

Basavarajeswari Group of Institutions  
**BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT**  
 (Autonomous Institute under Visvesvaraya Technological University, Belagavi)

USN 

--	--	--	--	--	--	--	--	--

Course Code 

2	2	E	C	3	2
---	---	---	---	---	---

Third Semester B.E. Degree Examinations, March/April 2024  
**DIGITAL CIRCUIT DESIGN USING VERILOG**

Duration: 3 hrs

Max. Marks: 100

**Note:** 1. Answer any FIVE full questions, choosing ONE full question from each module.  
 2. Missing data, if any, may be suitably assumed

<u>Q. No</u>	<u>Question</u>	<u>Marks</u>	<u>(RBTL:CO:PI)</u>
<b><u>MODULE – 1</u></b>			
1. a.	Simplify using K-Map $F(A, B, C, D) = \sum m(0, 7, 8, 9, 10, 12) + \sum d(2, 5, 13)$	06	(3 :1: 1.7.1)
b.	Convert the given expression into canonical form (i) $F(A, B, C) = AC + AB + \overline{BC}$ (ii) $F(A, B, C) = (\overline{A} + \overline{B})(B + C)(A + C)$	06	(3 :1: 1.7.1)
c.	A combinational logic circuit has three inputs and one output, that produces logic 1 output when 2 or more inputs are at logic 1. Derive and simplify the Boolean function and design the combinational logic circuit using logic gates	08	(4 :1: 1.7.1)
<b>(OR)</b>			
2. a.	Simplify using K-Map $F(A, B, C, D) = \pi M(0, 2, 3, 8, 9, 12, 13, 15)$	06	(3 :1: 1.7.1)
b.	Simplify using 3 variable MEV map. $F(A, B, C, D) = \sum m(3, 4, 5, 7, 8, 11, 12, 13, 15)$	06	(3 :1: 1.7.1)
c.	Define the following (i) Literal (ii) sum term (iii) Minterms (iv) Canonical POS v. SOP (vi) Maxterm (vii) Canonical SOP (viii) Prime implicants	08	(2 :1: 1.6.1)
<b><u>MODULE – 2</u></b>			
3. a.	Define Lexical Conventions. Explain the following lexical Convention (i) White Space (ii) Number Specification (iii) Identifiers and Keywords	06	(3 :4: 1.6.1)
b.	Define bufif and notif and write gate instantiation of bufif and notif gates.	06	(3 :4: 1.6.1)
c.	Explain the <b>Typical design flow</b> for designing VLSI IC circuit, using flow chart.	08	(2 :4:1.6.1)
<b>(OR)</b>			
4. a.	What are the four values and 8 strength levels supports in Verilog HDL? List out in a neat table.	06	(3 :4: 1.6.1)
b.	Define module. Explain the basic component of a Verilog module	06	(3 :4: 1.6.1)
c.	Write the Verilog code for 4 bit ripple carry full adder using gate level modeling	08	(2 :4: 1.7.1)

**Note: (RBTL - Revised Bloom's Taxonomy Level: CO - Course Outcome: PI- Performance Indicator)**

### MODULE – 3

- |    |  |           |               |
|----|--|-----------|---------------|
| 5. | a. Define multiplexer. Design 8:1 Multiplexer using logic gates.       | <b>06</b> | (3 :2: 1.6.1) |
|    | b. Define Full Adder. Implement full adder using IC74LS138.            | <b>06</b> | (3 :2:1.7.1)  |
|    | c. Define Look Ahead Carry adder. Design 4-bit Look Ahead Carry adder. | <b>08</b> | (3 :2: 1.6.1) |

**(OR)**

- |    |  |           |               |
|----|--|-----------|---------------|
| 6. | a. Implement the following Boolean function using 4:1 mux<br>$F(A, B, C) = \overline{A}B + AC + \overline{B}C$ | <b>06</b> | (3 :2:1.7.1)  |
|    | b. Explain encoder. Design 4:2 priority encoder.   | <b>06</b> | (3 :2: 1.6.1) |
|    | c. Define comparator. Design 2 bit comparator.   | <b>08</b> | (3 :2: 1.6.1) |

### MODULE – 4

- |    |   |           |               |
|----|---|-----------|---------------|
| 7. | a. Define sequential circuit and explain the universal sequential model                 | <b>06</b> | (2 :3: 1.6.1) |
|    | b. Define Asynchronous counter. Explain Divide by 8 Ripple counter with timing diagram. | <b>06</b> | (2 :3: 1.6.1) |
|    | c. Design a Mealy type detector to detect a serial input sequence of 101.               | <b>08</b> | (3 :3: 1.7.1) |

**(OR)**

- |    |   |           |               |
|----|---|-----------|---------------|
| 8. | a. Define Shift register. With a neat block diagram and truth table explain 4-bit SISO Register (Assume the serial data 1001) | <b>06</b> | (2 :3: 1.6.1) |
|    | b. Explain Mealy and Moore Models with neat block diagram   | <b>06</b> | (2 :3: 1.6.1) |
|    | c. Define Latch. With a neat circuit and timing diagram explain SR Latch using NOR gate.                                      | <b>08</b> | (2 :3: 1.6.1) |

### MODULE – 5

- |    |   |           |               |
|----|---|-----------|---------------|
| 9. | a. Explain the following Operator types with example.<br>(i) Arithmetic Operator (ii) Bitwise Operator (iii) Shift operator | <b>06</b> | (2 :4: 1.6.1) |
|    | b. Explain the initial statement with an examples.  | <b>06</b> | (2 :4: 1.6.1) |
|    | c. Design 4:1 mux and write the Verilog code in Data-flow (i) using logic equation (ii) conditional operator.               | <b>08</b> | (3 :4: 1.6.1) |

**(OR)**

- |     |   |           |               |
|-----|---|-----------|---------------|
| 10. | a. What would be the output of the following: a=4'b1001, b=4'b1101<br>(i) a & b (ii) &a (iii) a>>2 (iv) y= {a, b} (v) {2{a}} (vi) a ^ b | <b>06</b> | (3 :4: 1.6.1) |
|     | b. Write a Verilog code for 4:1 MUX using Case statement.   | <b>06</b> | (3 :4: 1.6.1) |
|     | c. Explain the Looping statements with an example for each: FOR, WHILE, REPEAT, FOREVER.  | <b>08</b> | (2 :4: 1.6.1) |

\*\* \*\* \*