Sena EH504

040200434

EEE205E - HW1

CRN: 10093

1. Convert the hexadecimal number 64CD to binary, to octal and decimal

$$64CD$$

$$6\times16^3 + 4\times16^2 + 12\times16 + 13\times16^0 = (25.805)_{10}$$

25 805 | 8
3225 | 8
403 | 8
50 | 8
6
2
1
3
5

$(62315)_8$

25805 | 2
12902 | 2
6455 | 2
3225 | 2
1612 | 2
806 | 2
403 | 2
201 | 2
100 | 2
50 | 2
25 | 2
12 | 2
6 | 2
3 | 2
1 | 1
0
1
0
0
1
1
0
0
1
0
1
0
0
1

OR

$(6)_{16} = (0110)_2$

$(4)_{16} = (0100)_2$

$(C)_{16} = (1100)_2$

$(D)_{16} = (1101)_2$

$(1100100111001101)_2$

2. Convert the decimal number 431 to binary, hexadecimal and octal.

431 | 2
215 | 2
107 | 2
53 | 2
26 | 2
13 | 2
6 | 2
3 | 2
1
1
1
0
1
0
1
1

$(110101111)_2$

$(110)_2 = (6)_8$

$(101)_2 = (5)_8$

$(111)_2 = (7)_8$

$(657)_8$

$(0001)_2 = (1)_{16}$

$(1010)_2 = (A)_{16}$

$(1111)_2 = (F)_{16}$

$(1AF)_{16}$

3. Express the following numbers in decimal

a. $(10110.0101)_2$

$(22,3125)_{10} = 1\times2^4 + 0\times2^3 + 1\times2^2 + 1\times2^1 + 0\times2^0 + 0\times2^{-1} + 1\times2^{-2} + 0\times2^{-3} + 1\times2^{-4}$

b. $(16.5)_{16}$

$(0001)_2 \quad (0110)_2 \cdot (0101)_2$

10110.0101

$16 + 4 + 2 + 0,25 + 0,0625 = (22.3125)_{10}$

c. $(26.24)_8$

$010 \quad 110 \cdot 010 \quad 100$

$(10110.0101)_2$

$= (22.315)_{10}$

d.

$(DADA.B)_{16}$

$13\times16^{3} + 10\times16^{2} + 13\times16^{1} + 10\times16^{0} + 11\times16^{-1} = (56026.6875)_{10}$

e.

$(1010.1101)_{2}$

$(10,8125)_{10} = 2^{3} + 0 + 2^{1} + 0 + 2^{-1} + 2^{-2} + 0 + 2^{-4}$

4. Convert the following binary numbers to hex and to decimal

a.

$(1.10010)_{2}$

$(1)_{16}$ $(9)_{16}$

$(1.9)_{16}$

$1\times2^{0} + 1\times2^{-1} + 1\times2^{-4} = (1,5625)_{10}$

b.

$(110.010)_{2}$

$(6)_{16}$ $(4)_{16}$ $(6.4)_{16}$

$1\times2^{2} + 1\times2^{1} + 1\times2^{-2} = (6.25)_{10}$

Fraction point in b is two digit right then a, meaning that there are two more digit multiplication. Because we are working with binary numbers, we multiply with 2 twice.

X	Y	Z	F(X,Y,Z)
0	0	0	0
0	0	1	1
0	1	0	1
1	0	0	0
0	1	1	1
1	1	0	0
1	0	1	1
1	1	1	1

I downloaded VirtualBox and Xilinx ISE and run the project navigator on VirtualBox. At first I was unable to share my folders due to drag and drop extension problems with my virtualbox, therefore I came up with the solution of cutting the power to my operating system and adding the folders manually from my host computer. After implementing the .vhd files, I viewed the RTL schematics to see the data flow in and out from the curcuit. The OR_gate_tb.vhd file had a 38 line code that's basically portrays the truth table for 1+0 with 15 ns pauses. I was not familiar with the coding so I did some research on that and found out it was developed to define basic std_logic data type and a few functions. Under the Simulation tab, I then simulate the OR_gate_tb.vhd file using iSim Simulator. I could see the 15 ns pauses on the simulator.