1. Why is the binary number system used in digital systems?

A. The Binary number system is used in digital systems because the devices used in digital systems operate in two states (ON and OFF) and the signals have two levels which are conveniently represented using binary number system.

2. Why are BCD codes required?

A. The binary number system is widely used in digital computers, but in many cases, it is very convenient to work with decimal numbers especially when communication between man and machine is extensive. Since most of the numerical data generated by man are in decimal numbers, to simplify the communication process between man and machine BCD codes are used.

3. Why AND, OR & NOT gates are called basic gates and NAND & NOR gates called universal gates?

A. AND, OR, and NOT gates are called basic gates or basic building blocks because any digital circuit of any complexity can be built by using only these three gates.

NAND and NOR gates are called universal gates or universal building blocks because any digital circuit of any complexity can be built by using only NAND gates or only NOR gates.

4. Which gate(s) is called an all/nothing gate and any/all gate?

A. An AND gate is called an all or nothing gate; because it produces a 1 only in one case when all its inputs are a 1. In all other cases its output is a zero.

An OR gate is called any or all gate; because it produces a 1 even if one of its inputs is a 1. It produces a 0 only when all the inputs are a 0.

5. What are the basic operations in Boolean algebra?

A. The basic operations in Boolean algebra are as follows.

(i) AND operation. It is the same as logical multiplication. It is denoted by '.' or no symbol at all.

(ii) OR operation. It is the same as logical addition. It is denoted by +.

(iii) NOT operation. It is the same as inversion or complementation. It is denoted by a bar or prime.

6. State & Prove De Morgan's theorem.

A. De Morgan's theorem states that

(i) the complement of a sum of variables is equal to the product of their individual complements

Law 1 : $\overline{\mathbf{A}+\mathbf{B}} = \bar{\mathbf{A}}\,\bar{\mathbf{B}}$

A	B	A + B	$\overline{A+B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

=

A	B	$\bar{A}$	$\bar{B}$	$\bar{A}\cdot\bar{B}$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

(ii) the complement of a product of variables is equal to the sum of their individual complements.

$$\textbf{Law 2}: \overline{\mathbf{AB}} = \overline{\mathbf{A}} + \overline{\mathbf{B}}$$

A	B	$\overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

=

A	B	$\overline{A}$	$\overline{B}$	$\overline{A}+\overline{B}$
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

7. What is a half-adder and a full adder?

A. A half-adder is an arithmetic circuit that adds two binary digits. It has two inputs and two outputs (sum and carry).

A full-adder is an arithmetic circuit that adds two binary digits and a carry, i.e. three bits. It has three inputs and two outputs (sum and carry).

8. Distinguish between combinational and sequential switching circuits.

A. Basically, switching circuits may be combinational switching circuits or sequential switching circuits.

Combinational switching circuits are those whose output levels at any instant of time are dependent only on the levels present at the inputs at that time. Any prior input level conditions have no effect on the present outputs, because combinational logic circuits have no memory.

Sequential switching circuits are those whose output levels at any instant of time are dependent not only on the levels present at the inputs at that time, but also on the prior input level conditions. It means that sequential circuits have memory. Sequential circuits are thus made of combinational circuits and memory elements.

9. What is a flip-flop and write its applications?

A. A flip-flop is the basic memory element used to store one bit of information. It can store a 0 or a 1. The name flip-flop is because this circuit shifts back and forth between its two stable states upon application of proper inputs.

Flip-flops have innumerable applications. They are used for data storage, transfer of data, counting, frequency division, parallel-to-serial and serial-to-parallel data conversion, etc.

10. How does a J-K flip-flop differ from an S-R flip-flop in its operation? What is its advantage over an S-R flip-flop?

A. In an S-R flip-flop, the condition both inputs are equal to 1 is invalid, whereas in a J-K flip-flop both inputs are equal to 1 result in toggle mode. The advantage is in ripple counters, the flip-flops are to be in toggle mode.

11. What are shift registers?

A. A number of flip-flops connected together such that the data may be shifted into and shifted out of them is called a shift register. In other words, registers in which shifting of data takes place are called shift registers. They are used basically for the storage and transfer of digital data. Data may be shifted into and out of the register either in serial form or in parallel form.

12. What is a counter?

A. A digital counter is a set of flip-flops interconnected such that their combined state at any time is the binary equivalent of the total number of pulses applied up to that time, i.e. it is a logic circuit used to count the number of pulses.

Counters are used to count pulses. They can also be used as frequency dividers. They are also used to perform the timing function as in digital watches, to create time delays, to produce non-sequential binary counts, to generate pulse trains, and to act as frequency counters, etc.

<u>Subjective Type</u>

1. Convert the following hexadecimal numbers to binary.

   (a) C20 (b) F297 (c) AF9.B0D (d) E79A.6A4

2. Convert the following binary numbers to hexadecimal.

   (a) 10110 (b) 1011011011 (c) 110110111.01111 (d) 1101101101101.101101

3. Convert the following binary numbers to decimal.

   (a) 1011 (b) 1101101 (c) 1101.11 (d) 1101110.011

4. Convert the following decimal numbers to binary.

   (a) 37 (b) 28 (c) 197.56 (d) 205.05

5. Convert the following hexadecimal numbers to decimal.

   (a) AB6 (b) 2EB7 (c) A08F.EA (d) 8E47.AB

6. Convert the following decimal numbers to hexadecimal.

   (a) 452 (b) 4796 (c) 1248.56 (d) 8957.75

7. Draw the logic symbols, construct the truth tables, and explain the working of the following gates:

   (i) AND (ii) OR (iii) NOT (iv) NAND (v) NOR (vi) Ex-OR (vii) Ex-NOR

8. State and prove (with the help of Truth tables)

(a) commutative, (b) associative, (c) distributive, and (d) idempotence laws of Boolean algebra.

9. Show that both NAND gate and NOR gate are universal gates.

10. Prove that following Boolean expressions

(a) $\overline{\overline{AB} + \overline{A} + AB} = 0$

(b) $AB + \overline{AC} + A\overline{B}C(AB + C) = 1$

(c) $\overline{\overline{A\overline{B} + ABC} + A(B + A\overline{B})} = 0$

(d) $AB + A(B + C) + B(B + C) = B + AC$

(e) $A\overline{B}(C + BD) + \overline{A}\,\overline{B} = \overline{B}C + \overline{A}\,\overline{B}$

(f) $\overline{A}\,\overline{B}C + \overline{(A + B + \overline{C})} + \overline{A}\,\overline{B}\,\overline{C}D = \overline{A}\,\overline{B}(C + D)$

(g) $ABCD + AB(\overline{CD}) + (\overline{AB})CD = AB + CD$

(h) $(A + \overline{A})(AB + AB\overline{C}) = AB$

(i) $(A\overline{B} + A\overline{C})(BC + B\overline{C})(ABC) = 0$

(j) $A\overline{B}C + \overline{A}BC + ABC = C(A + B)$

(k) $A[B + C(\overline{AB + AC}) = AB$

(l) $(\overline{A + \overline{\overline{BC}}})(A\overline{B} + \overline{ABC}) = \overline{A}BC$

(m) $A + \overline{B}C(A + \overline{\overline{B}C}) = A$

(n) $\overline{ABC}(\overline{A + B + C}) = \overline{A}\,\overline{B}\,\overline{C}$

(o) $\overline{\overline{ABC + \overline{A}\overline{B}} + BC} = \overline{A}\overline{B}$

11. Explain the half adder circuit operation with the truth table and draw the logic diagram using basic logic gates.

12. Explain the full adder circuit operation with the truth table and draw the logic diagram using basic logic gates.

13. Draw the circuit diagram of a SR flip-flop and explain its operation with the help of a truth table.

14. Draw the circuit diagram of a JK flip-flop and explain its operation with the help of a truth table.

15. With neat diagrams explain the working of the 2-bit register.

16. With neat diagrams explain the working of the 2-bit counters.

Note: Refer problems solved in class & given in assignments, internal examination and also practice unsolved problems in the Textbooks. In addition, follow Model question paper.