Solutions - Midterm Exam

(February 15th @ 5:30 pm)

Presentation and clarity are very important! Show your procedure!

Problem 1 (20 pts)

a) Complete the following table. The decimal numbers are unsigned: (5 pts.)

Decimal	BCD	Binary	Reflective Gray Code
51	01010001	110011	101010
98	10011000	1100010	1010011
576	010101110110	1001000000	1101100000

b) Complete the following table. Use the fewest number of bits in each case: (12 pts.)

	REPRESENTATION		
Decimal	**Sign-and-magnitude**	**1's complement**	**2's complement**
-38	1100110	1011001	1011010
0	00	11111	0
-32	1100000	1011111	100000
69	01000101	01000101	01000101
-64	11000000	10111111	1000000
-24	111000	100111	101000

c) Convert the following decimal numbers to their 2's complement representations. (3 pts.)

- ✓ -16.3125
  +16.3125 = 010000.0101
  ⇒ -16.3125 = 101111.1011
- ✓ 18.375
  +18.375 = 010010.011

Problem 2 (11 pts)

- The figure below depicts the entire memory space of a microprocessor. Each memory address occupies one byte. 1KB = 2^{10} bytes, 1MB = 2^{20} bytes, 1GB = 2^{30} bytes
  - ✓ What is the size (in bytes, KB, or MB) of the memory space? What is the address bus size of the microprocessor? (2 pts.)

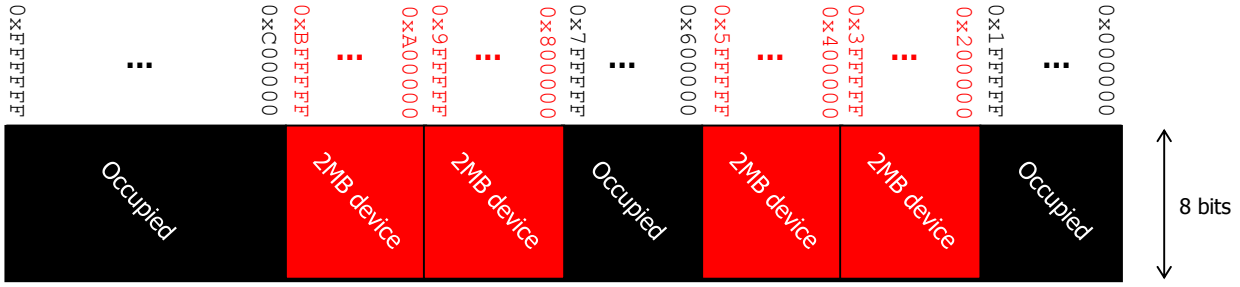
    Address space: 0x000000 to 0xFFFFFF. To represent all these addresses, we require 24 bits. So, the address bus size of the microprocessor is 24 bits. The size of the memory space is 2^{24} = 16 MB.

  - ✓ If we have a memory chip of 2 MB, how many bits do we require to address those 2 MB of memory? (1 pt.)

    2 MB = 2^{21} bytes. Thus, we require 21 bits to address the memory device.

  - ✓ We want to connect the 2 MB memory chip to the microprocessor. For optimal implementation, we must place those 2 MB in an address range where every single address shares some MSBs. Provide a list of all the possible address ranges that the 2 MB chip can occupy. You can only use the non-occupied portions of the memory space as shown below.

    - 0x200000 to 0x3FFFFF
    - 0x400000 to 0x5FFFFF
    - 0x800000 to 0x9FFFFF
    - 0xA00000 to 0xBFFFFF

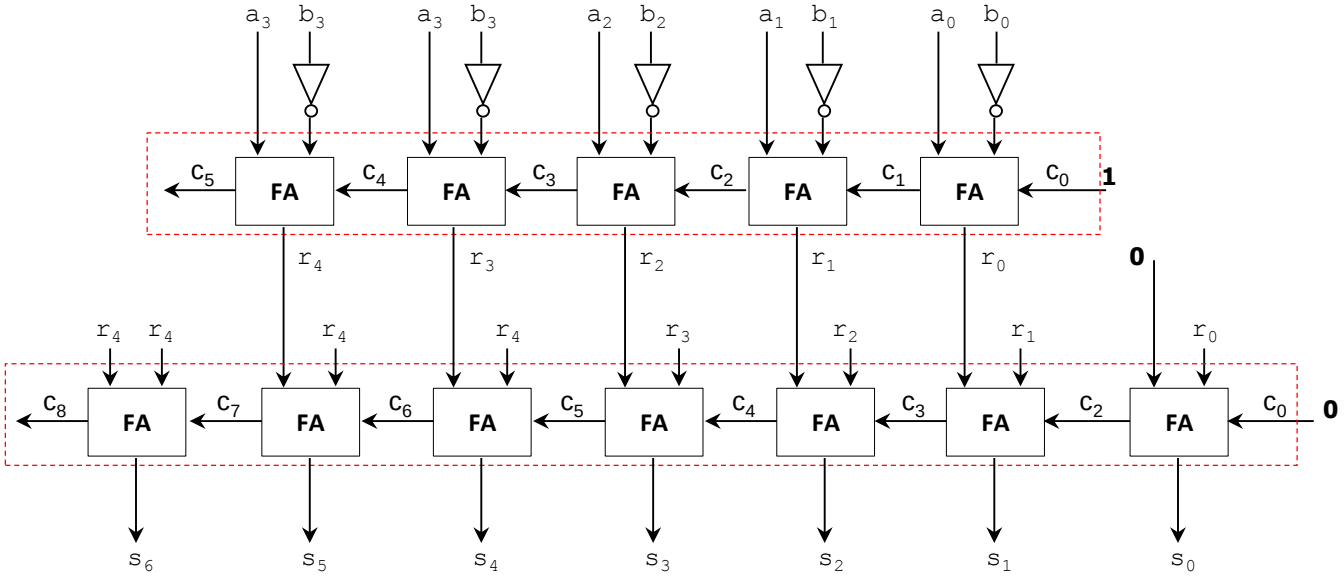


PROBLEM 3 (10 PTS)

- Given two 4-bit signed (2's complement) numbers A, B, sketch the circuit that computes $(A - B) \times 3$. You can only adder units (or full adders if you prefer) and logic gates. Make sure your circuit avoids overflow.

$$(A - B) \times 3 = (A - B) \times 2 + (A - B)$$

Worst case: $15 \times 3 = 45$. 45 requires 7 bits, thus, we need to sign extend the operand $(A - B) \times 2$ on the last addition.



PROBLEM 4 (17 PTS)

a) Perform the following additions and subtractions of the following unsigned integers. Use the fewest number of bits n to represent both operators. Indicate every carry (or borrow) from c_0 to c_n (or b_0 to b_n). For the addition, determine whether there is an overflow. For the subtraction, determine whether we need to keep borrowing from a higher byte. (6 pts.)

✓ 29 - 51

Borrow out! → b_6=1 b_5=0 b_4=0 b_3=0 b_2=1 b_1=0 b_0=0

```
29 = 0x1D =  0 1 1 1 0 1 -
51 = 0x33 =  1 1 0 0 1 1
-------------------------
             1 0 1 0 1 0
```

✓ 41 + 37

c_6=1 c_5=0 c_4=0 c_3=0 c_2=0 c_1=1 c_0=0

```
41 = 0x29 =  1 0 1 0 0 1 +
37 = 0x25 =  1 0 0 1 0 1
-------------------------
Overflow! → 1 0 0 1 1 1 0
```

b) Perform the following operations, where numbers are represented in 2's complement. Indicate every carry from c_0 to c_n. For each case, use the fewest number of bits to represent the summands and the result so that overflow is avoided. (8 pts.)

✓ 62 - 79

n = 8 bits c_8=0 c_7=0 c_6=1 c_5=1 c_4=0 c_3=0 c_2=0 c_1=0 c_0=0

$c_8 \oplus c_7$=0 No Overflow

```
 62 = 0 0 1 1 1 1 1 0 +
-79 = 1 0 1 1 0 0 0 1
---------------------
-17 = 1 1 1 0 1 1 1 1
```

-62 + 79 = -17 ∈ $[-2^7, 2^7-1]$ → no overflow

✓ -53 - 26

n = 7 bits c_7=1 c_6=0 c_5=0 c_4=1 c_3=1 c_2=1 c_1=0 c_0=0

$c_7 \oplus c_6$=1 Overflow!

```
-53 = 1 0 0 1 0 1 1 +
-26 = 1 1 0 0 1 1 0
-------------------
      0 1 1 0 0 0 1
```

-53 -26 = -79 ∉ $[-2^6, 2^6-1]$ → overflow!

To avoid overflow: n = 8 bits (sign-extension)

$c_8 \oplus c_7$=0 No Overflow c_8=1 c_7=1 c_6=0 c_5=0 c_4=1 c_3=1 c_2=1 c_1=0 c_0=0

```
-53 = 1 1 0 0 1 0 1 1 +
-26 = 1 1 1 0 0 1 1 0
---------------------
      1 0 1 1 0 0 0 1
```

-53 -26 = -79 ∈ $[-2^7, 2^7-1]$ → no overflow

c) Get the multiplication result of the following numbers that are represented in 2's complement arithmetic with 4 bits. (3 pts.)

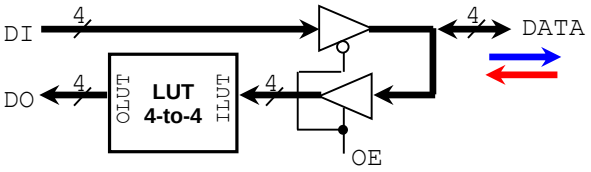
✓ -5 x 7

```
1 0 1 1 x        0 1 0 1 x
0 1 1 1          0 1 1 1
                 -------
                   0 1 0 1
                 0 1 0 1
               0 1 0 1
             0 0 0 0
             -------------
             0 1 0 0 0 1 1
                   ↓
             1 0 1 1 1 0 1
```

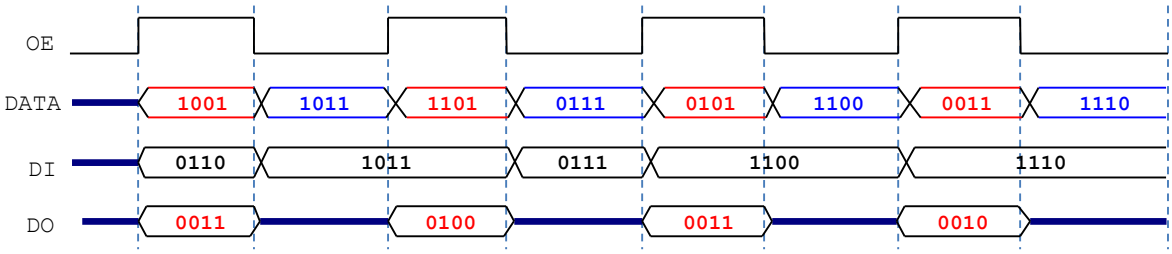
PROBLEM 5 (10 PTS)

- Given the following circuit, complete the timing diagram (signals DO and $DATA$).
  The LUT 4-to-4 implements the following function: $OLUT = \lceil sqrt(ILUT) \rceil$. For example: $ILUT = 1100 \rightarrow OLUT = 0100$

Input data to LUT is treated as an unsigned number.




```
ceil(sqrt(9))= 3
ceil(sqrt(13))= 4
ceil(sqrt(5)) = 3
ceil(sqrt(3))= 2
```

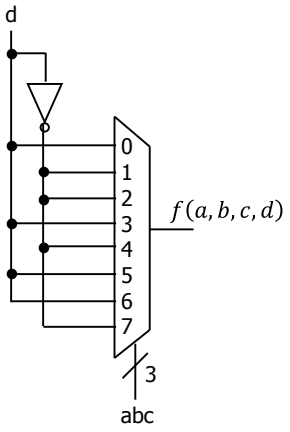


PROBLEM 6 (17 PTS)

- Sketch the circuit that implements the following Boolean function: $f = a \oplus b \oplus c \oplus d$
  Recall that $a \oplus b \oplus c \oplus d = (a \oplus b) \oplus (c \oplus d)$

✓ Using ONLY an 8-to-1 MUX and 'NOT' gates. (3 pts.)

a	b	c	d	f
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0



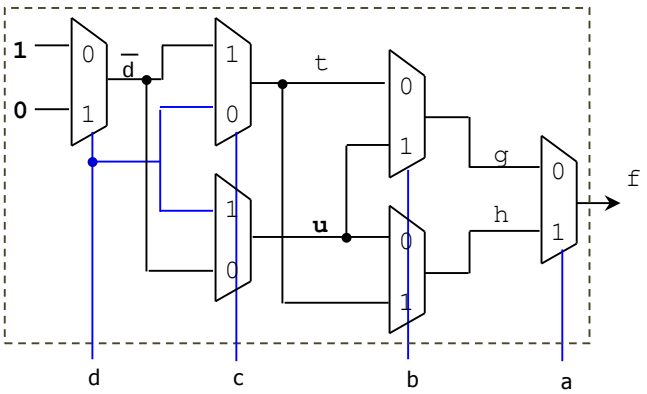
✓ Implement the previous circuit using ONLY 2-to-1 MUXs (AND, OR, NOT, XOR gates are not allowed). (14 pts.)

$f(a,b,c,d) = \bar{a}f(0,b,c,d) + af(1,b,c,d) = \bar{a}\left(b \oplus (c \oplus d)\right) + a\left(\overline{b \oplus (c \oplus d)}\right) = \bar{a}g(b,c,d) + ah(b,c,d)$

$g(b,c,d) = \bar{b}g(0,c,d) + bf(1,c,d) = \bar{b}(c \oplus d) + b\left(\overline{c \oplus d}\right)$
$h(b,c,d) = \bar{b}h(0,c,d) + bh(1,c,d) = \bar{b}\left(\overline{c \oplus d}\right) + b(c \oplus d)$

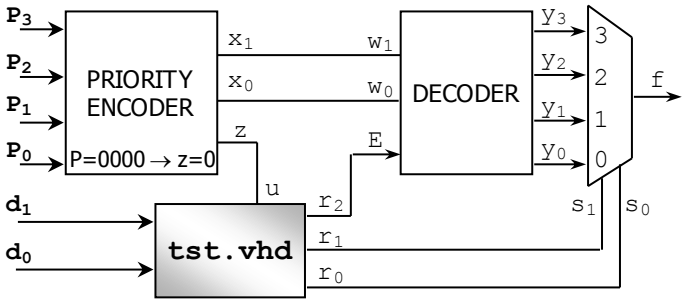
$t(c,d) = c \oplus d = \bar{c}t(0,d) + ct(1,d) = \bar{c}(d) + c(\bar{d})$
$u(c,d) = \overline{c \oplus d} = \bar{c}u(0,d) + cu(1,d) = \bar{c}(\bar{d}) + c(d)$

Also: $\bar{d} = \bar{d}(1) + d(0)$



PROBLEM 7 (15 PTS)

- Complete the timing diagram of the following circuit. The VHDL code (`tst.vhd`) corresponds to the shaded circuit.
  $d = d_1d_0$, $w = w_1w_0$, $r = r_2r_1r_0$, $y = y_3y_2y_1y_0$




```
library ieee;
use ieee.std_logic_1164.all;
entity tst is
  port (d: in std_logic_vector(1 downto 0);
        r: out std_logic_vector(2 downto 0);
        u: in std_logic);
end tst;
```

```
architecture bhv of tst is

begin

  process (d, u)
  begin

      r <= d&'0';
      if u = '1' then
         r <= '1'&d;
      end if;

  end process;

end bhv;
```