

## EE 208: Tutorial#1

$$1-(1A2.4)_{16}=(?)_{10} \Rightarrow 1 \times 16^2 + 10 \times 16^1 + 2 \times 16^0 + 4 \times 16^{-1} = (418.25)_{10}$$

$$2-(1011)_2=(?)_{10} \Rightarrow 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (11)_{10}$$

$$3-(653.52)_{10}=(?)_8$$

4-Convert  $(418.25)_{10}$  to Binary then to Octal and Hex

5-Convert  $(E8B)_{16}$  to Binary and Octal

6-Convert  $(542)_8$  to Binary then to Hex

7-1 1 0 0 1 1 - 0 1 1 1 0 1



8-  $110 \times 101$  ,  $(AB)_{16} \times (2F)_{16}$

9- 30-10, 10-30 using signed binary

Overflow & Underflow:

10-  $70+80$ ,  $-70-80$  8 bit signed number

11- BCD

$$\begin{array}{rcl}
 (694)_{10} & = & (0110\ 1001\ 0100)_{\text{BCD}} \\
 (835)_{10} & = & (1000\ 0011\ 0101)_{\text{BCD}} \\
 & & \begin{array}{c} 1 \leftarrow \\ \begin{array}{rcl}
 0110 & 1001 & 0100 \\
 \underline{+1000} & \underline{+0011} & \underline{+0101} \\
 1111 & 1100 & 1001 \\
 \underline{+0110} & \underline{+0110} & \underline{+0000} \\
 0001\ 0101 & 1\ 0010 & 1001
 \end{array}
 \end{array}
 \end{array}$$



**1.30** The following is a string of ASCII characters whose bit patterns have been converted into hexadecimal for compactness: 73 F4 E5 76 E5 4A EF 62 73. Of the eight bits in each pair of digits, the leftmost is a parity bit. The remaining bits are the ASCII code.

(a) Convert the string to bit form and decode the ASCII.

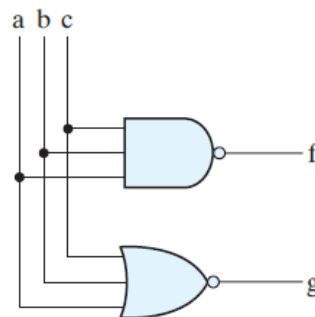
(b) Determine the parity used: odd or even?

73 F4 E5 76 E5 4A EF 62 73

73: 0\_111\_0011 s  
 F4: 1\_111\_0100 t  
 E5: 1\_110\_0101 e  
 76: 0\_111\_0110 v  
 E5: 1\_110\_0101 e  
 4A: 0\_100\_1010 j  
 EF: 1\_110\_1111 o  
 62: 0\_110\_0010 b  
 73: 0\_111\_0011 s

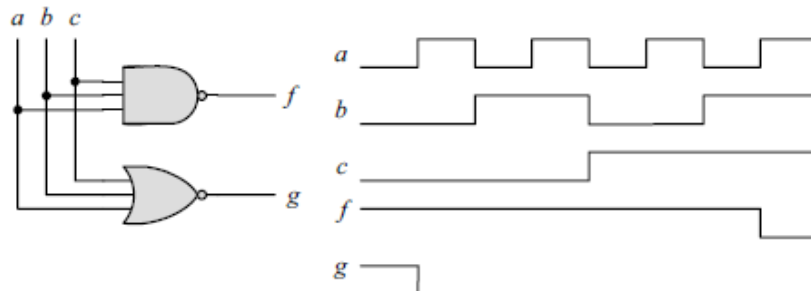
b) Odd

**1.35** By means of a timing diagram similar to Fig. 1.5, show the signals of the outputs f and g in Fig. P1.35 as functions of the three inputs a, b, and c. Use all eight possible combinations of a, b, and c.



**FIGURE P1.35**

Solution:





## Exercises

Q.1: Convert the following to Binary, Hex and Octal

$(166.5)_{10}$  :

Q.2 :

$1100110 - 0111101$

Q.3 :  $112+50$  using 8 bit signed number, detect if there is an overflow and if there is, fix the problem.

Q.4 : What is the largest positive number that can be represented in sign magnitude format using 16-bits? Express in hexadecimal.