

Controlling a 7 - Segment Display Using a PIC Microcontroller and 7 – Segment Driver

VINOD NATLA

11/07/2008

Application Notes

Team - I

Executive Summary:

One universal requirement for most of the digital devices is an image numeric display. A seven segment display, which is also known as seven segment indicator is an electronic device that are used to display decimal values. These devices are used in digital meters, digital clocks, watches, house hold appliances and other electric parts for displaying numerical information. These are implemented in many circuit applications to perform a desired task. This application note will describe how to design a circuit to display the any of the ten decimal digits from 0 – 9 using a PIC microcontrollers and drivers [1].

Keywords:

Dual digit 7 segment Display, 18f4520 PIC Microcontroller, BCD to 7 – segment driver, switch

Objective:

The purpose of this document is for the reader to get familiar on how to operate dual digit 7 segment display using a microcontroller with the help of driver. This document will help to identify the pins of parts and design will be discussed along with some factors that should be taken into account.

Introduction:

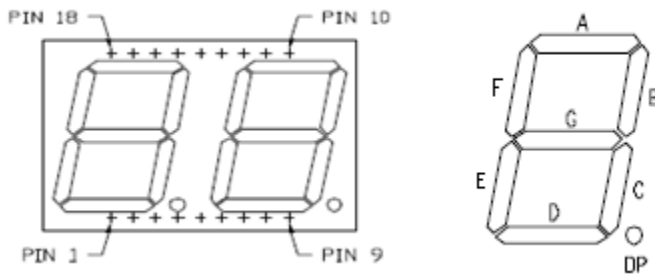
An 18f4520 PIC microcontroller is used to display decimal values on the 7 segment display. Using the data pins on the PIC we can send the information to driver data pins to display the decimal values as an output.

Hardware:

Before I explain the design process, I want to discuss about the parts that I will be using and their pin structure.

In the present market there are different parts available with same functionality, but the basic idea of functionality is same for all the parts. The parts you might be using may have different pin structure, so it would be good idea to check the datasheet to identify the right pins before you start with your designing process. In this particular example, I used a LDD-C814RI dual digit 7 segment display. Following picture shows detailed information about the pins.

a) Dual Digit 7 Segment Display [2]



PIN 16 – DIGIT 1 – A

PIN 15 – DIGIT 1 – B

PIN 3 – DIGIT 1 – C

PIN 2 – DIGIT 1 – D

PIN 1 – DIGIT 1 – E

PIN 18 – DIGIT 1 – F

PIN 17 – DIGIT 1 – G

PIN 4 – DIGIT 1 – 1st Decimal point

PIN 14 – DIGIT 1 – Ground

PIN 11 – DIGIT 2 – A

PIN 10 – DIGIT 2 – B

PIN 8 – DIGIT 2 – C

PIN 6 – DIGIT 2 – D

PIN 5 – DIGIT 2 – E

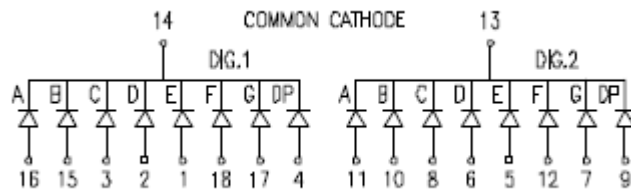
PIN 12 – DIGIT 2 – F

PIN 7 – DIGIT 2 – G

PIN 9 – DIGIT 2 – 2nd Decimal point

PIN 13 – DIGIT 2 – Ground

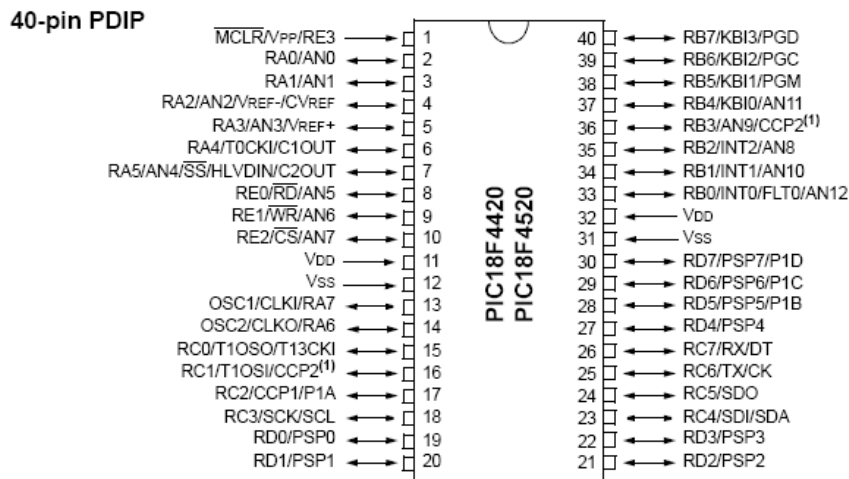
Internal structure (wiring) of the above mentioned dual 7 segment display. In other words functional diagram



For more information regarding to voltage and current measurements follow the data sheet [2]

<http://www.lumex.com/pdf/LDD-C814RI.pdf>

b) 18f4520 PIC Microcontroller.

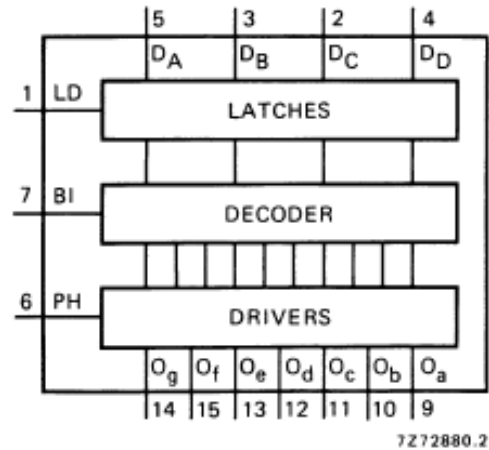
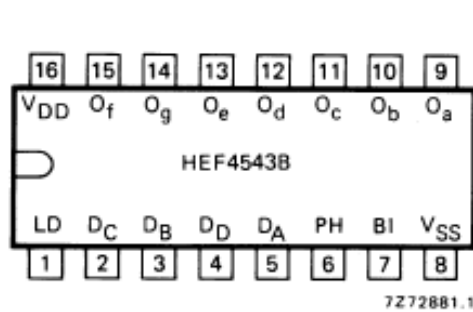


The data sheet and some basic information and sample programs can be found at the following website. This website will provide detailed information on the pin structures and also tells us how to use MPLAB software to program the PIC [3].

<http://www.egr.msu.edu/classes/ece480/goodman/ForMiniprojects/Lab3.pdf>

c) BCD to 7 segment latch/decoder/driver

The driver used in this example is HEF4543B. These driver acts as a mediator between the PIC and the display. It takes binary information from the PIC and helps to output a decimal number on the display. In other words, it converts the binary value to decimal value. Following picture shows detailed information about the pins.



Pin 1 - LD – Latch disable Input – Connected to VDD

Pin 2 – Pin 5 – Data pins – These pins will be connected to PIC to get binary information

Pin 6 – PH – Phase input – For this example we can ground this pin.

Pin 7 – BI – Blanking Input - For this example we can ground this pin.

Pin 8 – VSS – GND

Pin 9 – Pin 15 – Output data pins

Each of these pins will be connected to 7 segment display accordingly.

Pin 16 – VDD

For more information regarding to voltage and current measurements follow the data sheet [4].

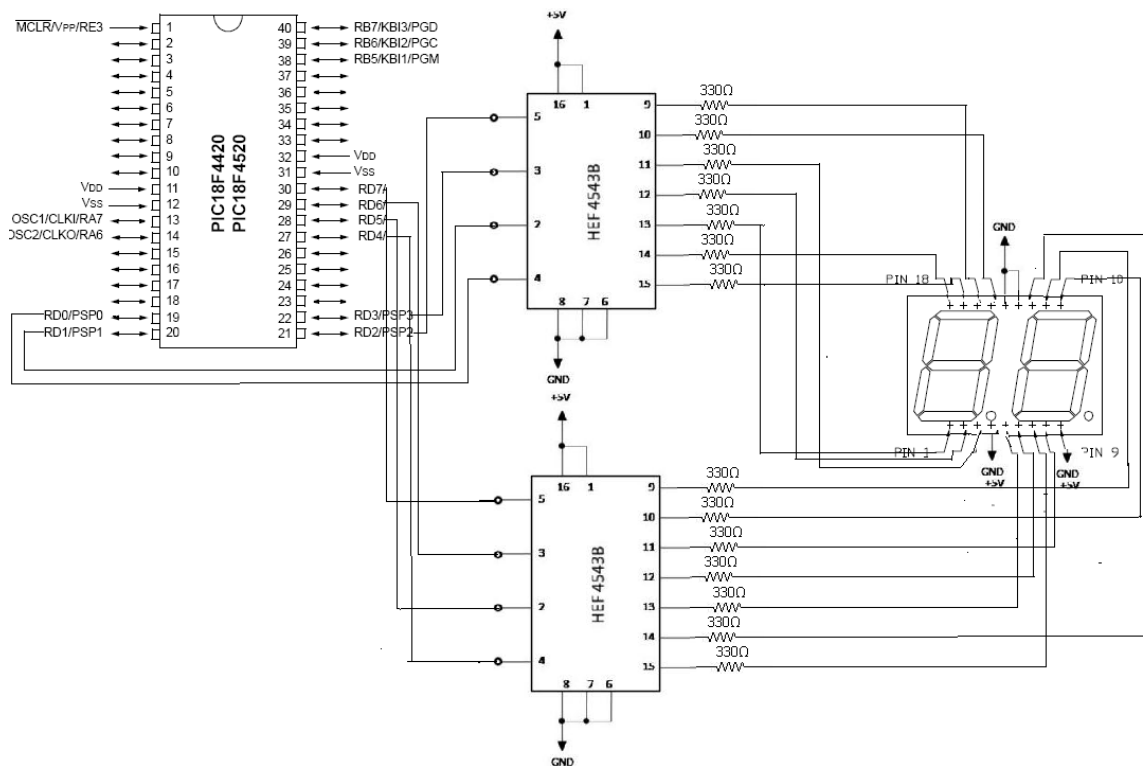
<http://www.chipcatalog.com/Datasheet/C1A7DE08130802F8600978D0DD17279B.htm>

Design process:

Use port D pins to transmit the data to drivers. We need 4 pins from port D to display one digit. In this particular example, we are displaying 2 digits, so we will be using 8 pins of port D. The PIC sends the data in a 4 bit binary form to the decoder to display the desired number in a decimal form on a 7 segment display. In order to display the decimal number on the 7 segment display, all the independent outputs are connected to the decoder appropriately. Use the 300 ohm resistors between driver and display to have a bright display. For clear understanding of wiring, look at the picture provided below. We

use the same method to display more than two digits; the only concern could be we will be using most of the I/O pins on PIC. This could be solved by using a multiplexer. In this case, PIC will output the binary number, and the multiplexer will activate the driver to display the appropriate digit on user's choice. In this example we used V_{dd} as +5v and V_{ss} is GND. It would be easy and good, if you follow the color code while doing the wiring. This helps you to figure out if there are any problems in your circuit. Once you are done building the circuit, you have to write the program for the PIC to display the numbers. You could also use a switch to control the numbers. The switch could be connected to Port B. In our 480 project we used this display for the user to input the length that could be viewed on the display to cut the material.

Wiring Diagram



Software:

This program will display the numbers upto 17 on the multi digit 7 segment display using a switch. Every time you press a switch the number will increment by 1.

```
#include <p18cxxx.h>
#include <timers.h>
#pragma config WDT=OFF
#pragma config OSC = INTIO67
long int count;
int value,ones,tenths,teen_count,loop,address;
void main()
{
    TRISA = 0x00;
    TRISB = 0xF0;
    TRISC = 0x01;
    TRISD = 0x00;
    TRISE = 0xF;
    ones = 0;
    tenths = 0;
    teen_count = 0;
    address = 0x0C;
    PORTDbits.RD0 = 0;
    PORTDbits.RD2 = 0;
    PORTA = 0x0C;
    PORTB = 0x01;
    PORTC = 0x00;
    PORTD = 0x00;
    while(1)
    {
        value = ( (ones*16) + tenths );
        PORTD = value;
        //-----Whole Inches-----
        if(PORTBbits.RB5 == 0)
        {
            //---INCREMENT VALUE BLOCK-----
            ones++;
            teen_count++;
            for(count=1;count<4000;count++);
            if(ones == 10)
            {
                ones = 0;
            }
        }
    }
}
```

```

        PORTBbits.RB3 = 1;
    }
    if(teen_count == 18)
    {
        ones = 0;
        teen_count = 0;
        PORTA = 0x06;
        PORTD = ( (ones*16) + tenths );
        PORTBbits.RB3 = 0;
    }
    //---END INCREMENT VALUE-----
//-----Tenths Inches-----
    if(PORTCbits.RC0 == 0)
    {
        //---INCREMENT VALUE BLOCK-----
        tenths++;
        for(count=1;count<4000;count++);
        if(tenths == 10)
        {
            tenths = 0;
            PORTA = 0x06;
        }
        //---END INCREMENT VALUE-----
        PORTD = ( (ones*16) + tenths );
    }
}

```

Recommendations:

Rather than using 4 data pins on the PIC for each driver. We could use a demux to send the data to all the drivers. This could take 6 pins on the PIC, 4 pins to transmit the binary data to driver to display the number and two pins to select the driver. This only disadvantage of this, programming part gets little bit complex compared to the other.

Conclusions:

7 segment displays can be used in most of the electronic instruments, residential appliances and other electric parts for displaying numerical information. This application note describes how to use a PIC and a driver to display a desired numbers on the 7 segment display

Work Cited Page

1. 7 segment display

http://en.wikipedia.org/wiki/Seven-segment_display

2. Dual Digit 7 segment display data sheet

<http://www.lumex.com/pdf/LDD-C814RI.pdf>

3. 18F4520 PIC Microcontroller

<http://www.egr.msu.edu/classes/ece480/goodman/ForMiniprojects/Lab3.pdf>

3. BCD 7 segment driver datasheet

[http://www.chipcatalog.com/Datasheet/C1A7DE08130802F8600978D0DD17279B.h
tm](http://www.chipcatalog.com/Datasheet/C1A7DE08130802F8600978D0DD17279B.htm)