SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

QUESTION BANK

18ECE324 –DIGITAL DESIGN THROUGH VERILOG HDL

Question No.	Questions	PO Attainment
	UNIT – 1	
	PART A (2 Marks)	
1	Define verilog HDL?	PO1,PO2
2	List levels of design description in verilog HDL?	PO1,PO2
3	Describe is concurrency?	PO1,PO2
4	What is simulation and synthesis?	PO1,PO2
5	What is functional verification?	PO1,PO2
6	What are system tasks?	PO1,PO2
7	Write short notes on programming language interface (PLI).	PO1,PO2
8	What is module?	PO1,PO2
9	What is a simulation and synthesis tool?	PO1,PO2
10	What is test bench?	PO1,PO2
11	Define keywords and identifiers?	PO1,PO2
12	What are white space characters?	PO1,PO2
13	Define comments and numbers?	PO1,PO2
14	Define strings and logic values?	PO1,PO2
15	What is a data types? And what are those?	PO1,PO2
16	Define scalars and vectors?	PO1,PO2
23	Give the functioning of "$ monitor" on system task?	PO1,PO2
24	Define verilog HDL?	PO1,PO2
25	List levels of design description in verilog HDL?	PO1,PO2
26	Describe is concurrency?	PO1,PO2
	PART-B (10 Marks)	
1	Compare the advantages of Verilog HDL with traditional schematic-based design.	PO1,PO2
2	Label the useful features of Verilog HDL for hardware design.	PO1,PO2
3	Explain in detail about top-down and bottom-up design methodologies for digital design with an example.	PO1,PO2
4	List the differences between modules and module instances in Verilog.	PO1,PO2
5	Define a stimulus block and design block in Verilog HDL.	PO1,PO2
6	Describe the various levels of abstraction in Verilog HDL.	PO1,PO2
7	Identify the components required for the simulation of a digital design in Verilog HDL.	PO1,PO2
8	Express various lexical conventions for operators, comments, whitespace, numbers, strings and identifiers in Verilog HDL.	PO1,PO2
9	Name the basic compiler directives to define macros and include files.	PO1,PO2
10	Discuss in detail about the various data types used in Verilog.	PO1,PO2

11	Classify the different categories of operators used in Verilog HDL.	PO1,PO2
12	Summarize useful system tasks for displaying and monitoring information in Verilog.	PO1,PO2
13	Label the useful features of Verilog HDL for hardware design.	PO1,PO2
14	Explain in detail about top-down and bottom-up design methodologies for digital design with an example.	PO1,PO2
15	Define the following terms relevant to Verilog HDL. (a) Module (b) Test bench.	PO1,PO2
16	Define the following terms relevant to Verilog HDL, (a) Simulation versus synthesis. (b) PLI	PO1,PO2
UNIT – 2		
PART A (2 Marks)		
1	What is gate level modeling?	PO1 – PO5
2	What is AND gate primitive?	PO1 – PO5
3	Define tri-state gate?	PO1 – PO5
4	What is array of instances of primitives?	PO1 – PO5
5	Define delay?	PO1 – PO5
6	Define strengths and content resolution?	PO1 – PO5
7	Write a verilog code using data flow model.	PO1 – PO5
8	What is meant by assignment to vectors in Verilog HDL?	PO1 – PO5
9	Explain operators in data flow?	PO1 – PO5
10	Write a verilog code using data flow model.	PO1 – PO5
11	What is continuous assignment structure?	PO1 – PO5
12	What is assignment to vectors?	PO1 – PO5
13	Define operators in verilog?	PO1 – PO5
14	If a= 3'b111 , b=3'b100 then find Y for the following (i) Y= a ^ b (ii) Y= a&&b	PO1 – PO5
15	Explain about concatenation operator.	PO1 – PO5
16	What is the difference between & and &&?	PO1 – PO5
17	Write a verilog code for half adder using in data flow modeling?	PO1 – PO5
PART-B (10 Marks)		
1	Design a Verilog module for a 4-to-1 multiplexer using basic logic gates.	PO1 – PO5
2	Develop Verilog module of a clocked RS flip-flop with NAND gates.	PO1 – PO5
3	Construct a 4-bit ripple carry full adder from four 1-bit full adders using Verilog description.	PO1 – PO5
4	Derive the stimulus for 4-bit ripple carry full adder from four 1-bit full adders.	PO1 – PO5
5	Formulate various logic gate primitives provided in Verilog.	PO1 – PO5

6	Explain min, max and typical delays in the gate-level design.	PO1 – PO5
7	Determine the types of delays that are associated with a primitive gate from its input to output.	PO1 – PO5
8	Categorize tri-state buffers that are available in Verilog as primitives.	PO1 – PO5
9	Design a Verilog description for module D with Delay.	PO1 – PO5
10	Develop a Verilog module of a positive edge-triggered D flip-flop.	PO1 – PO5
11	Illustrate characteristics of Continuous assignment in Data flow modelling.	PO1 – PO5
12	Contrast an implicit continuous assignment with a regular continuous assignment with an example.	PO1 – PO5
13	Explain Implicit Net Declaration in Data flow modelling.	PO1 – PO5
14	Interpret the design in terms of expressions instead of primitive gates in Data flow modelling.	PO1 – PO5
15	Exhibit different ways of specifying delays in continuous assignment statements in Data flow modelling.	PO1 – PO5
16	Show the different type of operators provided by Verilog in Data flow modelling.	PO1 – PO5
17	Develop a Verilog description for a 4-bit Full Adder using Dataflow Operators.	PO1 – PO5
18	Determine a Verilog description for a 4-bit Full Adder with Carry Lookahead adder in Data flow modelling.	PO1 – PO5
	UNIT – 3	
	PART A (2 Marks)	
1	What is behavioral modeling?	PO1 – PO5
2	What are operations and assignments?	PO1 – PO5
3	Define functional Bifurcation.	PO1 – PO5
4	Define initial construct.	PO1 – PO5
5	Define always construct.	PO1 – PO5
6	Explain assignments with delays	PO1 – PO5
7	Define wait construct	PO1 – PO5
8	Explain multiple always blocks	PO1 – PO5
9	Define blocking and non-blocking assignments	PO1 – PO5
10	Explain the case statement	PO1 – PO5
11	Explain if and if-else construct	PO1 – PO5
12	Explain assign and de-assign construct.	PO1 – PO5
13	Define repeat construct	PO1 – PO5
14	Write the syntax for a for loop	PO1 – PO5
	PART-B (10 Marks)	
1	Explain the significance of structured procedures **always** and	PO1 – PO5

	initial in behavioural modelling.	
2	Examine blocking and nonblocking procedural assignments with an example.	PO1 – PO5
3	Analyze the following timing control mechanisms with examples in behavioural modelling: i) Delay-Based ii) Event-Based iii) Level-Sensitive	PO1 – PO5
4	Design a Verilog description for 4-to-1 multiplexer with Case Statement.	PO1 – PO5
5	Compare the two variations of the **case** statement with an example.	PO1 – PO5
6	Develop a Verilog code to increment count from 0 to 127 and exit count at 128 using While Loop.	PO1 – PO5
7	Explain the use of the Forever statement with an example.	PO1 – PO5
8	Develop a Verilog code for 4-bit Binary Counter using behavioural modelling.	PO1 – PO5
9	Write the differences between begin-end and fork-blocks with examples.	PO1 – PO5
10	Design up counter coding procedural assignment.	PO1 – PO5
11	Write up counter test bench, simulation results	PO1 – PO5
	UNIT – 4	
	PART A (2 Marks)	
1	Explain about CMOS inverter.	PO1 – PO5
2	Define resistive switches?	PO1 – PO5
3	Design half subtractor using CMOS switch?	PO1 – PO5
4	Explain the operation of NMOS switch?	PO1 – PO5
5	How to instante with strength and delays?	PO1 – PO5
6	What is continuous assignment structure?	PO1 – PO5
7	Explain assignment with delays?	PO1 – PO5
8	Define UDP?	PO1 – PO5
9	List the types of UDP?	PO1 – PO5
10	Compare combinational and sequential UDP	PO1 – PO5
	PART-B (10 Marks)	
1	Illustrate the modelling procedure of a CMOS device with nmos and pmos devices.	PO1 – PO5
2	Sketch the symbols of Bidirectional Switches.	PO1 – PO5
3	Categorize the main differences between regular switches and resistive switches.	PO1 – PO5
4	Interpret Delay Specification on MOS and CMOS Switches.	PO1 – PO5
5	Develop a Switch-Level Verilog Description of 2-to-1 Multiplexer.	PO1 – PO5
6	Sketch the gate and the switch-level circuit diagram for Nor Gate.	PO1 – PO5
7	Design Switch-Level Verilog description for CMOS inverter by using MOS switches.	PO1 – PO5

8	Determine User-Defined Primitives (UDP) with examples.	PO1 – PO5
UNIT – 5		
PART A (2 Marks)		
1	Define system task	PO1 – PO5
2	Define system function.	PO1 – PO5
3	Explain $display Task.	PO1 – PO5
4	What are system tasks?	PO1 – PO5
5	Do function require minimum number of argument. Justify it	PO1 – PO5
6	Explain $random.	PO1 – PO5
7	What are task. List out its syntax.	PO1 – PO5
8	Define$ readmem	PO1 – PO5
9	Define Compiler Directive?	PO1 – PO5
10	What are Compiler Directives?	PO1 – PO5
PART-B (10 Marks)		
1	Compare the differences between tasks and functions.	PO1 – PO5
2	Identify the conditions required for task to be defined.	PO1 – PO5
4	Explain the conditions necessary for functions to be defined.	PO1 – PO5
6	Develop a Verilog Description for the definition and invocation of the function calc_parity.	PO1 – PO5
7	Implement Verilog code for a function that shifts a 32-bit value to the left or right by one bit, based on a control signal.	PO1 – PO5
8	Demonstrate the use of the disable statement using an OR gate Verilog module.	PO1 – PO5
9	Explain the concept of Name Events with an example.	PO1 – PO5
10	Sketch the module hierarchy of the path name.	PO1 – PO5
11	Identify types of delay models used in Verilog simulation.	PO1 – PO5
12	Explain how distributed delays are specified in gate level and dataflow description.	PO1 – PO5
13	Discuss in detail about various aspects of path delay modelling.	PO1 – PO5
14	Develop a Verilog module to illustrate the use of the **‘define** directive.	PO1 – PO5
15	Generalize the components of **‘timescale** directive.	PO1 – PO5
16	Develop a Verilog module for wallace tree multiplier and write test bench program for it	PO1 – PO5
17	Design of Carry Look Ahead Adder and write its Verilog module and testbench program	PO1 – PO5
18	Develop a Verilog module for floating point encoder and write its testbench program	PO1 – PO5