Solution CS 2115 Midterm 2017

Computer Organization (City University of Hong Kong)

Student Name: SOLUTION
Student ID:

CS 2115 Midterm (100 marks)

1. Convert the decimal number 11.09375 to binary, octal and hexadecimal numbers. [10 marks]

Solution:

$(11.09375)_{10} = (1011.\ 00011)_2 = (13.06)_8 = (B.18)_{16}$

[Notice: 0.09375 (not 0.9375)]

2. Suppose you have a thirty-two bit computer. Convert the negative decimal number -30 to two's complement representation on this computer. [10 marks]

Solution:

To convert the decimal number -30 into a representation of the 2's complement, we first take the binary representation of 30 (written out in thirty-two bits):

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

Invert the digits.

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

And add one.

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

3. Convert the decimal number $(-36.789)_{10}$ to
   a. single binary floating point numbers, [10 marks]
   b. double binary floating point numbers. [10 marks]

   (Note: For fraction part, when converting to binary, calculate up to 5 digits)

Solution:

$(-36.789)_{10} \approx (-100100.11001)_2$
$(5)_{10} = (101)_2$

1	1000 0100	0010 0110 0100… (**23**-12=11)
1	100 0000 0100	0010 0110 0100… (**52**-12=40)

[Notice: 2^(k-1) – 1 + 5]

4. The place values in a base 3 *balanced ternary number system* are powers of 3, namely 1, 3, 9, 27, 81 and so on. Suppose the digits in this balanced ternary number are `1`, `0` and `T`, where `T` is the symbol to mean negative one. For example, the base 10 number 35 is written as `110T` in this base 3 system. Show your workings for the following ternary number addition:

```
    0   1   1
+   0   0   1
----------
```

[10 marks]

Solution:

```
 4  = 0  1  1
+ 1 = 0  0  1
---   -------
      0  1  2 <-- decimal sum of each column
      0  1 1T <-- ternary sum of each column
      0 1+1 T <-- carry: move the extra digit to the next column
      0 1T  T <-- add; but we have to carry again!
      1  T  T <-- final answer, 5
```

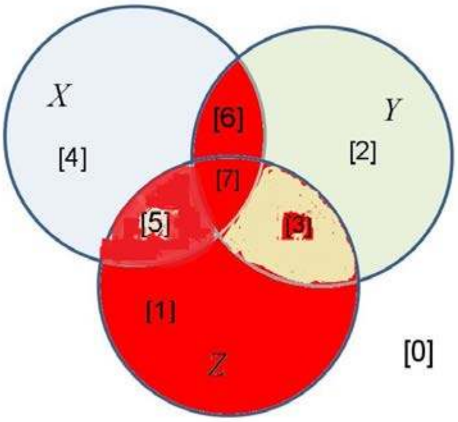
Observe that in order to have only a single digit in each "ternary place," we have to rewrite the two-digit sums so that the digit is added to the column on the left.

5. Use a *Venn diagram* to prove the *Consensus Theorem:*

$$xy + \overline{y}z + xz = xy + \overline{y}z$$

[20 marks]

Solution:



The Venn diagram can be used to visualize boolean values as sets. A point in the Venn diagram means either be a member of a certain set or not.

Three boolean variables lead to eight different areas in the Venn diagram above. They are also illustrated as the eight corresponding cases as shown in the following truth table:

#	x	y	z	$xy \vee \bar{y}z \vee xz$	$xy \vee \bar{y}z$
[0]	0	0	0		
[1]	0	0	1	1	1
[2]	0	1	0		
[3]	0	1	1		
[4]	1	0	0		
[5]	1	0	1	1	1
[6]	1	1	0	1	1
[7]	1	1	1	1	1

In the Venn diagram xy is depicted by the overlap area of circle x and circle y. Both have to be *true* to get a *true* value for xy. That corresponds to [6] and [7] in the table. The second element (y)¯z ([1] and [5]) stands for the area not in circle y but in circle z. Lastly, the third element xz ([5] and [7]) is shown as the area overlapped by circle y and by circle z.

You can see from the diagram that the Consensus Theorem holds: Right-hand-side and left-hand-side are in fact equivalent. The xz (= terms [5] + [7]) is covered by xy∨(y)¯z (= terms [1]+[5]+[6]+[7]) and thus does not add anything to the left-hand-side.

6. [CS2115 2013/2014 Final Exam Question] Design a digital circuit that converts from the natural binary code to the Gray code. You should use a Karnaugh Map (K-map) to minimize the Boolean logic function in your digital circuit. [30 marks]

Binary	Gray Code
000	000
001	001
010	011
011	010
100	110
101	111
110	101
111	100

Solution: Let A, B and C be left-most bit, center bit and right-most bit of the natural binary code respectively.
And let X, Y and Z be left-most bit, center bit and right-most bit of the Gray code respectively.

X = A

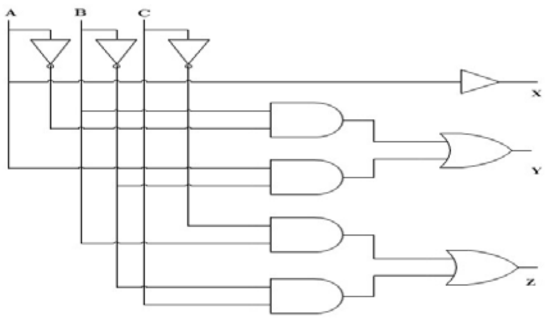
X=A\BC	00	01	11	10
0	0	0	0	0
1	1	1	1	1

- **Y =** $\overline{A}B + A\overline{B}$

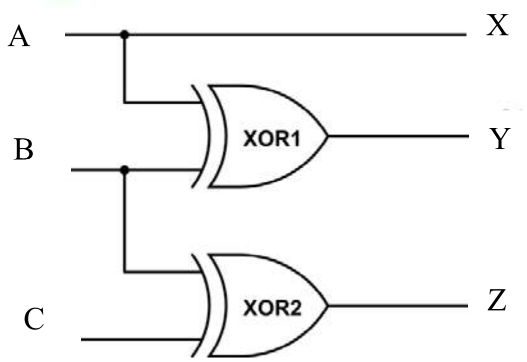
Y=A\BC	00	01	11	10
0	0	0	1	1
1	1	1	0	0

- **Z =** $\overline{B}C + B\overline{C}$

Z=A\BC	00	01	11	10
0	0	1	0	1
1	0	1	0	1



OR



[Notice: the Binary to Gray Code circuit is an interconnection of XOR gates]

- END -