

University of Massachusetts Dartmouth  
Department of Electrical and Computer Engineering

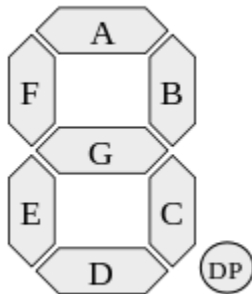
ECE 160  
ATmega Lab 3

Name: single7Seg\_main.c  
Due: see <http://ece160.org>

NOTE: The syllabus says Lab 3 is a PWM lab. PWM is being moved later in the semester.

Write a program the interfaces the buttons with a single character of the 7-segment display.

A 7-segment display is a matrix of leds than can be used to create any hex character (0-F). The segments are each labeled by a letter (A - G + dp for decimal point). The display on the dev board has a 4 character display on it.



To turn a segment on you set the pin connected to the common for that character HIGH and the pin for that segment (common across all characters) HIGH.

Segment (common across characters)	Bit
A	d0
B	d1
C	d2
D	d3
E	d4
F	d5
G	d6

Common for character	Bit
Char 0	b0
Char 1	b1
Char 2	b2
Char 3	b3

For this first seven segment lab, only one of the commons (B0) is connected to the display. In the next lab, you will learn how to enable all the digits. Remember to set all these pins as outputs before trying to write a HIGH to any of the outputs.

Problem statement:

Write a program which causes the single digit to reset (to zero), count up by one, or count down by one, depending on which button is pressed as follows:

Left Button: Count down by one; if zero is already displayed, stay at zero.

Center Button: Reset the counter to zero;

Right Button: Count up by one; if nine is already displayed, stay at nine.

Displaying numbers:

By turning on certain segments and leaving others off, any digit (as well as A, b, C, d, E, and F) may be displayed. For example, the pattern for digits 0, 1, and 2, is shown below. You need to complete the patterns for 3-9.

Digit	G	F	E	d	C	B	A
0	0	1	1	1	1	1	1
1	0	0	0	0	1	1	0
2	1	0	1	1	0	1	1

You should consider writing a ShowDigit(uint8\_t val) function. uint8\_t stands for unsigned integer 8 bit type. This function will display whatever val is on the display. For example, if val is 1, then you should write 0b00000110 to the appropriate port to display a 1.

You should also consider some sort of uint8\_t CheckButtons() function that returns a value indicating which button is pressed.

NAME: \_\_\_\_\_

TA: \_\_\_\_\_

Date: \_\_\_\_\_

Checksum: \_\_\_\_\_