Solutions - Midterm Exam

(February 14th @ 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
42	01000010	101010	111111
54	01010100	110110	101101
169	000101101001	10101001	11111101

b) Complete the following table. The decimal numbers are signed. Use the fewest number of bits in each case: (12 pts.)

	REPRESENTATION		
Decimal	Sign-and-magnitude	1's complement	2's complement
-21	110101	101010	101011
-32	1100000	1011111	100000
45	0101101	0101101	0101101
-64	11000000	10111111	1000000
-24	111000	100111	101000
-38	1100110	1011001	1011010

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

✓ -19.375

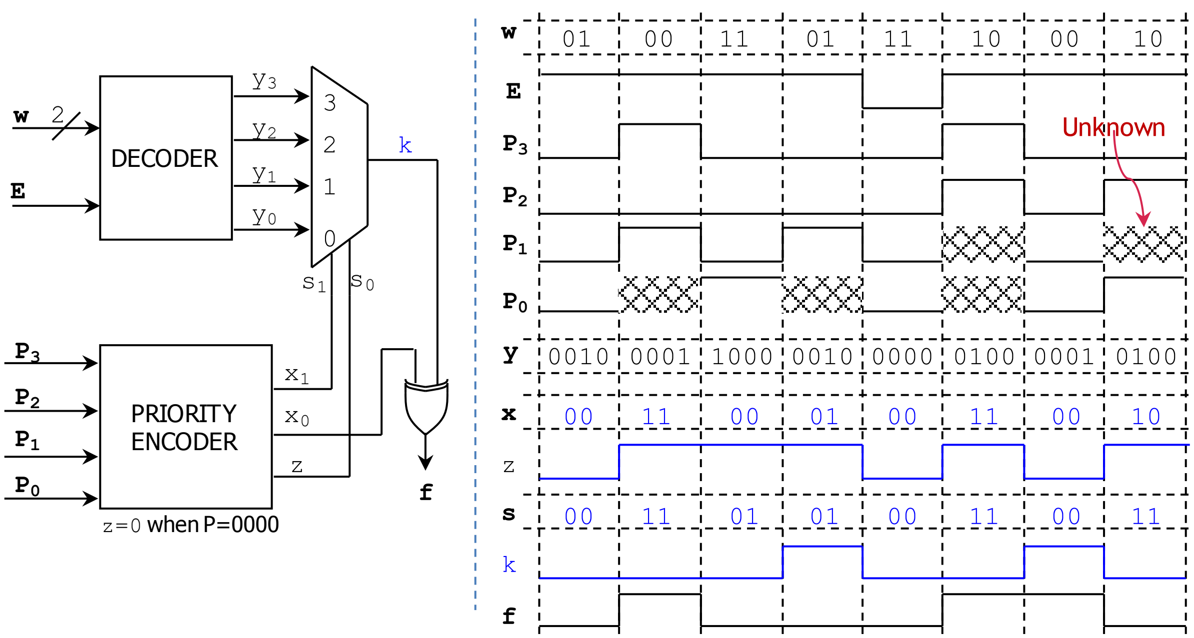
+19.375 = 010011.011 ⇒ -19.375 = 101100.101

✓ 16.125

16.125 = 010000.001

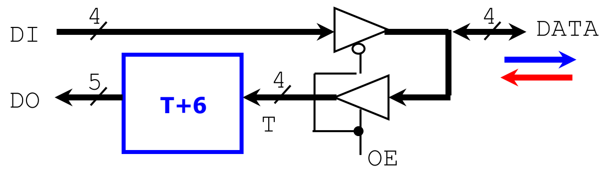
PROBLEM 2 (15 PTS)

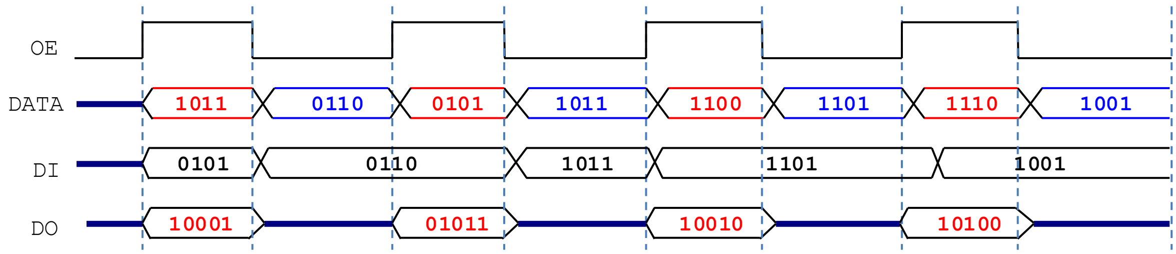
- Complete the timing diagram of the circuit shown below. $y = y_3y_2y_1y_0$, $x = x_1x_0$, $s = s_1s_0$



PROBLEM 3 (8 PTS)

- Complete the timing diagram (signals DO and $DATA$) of the following circuit. The circuit in the blue box computes the summation T+6, with the result having 5 bits. T is an unsigned 4-bit number.





PROBLEM 4 (12 PTS)

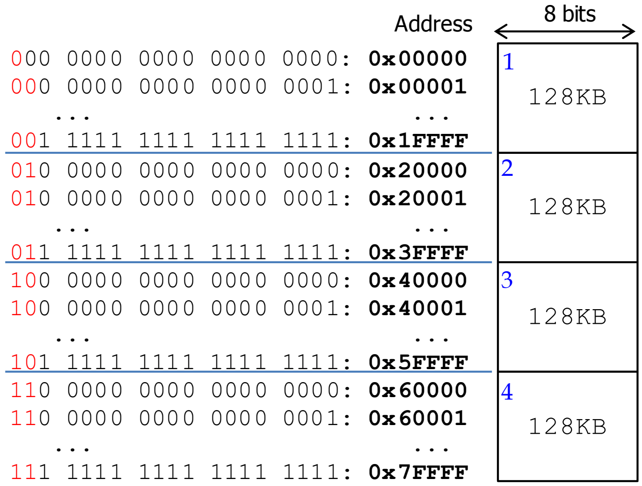
- A microprocessor has a memory space of 512 KB. Each memory address occupies one byte. 1KB = 2^{10} bytes, 1MB = 2^{20} bytes, 1GB = 2^{30} bytes. We want to connect four 128 KB memory chips to this microprocessor.
  - ✓ What is the address bus size (number of bits of the address) of the microprocessor? (1 pt).

    Size of memory space: 512 KB = 2^{19} bytes. Thus, we require 19 bits to address the memory space.

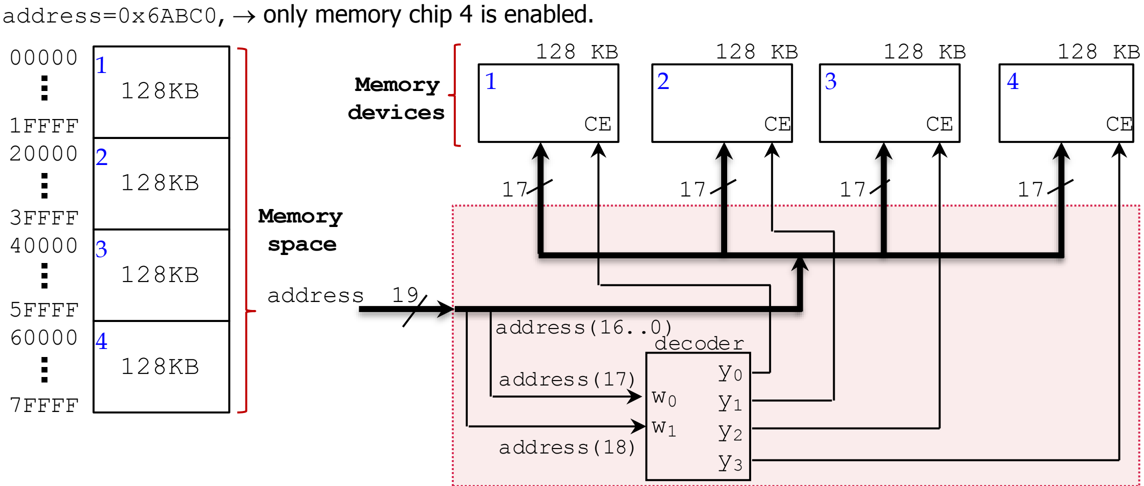
  - ✓ For a memory chip of 128 KB, how many bits do we require to address 128 KB of memory? (1 pt).

    128 KB memory device: 128 KB = 2^{17} bytes. Thus, we require 17 bits to address the memory device.

  - ✓ Complete the address ranges (lowest to highest, in hexadecimal) for each of the memory chips in the figure. (4 pts).



  - ✓ Sketch the circuit that: i) addresses the memory chips, and ii) enables only one memory chip (via CE: chip enable) when the address falls in the corresponding range. Example: if `address=0x2FFFF`, → only memory chip 2 is enabled (CE=1). If `address=0x6ABC0`, → only memory chip 4 is enabled.



PROBLEM 5 (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)

✓ 49 + 18

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

49 = 0x31 = 1 1 0 0 0 1 +
18 = 0x12 = 0 1 0 0 1 0

Overflow! → 1 0 0 0 0 1 1

✓ 38 − 42

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

38 = 0x26 = 1 0 0 1 1 0 −
42 = 0x2A = 1 0 1 0 1 0

1 1 1 1 0 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)

✓ -37 + 50

n = 7 bits

$c_7=1$ $c_6=1$ $c_5=1$ $c_4=0$ $c_3=0$ $c_2=1$ $c_1=0$ $c_0=0$

$c_7 \oplus c_6=0$
No Overflow

-37 = 1 0 1 1 0 1 1 +
50 = 0 1 1 0 0 1 0

13 = 0 0 0 1 1 0 1

-37 + 50 = 13 $\in [-2^6, 2^6-1]$ → no overflow

✓ -26 − 40

n = 7 bits

$c_7=1$ $c_6=0$ $c_5=0$ $c_4=0$ $c_3=0$ $c_2=0$ $c_1=0$ $c_0=0$

$c_7 \oplus c_6=1$
Overflow!

-40 = 1 0 1 1 0 0 0 +
-26 = 1 1 0 0 1 1 0

0 1 1 1 1 1 0

-40 -26 = -66 $\notin [-2^6, 2^6-1]$ → overflow!

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

$c_8 \oplus c_7=0$
No Overflow

$c_8=1$ $c_7=1$ $c_6=0$ $c_5=0$ $c_4=0$ $c_3=0$ $c_2=0$ $c_1=0$ $c_0=0$

-40 = 1 1 0 1 1 0 0 0 +
-26 = 1 1 1 0 0 1 1 0

1 0 1 1 1 1 1 0

-40 -26 = -66 $\in [-2^7, 2^7-1]$ → no overflow

c) Perform binary multiplication of the following numbers that are represented in 2's complement arithmetic. (3 pts)

✓ -6 x 9

1 0 1 0 x
0 1 0 0 1

➡

1 0 0 1 x
0 1 1 0

0 0 0 0
1 0 0 1
1 0 0 1
0 0 0 0

0 1 1 0 1 1 0

⬇

1 0 0 1 0 1 0

PROBLEM 6 (10 PTS)

- Given two 4-bit <u>signed</u> numbers A, B, sketch the circuit that computes $|A - 2B|$. For example: $A = 1010, B = 0110 \rightarrow |A - 2B| = |-6 - 2 \times 6| = 18$. You can only use full adders and logic gates. Your circuit must avoid overflow: design your circuit so that the result and intermediate operations have the proper number of bits.

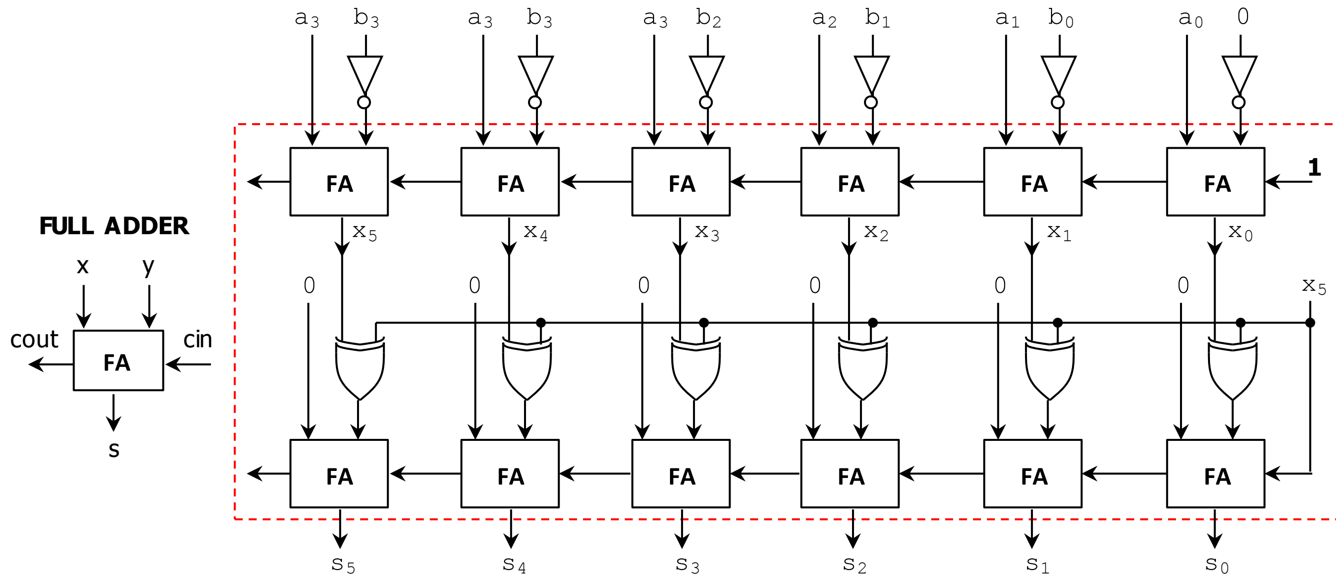
$A = a_3a_2a_1a_0$, $B = b_3b_2b_1b_0$: signed numbers

$A, B \in [-8,7] \rightarrow 2B \in [-16,14]$ requires 5 bits in 2C. $2B = b_3b_2b_1b_00$

✓ $X = A - 2B \in [-22,23]$ requires 6 bits in 2C. Thus, the operation $A - 2B$ requires 6 bits (we sign-extend A and $2B$).

$$A - 2B = a_3a_3a_3a_2a_1a_0 - b_3b_3b_2b_1b_00$$

✓ $|X| = |A - 2B| \in [0,23]$ requires 6 bits in 2C. Thus, the second operation $0 \pm X$ only requires 6 bits.
  - If $x_5 = 1 \rightarrow X < 0 \rightarrow$ we do $0 - X$.
  - If $x_5 = 0 \rightarrow X \geq 0 \rightarrow$ we do $0 + X$.



PROBLEM 7 (18 PTS)

- Sketch the circuit that implements the following Boolean function: $f(a,b,c,d) = \left(\overline{a\oplus b}\right)\left(\overline{c\oplus d}\right)$
  - ✓ Using ONLY 2-to-1 MUXs (AND, OR, NOT, XOR gates are not allowed). (12 pts)

$f(a,b,c,d) = \left(\overline{a\oplus b}\right)\left(\overline{c\oplus d}\right)$

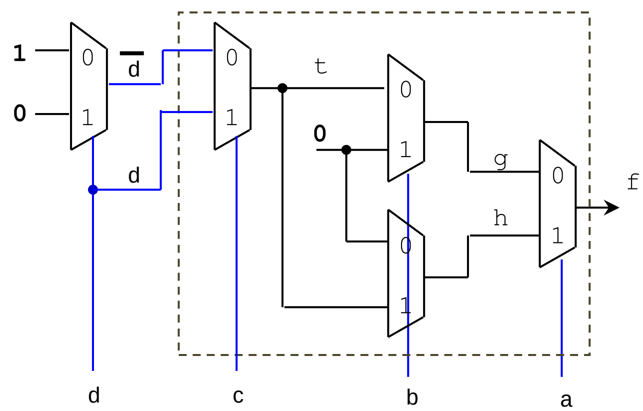
$f = \bar{a}f(0,b,c,d) + af(1,b,c,d) = \bar{a}\left(\bar{b}\left(\overline{c\oplus d}\right)\right) + a\left(b\left(\overline{c\oplus d}\right)\right) = \bar{a}g(b,c,d) + ah(b,c,d)$

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

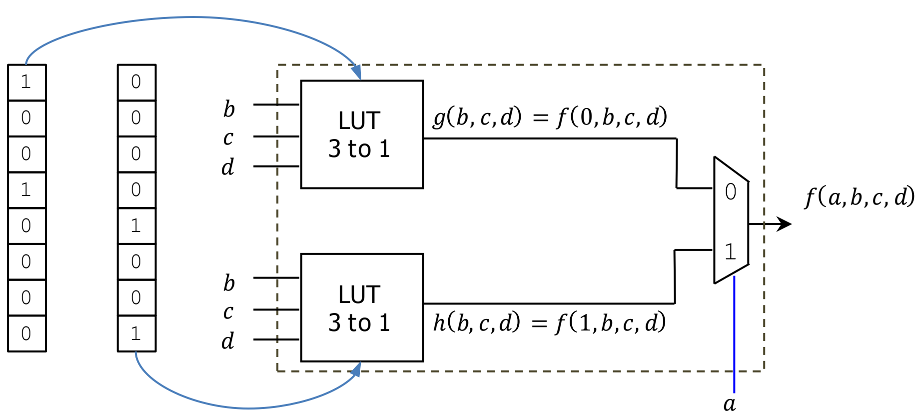
$h(b,c,d) = \bar{b}(0) + b\left(\overline{c\oplus d}\right)$

$t(c,d) = \left(\overline{c\oplus d}\right) = \bar{c}\left(\bar{d}\right) + c(d)$

Also: $\bar{d} = \bar{d}(1) + d0$



  - ✓ Using two 3-to-1 LUTs and a 2-to-1 MUX. Specify the contents of each of the 3-to-1 LUTs. (6 pts)



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