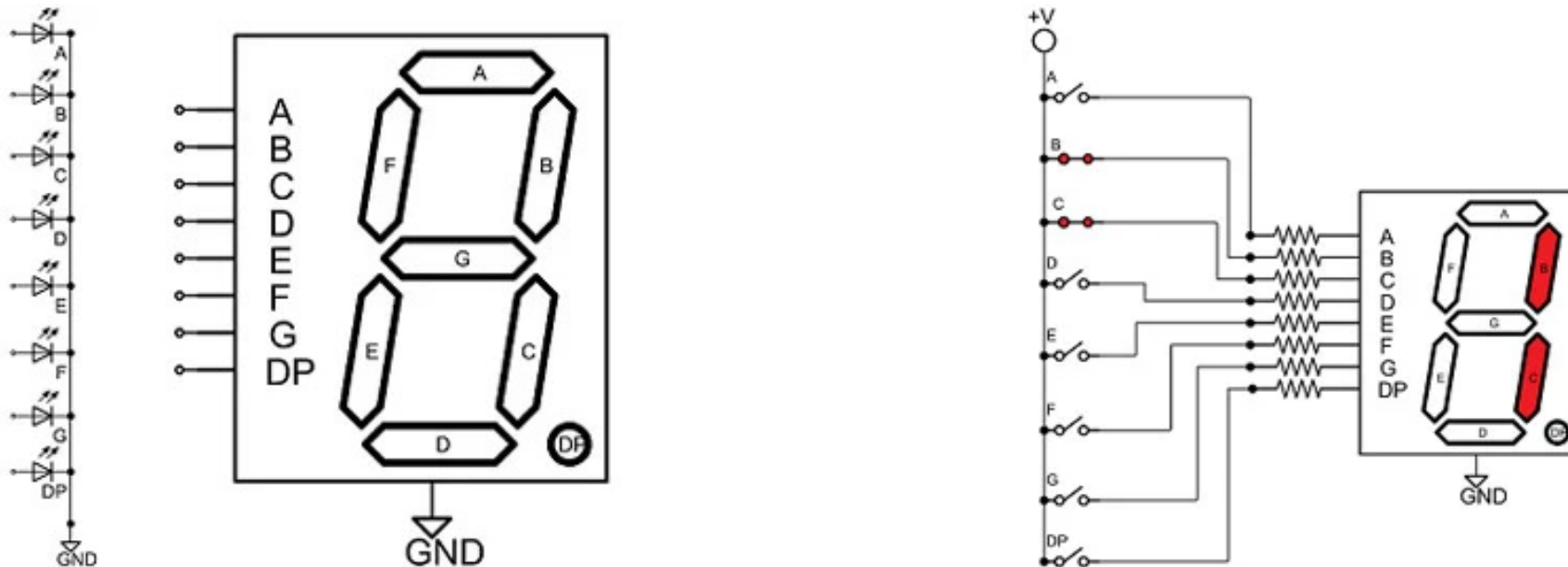
Seven Segment Led Display



<https://embedjournal.com/interface-7-segment-display-with-microcontroller/>

Seven Segment Led Display

- Common Cathode (CC)

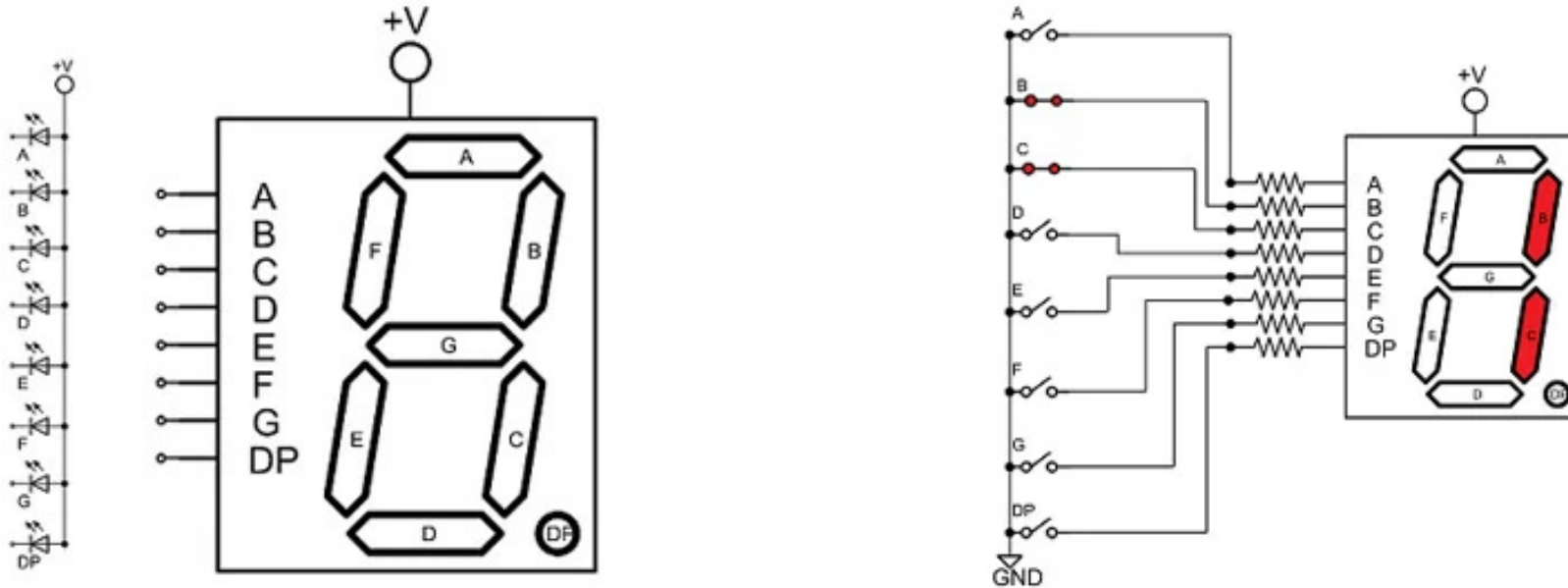


<https://www.jameco.com/Jameco/workshop/TechTip/working-with-seven-segment-displays.html>

Microcontrollers and Industrial Applications 1 KOM3722

Seven Segment Led Display

- Common Anode (CA)

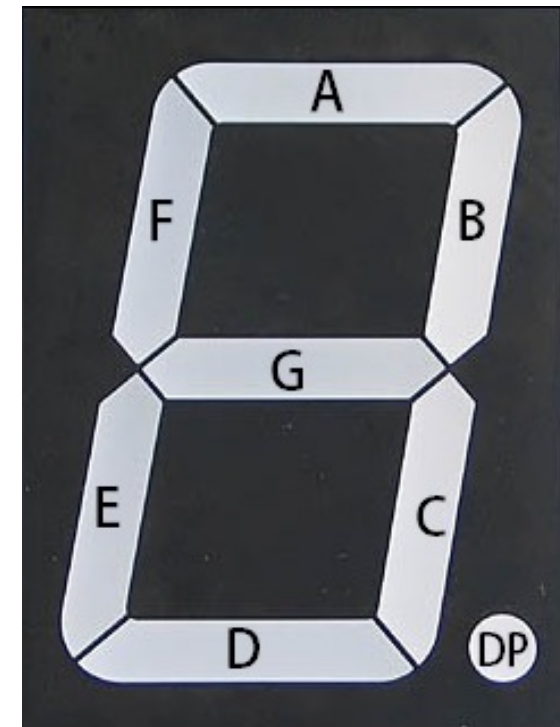


<https://www.jameco.com/Jameco/workshop/TechTip/working-with-seven-segment-displays.html>

Microcontrollers and Industrial Applications 1 KOM3722

Common cathode configuration lookup table

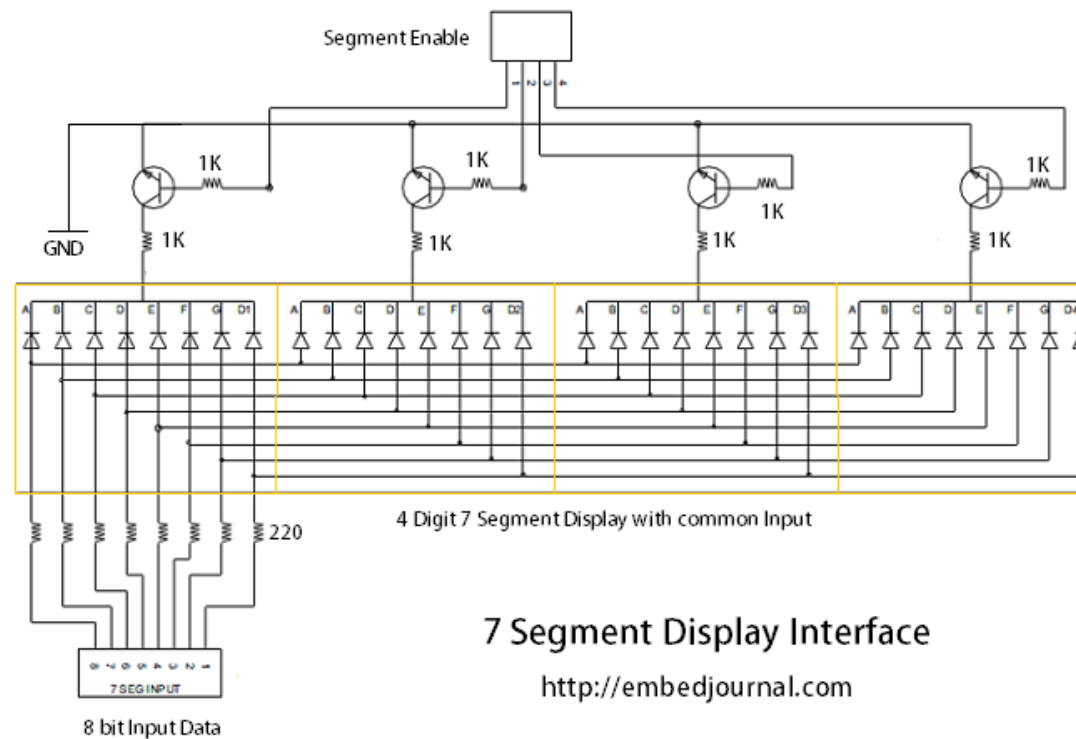
Number	DP	g	f	e	d	c	b	a	Hex
0	0	0	1	1	1	1	1	1	0x3F
1	0	0	0	0	0	1	1	0	0x06
2	0	1	0	1	1	0	1	1	0x5B
3	0	1	0	0	1	1	1	1	0x4F
4	0	1	1	0	0	1	1	0	0x66
5	0	1	1	0	1	1	0	1	0x6D
6	0	1	1	1	1	1	0	1	0x7D
7	0	0	0	0	0	1	1	1	0x07
8	0	1	1	1	1	1	1	1	0x7F
9	0	1	1	0	1	1	1	1	0x6F



Exercise

- Start from scratch and generate common cathode configuration lookup table
- Enlarge common cathode configuration lookup table with remaining hex digits A, b, C, d, E and F.
- Start from scratch and generate common anode configuration lookup table

Scan More Than One Seven Segment Display



<https://embedjournal.com/interface-7-segment-display-with-microcontroller/>

Microcontrollers and Industrial Applications 1 KOM3722

Chronometer Application

```
/*
 * File: newmain.c
 * Author: bahadircatalbas
 *
 * Created on 23 Mart 2023 Per?embe, 00:01
 */

#include <stdio.h>
#include <stdlib.h>
#include <pic18f45k22.h>
#include <htc.h>

#define _XTAL_FREQ 8000000

#pragma config FOSC = HSHP
#pragma config WDTEN = OFF

unsigned char seg_char[10] =
{0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
unsigned long counter = 0;
unsigned int value = 0;
unsigned char digits[10]={0,0,0,0};

void main(void) {
    ANSELA = 0x00;
    TRISA = 0xF0;
    PORTA = 0x00;

    ANSELD = 0x00;
    TRISD = 0x00;
    PORTD = 0x00;

    while(1){
        PORTA = 0x00;
        PORTD = seg_char[digits[0]];
        PORTA = 0x01;
        __delay_ms(8/4);

        PORTA = 0x00;
        PORTD = seg_char[digits[1]];
        PORTA = 0x02;
        __delay_ms(8/4);

        PORTA = 0x00;
        PORTD = seg_char[digits[2]];

        PORTA = 0x04;
        __delay_ms(8/4);

        PORTA = 0x00;
        PORTD = seg_char[digits[3]];
        PORTA = 0x08;
        __delay_ms(8/4);

        counter++;
        if(counter >= (30000*4)){
            counter = 0;
        }
        value = counter/3/4;

        digits[3] = value/1000;
        digits[2] = (value%1000)/100;
        digits[1] = (value%100)/10;
        digits[0] = value%10;
    }
    return;
}
```

