
B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

June 2025 Semester End Main Examinations

Programme: B.E. **Semester: III**

Branch: Electronics and Communication Engineering **Duration: 3 hrs.**

Course Code: 23EC3ESHDL **Max Marks: 100**

Course: HDL Programming

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

			UNIT -- I	*CO*	*PO*	Marks
1	a)		Illustrate with a neat flow diagram, the steps involved in a typical designing of VLSI IC circuits.	--	--	**10**
	b)		Briefly describe the various operators used in Verilog.	--	--	**10**
			OR			
2	a)		Explain top-down and bottom-up design methodology with the help of a suitable example.	--	--	**10**
	b)		Describe the components of a Verilog module. Also indicate the port connections between design and stimulus blocks.	--	--	**10**
			UNIT -- II			
3	a)		Given reg [7:0] C; reg signed [7:0] D; reg E; reg [8:0] Y; reg signed [7:0] A = 8'hD5; Evaluate the following i. C = A << 4 ii. C = A >>> 4 iii. D = A >> 4 iv. D = A <<< 4 v. E = ~(|A) vi. Y = {A[3:0], 4'b0011, A[7:4]};	*CO1*	*PO1*	**6**

		b)	Design a Verilog code for 4:1 multiplexer with gate-level modeling. Also write the stimulus to test the design.	*CO3*	*PO3*	**8**
		c)	Describe Gate Delays with Verilog format.	--	--	**6**
			OR			

	4	a)	Given
			wire a = 1'b0;
			wire [1:0] b = 2'b10; wire [2:0] c = 3'b101;
			Evaluate
			i. {4{a}, &b}
			ii. {4{a}, 2{b}, |c}
			iii. {4{a}, c, (a && b)}
				CO1	*PO1*	**6**

		b)	Design a Verilog code for 4-bit Ripple Carry Adder with full adder as building block. Also realize the full adder using dataflow modeling.	*CO3*	*PO3*	**8**
		c)	Explain implicit continuous assignments using logical operators as example.	--	--	**6**
			UNIT - III			
	5	a)	Discuss the Verilog case statements with syntax. Apply the concept of Verilog case on output Q for a JK flip-flop to describe its behaviour. Write the stimulus block to test the code.	*CO1*	*PO1*	**10**
		b)	Explain Conditional Statements with suitable example.	--	--	**10**
			OR			
	6	a)	Explain Verilog loop statements with syntax and example. Generate the clock pulse of time period 40 ns with 10% duty cycle using forever statement. Initially clock is at logic 1 at 0 ns.	*CO1*	*PO1*	**10**
		b)	Briefly discuss blocking and non-blocking statements, sequential and parallel blocks in the context of behavioural modeling.	--	--	**10**
			UNIT - IV			
	7	a)	With the help of a neat flowchart, describe the logic synthesis process of converting RTL to gates.	--	--	**10**

		b)	Analyze the following snippet of Verilog code, write the complete Verilog description and interpreted logic diagram with combination of multiplexers and gates. Also write the stimulus block to test the design.
			`always @(x)`
			`begin`
			`  if (x[0]) y = 2'b00;`
			`  else if (x[1]) y = 2'b01;`
			`  else if (x[2]) y = 2'b10;`
			`  else if (x[3]) y = 2'b11;`
			`  else y = 2'bxx;`
			`end`
				CO2	*PO2*	**10**

			OR			
	8	a)	Discuss Verilog modeling tips with the intention of getting optimized hardware as the output of logic synthesis.	--	--	**10**

		b)	Analyze the Verilog code snippet given below. Complete the code and draw the synthesized logic diagram. Also write the stimulus block to test the design.
			`always @ (a, b, ct)`
			`begin`
			`  case (ct)`
			`  1'b0: d = a + b;`
			`  1'b1: ;`
			`  endcase`
			`end`
				CO2	***PO2***	**10**

			UNIT - V			
	9	a)	With the help of neat diagrams, briefly describe the building blocks of an FPGA.	--	--	**10**
		b)	Develop the Moore type FSM to detect the sequence 101 with overlapping allowed. Develop a synthesizable Verilog module for the same.	***CO3***	***PO3***	**10**
			OR			
	10	a)	With the help of neat flowchart, describe the VLSI System design using FPGA.	--	--	**10**
		b)	Develop a synthesizable Verilog module for serial adder using Moore machine.	***CO3***	***PO3***	**10**

****