Reg. No. : | | | | | | | | | | | | |

Question Paper Code : 21353

B.E./B.Tech. DEGREE EXAMINATION, MAY/JUNE 2013.

Third Semester

Electronics and Communication Engineering

EC 2203 / EC 34 / 10144 EC 304 / 080290010 – DIGITAL ELECTRONICS

(Regulation 2008 / 2010)
(Common to PTEC 2203 – Digital Electronics for Third Semester B.E. (Part-Time) Electronics and Communication Engineering – Regulation 2009)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. State De Morgan's theorem.

2. What are Don't care terms?

3. Design a Half subtractor using basic gates.

4. Draw the logic diagram of a 4 line to 1 line Multiplexer.

5. Convert D flipflop to T flipflop.

6. How many flipflops are required to build a binary counter that counts from 0 to 1023?

7. What are the different types of programmable logic devices?

8. Distinguish between PLA and PAL.

9. What are Hazards?

10. Distinguish between a flowchart and an ASM chart.

PART B — (5 × 16 = 80 marks)

11. (a) Minimize the given switching function using Quine – Mcclusky method.

$f(x_1, x_2, x_3, x_4) = \Sigma(0,5,7,8,9,10,11,14,15)$. (16)

Or

(b) Simplify the given Boolean function into

(i) Sum of products form (8)

(ii) Product of sum form and implement if using basic gates. (8)

F (A, B, C, D) = Σ (0,1, 2, 5, 8, 9, 10).

12. (a) Design a BCD adder and explain its working with necessary circuit diagram. (16)

Or

(b) Design a 4 bit magnitude comparator and draw the circuit. (16)

13. (a) Design a counter to count the sequence 0, 1, 2, 4, 5, 6 using SRFFs. (16)

Or

(b) Design a 4 bit Asynchronous Ripple counter and explain its operation with timing diagrams. (16)

14. (a) Design using PAL the following Boolean functions.

W (A, B, C, D) = Σ (2, 12, 13)

X (A, B, C, D) = Σ (7, 8, 9, 10, 11, 12, 13, 14, 15)

Y (A, B, C, D) = Σ (0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 15)

Z (A, B, C, D) = Σ (1, 2, 8, 12, 13). (16)

Or

(b) Design and explain a 32 × 8 ROM. (16)

15. (a) Design a hazard-free asynchronous circuit that changes state whenever the input goes from logic 1 to logic 0. (16)

Or

(b) (i) Design a full adder using two half adders by writing verilog program. (10)

(ii) Write Explanatory notes on Algorithmic state machines. (6)