Subject Code: 24EC11RC17 **R-24** **Reg No:** | | | | | | | | | | | |

GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING FOR WOMEN

(Autonomous)

(Affiliated to Andhra University, Visakhapatnam)

II B.Tech. - II Semester Regular Examinations, April – 2026

DIGITAL SYSTEM DESIGN

[Electronics and Communication Engineering]

1. All questions carry equal marks
2. Must answer all parts of the question at one place

Time: 3Hrs. **Max Marks: 70**

UNIT-I

1. a) Explain the concept of top-down design in System Verilog with an example. [7M]
   b) Compare dynamic arrays and associative arrays in System Verilog with examples. [7M]

(OR)

2. a) Compare gate-level and switch-level modelling in System Verilog. Show with examples how a 2-input AND gate is described at the gate level and a CMOS inverter at the switch level. [7M]
   b) Discuss the guidelines for using 2-state vs. 4-state logic values in Verilog. Why are 4-state values critical in verification? [7M]

UNIT-II

3. a) Construct an 8-bit binary adder in System Verilog and develop a testbench using basic constructs to verify its correctness with multiple input combinations. [7M]
   b) Describe the role of mod-ports in System Verilog interfaces. Provide an example of how mod-ports control signal direction between modules. [7M]

(OR)

4. a) Explain the purpose of the program block in a System Verilog testbench and how it differs from a module. Additionally, describe the role of the final block and provide a scenario where it is useful. [7M]
   b) Construct SR flip-flop and test bench using System-Verilog [7M]

UNIT-III

5. a) Differentiate between case, caseX, and caseZ statements in System Verilog. Illustrate with suitable examples. [7M]
   b) Develop a 4-to-1 multiplexer in System Verilog using if-else and priority if statements. [7M]

(OR)

6. a) Discuss arguments in System Verilog tasks/functions. Provide a simple example how arrays, structures, and unions can be used as formal arguments. [7M]
   b) Explain the difference between tasks and functions in System Verilog. Mention at least two key distinctions with examples. [7M]

UNIT-IV

7. a) Construct 1010 mealy type overlapped finite state machine using user defined types. [7M]
   b) Develop a System Verilog FSM using reversed case statements with enumerated types. Demonstrate how unique case improves reliability in state transitions. [7M]

(OR)

8. a) Model a Mealy FSM in System Verilog using enumerated types. Show how outputs depend on both present state and input. [7M]
   b) Model a FSM in System Verilog that demonstrates reset behavior using 2-state types. Show how unused state values can be specified and handled in the design. [7M]

UNIT-V

9.	a)	Draw and explain the architecture and components of a System Verilog test bench.	[7M]
	b)	Discuss the importance of interfaces and clock/reset generation in System Verilog testbenches. Provide an example of an interface with clock and reset signals.	[7M]

(OR)

10.	a)	Construct a self-checking testbench in System Verilog for a 4-bit adder. Show how the testbench automatically verifies correctness of outputs.	[7M]
	b)	Explain in detail about Randomization concept in System Verilog.	[7M]