JEPPIAAR ENGINEERING COLLEGE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CS8352– Digital Principles and system Design

Question Bank

II YEAR A & B / BATCH : 2017 -2021

(common for CSE &IT)

Vision of Institution

To build Jeppiaar Engineering College as an Institution of Academic Excellence in Technical education and Management education and to become a World Class University.

Mission of Institution

M1	To excel in teaching and **learning, research and innovation** by promoting the principles of scientific analysis and creative thinking
M2	To participate in the production, **development and dissemination of knowledge** and interact with **national and international communities**
M3	To equip students with **values, ethics and life skills** needed to enrich their lives and enable them to meaningfully contribute to the **progress of society**
M4	To prepare students **for higher studies and lifelong learning**, enrich them with the **practical and entrepreneurial skills** necessary to excel as future professionals and contribute to **Nation's economy**

Program Outcomes (POs)

PO1	**Engineering Knowledge**: Apply the Knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	**Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations
PO4	**Conduct investigations of complex problems**: Use research-based Knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	**Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex

	engineering activities with an understanding of the limitations.
PO6	**The engineer and society**: Apply reasoning informed by the contextual Knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7	**Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the Knowledge of, and need for sustainable development.
PO8	**Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	**Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	**Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	**Project management and finance**: Demonstrate Knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	**Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Vision of Department

To emerge as a globally prominent department, developing ethical computer professionals, innovators and entrepreneurs with academic excellence through quality education and research**.**

Mission of Department

M1	To create **computer professionals** with an ability to identify and **formulate the engineering problems** and also to provide **innovative solutions** through **effective teaching learning process.**
M2	To **strengthen the core-competence** in computer science and engineering and to create an ability to **interact** effectively with industries.
M3	To produce engineers with good professional sKills, **ethical values** and life skills for the **betterment of the society.**
M4	To encourage students towards **continuous and higher level learning** on technological advancements and provide a platform for **employment and self-employment.**

Program Educational Objectives (PEOs)

PEO1	**To address the real time complex engineering problems using innovative approach with strong core computing skills.**
PEO2	**To apply core-analytical Knowledge and appropriate techniques and provide solutions to real time challenges of national and global society**
PEO3	**Apply ethical Knowledge for professional excellence and leadership for the betterment of the society.**
PEO4	**Develop life-long learning skills needed for better employment and entrepreneurship**

SYLLABUS

UNIT I BOOLEAN ALGEBRA AND LOGIC GATES 12

Number Systems - Arithmetic Operations - Binary Codes- Boolean Algebra and Logic Gates - Theorems and Properties of Boolean Algebra - Boolean Functions - Canonical and Standard Forms - Simplification of Boolean Functions using Karnaugh Map - Logic Gates – NAND and NOR Implementations.

UNIT II COMBINATIONAL LOGIC 12

Combinational Circuits – Analysis and Design Procedures - Binary Adder-Subtractor - Decimal Adder - Binary Multiplier - Magnitude Comparator - Decoders – Encoders – Multiplexers - Introduction to HDL – HDL Models of Combinational circuits.

UNIT III SYNCHRONOUS SEQUENTIAL LOGIC 12

Sequential Circuits - Storage Elements: Latches , Flip-Flops - Analysis of Clocked Sequential Circuits - State Reduction and Assignment - Design Procedure - Registers and Counters - HDL Models of Sequential Circuits.

UNIT IV ASYNCHRONOUS SEQUENTIAL LOGIC 12

Analysis and Design of Asynchronous Sequential Circuits – Reduction of State and Flow Tables – Race-free State Assignment – Hazards.

UNIT V MEMORY AND PROGRAMMABLE LOGIC 12

RAM – Memory Decoding – Error Detection and Correction - ROM - Programmable Logic Array – Programmable Array Logic – Sequential Programmable Devices.

TEXT BOOK:

1. M. Morris R. Mano, Michael D. Ciletti, "Digital Design: With an Introduction to the Verilog HDL, VHDL, and SystemVerilog", 6th Edition, Pearson Education, 2017.

REFERENCES:

1. G. K. Kharate, Digital Electronics, Oxford University Press, 2010
2. John F. Wakerly, Digital Design Principles and Practices, Fifth Edition, Pearson Education, 2017.
3. Charles H. Roth Jr, Larry L. Kinney, Fundamentals of Logic Design, Sixth Edition, CENGAGE Learning, 2013
4. Donald D. Givone, Digital Principles and Design‖, Tata Mc Graw Hill, 2003.

Course Outcomes (COs)

CO202.1	Simplify Boolean functions using KMap Design
CO202.2	Analyze Combinational and Sequential Circuits
CO202.3	Implement designs using Programmable Logic Devices
CO202.4	Write HDL code for combinational Circuits
CO202.5	Write HDL code for Sequential Circuits

BLOOM TAXANOMY LEVELS

BTL1: Creating., BTL2: Evaluating., BTL3: Analyzing., BTL4: Applying., BTL5: Understanding., BTL6: Remembering

JEPPIAAR ENGINEERING COLLEGE
Jeppiaar Nagar, Rajiv Gandhi Salai – 600 119

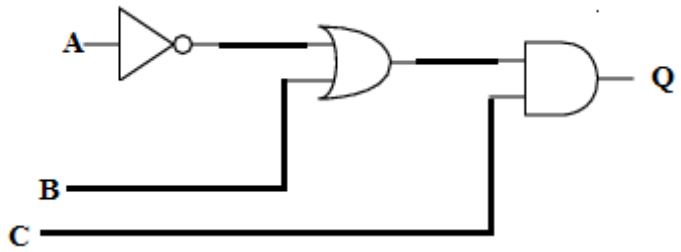
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
QUESTION BANK

SUBJECT: CS8351 DIGITAL PRINCIPLES AND SYSTEM DESIGN

YEAR /SEM: II/III

UNIT I BOOLEAN ALGEBRA AND LOGIC GATES				
Number Systems - Arithmetic Operations - Binary Codes- Boolean Algebra and Logic Gates - Theorems and Properties of Boolean Algebra - Boolean Functions - Canonical and Standard Forms - Simplification of Boolean Functions using Karnaugh Map - Logic Gates – NAND and NOR Implementations.				
PART – A				
CO Mapping : CO202.1				
S. No.	**Question**	Blooms Taxanomy Level	**Competence**	**PO**
1	**Find the Octal equivalent of the hexadecimal number DC.BA. (May/June 2016)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
2	**What is meant by multilevel gates networks?(May/June 2016)**	**BTL-1**	**Remembering**	**PO1**
3	**Discuss the NOR operation with a truth table. (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1**
4	**Write short notes on weighted binary codes. (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1**
5	**Convert $(126)_{10}$ to Octal number and binary number. (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1**
6	**Prove the following using Demorgan' theorem [(X+Y)'+(X+Y)']'= X+Y (May 2015)**	**BTL-1**	**Remembering**	**PO1**
7	**Convert $(0.6875)_{10}$ to binary. (May 2015)**	**BTL-1**	**Remembering**	**PO1**
8	**Implement AND gate using only NOR gate (December 2014)**	**BTL-1**	**Remembering**	**PO1**
9	**State the principle of duality (December 2014)**	**BTL-1**	**Remembering**	**PO1**

10	**State and prove the consensus theorem. (June 2014)**	**BTL-1**	**Remembering**	**PO1**
11	**Find the octal equivalent of hexadecimal numbers AB.CD. (June 2014)**	**BTL-1**	**Remembering**	**PO1**
12	**Realize XOR gate using only 4 NAND gates. (Dec 2013)**	**BTL-2**	**Understanding**	**PO1, PO2**
13	**Realize JK flip flop using D flip flop. (Dec 2013)**	**BTL-1**	**Remembering**	**PO1**
14	**Convert the following hexadecimal numbers into decimal numbers: (Dec 2012)** **a)263, b)1C3**	**BTL-1**	**Remembering**	**PO1**
15	**What is the significance of BCD code. (Dec 2012)**	**BTL-1**	**Remembering**	**PO1**
16	**Simplify the expression: X = (A'+B)(A+B+D)D'.**	**BTL-1**	**Remembering**	**PO1**
17	**Convert $(11001010)_2$ into gray code.** **b) Convert a Gray code 11101101 into binary code.**	**BTL-1**	**Remembering**	**PO1**
18	**State & prove De-Morgan's theorem**.	**BTL-1**	**Remembering**	**PO1**
19	**Describe the canonical forms of the Boolean function.**	**BTL-1**	**Remembering**	**PO1**
20	**Describe the importance of don't care conditions.**	**BTL-1**	**Remembering**	**PO1**
21	**What is a prime implicant?**	**BTL-1**	**Remembering**	**PO1**
22	**Define the following: minterm and maxterm?**	**BTL-1**	**Remembering**	**PO1**
23	**Minimize the function using K-map: $F=\sum m(1,2,3,5,6,7)$.**	**BTL-1**	**Remembering**	**PO1**
24	**Define Karnaugh map.**	**BTL-1**	**Remembering**	**PO1**
25	**Plot the expression on K-map: $F(w,x,y) = \sum m(0, 1, 3, 5, 6) + d(2, 4)$.**	**BTL-1**	**Remembering**	**PO1**
26	**Express x + yz as the sum of minterms**	**BTL-1**	**Remembering**	**PO1**
27	**Simplify: a) Y = AB'D + AB'D' b) Z = (A'+B)(A+B).**	**BTL-1**	**Remembering**	**PO1**
28	**What are Universal Gates? Why are they called so?**	**BTL-1**	**Remembering**	**PO1**
29	**Implement OR using NAND only.**	**BTL-1**	**Remembering**	**PO1**
30	**Implement NOR using NAND only.**	**BTL-1**	**Remembering**	**PO1**

	PART B			
1	Reduce the expression using Quine McCluskey's method $F(x_1, x_2, x_3, x_4, x_5) = \sum m (0, 2, 4, 5, 6, 7, 8, 10, 14, 17, 18, 21, 29, 31) + \sum d (11, 20, 22)$ **(May/June 2016)**	**BTL-6**	**Creating**	**PO1, PO2, PO3, PO4**
2	Simplify the following switching functions using Quine McCluskey's tabulation method and realize expression using gates F(A,B,C,D) = Σ(0,5,7,8,9,10, 11, 14,15). **(Nov/Dec 2015)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
3	Simplify the following switching functions using Karnaugh map method and realize expression using gates *F(A,B,C,D) = Σ(0,3,5,7,8,9,10,12,15).* **(Nov/Dec 2015)**	**BTL-1**	**Remembering**	**PO1**
4	(a) Express the following function in sum of min-terms and product of max-terms F(X,Y,Z)=X+YZ **(May 2015)** (b) convert the following logic system into NAND gates only. **(May 2015)**  	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
5	Simply the following Boolean expression in (i) sum of product (ii) product of sum using k-map AC'+B'D+A'CD+ABCD **(May 2015)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
6	Simplify the Boolean function in SOP and POS F(A,B,C,D)=∑m(0,1,2,5,8,9,10) (**Dec2014)** (ii) plot the following Boolean function in k-map and simplify it. F(w,x,y,z) = ∑m(0,1,2,4,5,6,8,9,12,13,14). (**Dec2014)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
7	Simply the function F(w,x,y,z)= ∑m(2,3,12,13,14,15) using tabulation method .Implement the simplified using gates.(**Dec2014)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
8	Minimize the expression using quineMccluskey(tabulation) F=∑m(0,1,9,15,24,29,30)	**BTL-6**	**Creating**	**PO1, PO2,**

	+∑d(8,11,31). method **(June 2014)**			**PO3, PO4**
9	Simplify the following functions using K-map technique **(June 2014)** G=∑m (0,1,3,7,9,11) (ii) f(w,x;y,z)=∑m(0,7,8,9,10,12)+∑d(2,5,13).	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
10	Simplify the given boolean function in POS form using K-map and draw the logic diagram using Only NOR gates F(A,B,C,D)= ∑m (0,1,4,7,8,10,12,15)+d(2,6,11,14). **(Dec2013)** ii)Convert 78.5_{10} into binary. iii)Find the dual and complement of the following Boolean expression Xyz'+x'yz+z(xy+w).	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
11	3.Simplify the Boolean function using QuineMcCluskey method: F (A, B, C, D,E) = ∑m (0,1,3,7,13,14,21,26,28) + ∑d(2,5,9,11,17,24) **(Dec 2013)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
12	Reduce the following function using K-map technique. **(Dec 2012)** i) f (A, B, C) = ∑m (0,1,3,7) + ∑d (2,5) ii) F (w,x,y,z) = ∑m (0,7,8,9,10,12) + ∑d (2,5,13)	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
13	Simlify the following Boolean function F using Tabulation method. i) F (A, B, C, D) = ∑m (0,6,8,13,14) ,d (A, B, C, D)= ∑m (2,4,10) **(Dec 2012)** ii) F (A, B, C, D) = ∑m (1,3,5,7,9,15) ,d (A, B, C, D)= ∑m (4,6,12,13)	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**

UNIT II

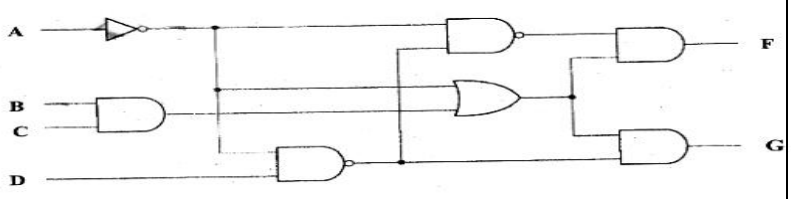
CCOMBINATIONAL LOGIC

Combinational Circuits – Analysis and Design Procedures - Binary Adder-Subtractor - Decimal Adder - Binary Multiplier - Magnitude Comparator - Decoders – Encoders – Multiplexers - Introduction to HDL – HDL Models of Combinational circuits.

PART – A

CO Mapping : CO202. 2

S. No.	Question	Blooms Taxanomy Level	Competence	PO
1	**Design the combinational circuit with 3 inputs and 1 output. The output is 1 when the binary value of the input is less than 3. The output is 0 otherwise. (May/June 2016)**	**BTL-1**	**Remembering**	**PO1**
2	**Define Combinational circuits. (May/June 2016)**	**BTL-1**	**Remembering**	**PO1**
3	**Draw the truth table of half adder. (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1**
4	**Write the Data flow description of a 4-bit Comparator. (April/May 2015)**	**BTL-1**	**Remembering**	**PO1**
5	**Implement a 4 bit even parity generator.**	**BTL-1**	**Remembering**	**PO1**
6	**Implement a 4 bit even parity checker.**	**BTL-1**	**Remembering**	**PO1**
7	**Write the data flow description of a 4-bit comparator. (May 2015)**	**BTL-1**	**Remembering**	**PO1**
8	**Implement a full adder with 4×1 multiplexer. (May 2015)**	**BTL-1**	**Remembering**	**PO1**
9	**Implement the following Boolean function using 8:1 multiplexer F(A,B,C)= ∑m(1,3,5,6)(Dec 2014)**	**BTL-1**	**Remembering**	**PO1**
10	**Draw a 2 to 1 multiplexer circuit. (June 2014)**	**(June 2014)**	**Remembering**	**PO1**
11	**What is priority encoder? (Dec 2014)**	**BTL-1**	**Remembering**	**PO1**
12	**Draw the truth table and circuit diagram of 4 to 2 encoder. (Dec 2013)**	**BTL-1**	**Remembering**	**PO1**
13	**Obtain the truth table for BCD to Excess-3 code converter. (Dec 2013)**	**BTL-1**	**Remembering**	**PO1**
14	**Write the stimulus for 2 to 1 line MUX. (June 2012)**	**BTL-1**	**Remembering**	**PO1**
15	**Distinguish between a decoder and a demultiplexer. (June 2012)**	**BTL-1**	**Remembering**	**PO1**
16	**Design a 2-bit binary to gray code converter.**	**BTL-1**	**Remembering**	**PO1**
17	**Draw the 4 bit Gray to Binary code converter.**	**BTL-1**	**Remembering**	**PO1**
18	**Draw the 4 bit Binary to Gray code converter.**	**BTL-1**	**Remembering**	**PO1**
19	**Distinguish between combinational logic and sequential logic.**	**BTL-1**	**Remembering**	**PO1**

20	**Implement half Adder using NAND Gates.**	**BTL-1**	**Remembering**	**PO1**
21	**Design a half subtractor.**	**BTL-1**	**Remembering**	**PO1**
22	**Give the truth table for half adder and write the expression for sum and carry.**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
23	**Mention the different type of binary codes.**	**BTL-1**	**Remembering**	**PO1**
24	**What is meant by self-complementing code?**	**BTL-1**	**Remembering**	**PO1**
25	**Draw the logic diagram of a one to four line de-multiplexer.**	**BTL-1**	**Remembering**	**PO1**
26	**List the advantages and disadvantages of BCD code**	**BTL-1**	**Remembering**	**PO1**
27	**Implement a full adder with two half adder.**	**BTL-1**	**Remembering**	**PO1**
28	**Define Tristate gates.**	**BTL-5**	**Evaluating**	**PO4**
29	**Define logic synthesis and simulation.**	**BTL-1**	**Remembering**	**PO1**
	PART B			
1	Implement the following Boolean function with 4 X 1 multiplexer and external gates. Connect inputs A and B to the selection lines. The input requiremnts for the four data lines will be a function of variables C and D these values are obatined by expressing F as a function of C and D for each four cases when AB = 00, 01, 10 and 11. These functions may have to be implemented with external gates. F(A, B, C, D) = Σ (1, 2, 5, 7, 8, 10, 11, 13, 15). **(May/June 2016)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
2	Design a full adder with x, y, z and two outputs S and C. The circuits performs x+y+z, z is the input carry, C is the output carry and S is the Sum. **(May/June 2016)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
3	Design a code converter thet converts a 8421 to BCD code. **(Nov./Dec. 2015)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
4	(i) Explain the Analysis procedure. Analyze the following logic diagram. **(April/May 2015)** 	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**

	(ii) With neat diagram explain the 4-bit adder with carry lookahead.			
5	(a) Design 2-bit magnitude comparator and write a verilog HDL code. (**Dec 2015**) (b)Implement the following Boolean functions with a multiplexer: F(w,x,y,z)= ∑(2,3,5,6,11,14,15) (c) Construct a 5 to 32 line decoder using 3 to 8 line decoders and 2 to 4 line decoder. **(May 2015)**	**BTL-2**	**Understanding**	**PO1, PO2**
6	Design and implement a 8241 to gray code converter. Realize the converter using only NAND gates (**Dec 2014**)	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
7	Design a circuit that converts 8421 BCD code to Excess-3 **(June 2014)** (b) Implement the following using 8 to 1 multiplexer. **(June 2014)**	BTL4		
8	(i).Realize 4 x 16 decoder using two 3 x 8 decoders with enable input. (ii) Implement the following functions using a multiplexer. F(W,X,Y,Z)= ∑m (0,1,3,4,8,9,15). (Dec 2013)	**BTL-2**	**Understanding**	**PO1, PO2**
9	5.(i).Design a combinational circuit to perform BCD addition. (ii).Design a 4-bit magnitude comparator with three outputs :A<B ,A=B ,A>B. (Dec 2013)	**(Dec 2013)**	**Creating**	**PO1, PO2, PO3**
10	Construct a 4 to 16 line decoder with an enable input using five 2 to 4 line decoders with enable inputs. **(June 2012)**	F(W,X,Y, Z)= ∑m (0,1,3,4,8, 9,15).	**Understanding**	**PO1, PO2**
11	Design a BCD to 7 segment decoder and implement it by using basic gates. **(Dec 2012)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
12	1. Discuss the need and working principle of Carry Look ahead adder. **(Dec 2012)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**

13	Design a full adder using 2 half adders.	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**
14	Design a logic circuit that accepts a 4 bit Gray code and converts it into 4 bit binary code.	**BTL-5**	**Evaluating**	**PO1, PO2, PO3, PO4**

UNIT III SYNCHRONOUS SEQUENTIAL LOGIC

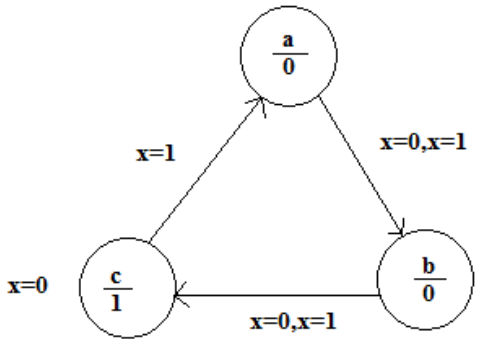
Sequential Circuits - Storage Elements: Latches , Flip-Flops - Analysis of Clocked Sequential Circuits - State Reduction and Assignment - Design Procedure - Registers and Counters - HDL Models of Sequential Circuits.

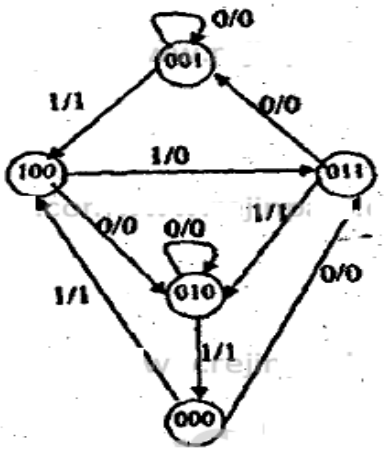
PART – A

CO Mapping : CO202. 3

S. No.	Question	Blooms Taxanomy Level	Competence	PO
1	**State the excitation table of JK Flip Flop. (May/June 2016)**	**BTL-1**	**Remembering**	**PO1**
2	**What is the minimum number of flip flops needed to build a counter of modulus 8? (May 2016)**	**BTL-1**	**Remembering**	**PO1**
3	**Write short notes on propagation delay. (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1**
4	**Draw the diagram of T flip flop and discuss its working. (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1**
5	**Give the block diagram of master-slave D flip- flop. (May 2015)**	**BTL-1**	**Remembering**	**PO1**
6	**What is ring counter? (May 2015)**	**BTL-1**	**Remembering**	**PO1**
7	**How many states are there in 3-bit ring counter? What are they? (Dec 2014)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
8	**With reference to a JK flip-flop, what is racing? (June/Dec 2014)**	**BTL-1**	**Remembering**	**PO1**
9	**What are Mealy and Moor machines? (Dec 2014)**	**BTL-1**	**Remembering**	**PO1**
10	**Write the characteristics table and equation of JK flip flop. (June 2014)**	**BTL-1**	**Remembering**	**PO1**
11	**Write any two applications of shift registers. (June 2014)**	**BTL-1**	**Remembering**	**PO1**
12	**Write the HDL code for up-down counter using behavioral model. (Dec 2013)**	**BTL-1**	**Remembering**	**PO1**

13	**Show D flip-flop implementation from a J-K flip-flop. (Dec 2013)**	**BTL-1**	**Remembering**	**PO1**
14	**Give the truth table for J-K flip-flop.**	**BTL-1**	**Remembering**	**PO1**
15	**Show the T-Flipflop implementation from SR flipflop.**	**BTL-1**	**Remembering**	**PO1**
16	**What is meant by triggering of Flip flop?**	**BTL-1**	**Remembering**	**PO1**
17	**Why D FF is known as Delay FF?**	**BTL-1**	**Remembering**	**PO1**
18	**What is the minimum number of flip-flops needed to build a counter of modulus 60?**	**BTL-1**	**Remembering**	**PO1**
19	**What is a universal shift register?**	**BTL-1**	**Remembering**	**PO1**
20	**What is meant by triggering of Flip flop?**	**BTL-1**	**Remembering**	**PO1**
21	**Differentiate between sequential and combinational circuits.**	**BTL-1**	**Remembering**	**PO1**
22	**Give difference between latch and flip-flop.**	**BTL-1**	**Remembering**	**PO1**
23	**How race around condition can be eliminated?**	**BTL-1**	**Remembering**	**PO1**
24	**How many flip flops are required to realize MOD 50 counter? (Dec 2012)**	**BTL-1**	**Remembering**	**PO1**
25	**What is a Mealy circuit?**	**BTL-2**	**Understanding**	**PO1, PO2**
26	**What is a state diagram?**	**BTL-1**	**Remembering**	**PO1**
27	**What is finite state machine?**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
28	**What do you meant by the term state reduction problem?**	**BTL-2**	**Understanding**	**PO1, PO2**
	PART B			
1	Design a modulo 5 synchronous counter using JK Flip Flop and implement it. Construct its timing diagram. **(May/June 2016)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**

				
2	Design a binary counter using T flip flops to count in the following sequences: (i) 000, 001, 010, 011, 100, 101, 111, 000 (ii) 000, 100, 111, 010, 011, 000 **(May/June 2016)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
3	Design three bit synchronous counter with T flip flop and draw the diagram. **(Nov./ Dec 2015)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
4	Design a sequence detector that detects a sequence of three or more consecutive 1's in a string of bits coming through an input line and produces an output whenever this sequence is detected. **(Nov./ Dec 2015)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
5	Consider the design of 4-bit BCD counter that counts in the following way: **(April/May 2015)** 0000,0010,0011,….,1001 and back to 0000 (i) Draw the state diagram (ii) List the next state table (iii)Draw the logic diagram of the circuit	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
6	i) A sequential circuit with two D flip-flops A and B, one input x and one output z is specified by the following next-state and output equations: **(April/May 2015)** A(t+1)= A′+B, B(t+1)=B′x, z=A+B′ (1) Draw the logic diagram of the circuit (2) Draw the state table (3) Draw the state diagram of the circuit ii) Explain the difference between a state table, characteristics table and excitation table.	**BTL-6**	**Creating**	**PO1, PO2, PO3**

7	(a) Design a MOD-10 synchronous counter using JK flip-flops. Write execution table and state table. **(Dec 2014)** (b) i) A sequential circuit with two D flips- flops A and B, one input x, and one output z is specified by the following next state and output equations: A (t+1) = A'+B, B(t+1)= B'x, z=A+B'. (1) Draw the logic diagram of the circuit. (2) Derive the state table (3) Draw the state diagram of the circuit. **(May 2015)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
8	(i)Design a shift register using JK flip flops .**(May 2015)** (ii) Explain the difference between a state table, characteristics table and an excitation table	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
9	(i) How race condition can be avoided in a flip flops? **(Dec 2014)** (ii) Realize the sequential circuit for the state diagram show below. **(Dec 2014)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
10	Design a synchronous counter that counts the sequence 000,001,010,011,100,101,110,111,000 Using D flipflop **(June 2014)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
11	Implement T flipflop using D flipflop and JK flipflop using D flipflop. **(June 2014)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
12	Design a sequential circuit by the following state diagram using T-flip flops. **(Dec 2013)**  	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**

13	Design a synchronous counter with the following sequence: 0,1,3,7,6,4 and repeats. **(Dec 2013)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
14	2. i) Write behavioural VHDL Description of 8 bit shift register with direct reset. ii)What is the difference serial and parallel transfer? Explain how to convert parallel data to serial and serial data to parallel. What type of register is needed? **(Dec 2012)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
15	Using D flip flops, design a synchronous counter which counts in the sequence, 000,001,010,011,100,101,110,111,000	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
16	Design synchronous mod 16 counter using JK flip flop. **(Dec 2012)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**

UNIT IV

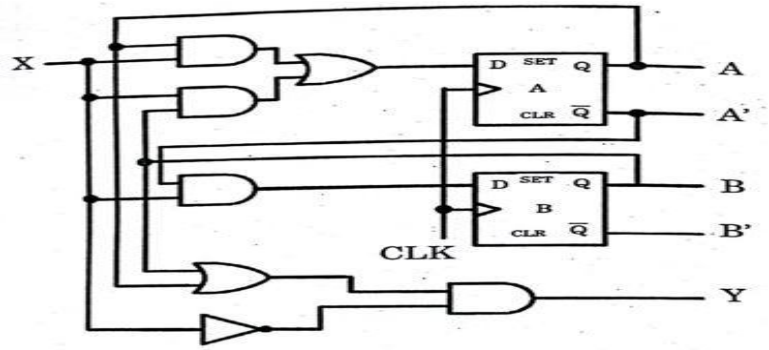
COMBINATIONAL LOGIC

PART – A

CO Mapping : CO202.4

S. No.	Question	Blooms Taxanomy Level	Competence	PO
	Define the critical race and non critical race. (May/June 2016)	**BTL-1**	**Remembering**	**PO1**
	What is lockout? How is avoided? (May/June 2016)	**BTL-1**	**Remembering**	**PO1**
	What is critical race condition? Give example. (APR/MAY 2015)	**BTL-2**	**Understanding**	**PO2**
	What is race condition? (Nov./Dec. 2015)	**BTL-1**	**Remembering**	**PO1**
	Define critical race in asynchronous sequential circuits.(May 2015)	**BTL-1**	**Remembering**	**PO1**
	Difference between synchronous and asynchronous sequential circuits (May 2015)			
	What are the types of hazards? (June 2014)	**BTL-1**	**Remembering**	**PO1**

	What are the types of hazards? (May/June 2014)	**BTL-2**	**Understanding**	**PO2**
	What is a Hazard? (June 2012/Dec 2014)	**BTL-2**	**Understanding**	**PO1**
	Difference between fundamental mode circuits and pulse-mode circuits. (Dec 2013)			
	What is Primitive Flow table? (Dec 2013)	**BTL-1**	**Remembering**	**PO1**
	What are cycles and races? (June 2012)	**BTL-1**	**Remembering**	**PO1**
	What are the different types of shift type? (Dec 2012)			
	What do you mean by Race condition? (Dec 2012/June 2014)	**BTL-1**	**Remembering**	**PO1**
	Why is the pulse mode operation of asynchronous sequential circuits not very popular?	**BTL-2**	**Understanding**	**PO1,PO2**
	Differentiae Static & Dynamic Hazard	**BTL-1**	**Remembering**	**PO1**
	What is ASM chart?	**BTL-1**	**Remembering**	**PO1**
	What is State Assignment?	**BTL-1**	**Remembering**	**PO1**
	Define Essential Hazard.	**BTL-1**	**Remembering**	**PO1**
	Define Flow table.	**BTL-1**	**Remembering**	**PO1**
	Define Merger diagram.	**BTL-2**	**Understanding**	**PO2**
	Explain Hazards in sequential circuits.	**BTL-2**	**Understanding**	**PO2**
	Explain Multiple row method.	**BTL-2**	**Understanding**	**PO2**
	What is the reason for essential hazard to occur?	**BTL-1**	**Remembering**	**PO1**
	Define the term Maximal compatible.	**BTL-1**	**Remembering**	**PO1**
	Define closed covering.	**BTL-1**	**Remembering**	**PO1**
	Explain Shared Row method.	**BTL-2**	**Understanding**	**PO2**
	What is the need of state reduction in sequential circuit design?	**BTL-2**	**Understanding**	**PO2**
	What is the use of flip-flop excitation table?	**BTL-1**	**Remembering**	**PO1**

<table>
<tr><td></td><td>List any two drawbacks of asynchronous circuits.</td><td>BTL-1</td><td>Remembering</td><td>PO1</td></tr>
<tr><td colspan="4">PART B</td><td></td></tr>
<tr><td>1</td><td>Discuss in detail the procedure for reducing the flow table with an example. (May/June 2016)</td><td>BTL-5</td><td>Evaluating</td><td>PO1,PO2, PO3</td></tr>
<tr><td></td><td>Design an asynchronous sequential circuit with 2 inputs X and Y and with one output Z Wherever Y is 1, input X is transferred to Z. When Y is 0; the output does not change for any change in X. Use SR latch for implementation of the circuit. (May/June 2016)</td><td>BTL-5</td><td>Evaluating</td><td>PO1,PO2, PO3</td></tr>
<tr><td></td><td>Design a serial adder using a full adder and a flip flop. (Nov./Dec. 2015)</td><td>BTL-5</td><td>Evaluating</td><td>PO1,PO2, PO3</td></tr>
<tr><td></td><td>Explain about hazards in digital systems. (May 2015)</td><td>BTL-2</td><td>Understanding</td><td>PO2</td></tr>
<tr><td></td><td>Analyze the following clocked sequential circuit and obtain the state equations and state diagram.
 

(Nov./Dec. 2015)</td><td>BTL-6</td><td>Creating</td><td>PO1,PO2, PO3</td></tr>
<tr><td></td><td>(a) Explain the Race- free state assignment procedure.
(b) Reduce the number of states in the following state diagram. Tabulated the reduced state table and Draw the reduced state diagram. (May 2015)
<table>
<tr><td>Present state</td><td>Next state</td><td>output</td></tr>
<tr><td colspan="2">x=0 x=1</td><td>x=0,x=1</td></tr>
<tr><td>a</td><td>a b</td><td>0 0</td></tr>
</table></td><td>BTL-6</td><td>Creating</td><td>PO1,PO2, PO3</td></tr>
</table>

		b	c d	0 0			
		c	a d	0 0			
		d	e f	0 1			
		e	a f	0 1			
		f	g f	0 1			
		g	a f	0 1			
5	Explain the steps in designing asynchronous sequential circuits. **(June 2014)**				**BTL-6**	**Creating**	**PO1,PO2, PO3**
6	Implement the switching function F=∑m(1,3,5,7,8,9,14,15) by a static hazard free two level AND OR gate network. **(June 2014)**				**BTL-5**	**Evaluating**	**PO1,PO2, PO3**
7	A synchronous sequential circuit is described by the following excitation and output function Y=X1X2+(X1+X2)Y, Z=Y. (i) Draw the logic diagram of the circuit. (ii) derive the transition table and output map.(iii) describe the behavior of the circuit. (**Dec 2014**)				**BTL-6**	**Creating**	**PO1,PO2, PO3**
8	Design a synchronous counter using JK-flip flop to count the following sequence 7, 4, 3, 15, 0, 7, **(Dec 2014)**				**BTL-6**	**Creating**	**PO1,PO2, PO3**
9	Design an asynchronous sequential circuit with inputs x1 and x2 and one output z. Initially and at any time if both the inputs are 0, output is equal to 0. When x1 or x2 becomes 1, z becomes 1. When second input also becomes 1, z=0; the output stays at 0 until circuit goes back to initial state. **(Dec 2013)**				**BTL-6**	**Creating**	**PO1,PO2, PO3**
10	(i) What is the objective of state assignment in asynchronous circuit? Explain race-free state assignment with an example. **(Dec2013)** ii) Discuss about static, dynamic and essential hazards in asynchronous sequential circuits.				**BTL-6**	**Creating**	**PO1,PO2, PO3**
11	Give the design Procedure for asynchronous sequential circuit. **(Dec 2012)**				**BTL-5**	**Evaluating**	**PO1,PO2, PO3**

UNIT V MEMORY AND PROGRAMMABLE LOGIC				
PART – A				
CO Mapping : CO202.5				
S. No.	Question	Blooms Taxanomy Level	**Competence**	**PO**
1	**Draw the waveforms showing static 1 hazard? (May/June 2016)**	**BTL-1**	**Remembering**	**PO1,PO2**
2	**Write short notes on PLA. (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1,PO2**
3	**What is memory address register? (Nov./Dec. 2015)**	**BTL-1**	**Remembering**	**PO1,PO2**
4	**How to detect double error and correct single error? (May 2015)**	**BTL-2**	**Understanding**	**PO1**
5	**Differentiate between EEPROM and PROM. (May 2015)**	**BTL-2**	**Understanding**	**PO1**
6	**What is a volatile memory? Give example. (Dec 2014)**	**BTL-1**	**Remembering**	**PO1,PO2**
7	**What is memory decoding? (June 2014)**	**BTL-1**	**Remembering**	**PO1,PO2**
8	**Define ASIC. (June 2014)**	**BTL-1**	**Remembering**	**PO1,PO2**
9	**Distinguish between PAL and PLA. (June 2012/Dec 2014)**			
10	**Distinguish EEPROM and flash memory. (Dec 2013)**	**BTL-1**	**Remembering**	**PO1,PO2**
11	**What is the difference between PROM and PLA?**	**BTL-1**	**Remembering**	**PO1,PO2**
12	**What is PLA and Its uses?**	**BTL-1**	**Remembering**	**PO1**
13	**Define Bit time & Word time.**	**BTL-1**	**Remembering**	**PO1**
14	**What is non- volatile memory?**	**BTL-1**	**Remembering**	**PO1**
15	**What are the major drawbacks of the EEPROM?**	**BTL-1**	**Remembering**	**PO1**

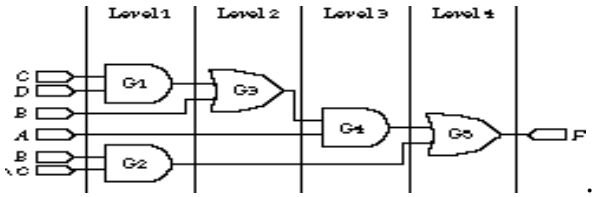
16	**Distinguish between EPROM and EEPROM**	**BTL-1**	**Remembering**	**PO1**
17	**What does burning a ROM mean?**	**BTL-1**	**Remembering**	**PO1**
18	**How many data inputs, data outputs and address inputs are needed for a 1024 *4 ROM?**	**BTL-1**	**Remembering**	**PO1**
19	**Describe the basic functions of ROM and RAM.**	**BTL-2**	**Understanding**	**PO1,PO2**
20	**How long will it take to erase UV erasable EPROM completely?**	**BTL-2**	**Understanding**	**PO1,PO2**
21	**What is Configurable Logic Block?**	**BTL-1**	**Remembering**	**PO1**
22	**Give the different types of RAM.**	**BTL-1**	**Remembering**	**PO1**
23	**What is dynamic RAM cell? Draw its basic structure.**	**BTL-2**	**Understanding**	**PO1,PO2**
24	**What is Memory refresh?**	**BTL-1**	**Remembering**	**PO1**
25	**What do you mean by PLD's?**	**BTL-1**	**Remembering**	**PO1**
26	**Compare SRAM and DRAM.**	**BTL-1**	**Remembering**	**PO1**
27	**List out the different types of ROM.**	**BTL-1**	**Remembering**	**PO1**
28	**A seven bit Hamming code is received as 1111110. What is the correct code?**	**BTL-1**	**Remembering**	**PO1**
	PART B			
1	Implement the switching functions. Z1=ab'd'e+a'b'c'd'e'+bc+de Z2=a'c'e Z3=bc+de+c'd'e'+bd Z4=a'c'e+ce using 5 x 8 x 4 PLA **(May/June 2016)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
2	(i) Write short notes on Address multiplexing. **(April/May 2015)** (ii) Briefly discuss the sequential programmable devices.	**BTL-6**	**Creating**	**PO1, PO2, PO3**
3	Implement the following two Boolean functions with a	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**

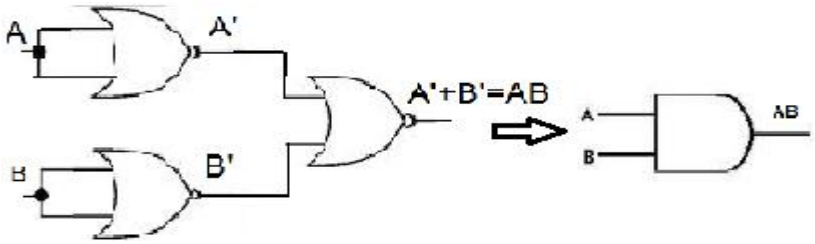
	PLA **(April/May 2015)** F1=AB′+AC+A′BC′ F2=(AC+BC)′ (ii) Give the internal block diagram of 4x4 RAM.			
4	Implement the following function using PAL F1 (A, B, C) = Σ(1, 2, 4, 6); F2 (A, B, C) = Σ(0, 1, 6, 7); F3 (A, B, C) = Σ(1, 2, 3, 5, 7). **(Nov/Dec 2015)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
5	. Design a combinational circuit using ROM that accepts a three bit binary number and outputs a binary number and outputs a binary number equal to the square of the input number. **(Nov/Dec 2015)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
6	Implement the following function using PLA A(x,y,z)= ∑m(1,2,4,6) B(x,y,z)= ∑m(0,1,6,7) C(x,y,z)= ∑m(2,6). **(May/June 2014)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
7	Design a BCD to Excess-3 code converter and implement using suitable PLA. **(Nov/Dec2014)**	**BTL-6**	**Creating**	**PO1, PO2, PO3**
8	Discuss on the concept of working and applications of semiconductor memories. **(Nov/Dec2014)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
9	i) Implement the following Boolean functions using 8 x 2 PROM. F1= ∑m (3,5,6,7) and F2= ∑m (1,2,3,4) ii) Implement the following Boolean functions using PLA with 3 inputs, 4 product terms and 2 outputs. ‘ F1= ∑m(3,5,6,7) and F2= ∑m (1,2,3,4) **(Nov/Dec 2013)**	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**
10	The following messages have been coded in the even parity hamming code and transmitted through a noisy Channel. Decode the messages, assuming that at most a single error has occurred in each code word. (i)1001001 (ii)0111001 (iii)1110110 (iv)0011011	**BTL-5**	**Evaluating**	**PO1, PO2, PO3**

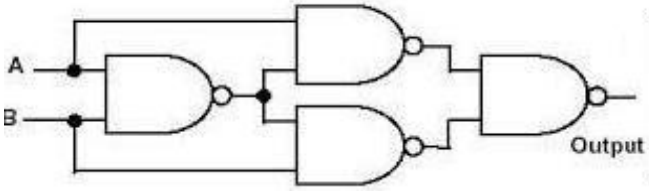
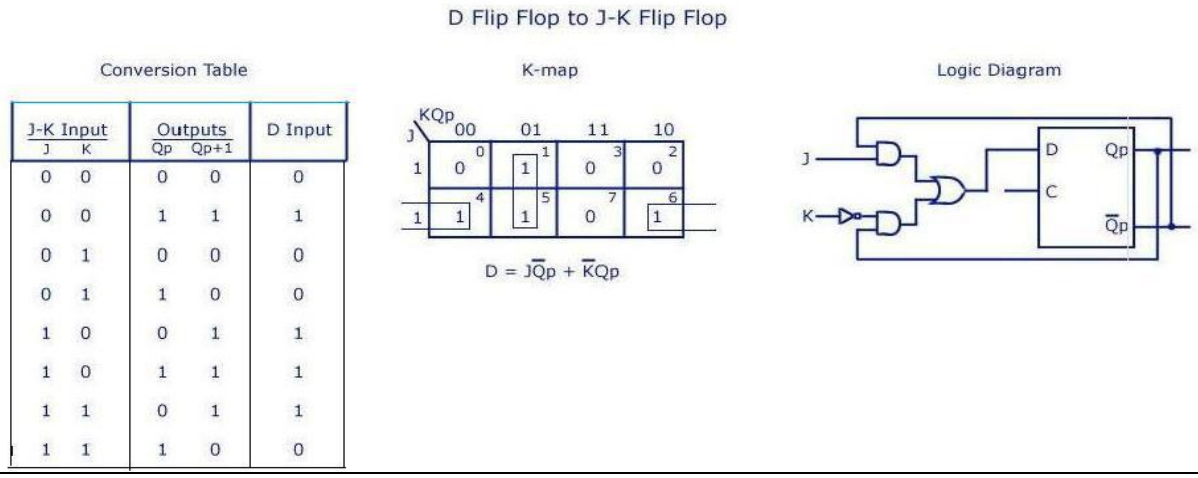
UNIT – I BOOLEAN ALGEBRA AND LOGIC GATES

PART A

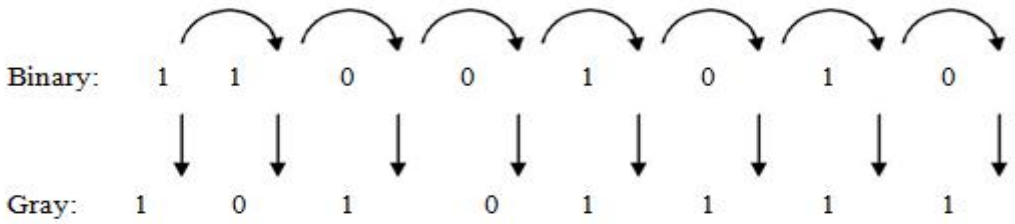
S.	**Question**

No	
1	**Find the Octal equivalent of the hexadecimal number DC.BA. (May/June 2016)** $DC.BA_{16} = 11011100.10111010_2 = 334.564_8$
2	**What is meant by multilevel gates networks?(May/June 2016)** A number of gates cascaded in series between a network input and output is referred to as the number of levels of gets. Don't count inverters as a level. Figure shows 4 level networks. 
3	**Discuss the NOR operation with a truth table. (Nov./Dec. 2015)** This is a NOT-OR gate which is equal to an OR gate followed by a NOT gate. The outputs of all NOR gates are low if any of the inputs are high. **2 Input NOR gate** A | B | $\overline{A+B}$ 0 | 0 | 1 0 | 1 | 0 1 | 0 | 0 1 | 1 | 0
4	**Write short notes on weighted binary codes. (Nov./Dec. 2015)** Weighted binary codes are those binary codes which obey the positional weight principle. Each position of the number represents a specific weight. Several systems of the codes are used to express the decimal digits 0 through 9.
5	**Convert $(126)_{10}$ to Octal number and binary number. (Nov./Dec. 2015)** $126_{10} = 1111110_2$ & 176_8
6	**Prove the following using Demorgan' theorem [(X+Y)'+(X+Y)']'= X+Y (May 2015)** = [(X+Y)'+(X+Y)']' = X+Y''**.** X+Y'' **=** (X+Y). (X+Y) = X+Y
7	**Convert $(0.6875)_{10}$ to binary. (May 2015)**

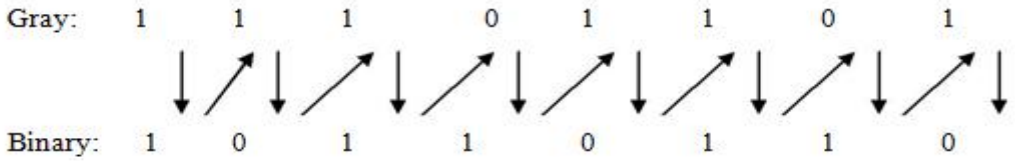
	INTEGER FRACTION COEFFICIENTS – 0.6875 x 2 = 1 + 0.3750 $a_1 = 1$ – 0.3750 x 2 = 0 + 0.7500 $a_2 = 0$ – 0.7500 x 2 = 1 + 0.5000 $a_3 = 1$ – 0.5000 x 2 = 1 + 0.0000 $a_4 = 1$ Answer: $(0.6875)_{10} = (0.a_1a_2a_3a_4)_2 = (0.1011)_2$
8	**Implement AND gate using only NOR gate (December 2014)** 
9	**State the principle of duality (December 2014)** The duality theorem states that starting with a Boolean relation we can drive another Boolean relation by changing OR operation i.e., + sign to an and operation i.e., dot and vice versa. Complement any 0 and 1appearing in the expression i.e., replacing contains 0 and 1 by 1 and o respectively
10	**State and prove the consensus theorem. (June 2014)** Theorem: AB+A'C+BC = AB+A'C **Proof:** AB+A'C+BC = AB+A'C+BC.1 =AB+A'C+BC(A+A) = AB+A'C+ABC+A'BC = AB(1+C) + A'C(1+B) = AB+A'C
11	**Find the octal equivalent of hexadecimal numbers AB.CD. (June 2014)** (i) Convert the hexadecimal to binary equivalent $(AB.CD)_{16}=(1010\ 1011.1100\ 1101)_2$ (ii) Then convert binary equivalent to octal number $(10101.1100\ 1101)_2=(\mathbf{253.315})_8$
12	**Realize XOR gate using only 4 NAND gates. (Dec 2013)**

	Exclusive OR (XOR)  A B Output 0 0 0 0 1 1 1 0 1 1 1 0
13	**Realize JK flip flop using D flip flop. (Dec 2013)** In this conversion, D is the actual input to the flip flop and J and K are the external inputs. J, K and Qp make eight possible combinations, as shown in the conversion table below. D is expressed in terms of J, K and Qp. The conversion table, the K-map for D in terms of J, K and Qp and the logic diagram showing the conversion from D to JK are given in the figure below. 
14	**Convert the following hexadecimal numbers into decimal numbers: (Dec 2012)** **(a) 263 (b)1C3.** 263H= 2x16x16+6x16+3x1=611 1C3H=1x16x16+12x16+3x1=451
15	**What is the significance of BCD code. (Dec 2012)** (i) Any large decimal number can be easily converted into corresponding binary number (i) A person needs to remember only the binary equivalents of decimal number from 0 to 9. Conversion from BCD into decimal is also very easy.
16	**Simplify the expression: X = (A'+B)(A+B+D)D'.** X = (A'+B)(A+B+D)D' = (AA'+A'B+A'D+AB+BB+BD)D' X = (0 +A'B +A'D+AB+B + BD)D' X = (A'D + B(A' + A + 1 + D))D' = (A'D + B)D' X = A'DD'+ BD'= 0 + BD' X = BD'
17	**Convert $(11001010)_2$ into gray code.** **b) Convert a Gray code 11101101 into binary code.**

(a) Binary to Gray code conversion:



(b) Gray to Binary code conversion:



18 **State & prove De-Morgan's theorem**.

De-Morgan's theorem 1: The complement of product of any number of variables is equivalent to sum of the individual complements.

De-Morgan's theorem 2: The complement of sum of any number of variables is equivalent to product of the individual complements.

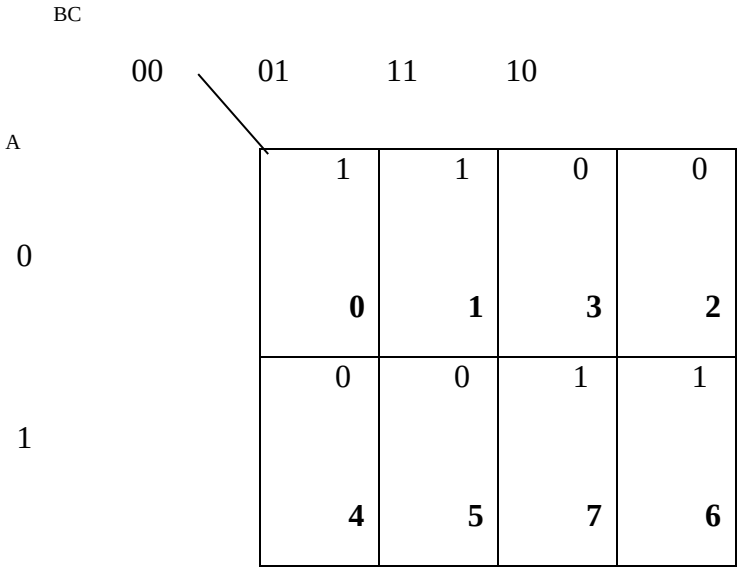
Proof:

a) (AB) '= A'+ B'

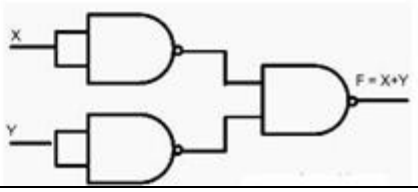
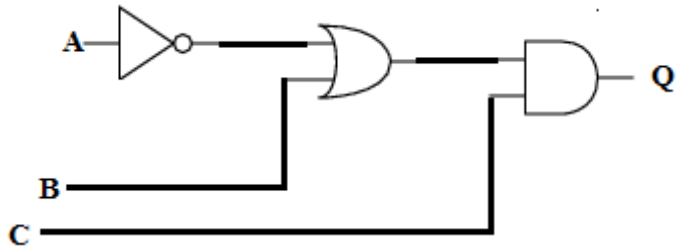
A	B	AB	**(AB)'**		A'	B'	**A'+B'**
0	0	0	**1**		1	1	**1**
0	1	0	**1**		1	0	**1**
1	0	0	**1**		0	1	**1**
1	1	1	**0**		0	0	**0**

b) (A+B)' = A'B'

A	B	A+B	**(A+B)'**		A'	B'	**A'B'**
0	0	0	**1**		1	1	**1**
0	1	1	**0**		1	0	**0**
1	0	1	**0**		0	1	**0**
1	1	1	**0**		0	0	**0**

19	**Describe the canonical forms of the Boolean function.** **a) Sum of minterms:** Combination of minterms using OR operation. Minterm (standard product) is a combination of n variables using AND operation for the function of n variables. Example for function of two variables A & B: F = A‘B + AB = $m_1 + m_3$ F = $\sum m(1,3)$ **b) Product of maxterms:** Combination of maxterms using AND operation. Maxterm (standard sum) is a combination of n variables using OR operation for the function of n variables.F = (A+B) (A‘+B) = $M_0 M_2$ F = $\prod M(0,2)$
20	**Describe the importance of don’t care conditions.** (i) Functions that have unspecified outputs for some input combinations are called incompletely specified functions. We simply don‘t care what value is assumed by the function for the unspecified minterms. (ii) The unspecified minterms are called don‘t care conditions. These don‘t care conditions can be used on a map to provide further simplification of the Boolean expression.
21	**What is a prime implicant?** A prime implicant is a product term obtained by combining the maximum possible number of adjacent squares in the map. Example: The possible prime implicants in the following K-Map are A‘B‘ & AB. 
22	**Define the following: minterm and maxterm?** **(i) Minterm**(standard product) is a combination of n variables using AND operation for the function of n variables. Possible minterms for a function of two variables A & B: A‘B‘, A‘B, AB‘, AB **(ii) Maxterm**(standard sum) is a combination of n variables using OR operation for the function of n variables. Possible maxterms for a function of two variables A & B: A+B, A+B‘, A‘+B, A‘+B‘

23	**Minimize the function using K-map: F=∑m(1,2,3,5,6,7).** BC A 00 01 11 10 0: 0 1 1 1 / **0 1 3 2** 1: 0 1 1 1 / **4 5 7 6** Quad (2,3,6,7) = B Quad (1,3,5,7) = C F = B + C
24	**Define Karnaugh map.** To simplify the Boolean expression that in canonical form, Karnaugh map is used.
25	**Plot the expression on K-map: F (w,x,y) =∑m (0, 1, 3, 5, 6) + d (2, 4).** xy w 00 01 11 10 0: 1 (0) 1 (1) 1 (3) X (2) 1: X (4) 1 (5) 0 (7) 1 (6)
26	**Express x + yz as the sum of minterms** x + yz = x(1) + (1)yz = x(y + y‘) + (x + x‘)yz = xy + xy‘ + xyz + x‘yz = xy(1) + xy‘(1) + xyz + x‘yz = xy(z + z‘) + xy‘(z + z‘) + xyz + x‘yz = xyz + xyz‘ + xy‘z + xy‘z‘ + xyz + x‘yz = xyz + xyz‘ + xy‘z + xy‘z‘ + x‘yz ----------(x + x = x) =111 + 110 + 101 + 100 + 011 =m7 + m6 + m5 + m4 + m3 x + yz =∑m(3, 4, 5, 6, 7).
27	**Simplify: a) Y = AB’D + AB’D’ b) Z = (A’+B)(A+B).** (a) Y = AB‘D + AB‘D‘ = AB‘(D+D‘) = (AB‘) (1) = AB‘ (b) Z = (A‘+B)(A+B) = AA‘+A‘B+AB+BB Z = 0+B(A‘+A)+B = B+B =B.
28	**What are Universal Gates? Why are they called so?** A Universal gates are NAND and NOR, they are called so because using these codes any logical gate

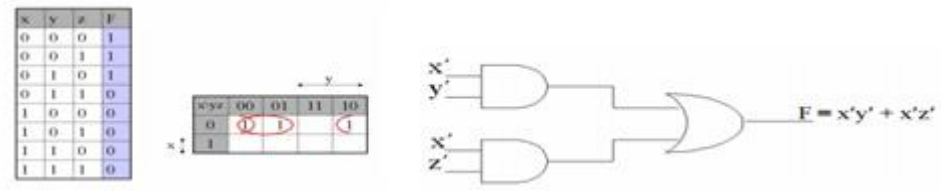
	or logical expression can be derived.
29	**Implement OR using NAND only.** **Input** **Output** **Rule** ((XX)'(YY)')' = (X'Y')' Idempotent = X''+Y'' DeMorgan = X+Y Involution 
30	**Implement NOR using NAND only.** **Input** **Output** **Rule** ((XX)'(YY))' =(X'Y')' Idempotent =X''+Y'' DeMorgan =X+Y Involution =(X+Y)' Idempotent
	PART B
1	Reduce the expression using Quine McCluskey's method $F(x_1, x_2, x_3, x_4, x_5) = \sum m (0, 2, 4, 5, 6, 7, 8, 10, 14, 17, 18, 21, 29, 31) + \sum d (11, 20, 22)$ **(May/June 2016)**
2	Simplify the following switching functions using Quine McCluskey's tabulation method and realize expression using gates F(A,B,C,D) = Σ(0,5,7,8,9,10, 11, 14,15). **(Nov/Dec 2015)**
3	Simplify the following switching functions using Karnaugh map method and realize expression using gates *F(A,B,C,D) = Σ(0,3,5,7,8,9,10,12,15).* **(Nov/Dec 2015)**
4	(a) Express the following function in sum of min-terms and product of max-terms F(X,Y,Z)=X+YZ **(May 2015)** (b) convert the following logic system into NAND gates only. **(May 2015)** 
5	Simply the following Boolean expression in (i) sum of product (ii) product of sum using k-map AC'+B'D+A'CD+ABCD **(May 2015)**
6	Simplify the Boolean function in SOP and POS F(A,B,C,D)=∑m(0,1,2,5,8,9,10) (**Dec2014)** (ii) plot the following Boolean function in k-map and simplify it. F(w,x,y,z) = ∑m(0,1,2,4,5,6,8,9,12,13,14). (**Dec2014)**

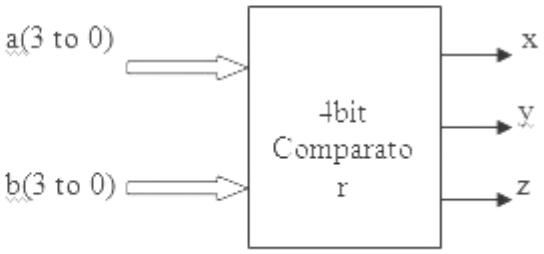
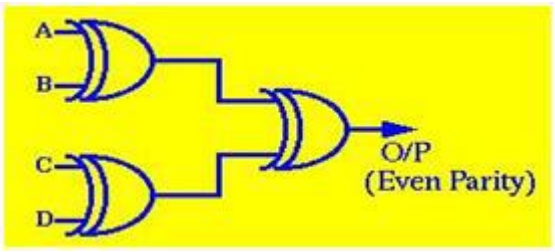
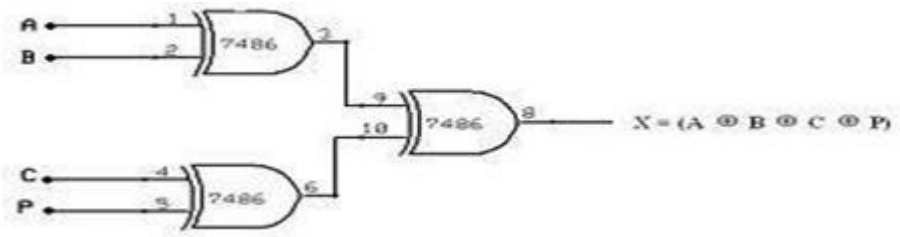
7	Simply the function F(w,x,y,z)= ∑m(2,3,12,13,14,15) using tabulation method .Implement the simplified using gates.(**Dec2014)**
8	Minimize the expression using quineMccluskey(tabulation) F=∑m(0,1,9,15,24,29,30) +∑d(8,11,31). method **(June 2014)**
9	Simplify the following functions using K-map technique **(June 2014)** G=∑m (0,1,3,7,9,11) (ii) f(w,x;y,z)=∑m(0,7,8,9,10,12)+∑d(2,5,13).
10	Simplify the given boolean function in POS form using K-map and draw the logic diagram using Only NOR gates F(A,B,C,D)= ∑m (0,1,4,7,8,10,12,15)+d(2,6,11,14). **(Dec2013)** ii)Convert 78.5_{10} into binary. iii)Find the dual and complement of the following Boolean expression Xyz'+x'yz+z(xy+w).
11	3.Simplify the Boolean function using QuineMcCluskey method: F (A, B, C, D,E) = ∑m (0,1,3,7,13,14,21,26,28) + ∑d(2,5,9,11,17,24) **(Dec 2013)**
12	Reduce the following function using K-map technique. **(Dec 2012)** iii) f (A, B, C) = ∑m (0,1,3,7) + ∑d (2,5) iv)F (w,x,y,z) = ∑m (0,7,8,9,10,12) + ∑d (2,5,13)
13	Simlify the following Boolean function F using Tabulation method. iii) F (A, B, C, D) = ∑m (0,6,8,13,14) ,d (A, B, C, D)= ∑m (2,4,10) **(Dec 2012)** iv)F (A, B, C, D) = ∑m (1,3,5,7,9,15) ,d (A, B, C, D)= ∑m (4,6,12,13)

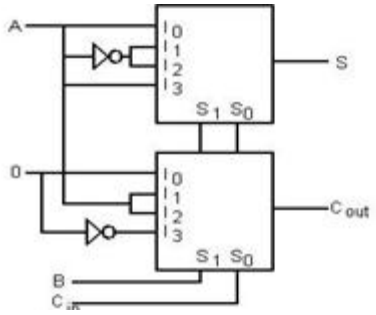
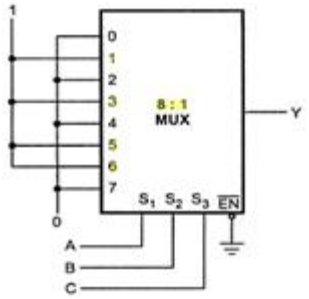
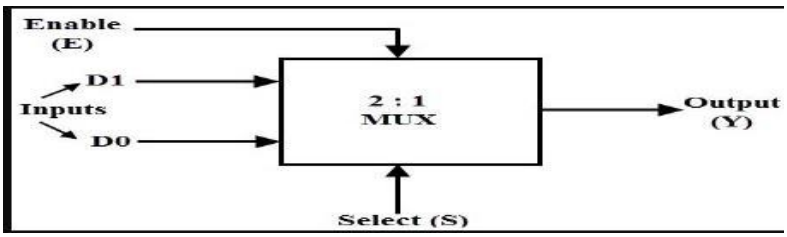
UNIT – II COMBINATIONAL LOGIC

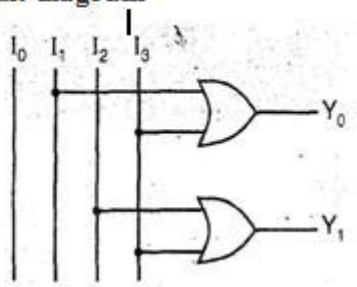
Combinational Circuits – Analysis and Design Procedures - Binary Adder-Subtractor - Decimal Adder - Binary Multiplier - Magnitude Comparator - Decoders – Encoders – Multiplexers - Introduction to HDL – HDL Models of Combinational circuits.

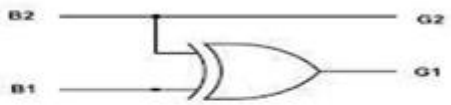
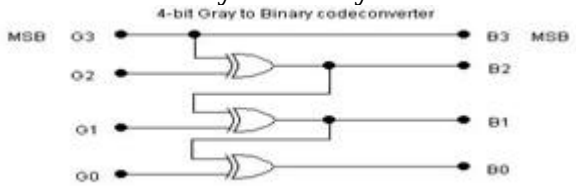
PART – A

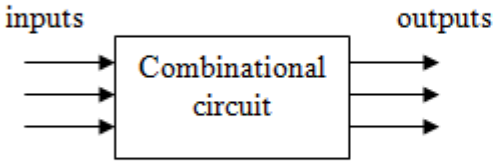
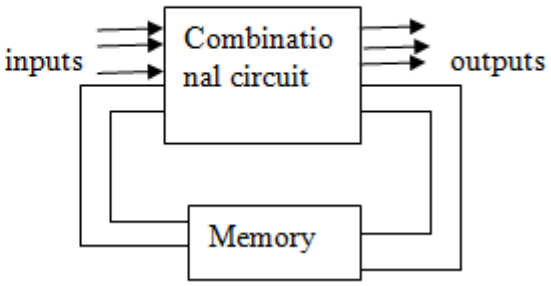
S. No	Question
1	**Design the combinational circuit with 3 inputs and 1 output. The output is 1 when the binary value of the input is less than 3. The output is 0 otherwise. (May/June 2016)** 

2	**Define Combinational circuits. (May/June 2016)** A combinational logic circuit consists of logic gates whose output is determined by the combination of current inputs.
3	**Draw the truth table of half adder. (Nov./Dec. 2015)** | INPUTS		OUTPUTS	
---	---	---	---
A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1	
4	**Write the Data flow description of a 4-bit Comparator. (April/May 2015)** ``` module comp(a,b,aeqb,agtb,altb); input [3:0] a,b; output aeqb,agtb,altb; reg aeqb,agtb,altb; always @(a or b) begin aeqb=0; agtb=0; altb=0; if(a==b) aeqb=1; else if (a>b) agtb=1; else altb=1; end end module ``` 
5	**Implement a 4 bit even parity generator.** 
6	**Implement a 4 bit even parity checker.** 
7	**Write the data flow description of a 4-bit comparator. (May 2015)**

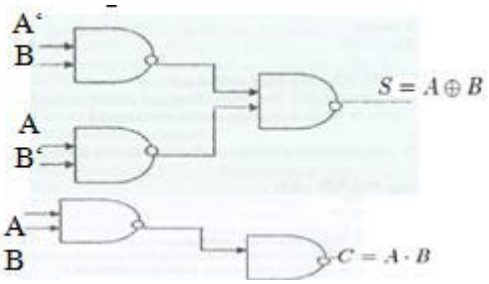
	module mag_comp(A,B,ALTB,AGTB,AETB); input [3:0] A,B; output ALTB,AGTB,AETB; assign ALTB=(A<B), AGTB=(A>B), AETB=(A= =B); end module
8	**Implement a full adder with 4×1 multiplexer. (May 2015)** 
9	**Implement the following Boolean function using 8:1 multiplexer F(A,B,C)= ∑m(1,3,5,6)(Dec 2014)** 
10	**Draw a 2 to 1 multiplexer circuit. (June 2014)** 

11	**What is priority encoder? (Dec 2014)** A priority encoder is an encoder circuit that includes the priority function. The operation of the priority encoder is such that if two or more inputs are activated at the same time, the output binary code will be generated to the highest-numbered input
12	**Draw the truth table and circuit diagram of 4 to 2 encoder. (Dec 2013)** **The Truth-table is as:** 	Input		Output	
---	---	---	---
		Y_0	Y_1
I_0	0	0	0
I_1	1	0	1
I_2	2	1	0
I_3	3	1	1	 **Circuit diagram** 
13	**Obtain the truth table for BCD to Excess-3 code converter. (Dec 2013)** | Decimal	BCD				Excess – 3 code			
---	---	---	---	---	---	---	---	---
	A	B	C	D	E_3	E_2	E_1	E_0
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	1	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	0	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0
14	**Write the stimulus for 2 to 1 line MUX. (June 2012)** module exm4_6(A,B,S,O); input A,B,S; output O; assign O=S ? A:B; end module
15	**Distinguish between a decoder and a demultiplexer. (June 2012)** 	S.NO	DEMUX	DECODER
---	---	---

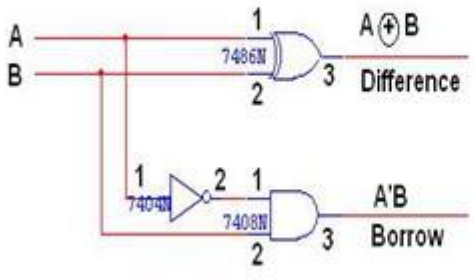
	1.	DEMUX Block diagram: Single input 1 X 2^n 2^n Outputs n selection inputs	DECODER n Intputs n X 2^n 2^n Outputs	
	2.	The demultiplexer is the circuit that receives information on a single line and transmits this information on one of many output lines.	The decoder accepts a set of binary inputs and activates only the output that corresponds to that input number.	
	3.	Data distributor	Decoder with enable input is used as Demultiplexer	
16	**Design a 2-bit binary to gray code converter.** Truth Table Binary B2B1 / Gray G2G1: 00/00, 01/01, 10/11, 11/10 Circuit Diagram 			
17	**Draw the 4 bit Gray to Binary code converter.** 			
19	**Distinguish between combinational logic and sequential logic.**			

S.no	Combinational circuit	Sequential circuit
1.		
2.	It consists of input signal, gates and output signals.	It consists of a combinational circuit to which memory elements are connected to form a feedback path.
3.	The outputs at any instant of time are entirely dependent upon the inputs present at that time.	The outputs dependent not only on the present input variable but they also depend upon the past value of the input variable.
4.	Combinational circuits are faster in speed.	Sequential circuits are slower than the combinational circuits.
5.	Combinational Circuits are easy to design	These are comparatively harder to design

20 **Implement half Adder using NAND Gates.**

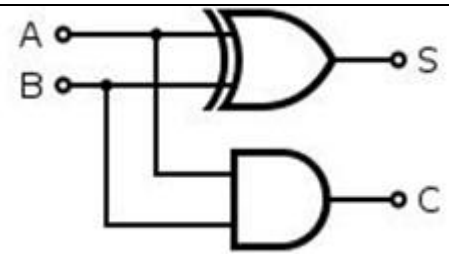


21 **Design a half subtractor.**



X	Y	D	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

22 **Give the truth table for half adder and write the expression for sum and carry.**



A half adder is a logical circuit that performs an addition operation on two binary digits. The half adder produces a sum and a carry value which is both binary digits. The drawback of this circuit is that in case of a multibit addition, it cannot include a carry. $S=A \oplus B$, $C=A.B$

Logic table for a half adder:

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

23 **Mention the different type of binary codes.**

The various types of binary codes are,

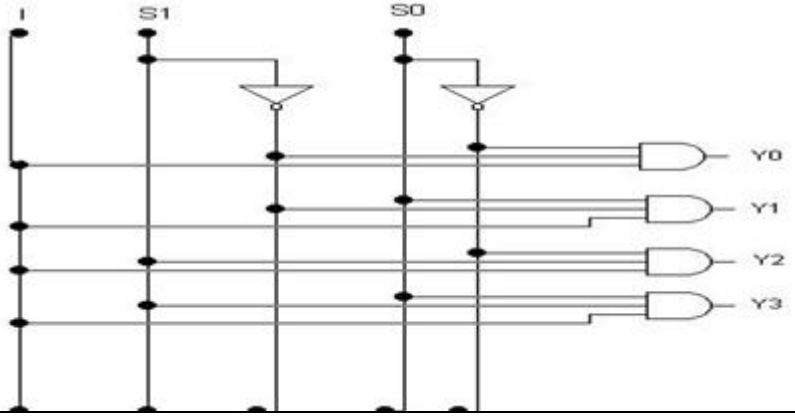
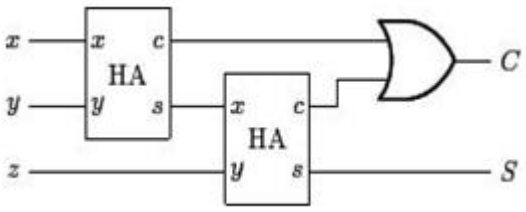
1. BCD code (Binary Coded decimal). 2. Self-complementing code. 3. The excess-3 (X‘s-3) code.4. Gray code. 5. Binary weighted code. 6. Alphanumeric code. 7. The ASCII code. 8. Extended binary coded decimal interchange code (EBCDIC). 9. Error-detecting and error-correcting code.10. Hamming code.

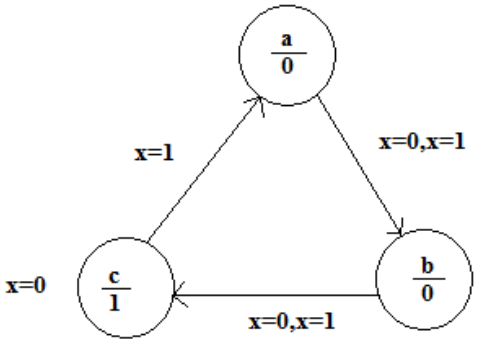
24 **What is meant by self-complementing code?**

A self-complementing code is the one in which the members of the number system complement on themselves. This requires the following two conditions to be satisfied.(i)The complement of the number should be obtained from that number by replacing 1s with 0s and 0s with 1s. (ii)The sum of the number and its complement should be equal to decimal 9.

Example of a self-complementing code is i. 2-4-2-1 code. ii. Excess-3 code

25 **Draw the logic diagram of a one to four line de-multiplexer.**

	
26	**List the advantages and disadvantages of BCD code** The advantages of BCD code are (i)Any large decimal number can be easily converted into corresponding binary number (ii)A person needs to remember only the binary equivalents of decimal number from 0 to 9. (iii)Conversion from BCD into decimal is also very easy. The disadvantages of BCD code are (i) The code is least efficient. It requires several symbols to represent even small numbers. (ii)Binary addition and subtraction can lead to wrong answer. (iii)Special codes are required for arithmetic operations. (iv)This is not a self-complementing code. (v)Conversion into other coding schemes requires special methods.
27	**Implement a full adder with two half adder.** 
28	**Define Tristate gates.** In digital electronics **three-state**, **tri-state**, or **3-state** logic allows an output port to assume a high impedance state in addition to the 0 and 1 logic levels, effectively removing the output from the circuit. This allows multiple circuits to share the same output line or lines (such as a bus which cannot listen to more than one device at a time).

29	**Define logic synthesis and simulation.** Logic synthesis is an automatic process of transforming a high level language description such HDL into an optimized net list of gates that perform the operations specified by the source code. Simulation is the representation of the structure and behavior of a digital logic system through the use of a computer. A simulator interprets the HDL description and produces output, such as timing diagram, that predicts how the hardware will behave before it is actually fabricated.
	PART – B
1	Design a modulo 5 synchronous counter using JK Flip Flop and implement it. Construct its timing diagram. **(May/June 2016)** 
2	Design a binary counter using T flip flops to count in the following sequences: (i) 000, 001, 010, 011, 100, 101, 111, 000 (ii) 000, 100, 111, 010, 011, 000 **(May/June 2016)**
3	Design three bit synchronous counter with T flip flop and draw the diagram. **(Nov./ Dec 2015)**
4	Design a sequence detector that detects a sequence of three or more consecutive 1's in a string of bits coming through an input line and produces an output whenever this sequence is detected. **(Nov./ Dec 2015)**
5	Consider the design of 4-bit BCD counter that counts in the following way: **(April/May 2015)** 0000,0010,0011,….,1001 and back to 0000 (iv) Draw the state diagram (v) List the next state table (vi) Draw the logic diagram of the circuit
6	Design and implement a 8241 to gray code converter. Realize the converter using only NAND gates (**Dec 2014**)
7	Design a circuit that converts 8421 BCD code to Excess-3 **(June 2014)**

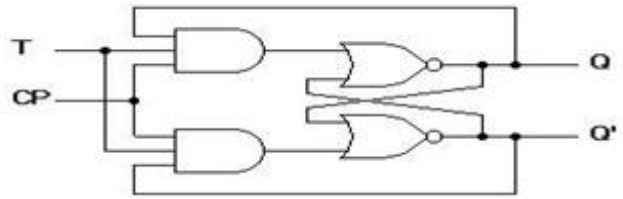
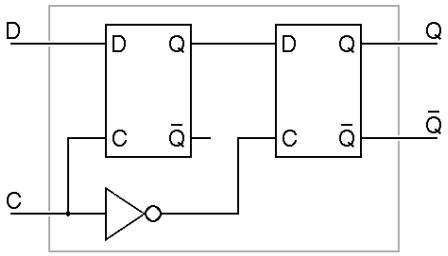
	(b) Implement the following using 8 to 1 multiplexer. **(June 2014)**
8	(i).Realize 4 x 16 decoder using two 3 x 8 decoders with enable input. (ii) Implement the following functions using a multiplexer. F(W,X,Y,Z)= ∑m (0,1,3,4,8,9,15). (Dec 2013)
9	5.(i).Design a combinational circuit to perform BCD addition. (ii).Design a 4-bit magnitude comparator with three outputs :A<B ,A=B ,A>B. (Dec 2013)
10	Construct a 4 to 16 line decoder with an enable input using five 2 to 4 line decoders with enable inputs. **(June 2012)**
11	Design a BCD to 7 segment decoder and implement it by using basic gates. **(Dec 2012)**
12	2. Discuss the need and working principle of Carry Look ahead adder. **(Dec 2012)**
13	Design a full adder using 2 half adders.
14	Design a logic circuit that accepts a 4 bit Gray code and converts it into 4 bit binary code.

UNIT – III SYNCHRONOUS SEQUENTIAL LOGIC

Sequential Circuits - Storage Elements: Latches , Flip-Flops - Analysis of Clocked Sequential Circuits - State Reduction and Assignment - Design Procedure - Registers and Counters - HDL Models of Sequential Circuits.

PART – A

S. No	Question
1	**State the excitation table of JK Flip Flop. (May/June 2016)** Q_n | Q_{n+1} | J | K 0 | 0 | 0 | X 0 | 1 | 1 | X 1 | 0 | X | 1 1 | 1 | X | 0
2	**What is the minimum number of flip flops needed to build a counter of modulus 8? (May 2016)** 3 Flip Flops
3	**Write short notes on propagation delay. (Nov./Dec. 2015)** Propagation delay is the amount of time it takes for the head of the signal to travel from the sender to the receiver.

4	**Draw the diagram of T flip flop and discuss its working. (Nov./Dec. 2015)** The T flip flop has two possible values. When T = 0, the flip flop does a hold. A hold means that the output, Q is kept the same as it was before the clock edge. When T = 1, the flip flop does a toggle, which means the output Q is negated after the clock edge, compared to the value before the clock edge. 
5	**Give the block diagram of master-slave D flip- flop. (May 2015)** 
6	**What is ring counter? (May 2015)** A ring counter is a type of counter composed of a type circular shift register. The output of the last shift register is fed to the input of the first register.
7	**How many states are there in 3-bit ring counter? What are they? (Dec 2014)** Three states-001,010,100
8	**With reference to a JK flip-flop, what is racing? (June/Dec 2014)** (i) Because of the feedback connection in the JK flip-flop,when both J & K are equal to 1 at the same time, the output will be complemented while activating the clock pulse. (ii) the output is complemented again and again if the pulse duration of the clock signal is greater than the signal propagation delay of the JK flipflop for this particular input combination (J=K=1).(iii) there is a race between 0 and 1 within a single clock pulse.this condition of the JK FF is called race-around condition or racing.
9	**What are Mealy and Moor machines? (Dec 2014)** Mealy and Moor machines are two models of clocked or synchronous sequential circuit. **Mealy machine:** The output depends on both the present state of the flip-flops and on the inputs. **Moore machine:** The output depends only on the present state of the flip-flops.
10	**Write the characteristics table and equation of JK flip flop. (June 2014)**

("X" is "don't care")

	Present state	J	K
0	0	0	x
0	1	0	x
1	0	x	1
1	1	x	0

Characteristic equation Q(next) = JQ' + K'Q

11 **Write any two applications of shift registers. (June 2014)**

(i)Parallel to serial conversion for signal transmission

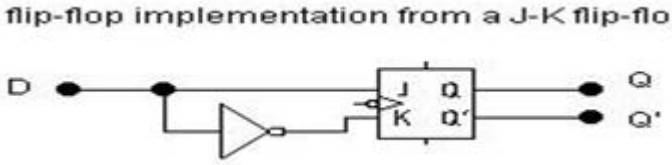
(ii)Pattern recognition

12 **Write the HDL code for up-down counter using behavioral model. (Dec 2013)**

```
module behav_counter( d, clk, clear, load, up_down, qd);
input [7:0] d;
input clk;
input clear;
input load;
input up_down;
output [7:0] qd;
reg [7:0] cnt;
assign qd=cnt;
always @ (posedgeclk) begin
 if (!clear)
            cnt = 8'h00;

else if (load)
        cnt = d;
else if (up_down)
        cnt = cnt + 1;
else
        cnt = cnt - 1;
end
endmodule
```

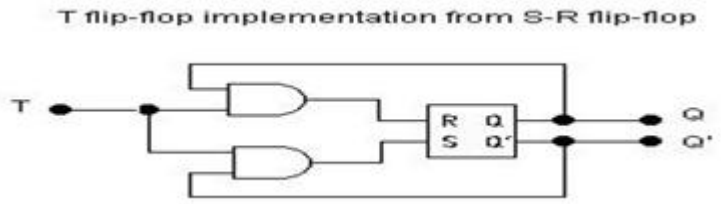
13 **Show D flip-flop implementation from a J-K flip-flop. (Dec 2013)**



14 **Give the truth table for J-K flip-flop.**

Q	J	K	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

15 **Show the T-Flipflop implementation from SR flipflop.**



16 **What is meant by triggering of Flip flop?**
The state of a flip-flop is switched by a momentary change in the input signal. This momentary change is called a trigger and the transition it causes is said to trigger the flip-flop.

17 **Why D FF is known as Delay FF?**
The binary information present at the data i/p of the D FF is transferred to the Q o/p when the cp input is enabled. The o/p follows the data i/p as long as the pulse remains in its 1 state. When the pulse goes to 0, the binary information that was present at the data i/p at the time the pulse transition occurred is retained at the Q o/p until the pulse i/p is enabled again. So D FF is known as Delay FF.

18 **What is the minimum number of flip-flops needed to build a counter of modulus 60?**
Modulus N $\leq 2^k$, where k is the number of flip-flops

Modulus $60 < 2^6 = 64$, k = 6

19 **What is a universal shift register?**

(i) A register may operate in any of the following five modes

1. SISO 2. SIPO 3. PIPO 4.PISO 5. Bidirectional

(ii)If a register can be operated in all the five possible ways, it is known as Universal Shift Register

20 **What is meant by triggering of Flip flop?**
The state of a flip-flop is switched by a momentary change in the input signal. This momentary change is called a trigger and the transition it causes is said to trigger the flip-flop.

21 **Differentiate between sequential and combinational circuits.**

Combinational circuits	**Sequential circuits**
Output depends only on the past values of input.	Output depends on the present and past

			values of input.
		Feedback path is not used in combinational circuits.	Feedback path is used for sequential circuits.
		Memory element is not present	Memory element is present.
		Clock is not used in this circuit.	Clock is used in sequential circuits.
		Examples: adder, subtractors, code converters, comparators, Mux,etc	Examples: flip-flops,counters,registers,etc

22 **Give difference between latch and flip-flop.**

Latch	**Flip-Flops**
Latch has an enable input.	Flip-Flops have a clock signal.
As long as enable input is active, the latch output will keep changing according to input.	Flip-flop samples its inputs and changes its outputs only at a particular instant of time i.e., when clock is provided.

23 **How race around condition can be eliminated?**

Race around condition can be eliminated in JK latch by two ways

1.Using the edge triggered J-K flip-flop.

2.Using the master slave J-K flip-flop.

24 **How many flip flops are required to realize MOD 50 counter? (Dec 2012)**

6 flip flops

25 **What is a Mealy circuit?**
Mealy circuit is a clocked or synchronous sequential circuit.

The output depends on both the present state of the flip-flops and on the inputs.

26 **What is a state diagram?**

(i) State diagram is the graphical representation of state table of sequential logic circuits.

(ii)In the state diagram, a state is represented by a circle and the transition between states is indicated by directed lines connecting the circles.

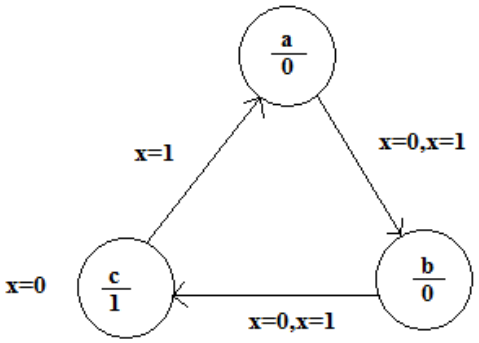
(iii)The directed lines are labeled with two binary numbers separated by a slash. The input value during the present state is labeled first and the number after the slash gives the output during the present state.

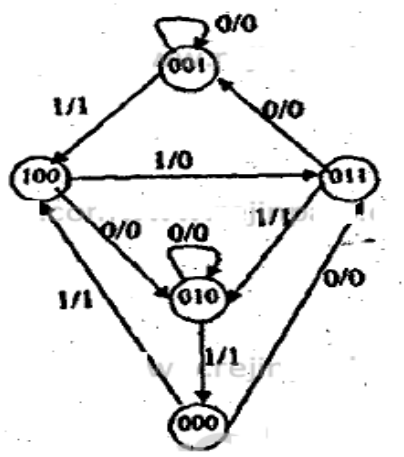
27 **What is finite state machine?**

A finite state machine (or finite automation) is an abstract model describing the synchronous sequential machine and its spatial counter, part, the iterative network

28	**What do you meant by the term state reduction problem?** The reduction of the number of flip-flops in a sequential circuit is referred to as the state – reduction problem. State – reduction algorithms are concerned with procedures for reducing the number of states in a state table while keeping the external input – output requirements unchanged.

PART – B

1	Design a modulo 5 synchronous counter using JK Flip Flop and implement it. Construct its timing diagram. **(May/June 2016)** 
2	Design a binary counter using T flip flops to count in the following sequences: (i) 000, 001, 010, 011, 100, 101, 111, 000 (ii) 000, 100, 111, 010, 011, 000 **(May/June 2016)**
3	Design three bit synchronous counter with T flip flop and draw the diagram. **(Nov./ Dec 2015)**
4	Design a sequence detector that detects a sequence of three or more consecutive 1's in a string of bits coming through an input line and produces an output whenever this sequence is detected. **(Nov./ Dec 2015)**
5	Consider the design of 4-bit BCD counter that counts in the following way: **(April/May 2015)** 0000,0010,0011,….,1001 and back to 0000 (vii) Draw the state diagram (viii) List the next state table (ix) Draw the logic diagram of the circuit
6	i) A sequential circuit with two D flip-flops A and B, one input x and one output z is specified by the following next-state and output equations: **(April/May 2015)** A(t+1)= A′+B, B(t+1)=B′x, z=A+B′ (4) Draw the logic diagram of the circuit

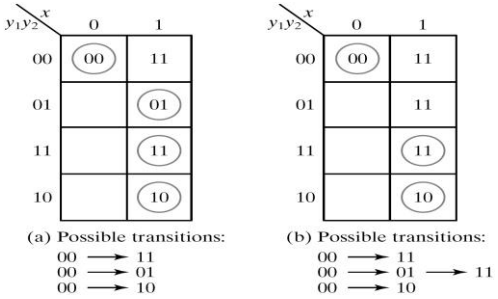
	(5) Draw the state table (6) Draw the state diagram of the circuit ii) Explain the difference between a state table, characteristics table and excitation table.
7	(a) Design a MOD-10 synchronous counter using JK flip-flops. Write execution table and state table. **(Dec 2014)** (b) i) A sequential circuit with two D flips- flops A and B, one input x, and one output z is specified by the following next state and output equations: A (t+1) = A'+B, B(t+1)= B'x, z=A+B'. (1) Draw the logic diagram of the circuit. (2) Derive the state table (3) Draw the state diagram of the circuit. **(May 2015)**
8	(i)Design a shift register using JK flip flops .**(May 2015)** (ii) Explain the difference between a state table, characteristics table and an excitation table
9	(i) How race condition can be avoided in a flip flops? **(Dec 2014)** (ii) Realize the sequential circuit for the state diagram show below. **(Dec 2014)**
	Design a synchronous counter that counts the sequence 000,001,010,011,100,101,110,111,000 Using D flipflop **(June 2014)**
	Implement T flipflop using D flipflop and JK flipflop using D flipflop. **(June 2014)**
	Design a sequential circuit by the following state diagram using T-flip flops. **(Dec 2013)** 
	Design a synchronous counter with the following sequence: 0,1,3,7,6,4 and repeats. **(Dec 2013)**
	2. i) Write behavioural VHDL Description of 8 bit shift register with direct reset.

	ii)What is the difference serial and parallel transfer? Explain how to convert parallel data to serial and serial data to parallel. What type of register is needed? **(Dec 2012)**
	Using D flip flops, design a synchronous counter which counts in the sequence, 000,001,010,011,100,101,110,111,000
	Design synchronous mod 16 counter using JK flip flop. **(Dec 2012)**

UNIT – IV ASYNCHRONOUS SEQUENTIAL LOGIC

Analysis and Design of Asynchronous Sequential Circuits – Reduction of State and Flow Tables – Race-free State Assignment – Hazards.

PART – A

S. No	Question
1	**Define the critical race and non critical race. (May/June 2016)** Critical race in asynchronous circuits occur between two signals that are required to change at the same time when the next stable state is dependent on the delay paths in the circuit. Non Critical race The final stable state does not depend on the change order of state variables.
2	**What is lockout? How is avoided? (May/June 2016)** Lockout condition is that condition wherein a counter gets onto a forbidden state and rather than coming out of it to another acceptable state or initial state, the counter switches to another forbidden state and gets stuck up in the cycle of forbidden states only. The counter should be provided with an additional circuit. This will force the counter from an unused state to the next state as initial state. It is not always necessary to force all unused states into an initial state. This frees the circuit from the Lock out condition
3	**What is critical race condition? Give example.** (APR/MAY 2015) A *critical race condition* occurs when the order in which internal variables are changed determines the eventual state that the state machine will end up in.  (a) Possible transitions: 00 ⟶ 11 00 ⟶ 01 00 ⟶ 10 (b) Possible transitions: 00 ⟶ 11 00 ⟶ 01 ⟶ 11 00 ⟶ 10

4 **What is race condition? (Nov./Dec. 2015)**

Two or more binary state variables will change value when one input variable changes

Difference between synchronous and asynchronous sequential circuits (May 2015)

S.No	Synchronous sequential circuits	Asynchronous sequential circuits
1	The change of internal state occurs in response to a clock pulse.	The change in internal state occurs whenever there is a change in input Variable.
2	Memory elements are clocked flip-flops	Memory elements are unclocked flip-flops or Time delay units.
3	The present state is totally specified by FF values and does not change if input changes while clock pulse is inactive	There is no clock pulse. Because of absence of clock, asynchronous circuits are faster than synchronous circuits.
4	Design is easy.	Design is more difficult because of the timing problems involved in the feedback path.

5 **Define critical race in asynchronous sequential circuits.(May 2015)**

Critical race in asynchronous circuits occur between two signals that are required to change at the same time when the next stable state is dependent on the delay paths in the circuit

6 **What are the types of hazards? (June 2014) (May/June 2014)**

(i) Static hazards (ii) Dynamic hazards

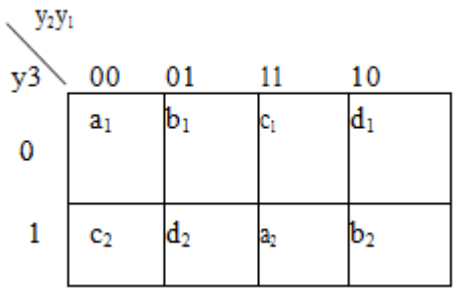
8 **What is a Hazard? (June 2012/Dec 2014)**

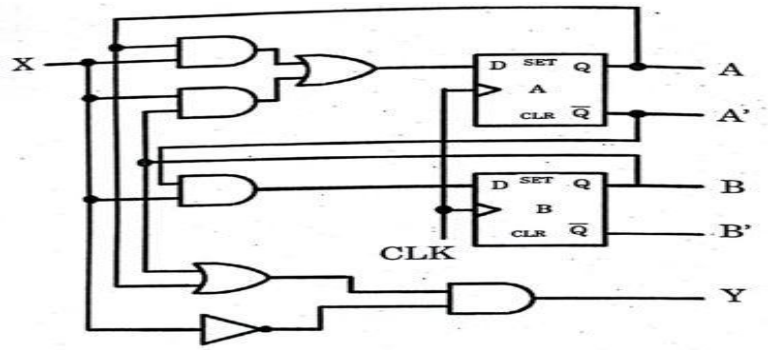
Hazards are unwanted switching transients that may appear at the output of a circuit because different paths exhibit different propagation delays. Hazards occur in combinational circuits, where they may cause a temporary false output value. When this condition occurs in asynchronous sequential circuits, it may result in a transition to a wrong stable state. Steps must be taken to eliminate this effect.

9 **Difference between fundamental mode circuits and pulse-mode circuits. (Dec 2013)**

	Fundamental Mode Circuit (i) The input variables change only when the circuit is stable. (ii) Only one input variable can change at a given time (iii) Inputs are levels and not pulses. **Pulse Mode Circuits** (i) The input variables are pulses instead of levels. (ii) The width of the pulses is long enough for the circuit to respond to the input. (iii) The pulse width must not be so long that it is still present after the new state is reached and cause a faulty change of state.(iv) No two pulses should arrive at the input lines simultaneously.
10	**What is Primitive Flow table? (Dec 2013)** A primitive flow table is a flow table with only one stable total state in each row.
11	**What are cycles and races? (June 2012)** A cycle occurs when an asynchronous circuit makes a transition through a series of unstable states. If a cycle does not contain a stable state, the circuit will go from one unstable to stable to another, until the inputs are changed. When 2 or more binary state variables change their value in response to a change in an input variable, race condition occurs in an asynchronous sequential circuit. In case of unequal delays, a race condition may cause the state variables to change in an unpredictable manner.
12	**What are the different types of shift type? (Dec 2012)** There are five types. They are, (i) Serial In Serial Out Shift Register (ii) Serial In Parallel Out Shift Register (iii) Parallel In Serial Out Shift Register (iv) Parallel In Parallel Out Shift Register
13	**What do you mean by Race condition? (Dec 2012/June 2014)** A race condition is said to exist in an asynchronous sequential circuit when two or more binary state variables change value in response to a change in an i/p variable. When unequal delays are encountered, a race condition may cause the state variables to change in an un predictable manner
14	**Why is the pulse mode operation of asynchronous sequential circuits not very popular?** Because of the input variable pulse width restrictions, pulse mode circuits are difficult to design. For this reason the pulse mode operation of asynchronous sequential circuits is not very popular.
15	**Differentiae Static & Dynamic Hazard** **Static 1-hazard:** The output may momentarily go to 0 when it should remain. **Static 0-hazard:** The output may momentarily go to 1 when it should remain 0. **Dynamic hazard** causes the output to change three or more times when it should change from 1 to 0 or from 0 to 1

16	**What is ASM chart?** i) Algorithmetic State Machine (ASM) chart is a special type of flow chart suitable for describing the sequential operation in a digital system. (ii)A state machine is another term for a sequential circuit, which is the basic structure of a digital system. (iii) The ASM chart is composed of three basic elements: the state box, the decision box and the conditional box.
17	**What is State Assignment?** (i)Assigning binary values to each state that is represented by letter symbol in the flow table of sequential circuit is called state assignment. (ii)The primary objective in choosing a proper binary state assignment in asynchronous circuit is the prevention of critical races
18	**Define Essential Hazard.** An essential Hazard is caused by unequal delays along two or more paths that originate from the same input. An excessive delay through an inverter circuit in comparison to the delay associated with the feedback path may cause such a hazard. Essential hazards cannot be corrected by adding redundant gates as in static hazards. To avoid essential hazard, each feedback loop must be handled with individual care to ensure that the delay in the feedback path is long enough compared to delays of other signals that originate from the input terminals
19	**Define Flow table.** During the design of asynchronous sequential circuits, it is more convenient to name the states by letter symbols without making specific reference to their binary values, such a table is called a Flow table.
20	**Define Merger diagram**. The merger diagram is a graph in which each state is represented by a dot placed along the circumference of a circle. Lines are drawn between any two corresponding dots that form a compatible pair. All possible compatibles can be obtained from the merger diagram by observing the geometrical patterns in which states are connected to each other.
21	**Explain Hazards in sequential circuits.** In normal combinational circuit design associated with synchronous sequential circuits, hazards are not of concern. Since momentary erroneous signals are not of generally troublesome. If a momentary incorrect signal is fed back in asynchronous sequential circuits, it may cause the circuit to go to the wrong stable state. The malfunction can be eliminated by adding an extra gate. To avoid static hazards, the asynchronous sequential circuits can be implemented with S R latches.
22	**Explain Multiple row method.** In the multiple row assignment each state in the original flow table is replaced by two or more combinations of state variables. The state assignment map shows the multiple row assignment that can

	be used with any four- row flow table. 
23	**What is the reason for essential hazard to occur?** Unequal delays along two or more paths that originate from the same input in the asynchronous sequential circuit is the reason for essential hazard to occur.
24	**Define the term Maximal compatible.** The maximal compatible is a group of compatibles that contains all the possible combinations of compatible states. The maximal compatible can be obtained from a merger diagram.
25	**Define closed covering.** The condition that must be satisfied for row merging is that the set of chosen compatibles must cover all the states that must be closed. The set will cover all the states if it includes all the states of the original state table. The closure condition is satisfied if there are no implied states or if the implied states are included within the set. A closed set of compatibles that covers all the states is called a closed covering.
26	**Explain Shared Row method**. The method of making race free assignment by adding extra rows in the flow table is sometimes referred to as Shared Row method
27	**What is the need of state reduction in sequential circuit design?** (i)To reduce the number of flip-flops (ii)To reduce the number of gates in the combinational circuit that drives the flip-flop inputs
28	**What is the use of flip-flop excitation table?** If the transition from present state to next state is known in the design of sequential circuit, the flip-flop excitation table is used to find the flip-flop input conditions that will cause the required transition.
29	**List any two drawbacks of asynchronous circuits.** Race condition and Hazards.
	PART - B
1	Discuss in detail the procedure for reducing the flow table with an example. **(May/June 2016)**

<table>
<tr><td>2</td><td>Design an asynchronous sequential circuit with 2 inputs X and Y and with one output Z Wherever Y is 1, input X is transferred to Z. When Y is 0; the output does not change for any change in X. Use SR latch for implementation of the circuit. (May/June 2016)</td></tr>
<tr><td>3</td><td>Design a serial adder using a full adder and a flip flop. (Nov./Dec. 2015)</td></tr>
<tr><td>4</td><td>Explain about hazards in digital systems. (May 2015)</td></tr>
<tr><td>5</td><td>Analyze the following clocked sequential circuit and obtain the state equations and state diagram.
 

(Nov./Dec. 2015)</td></tr>
<tr><td>6</td><td>(a) Explain the Race- free state assignment procedure.

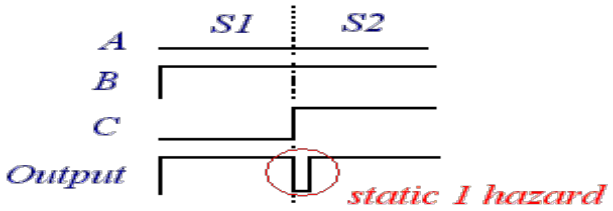
(b) Reduce the number of states in the following state diagram. Tabulated the reduced state table and Draw the reduced state diagram. (May 2015)
<table>
<tr><th>Present state</th><th>Next state</th><th>output</th></tr>
<tr><td></td><td>x=0 x=1</td><td>x=0,x=1</td></tr>
<tr><td>a</td><td>a b</td><td>0 0</td></tr>
<tr><td>b</td><td>c d</td><td>0 0</td></tr>
<tr><td>c</td><td>a d</td><td>0 0</td></tr>
<tr><td>d</td><td>e f</td><td>0 1</td></tr>
<tr><td>e</td><td>a f</td><td>0 1</td></tr>
<tr><td>f</td><td>g f</td><td>0 1</td></tr>
<tr><td>g</td><td>a f</td><td>0 1</td></tr>
</table></td></tr>
<tr><td>7</td><td>Explain the steps in designing asynchronous sequential circuits. (June 2014)</td></tr>
<tr><td>8</td><td>Implement the switching function F=∑m(1,3,5,7,8,9,14,15) by a static hazard free two level AND OR gate network. (June 2014)</td></tr>
</table>

9	A synchronous sequential circuit is described by the following excitation and output function Y=X1X2+(X1+X2)Y, Z=Y. (i) Draw the logic diagram of the circuit. (ii) derive the transition table and output map.(iii) describe the behavior of the circuit. (**Dec 2014**)
10	Design a synchronous counter using JK-flip flop to count the following sequence 7, 4, 3, 15, 0, 7, **(Dec 2014)**
11	Design an asynchronous sequential circuit with inputs x1 and x2 and one output z. Initially and at any time if both the inputs are 0, output is equal to 0. When x1 or x2 becomes 1, z becomes 1. When second input also becomes 1, z=0; the output stays at 0 until circuit goes back to initial state. **(Dec 2013)**
12	(i) What is the objective of state assignment in asynchronous circuit? Explain race-free state assignment with an example. **(Dec2013)** ii) Discuss about static, dynamic and essential hazards in asynchronous sequential circuits.
13	Give the design Procedure for asynchronous sequential circuit. **(Dec 2012)**

UNIT – V MEMORY AND PROGRAMMABLE LOGIC

RAM – Memory Decoding – Error Detection and Correction - ROM - Programmable Logic Array – Programmable Array Logic – Sequential Programmable Devices

PART - B

S. No	Question
1	**Draw the waveforms showing static 1 hazard? (May/June 2016)** 
2	**Write short notes on PLA. (Nov./Dec. 2015)** **Programmable Logic Array (PLA)** is a programmable logic device with a Programmable AND array and a programmable OR array. PLA can be used to implement complex logic circuits. It uses conventional symbol. It is more flexible than PAL
3	**What is memory address register? (Nov./Dec. 2015)**

MAR holds the memory location of data that needs to be accessed.

4 **How to detect double error and correct single error? (May 2015)**

Single Bit Error Correction using parity bits. Double Bit Error Detection, which is somehow related to the even or odd parity of the bit sequence.

5 **Differentiate between EEPROM and PROM. (May 2015)**

EEPROM	PROM
Reusable the programmable	One time programmable
Electrically erasable	Not erasable
Programmed in place (no need to remove from circuit board)	Using external for programming device

6 **What is a volatile memory? Give example. (Dec 2014)**

Volatile memory means that any storage memory location can be accessed to read or write operation. RAM is volatile memory, so data will lost if power is switched off.

7 **What is memory decoding? (June 2014)**

The memory IC used in a digital system is selected or enabled only for the range of addresses

Assigned to it and this process is called memory decoding

8 **Define ASIC. (June 2014)**

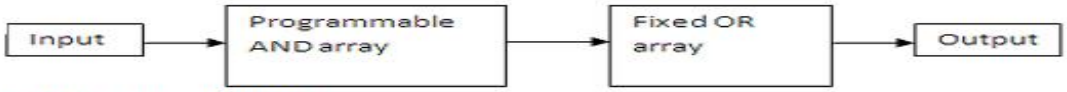
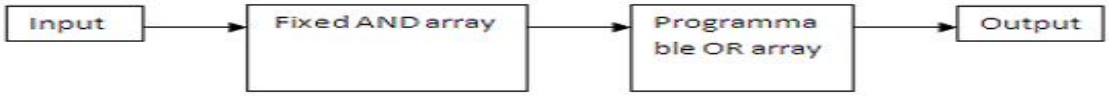
An ASIC (application-specific integrated circuit) is a microchip designed for a special application, such as a particular kind of transmission protocol or a hand-held computer.

9 **Distinguish between PAL and PLA. (June 2012/Dec 2014)**

PLA	PAL
In programmable logic array both AND and OR arrays are programmable.	In PAL OR arrays are fixed and AND arrays are programmable.
It is costlier as compared to PAL	It is cheaper.
It is complex than PAL	It is simple
It can't easily be programmed	It is easy to program a PAL

10 **Distinguish EEPROM and flash memory. (Dec 2013)**

Flash and EEPROM are very similar, but there is a subtle difference. Flash and EEPROM both use quantum cells to trap electrons. Each cell represents one bit of data. The cells have a finite life - every time a cell is erased, it wears out a little bit. In EEPROM, cells are erased one-by-one. The only cells erased are those which are 1 but need to be zero. (Writing a 1 to a cell that's 0 causes very little wear, IIRC). In Flash, a large block is erased all at once. In some devices, this "block" is the entire device. So in flash, cells are erased whether they need it or not. This cut down on the lifespan of the device, but is much, much faster than the EEPROM method of going cell-by-cell.

11	**What is the difference between PROM and PLA?** The programmable array logic(PAL) is a programmable logic device with a fixed OR array and a programmable AND array. The programmable Read only Memory(PROM) is a programmable logic device with a fixed AND array and programmable OR array. Architecture: PAL  Architecture: PROM 
12	**What is PLA and Its uses?** (i) PLA (Programmable Logic Array) is a Programmable Logic device with a programmable AND array and Programmable OR array.(ii) PLA can be used to implement complex logic circuits.(iii) It is more economical to use PLA rather than PROM to implement logic circuits that have more number of don‘t care conditions in order to reduce number of gates.
13	**Define Bit time & Word time.** The time interval between clock pulses is called the bit time, and the time required to shift the entire contents of a shift register are called the word time.
14	**What is non- volatile memory?** Memory units that retain its stored information after removal of power.Eg magnetic disk. This is because the data stored on magnetic components is manifested by the direction of magnetization, which is retained after power is turned off.
15	**What are the major drawbacks of the EEPROM?** (i)COST: In EEPROM, the erasing and programming of an EEPROM can be done in circuit.(Without using separate UV light source and special PROM programmer unit). Because of this on-chip support circuitry the EEPROM is available with more cost. (ii) DENSITY: The high level integration of the EEPROM occupies more space. For example, 1-Mbit EEPROM requires about twice as much silicon as a 1-Mbit EPROM.
16	**Distinguish between EPROM and EEPROM**

S.No	EPROM	EEPROM
1	Erasable Programmable Read Only Memory	Electrically Erasable Programmable Read Only Memory
2	Placing the EPROM chip under a special ultraviolet erases the stored Information.	Applying electrical signal erases the stored Information.
3	It can also be called as UV EPROM	It can also be called as Electrically Alterable ROM (EAROM).

17 **What does burning a ROM mean?**

The process of entering data into the ROM by burning internal fuses is called programming or burning a ROM.

18 **How many data inputs, data outputs and address inputs are needed for a 1024 *4 ROM?**

No. of data inputs and outputs = $4*1024 = 2^{10}$

No of address inputs = 10

19 **Describe the basic functions of ROM and RAM.**

ROM: Read only memory is used to store information permanently. The information cannot be altered.
RAM: Random Access Memory is used to store information. The information can be read form it and the new information can be written into the memory.

20 **How long will it take to erase UV erasable EPROM completely?**

15 to 20 min.

21 **What is Configurable Logic Block?**

The programmable logic blocks in the Xilinx family of FPGAs are called configurable logic blocks (CLBs). The CLB of Xilinx 3000 series can be configured to perform any logic function of up to a maximum of seven variables.

22 **Give the different types of RAM.**

RAM can be classified into two types:

(i)Static RAM: The storage elements used in this type RAM are latches (unclocked FFs).

(ii) Dynamic RAM: A dynamic RAM is one in which data are stored on capacitors which require periodic recharging (refreshing) to retain the data. RAMs are manufactured with either bipolar or MOS technologies. Bipolar RAMs are all static RAM. MOS RAM are available in both static and dynamic types .

23 **What is dynamic RAM cell? Draw its basic structure.**

	A dynamic RAM is one in which data are stored on capacitors which require periodic recharging (refreshing) to retain the data.
24	**What is Memory refresh?** Dynamic RAMs are fabricated using MOS technology. They store 1s and 0s as charges on a small MOS capacitor (typically a few Pico farads). Because if the tendency for these charges to leak of after a period of time, dynamics require periodic recharging of the memory cells This is called refreshing the dynamic RAM or memory refresh.
25	**What do you mean by PLD's?** **PLDs:** Programmable logic devices are the special type of IC's used by the USE and are programmed before use Different type of logic functions can be implemented using a single programmed IC chip of PLD's. PLD s can be reprogrammed because these are based on re-.writable memory technologies fuse links are used to programmed the PLD b the user according to the type of PLD to be manufactured.
26	**Compare SRAM and DRAM.** **SRAM:** Static RAM uses the flip-flop for its basic storage element. It is possible to store data as long as power is applied to the chip. It make use of cross coupled TTL multiemitter bipolar transistors or cross coupled MOSFETs for its construction. **DRAM: Dynamic** RAM make use of capacitive element for storing the data bit. Binary information is stored as charge. If charge is present at a capacitive element it represents a logic 1 and in the absence of the charge a logic 0 is stored. DRAM's consumes less power as compared to SRAM's..
27	**List out the different types of ROM.** ROM, PROM, EPROM, EEPROM
28	**A seven bit Hamming code is received as 1111110. What is the correct code?** C_1=1 C_2=1 C_4=1 The corrected code 1111111
	PART - B
1	Implement the switching functions. Z1=ab'd'e+a'b'c'd'e'+bc+de Z2=a'c'e Z3=bc+de+c'd'e'+bd Z4=a'c'e+ce using 5 x 8 x 4 PLA **(May/June 2016)**
2	(i) Write short notes on Address multiplexing. **(April/May 2015)** (ii) Briefly discuss the sequential programmable devices.

3	Implement the following two Boolean functions with a PLA **(April/May 2015)** F1=AB′+AC+A′BC′ F2=(AC+BC)′ (ii) Give the internal block diagram of 4x4 RAM.
4	Implement the following function using PAL F1 (A, B, C) = Σ(1, 2, 4, 6); F2 (A, B, C) = Σ(0, 1, 6, 7); F3 (A, B, C) = Σ(1, 2, 3, 5, 7). **(Nov/Dec 2015)**
5	. Design a combinational circuit using ROM that accepts a three bit binary number and outputs a binary number and outputs a binary number equal to the square of the input number. **(Nov/Dec 2015)**
6	Implement the following function using PLA **(May/June 2014)** A(x,y,z)= ∑m(1,2,4,6) B(x,y,z)= ∑m(0,1,6,7) C(x,y,z)= ∑m(2,6)
7	Design a BCD to Excess-3 code converter and implement using suitable PLA. **(Nov/Dec2014)**
8	Discuss on the concept of working and applications of semiconductor memories. **(Nov/Dec2014)**
9	i) Implement the following Boolean functions using 8 x 2 PROM. **(Nov/Dec 2013)** F1= ∑m (3,5,6,7) and F2= ∑m (1,2,3,4) ii) Implement the following Boolean functions using PLA with 3 inputs, 4 product terms and 2 outputs. ‘ F1= ∑m(3,5,6,7) and F2= ∑m (1,2,3,4)
10	The following messages have been coded in the even parity hamming code and transmitted through a noisy Channel. Decode the messages, assuming that at most a single error has occurred in each code word. (i)1001001 (ii)0111001 (iii)1110110 (iv)0011011