

# ECEN 2350: Digital Logic

## Assignment #7

1. [5 points.] Compute the two's complement of the following binary numbers:

- (a) 0010  
(b) 1101  
(c) 0000  
(d) 0001  
(e) 1111

Handwritten solutions for two's complement:

a) 
$$\begin{array}{r} 1101 \\ + 1110 \\ \hline 1110 \end{array}$$

b) 
$$\begin{array}{r} 0010 \\ + 1111 \\ \hline 0011 \end{array}$$

c) 
$$\begin{array}{r} 1111 \\ + 0000 \\ \hline 1111 \end{array}$$

d) 
$$\begin{array}{r} 1110 \\ + 1111 \\ \hline 1111 \end{array}$$

e) 
$$\begin{array}{r} 10000 \\ + 10001 \\ \hline 10001 \end{array}$$

2. [5 points.] Interpret the following numbers as a signed (two's complement) and unsigned binary number:

	signed	unsigned
0010	2	2
1101	-3	13
1111	-1	15
0111	7	7
111101	-3	61

-32 16 8 4 2 1

3. [5 points.] Carry out the following two's complement additions and subtractions. Do not show the carry out of the MSB (i.e. your answer should be 4 bits). For the subtractions, add the two's complement of the number to be subtracted (the subtrahend) to the other one (the minuend).

Handwritten solutions for two's complement operations:

1. 
$$\begin{array}{r} 1101 \quad 13 \\ + 0011 \quad 3 \\ \hline 0000 \Rightarrow 0, \text{ overflow} \end{array}$$

2. 
$$\begin{array}{r} 01101 \quad 13 \\ + 01101 \quad 13 \\ \hline 11010 \quad 26 \end{array}$$

3. 
$$\begin{array}{r} 10110 \quad 22 \\ + 00001 \quad 1 \\ \hline 10111 \quad 23 \end{array}$$

4. 
$$\begin{array}{r} 1111 \quad 15 \\ + 0001 \quad 1 \\ \hline 0000 \Rightarrow 0, \text{ Overflow} \end{array}$$

5. 
$$\begin{array}{r} 0101 \quad 5 \\ - 0101 \quad 5 \\ \hline 0000 \Rightarrow 0 \end{array}$$

6. 
$$\begin{array}{r} 1011 \quad 11 \\ - 0010 \quad 2 \\ \hline 1001 \quad 9 \end{array}$$

7. 
$$\begin{array}{r} 00101010 \quad 42 \\ - 00111000 \quad 56 \\ \hline 11000111 \quad -14 \end{array}$$

8. 
$$\begin{array}{r} 1011 \quad 11 \\ - 0111 \quad 7 \\ \hline 0100 \quad 4 \end{array}$$

Final calculation for problem 7:

$$\begin{aligned} & -128 + 64 + 32 + 16 + 2 \\ & = -128 + 114 \\ & = -14 \end{aligned}$$

Check your results by converting the operations to (signed-magnitude) base-10. Note if overflow occurs.

4. [5 points.] Write a verilog module that performs the two's complement on a 4-bit input. You should use the following prototype:

```
module negate(input [3:0] in, output [3:0] out);
```

Turn in your verilog module as `negate.txt`.

5. [5 points.] Using the Boolean Board, display the 4 right-most switches on the 7-segment hex display as a **signed** number. If the number is a negative number, you should display a negative sign (dash) on the left 7-segment display.

You may modify your solution from Lab 6, or use the solutions posted to Canvas.

For instance, if the switches were set to 1101, you should display this as -3:



A simple check to see if all is working is that all 1s (1111) should be -1:



Meanwhile, positive (signed) numbers should show their positive representation. E.g. 0101 should display 5:



Turn in your top-level Verilog file as `lab7-q5.txt`.