COMP – 3-5(RC)

S.E. (Computer Engineering) (Semester – III) (RC) Examination, Nov./Dec. 2012
LOGIC DESIGN

Duration : 3 Hours Max. Marks : 100

Instructions : 1) *Assume **suitable** data **if** necessary.*
2) *Answer **any five** questions, attempt atleast **one** question from **each** Module.*
3) *Draw **neat** diagrams **wherever** required.*
4) *Write question numbers **legibly** while answering.*
5) *Write answers for the questions based on marks **allotted**. Figures at the **right inside** "( )" indicate marks.*

MODULE – I

1. a) Represent the following decimal numbers in 8-bit 1) Sign magnitude form 2) Sign 1's complement form and 3) Sign 2's complement form :
A) +27 B) –36 **(6)**

b) Simplify expression AB' + ABC + A(B + AB') and verify. **(4)**

c) Express the following decimal numbers in Excess-3 form and verify.
1) 245 2) 532
3) 631 4) 739. **(4)**

d) Convert the following into Gray numbers and verify :
1) $(3A7)_{16}$
2) $(527)_8$
3) $(652)_{10}$. **(6)**

2. a) Detect and correct error, if any in the following even parity Hamming code word 0111101. Explain Hamming code concept. **(8)**

b) Implement AB + C'D' using 3 NAND gates only. Draw logic diagram. Assume inverted input is available. **(4)**

c) Expand A + BC' + ABD' + ABCD to minterms and maxterms. **(4)**

d) What is duality principle in Boolean Algebra ? Show that dual of EX-OR is same as its complement. **(4)**

P.T.O.

MODULE – II

3. A) Convert a D flip to JK flip flop. **(6)**

   B) Differentiate between the following :
      i) Synchronous and Asynchronous counters
      ii) Combinational and Sequential circuits
      iii) Level Triggered and edge triggered flip flops. **(6)**

   C) Design an active high S-R flip flop using NAND gates. Also, draw its timing diagram. **(6)**

   D) How many flip flops are required to construct mod 1024 synchronous counter ? **(2)**

4. A) Design a full adder using 2 Half Adders and one OR gate and verify. **(6)**

   B) Design a 4 bit Binary to Gray code converter using combinational logic. **(6)**

   C) Design a 4-bit even parity bit generator using combinational logic. **(4)**

   D) Draw the logic diagram of 3-Line to-8 Line Decoder which has Active High outputs. Also, present the truth table of the same. **(4)**

MODULE – III

5. A) Explain difference between counters and shift registers. **(2)**

   B) Design Mod-6 Asynchronous Up counter using T FF. Present the logic diagram, the timing diagram and table for Asynchronous Reset Inputs. **(9)**

   C) Design a synchronous counter to produce the following binary sequence 1, 3, 5, 7, 1..... Use JK flip flops. **(9)**

6. A) With a neat diagram explain the working of serial-in-parallel-out shift register. **(6)**

   B) The binary number 1101 is serially entered (right most bit first) into a 4-bit parallel out shift register that is initially clear. What are Q outputs after 2 clock pulses ? Explain functioning of 4 bit SIPO register. **(6)**

   C) With neat diagram, explain the operation of ring counter. Also sketch output waveform. **(6)**

   D) While designing a N state counter, how do you decide on the number of flip flops in the counter ? **(2)**

MODULE – IV

7. A) Design synchronous sequential circuit for Serial Binary Adder. Briefly explain its functioning. **(10)**

B) Design a sequential circuit with JK flip flops to satisfy the following state equations :

$A(t+1) = A' B' CD + A' B' C + ACD + AC' D'$

$B(t + 1) = A' C + CD' + A' BC'$

$C(t + 1) = B$

$D(t + 1) = D'$

A, B, C and D are present states of the four JK flip-flops and $A(t + 1)$, $B(t + 1)$, $C(t + 1)$ and $D(t + 1)$ are next states of the four flip flops. **(10)**

8. A) Write the PAL programming table used for implementation of the following Boolean functions using PAL with four inputs and 3-wide AND- OR structure.

$F1 (A, B, C, D) = \Sigma m(2, 12, 13)$

$F2 (A, B, C, D) = \Sigma m(7, 8, 9, 10, 11, 12, 13, 14, 15)$

$F3 (A, B, C, D) = \Sigma m(0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 15)$

$F4 (A, B, C, D) = \Sigma m(1, 2, 8, 12\ 13)$ **(10)**

B) Show how PLA circuit can be programmed to implement sum and carry output of full adder. **(10)**