Name: KEY **Id#**

COE 202, Term 052
Fundamentals of Computer Engineering

Quiz# 1

Date: Monday, Feb. 27

Q1. Represent the following numbers in **binary** and **hexadecimal**. Use as many bits as needed, and approximate the fraction to **4 <u>binary</u> digits**:

i. $(200.6)_{10}$

$= (C8.9)_{16}$
$=(11001000.1001)_2$

ii. $(54.5)_8$

$=(101100.101)_2$
$= (2C.A)_{16}$

Q2. Assume that an 8-bit register contains the following number 10111000. Determine the content of the register assuming that it represents:

i. an unsigned number.

$= 128+32+16+8=184$
OR $= 255 – (64+7) = 255 -71 = 184$

ii. a 2's complement signed number.

2's complement of 10111000 = 01001000 = 64+8=72
⇨ 10111000 represents -72

Q3. Perform the following operations assuming that numbers are represented using **8-bits** assuming **r's complement** representation, and determine if there is an **overflow** or not:

i. $(11101110)_2 + (11110111)_2$

$$\begin{array}{ll} & 11101110 \\ + & 11110111 \\ \hline & 11100101 \end{array} \qquad \text{Cout} = 1$$

There is no overflow since we are adding two negative numbers and we got a negative number. Also Cin $\oplus$ Cout = 0.

ii. $(4C)_{16}$ - $(D0)_{16}$

$= (4C)_{16}$ + 16's complement of $(D0)_{16}$
$= (4C)_{16} + (30)_{16} = (7C)_{16}$

There is no overflow since we are adding two positive numbers and we got a positive number.

Q4. Determine, in **binary** and **decimal**, the *smallest (negative)* number and the *largest (positive)* number that can be stored using the 2`s complement notation, assuming ***10-bit*** *representation*.

	Binary	**Decinal**
Smallest (negative)	10 0000 0000	$= -2^9 = -512$
Largest (Positive)	01 1111 1111	$= +(2^9-1) = +511$