# EE 231 – Homework 6
## Due October 7, 2009

1. Problem 4.16

2. Using a decoder and external gates, design the combinational circuit defined by the following three Boolean functions:

   (a) $F_1 = xyz + x'z'$
   $F_2 = x'yz + xy'$
   $F_3 = xyz' + xy$

   (b) $F_1 = (x' + y)z$
   $F_2 = yz + x'y + y'z$
   $F_3 = (x + y')z$

3. Implement the following Boolean functions with a multiplexer:

   (a) $F(w, x, y, z) = \Sigma(1, 3, 6, 8, 10, 15)$

   (b) $F(w, x, y, z) = \Pi(4, 11, 12)$

4. Write a Verilog dataflow description to implement the Boolean functions of Problem 3.

5. Implement a full subtractor with two 4x1 multiplexers. Note: the truth table for the full subtractor is:

| $x$ | $y$ | $B_{in}$ | $B_{out}$ | $Diff$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

6. An 8x1 multiplexer has inputs $A$, $B$ and $C$ connected to the selection inputs $S_2$, $S_1$, and $S_0$ respectively. The data inputs through $I_0$ through $I_7$ are as follows:

   (a) $I_1 = I_3 = I_6 = 0; I_2 = I_5 = 1; I_0 = I_7 = D$; and $I_4 = D'$.

   (b) $I_2 = I_4 = 0; I_3 = I_6 = I_7 = 1; I_0 = I_5 = D$; and $I_1 = D'$.

   Determine the Boolean function that the multiplexer implements.

7. Problem 4.38. Use a dataflow description to implement the truth table of Figure 4.26. Do not write a gate level description.

8. Problem 4.43.

9. Using a case statement, write an HDL behavioral description of an eight-bit arithmetic-logic unit (ALU). The ALU needs to implement the 10 functions listed below. The inputs are two eight-bit numbers $A$ and $B$, and select inputs $S$ (where $S$ has enough bits to select the ten functions). The outputs are the eight-bit result $R$, a zero-bit $Z$, and a carry bit $C$. The $C$ bit is described in the table below. (X means Don't Care.) The zero bit $Z$ is 1 if all the bits of the eight-bit result are 0, and is 0 otherwise.

| Name | Description | R | C | Z |
|---|---|---|---|---|
| LOAD | Load input A | $A$ | X | 1 if $R == 0$ |
| ADDA | Add inputs | $A + B$ | Carry | 1 if $R == 0$ |
| SUBA | Subtract inputs | $A - B$ | Borrow | 1 if $R == 0$ |
| ANDA | AND inputs | $A\&B$ | X | 1 if $R == 0$ |
| ORAA | OR inputs | $A\|B$ | X | 1 if $R == 0$ |
| COMA | Bitwise Complement input A | $\sim A$ | 1 | 1 if $R == 0$ |
| INCA | Increment input A | $A + 1$ | X | 1 if $R == 0$ |
| LSRA | Logical Shift Right input A | $0 => R[7]$<br>$A[7:1] => R[6:0]$ | $A[0]$ | 1 if $R == 0$ |
| LSLA | Logical Shift Left input A | $0 => R[0]$<br>$A[6:0] => R[7:1]$ | $A[7]$ | 1 if $R == 0$ |
| ASRA | Arithmetic Shift Right input A | $A[7] => R[7]$<br>$A[7:1] => R[6:0]$ | $A[0]$ | 1 if $R == 0$ |