SRN

PES University, Bengaluru
(Established under Karnataka Act No. 16 of 2013)

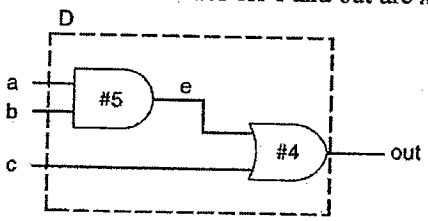
UE16EC532

MAY 2017: END SEMESTER ASSESSMENT (ESA) M.TECH. II SEMESTER

UE16EC532 - Digital System Design Using HDL

Time:3 Hrs | Answer All Questions | Max Marks: 100

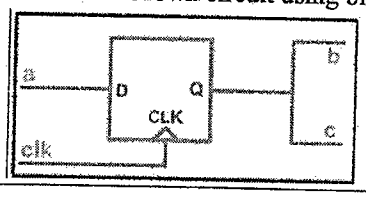
1	a	**Identify the little endian and big endian declaration in the below given Verilog code and write the output.**
		module lilbigendian;
		reg [2:0] a;
		reg [0:2] b;
		initial begin
		a=3'b100;
		b=3'b001;
		$display(" a[0]=%b a[1]=%b a[2]=%b\n",a[0],a[1],a[2]);
		$display(" b[0]=%b b[1]=%b b[2]=%b\n",b[0],b[1],b[2]);
		$display("a=%d \t b=%d",a,b);
		end
			05

	b	The ripple carry counter is designed using negative edge-triggered toggle flip-flops (T_FF). Each of the T_FFs is created from negative edge-triggered D-flip-flops (D_FF) and inverters (assuming q_bar output is not available on the D_FF), Write structural Verilog code to implement 4 bit ripple carry counter and T_FF. Assume module D_FF is available in the library.	10
	c	Write test bench for ripple counter designed in question 1b.	05
2	a	For the below given logic network, write structural level Verilog code incorporating the delay mentioned for each gate. Write the waveform for the same module using the test bench given below. Initial values for e and out are x.  Test bench: module stimulus; reg A, B, C; wire OUT; D d1(OUT, A, B, C); initial begin A= 1'b0; B= 1'b0; C= 1'b0; #10 A= 1'b1; B= 1'b1; C= 1'b1; #10 A= 1'b1; B= 1'b0; C= 1'b0; #20 $finish; end endmodule	10
	b	Write Verilog code for 2 digit BCD adder using dataflow model. Explain BCD addition with an example. Consider X, Y each 8 bits wide representing two digits primary inputs and Z , 12 bits wide representing 3 digit primary output.	10

P. To

SRN

3	a	Explain regular delay control and intra delay control statements with an example for each.
		Write the output of the following code:
		module delay_control_check;
		parameter L_DELAY = 10;
		reg a,b,c;
		reg[3:0] rval;
		initial begin
		a = 0; b=0; c=0; rval = L_DELAY - 5;
		#10 b = 1;
		#20 c = a + b;
		#5 a = c;
		#10 rval = L_DELAY + a;
		b= 0; c = 0;
		a = #5 (b + c);
		rval = #20 L_DELAY + a - 4;
		end
		initial begin
		$monitor($time,"Value a = %b, b= %b, c=%b rval=%d\n",a,b,c,rval);
		#500 $finish;
		end
		endmodule
			08

	b	Write veriog code to model Data buffer using behavioral model. It stores the data on every +ve clock edge . Data buffer is an array and can store upto 8 set of data, data width is 16 bits. When Data_start signal= 1 at every positive edge of the clock the data is stored in the data buffer for 8 clock cycles (Note: After it receives a data_start signal, reads data for next 8 cycles)	08
	c	Write Verilog code to implement the below shown circuit using blocking statements. 	04
4	a	Write Verilog code to design negative edge triggered D Flip Flop with asynchronous reset using procedural continuous assign, deassign statements.	10
	b	Define a Verilog module that contains the function shift, that shifts a 32-bit value to the left or right by one bit based on a control signal and returns a 32 bit shifted value.	06
	c	Write any four differences between Function and Task	04
5	a	Write UDP code for negative edge triggered D Flip Flop with active high synchronous clear signal.	05
	b	Write FSM for newspaper vending machine. Write Verilog code **with function** to implement FSM . The design specifications are given below: • Assume that the newspaper cost 15 cents. • The coin acceptor takes only nickels and dimes. • Exact change must be provided. The acceptor does not return extra money. • Valid combinations including order of coins are one nickel and one dime, three nickels, or one dime and one nickel. Two dimes are valid, but the acceptor does not return money.	10
	c	Explain functional and structural coverage that are used to quantify the verification progress.	05