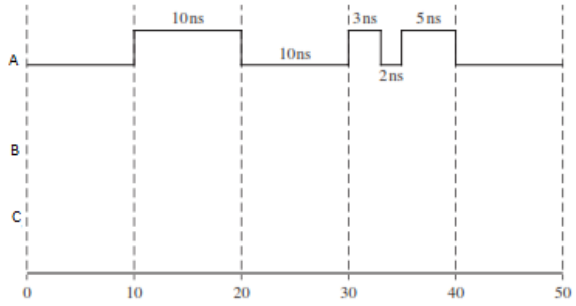
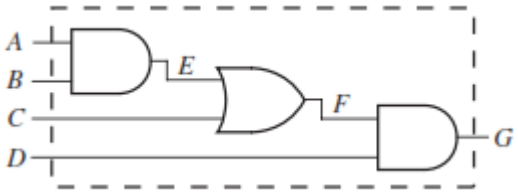
MODULE-1

1. Illustrate with a neat flow diagram, the steps involved in the modern digital system design.
2. Write a Verilog code for a full subtracter using logic equations and using this module as a component; Develop a Verilog code for 4 bit subtracter.
3. Briefly describe the various operators used in Verilog
4. Develop a Verilog code for Full adder using dataflow, behavioural and Structural modelling.
5. Discuss operators in Verilog
6. Write a Verilog code for 4:1 Mux using conditional operator.
7. Discuss delays in Verilog with example.
8. Write a Verilog code for 4:1 Mux using case operator.
9. Develop a Verilog code for JK flip flop with preset and clear.
10. Explain in detail Compilation, simulation and synthesis of Verilog Code with a neat block diagram
11. Develop a verilog code for Fullsubstactor using dataflow, behavioral and Structural modelling
12. Given the following timing waveform for A, draw B and C.

```
wire #3 B;
assign #5 B = A;
wire #5 C;
assign #3 C = A;
```



13. Develop a Verilog code using behavioural model for 8:1 MUX
14. Explain blocking and non-blocking statements with example.
15. Write a Verilog code using behavioural model for 2:4 Decoder
16. Write Verilog code for the following circuit.
    i) Using concurrent statements.
    ii) Using an always block with sequential statements. No latches should be generated
    Note: Assume that the gate delays are negligible.



17. Consider simulator command driving B is given by:
    force B 0 0, 1 10, 0 15, 1 20, 0 30, 1 35
    Draw a timing diagram illustrating *A*, *B*, and *C* if the following concurrent statements are executed:
    always @(B)

```
begin
A = #5 B;
end
assign #8 C = B;
```

18. Develop a Verilog structural description of a 4-bit adder using full adder as a component
19. Draw the circuit diagram at gate level

```
module Q3(A,B,C,F,Clk,E);
input A,B,C,F,Clk;
output reg E;
reg D,G;
initial
begin
E = 1'b0;
D = 1'b0;
G = 1'b0;
end
always @(posedge Clk)
begin
 D <= A & B & C;
G <= ~A & ~B;
 E <= D | G | F;
end
endmodule
```

20. Given

    wire a = 1'b0;
    wire [1:0] b = 2'b10;
    wire [2:0] c = 3'b101;
    Evaluate
        i. { 4{a} }
        ii. { 4{a}, 2{b} }
        iii. { 4{a}, c}

21. Given

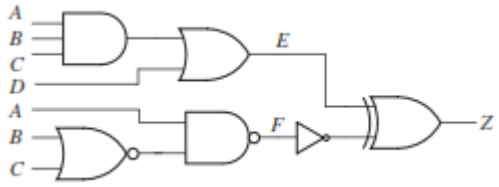
        reg [7:0] C;
        reg signed [7:0] D;
        reg signed [7:0] A = 8'hD5;
    evaluate
        i. C = A >> 4
        ii. C = A >>> 4
        iii. C = A << 4
        iv. C = A <<< 4
        v. D = A >> 4
        vi. D = A >>> 4
        vii. D = A << 4
        viii. D = A <<< 4

22. For the following Verilog code segment:

```
 reg a;
reg [2:0] b, c;
 reg [10:0] x;
  a = 3'b100; b = 1'b1; c = 3'b101;
  x = {{2{b}}, a,{2{c}}};
```

What will be the value of x?

23. An AB flip-flop responds to the falling clock edge as follows:
    If $A = B = 0$, the flip-flop changes state.
    If $A = 0$ and $B = 1$, the flip-flop output is set to 1.
    If $A = 1$ and $B = 0$, the flip-flop output is set to 0.
    If $A = B = 1$, no change of flip-flop state occurs.
    The flip-flop is cleared asynchronously if $CLRn = 0$.
    Write a complete Verilog module that implements an AB flip-flop.
24. Write a Verilog description of the following combinational circuit using concurrent statements. Each gate has a 5-ns delay, excluding the inverter, which has a 2-ns delay.



25. For the following Verilog code segment:

```
reg a;
reg [2:0] b, c;
reg [10:0] x;
a = 3'b100; b = 1'b1; c = 3'b101;
x = {{2{b}}, a,{2{c}}};
```

What will be the value of x?

26. Consider the fallowing Verilog code:

```
module bitwise_op ( );
 reg [2:0] a, b, c, x, y, z;
 initial
 begin
        a = 5;b = 3'b111; c =3'd x;
        x = a & b;
        y = a & c;
        z = b & 1;
 $display("x = %b, y = %b, z = %b",
       x,y,z);
end
endmodule
```

Write the final result for x,y and z ?

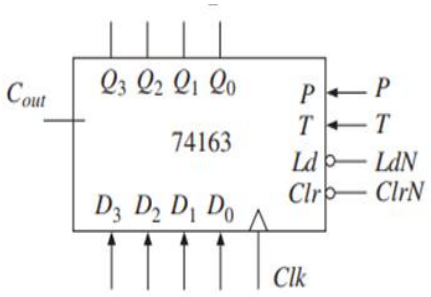
27. **Given**

```
reg signed [7:0] B = 8'hC7;
i)B>> 4
ii)B >>> 4
iii)B  << 4
iv)B <<< 4
```

MODULE-2

1. Write Verilog modules using conditional operator that is equivalent to the following code:
   A = B1 when C = 1 else B2 when C = 2 else B3 when C = 3 else 0;

2. Develop a Verilog code for 4-bit fully synchronous binary counter shown in Figure



Control Signals			Next State				
ClrN	*LdN*	*PT*	Q_3^+	Q_2^+	Q_1^+	Q_0^+	
0	X	X	0	0	0	0	(clear)
1	0	X	D_3	D_2	D_1	D_0	(parallel load)
1	1	0	Q_3	Q_2	Q_1	Q_0	(no change)
1	1	1	present state + 1				(increment count)

3. Write the Verilog code for the following 8-bit bidirectional synchronous shift register with parallel load capability. The notation used to represent the input/output pins is explained as follows:

   CLR Asynchronous Clear, which overrides all other inputs
   (7:0) 8-bit output
   D(7:0) 8-bit input
   S0,S1 mode control inputs
   LSI serial input for left shift
   RSI serial input for right shift
   The mode control inputs work as follows:

S_0	S_1	Action
0	0	No action
0	1	Right shift
1	0	Left shift
1	1	Load parallel data (i.e., $Q = D$)

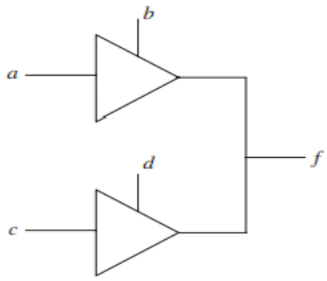
4. Write a Verilog code for Tristate buffer with always and assignment statements
5. Define $display, $write, $strobe, $setup
6. Illustrates a function using 'for loop'. Write a Verilog function for generating an even parity bit for a 4-bit number.
7. List the Compiler Directives in Verilog
8. Consider the following Verilog code segment, if the initial value of IR is 12345678 (in hexadecimal).

```
wire [31:0] IR;
wire [3:0] data;
wire [15:0] d1;
wire [31:16] d2;
assign d1 = IR[31:16];
 assign d2 = IR[15:0];
assign data = d1[7:4] + d2[19:16] + d2[31:28];
```

   the value of "data" in decimal will be _____
   (Note that "data" is a 4-bit variable **Note answer should be shown step by step**)

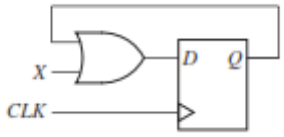
9. Write a Verilog code 4 bit asynchronous counter with test bench
10. Write a Verilog code 4 bit synchronous counter with test bench
11. Design a 4 bit subtractor using generate statement.
12. Develop a Verilog code 4 bit adder with test bench
13. Develop a Verilog code for n bit shift register using generate statements
14. Discuss system functions and Compiler directives in detail.
15. Design a 4 bit adder using generate statement
16. List the built in primitives. Explain any two with example.
17. List the Compiler Directives in Verilog.
18. Write a Verilog code for task which adds two 4 bit data with carry and returns n bit data and carry

19. Develop a Verilog code for the tri-state buffers with active-high output enable shown in Figure



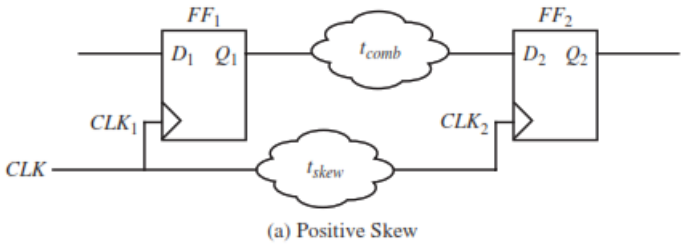
MODULE-3

1. Design a circuit that counts 16 clock cycles and produces a control signal, "ctrl", that is '1', during every fifth and Tenth cycle. Also, develop a Verilog model for the same Tone.
2. With suitable Verilog syntax, discuss the various D flip flop with features.
3. A digital alarm clock needs to generate a periodic signal at a frequency of approximately 1024Hz to drive the speaker for the alarm tone. Use a counter to divide the system's master clock signal, with a frequency of 2 MHz, to derive the alarm.
4. A D flip-flop has a setup time of 5 ns, a hold time of 3 ns, and a propagation delay from the rising edge of the clock to the change in flip-flop output in the range of 6 to 12 ns. An OR gate delay is in the range of 1 to 4 ns.



a) What is the minimum clock period for proper operation of the following circuit?
b) What is the earliest time after the rising clock edge at which *X* is allowed to change?

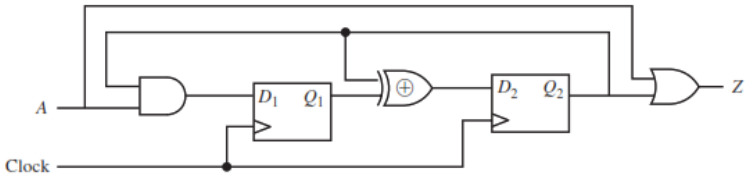
5. Discuss in detail timing rules for flip flop and flip flop paths.
6. Consider the circuit shown below with the following delays:
   CLK-to-Q for Flip-flops: 7 ns/9 ns
   Combinational Delay: 4 ns/6 ns
   Setup Time for Flip-Flops: 5 ns
   Hold Time for Flip-Flops: 2 ns



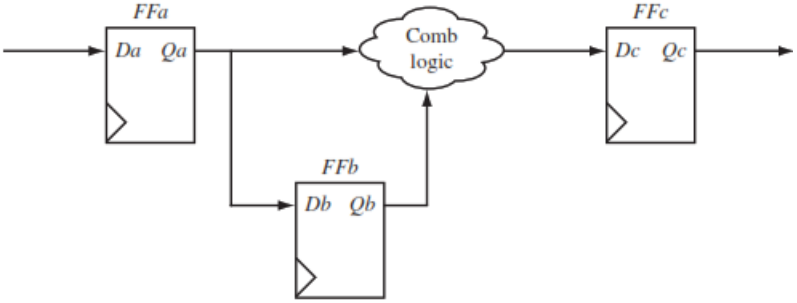
(a) Positive Skew

(a) If skew for the second flip-flop is 3 ns, what is the maximum clock frequency? Compare it with the clock frequency if no skew is present.
(b) What is the biggest skew that the above circuit shown in Figure can take while meeting the hold-time constraint for this circuit?

7. Consider the circuit shown below with the following minimum/maximum delays:
   CLK-to-Q for flip-flop 1: 5 ns/8 ns
   CLK-to-Q for flip-flop 2: 7 ns/9 ns
   XOR Gate: 4 ns/6 ns
   AND Gate: 1 ns/3 ns
   Setup time for flip-flops: 5 ns
   Hold time for flip-flops: 2 ns
   (a) What is the minimum clock period that this circuit can be safely clocked at?
   (b) What is the earliest time after the rising clock edge that input A can safely change?
   (c) What is the latest time before the rising clock edge that input A can safely change?

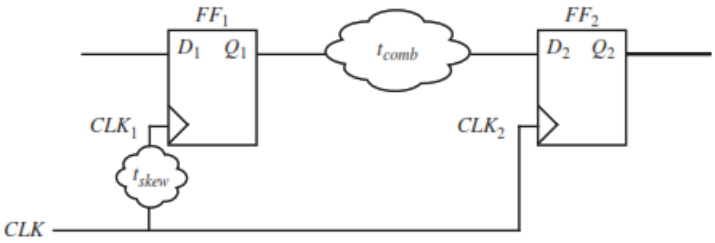


8. Consider the circuit shown in Figure with the following minimum/maximum delays: CLK-to- Q for flip-flop A: 7ns/9ns
   CLK-to-Q for flip-flop B: 8ns/10ns
   CLK-to-Q for flip-flop C: 9ns/11ns
   Combinational logic: 3ns/4ns
   Setup time for flip-flops: 2ns
   Hold time for flip-flops: 1ns



   Compute the delays for all timing paths in this circuit and determine the maximum clock frequency allowed in this circuit

9. Develop a Verilog model for a pipelined circuit that computes the average of corresponding values in three streams of input values, a, b, c and d. The pipeline consists of three stages: the first stage sums values of a, b and c and saves the value of d; the second stage adds on the saved value of d; and the third stage divides by four. The inputs and output are all signed fixed-point numbers indexed from 5 down to -8.
10. Consider the circuit shown below with the following delays:
    CLK-to-Q for Flip-flops: 7 ns/9 ns
    Combinational Delay: 4 ns/6 ns
    Setup Time for Flip-Flops: 5 ns
    Hold Time for Flip-Flops: 2 ns



(a) If skew for the first flip-flop in above circuit is 3 ns, what is the maximum clock frequency? Compare it with the clock frequency if no skew is present.

(b) What is the biggest skew that the circuit in shown above can take while meeting the Hold-time constraint for this circuit?

11. Design an n bit shift register using D flip-flop and Multiplexers.
12. Illustrate timing paths in synchronous circuits.
13. Develop a Verilog model for an accumulator that calculates the sum of a sequence of fixed-point numbers. Each input number is signed with 4 pre-binary-point and 12 post-binary-point bits. The accumulated sum has 8 pre-binary-point and 12 post-binary-point bits. A new number arrives at the input during a clock cycle when the data_en control input is 1. The accumulated sum is cleared to 0 when the reset control input is 1. Both control inputs are synchronous.
14. Discuss asynchronous and synchronous reset feature in D flip-flop with the suitable waveform.
15. A digital alarm clock needs to generate a periodic signal at a frequency of approximately 500Hz to drive the speaker for the alarm tone. Use a counter to divide the system's master clock signal, with a frequency of 1 MHz, to derive the alarm tone.
16. Design a circuit for a modulo-10 counter, otherwise known as a decade counter.
17. Develop a Verilog model for an interval timer that has clock, load and data input ports and a terminal-count output port. The timer must be able to count intervals of up to 1000 clock cycles.
18. Explain Glitches in sequential circuits?
19. Explain arrival time and required time with suitable diagrams? Brief about how memory modules are constructed?
20. Discuss coding Guidelines
21. Discuss guidelines for clocks and resets.
22. Compare ASIC and FPGA in detail with design flow diagrams
23. Discuss on Partitioning for synthesis with all necessary diagrams.
24. Explain Static Timing Analysis.
25. Design a circuit that counts 16 clock cycles and produces a control signal, "ctrl", that is '1', during every fifth and Tenth cycle. Also, develop a Verilog model for the same Tone.

MODULE-4

1. Explain unsigned and signed inters and their application in digital electronics.
2. Illustrate representation of unsigned and signed numbers in Verilog.
3. Give implementation of a zero extension and termination in a circuit. Represent a zero
4. Design a 4-bit unsigned adder using full adders/ ripple carry and write Verilog code for the same.
5. How the full adder equations are reformulated for faster operation of ripple carry adder circuit.
6. Explain the working cell of full-carry-chain adder or Manchester adder.
7. Develop a Verilog behavioral model of an adder/subtracter for 12-bit unsigned binary numbers. The circuit has data inputs x and y, a data output s, a control input mode that is 0 for addition and 1 for

subtraction, and an output ovf_unf that is 1 when an addition overflow or a subtraction underflow occurs.

8. Develop a verification test bench for the adder/subtracter that compares the result with the result of addition or subtraction performed on values of type integer.
9. Implement a combinational multiplier constructed from adders for partial products.
10. Design an adder/subtracter for both unsigned and 2s-complement numbers.
11. What number is represented by the signed fixed-point binary number 111101, assuming the binary point are four places from the right?
12. Write a behavioral Verilog Code for BCD to 7-Segment Decoder and draw the interface schematic.
13. Give the representation of Block Diagram of a 4-Bit CLA and extend this to 32 bit adder. Also, write a Verilog code for its FPGA execution.
14. Convert the 40.15625 decimal floating point number in to 32 bit single precision IEEE floating point representation.
15. Write behavioral model of a multiplier for unsigned binary numbers to multiply a 4-bit multiplicand by a 4-bit multiplier to give an 8-bit product.
16. Express the number 4.5 in floating-point format with 5 bits of exponent and 12 bits of mantissa magnitude.
17. Convert the following 32-bit binary number to its decimal floating point equivalent: 10111110101000000000000000000000
18. Resize the 2s-complement numbers 01110001 and 11110011 to 12 bits and 6 bits. In each case, does the result correctly represent the same value as the original?
19. Illustrate design of a binary multiplier control state graph.
20. Design and implement a serial-parallel multiplier or Shift-and-Add Multiplier and explain with the help of an example by incorporating Clock. (both are same)
21. Design an array multiplier and write Verilog Code for 4 × 4 Array Multiplier.
22. What is the range and precision of each of the following signed 2s-complement fixed-point representations, with *m* pre-binary-point and *f* post-binary-point bits:
    i) 14 bits, with m=6 and f =8
    ii) 8 bits, with m=4 and f=12
23. Draw the block diagram of 3X3 Array multiplier
24. Discuss the procedure used for Signed Integer/Fraction Multiplier operation.
25. Write a behavioral code for a signed multiplier and explain the logic used.
26. What numbers are represented by the following signed 2s-complement fixed-point numbers, assuming the binary point is four places from the right: 00101100 and 11111101?
27. Develop a Verilog model of a code converter to convert the 4-bit Gray code to a 4-bit unsigned binary integer
28. Draw the block diagram, state graph of Binary multiplier and write the Verilog code for same.
29. How to resize the unsigned Integers? Discuss in detail with suitable example**.**
30. Convert the given decimal floating point numbers in to 32 bit single precision IEEE floating point representation.
31. Show the multiplication of m x n using add and shift method with steps, generate the control state graph and table which defines the operation of a binary multiplier and Also write the Verilog code for same.

MODULE-5

1. Define a Mealy and Moore State Machines with neat block diagram
2. Bring out the difference between Mealy and Moore sequential circuits
3. Design a Mealy sequential circuit to convert given BCD code in to Ex-3 code using D or T or JK flip-flop with a suitable state diagram, state table, and excitation table.
4. Develop a Behavioural Model for BCD to Ex-3 code converter.
5. Develop a Verilog code for BCD to Ex-3 code converter using Equations
6. Design the state diagram of a Mealy machine to detect an input sequence 10110 which could overlap. An output 1 is generated when the sequences is detected
7. Design a Moore sequential circuit to convert non return –to-zero (NRZ) code to Manchester Code
8. Develop the Mealy type FSM for the input sequence 101 which could overlap. An output 1 is to be generated when the sequence is detected. Realize the circuit for this sequence.
9. Design the state diagram of a Mealy machine to detect an input sequence 10110 which could overlap. An output 1 is generated when the sequences is detected
10. Develop a Verilog code for serial adder with suitable logic diagram, state diagram, and state table.
11. Design a Serial adder using Mealy Type FSM and write the Verilog code for same with circuit diagram
12. Design a Serial adder using Moore Type FSM and write the Verilog code for same with circuit diagram
13. Discuss the signal integrity issues in PCB design and techniques used to solve the signal integrity issues.
14. A sequential Circuit has one input(X) and one Output (Z).The circuit examines groups of four consecutive inputs and produces an output Z=1, if the input sequence 0101 or 1001 occurs. The circuit resets after every four inputs. Find the mealy state graph

15. Develop the Mealy type FSM for the input sequence 1001 which could overlap. An output 1 is to be generated when the sequence is detected. Realize the circuit for this sequence.
16. Develop the Mealy type FSM for the input sequence 1100 which could overlap. An output 1 is to be generated when the sequence is detected. Realize the circuit for this sequence.
17. Develop the Mealy type FSM for the input sequence 0110 which could overlap. An output 1 is to be generated when the sequence is detected. Realize the circuit for this sequence
18. Design a 4-digit decimal counter with a 7-segment LED display: two 74LS390 dual decade counters, four 74LS47 BCD to 7-segment decoders, four 7-segment displays, plus any additional gates required.
19. Discuss about: i) Programmable array logic, ii) Complex PLDs
20. Discuss the salient features of the following with suitable diagram.
    i. Output circuit of a PAL16R8
    ii. Generic array logic (GAL) GAL22V10
21. With a neat diagram, explain the internal organization of FPGA
22. Explain typical organization of an FPGA I/O block with suitable block diagram.
23. Discuss packaging and circuit boards in detail.
24. Discuss internal organization of CPLD with neat diagram.
25. Design a mealy State machine for the sequence shown below. The output *Z* should be 1 if the input sequence ends in either 010 or 1001, and *Z* should be 0 otherwise.

$X = 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0$

$\uparrow\ \uparrow\ \uparrow\uparrow\ \uparrow\ \uparrow$

$a\ \ b\ \ c\ d\ \ e\ \ f$

$Z = 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0$

26. Develop the Mealy and Moore type FSM for the sequence detector sequence shown below. The circuit has one input X and Output Z. Realize the circuit for Mealy type FSM. Develop the Verilog code for same.

$X = 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0$

$Z = 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0$