Seat No.: _____ Enrolment No._____

GUJARAT TECHNOLOGICAL UNIVERSITY

ME – SEMESTER III (OLD) – • EXAMINATION – SUMMER 2016

Subject Code: 730303 **Date:05/05/2016**

Subject Name: VLSI Design

Time:10:30 am to 01:00 pm **Total Marks: 70**

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q.1	**(a)**	Explain the usage of net and register in verilog with appropriate example. How they differ from eachother?	**07**
	(b)	State rules for developing UDPs.	**07**
Q.2	**(a)**	With diagram explain FUSE and ANTIFUSE mechanism of FPGA programming	**07**
	(b)	Explain in detail delay-based, event-based and level sensitive timing controls in verilog with example.	**07**
		OR	
	(b)	Write a verilog program for 8-bit Universal shift register with shift right, shift left and parallel load capabilities.	**07**
Q.3		With the help of T-flipflop design a decade counter. Show all design steps. Write verilog program of the same using structural model style. Also write verilog program of all components used in your design.	**14**
		OR	
Q.3		With the help of T-flipflop design a modulo 13 counter. Show all design steps. Write verilog program of the same using structural model style. Also write verilog program of all components used in your design.	**14**
Q.4		Explain the architecture of Spartan3e FPGA family	**14**
		OR	
Q.4		Consider Mealy Finite State Machine (FSM), with one input X and one output Z. The FSM asserts its output Z when it recognizes the “10011” input bit sequence. Implement the state diagram for above & write verilog code for it.	**14**
Q.5	**(a)**	Write a verilog program for full-subtractor using half-subtractor and logic gates. Write modules for half-subtractor and the gates used. Draw the circuit diagram of the whole design.	**07**
	(b)	Explain mux- versus LUT-based logic blocks.	**07**
		OR	
Q.5	**(a)**	Write a function that calculates the even parity of 16 bit address & returns the value.	**07**
	(b)	Compare the following: (1) Tasks and Functions (2) CPLD and FPGA	**07**

****