Birla Institute of Technology & Science, Pilani

I[st] Semester 2017-18

MEL G626 VLSI Test and Testability

Mid Semester Test (Closed Book)

MM:30 **Duration: 1.30 Hrs** **Date: 09/10/2017**

Q1:

(A) Draw the diagram for an IC which follows the dual diode protection to provide ESD protection for each of input and output pin. Device the test strategy test pin to pin short. **(3 M)**

(B) Discuss the testing related issues in the ripple counter. Also discuss the Guard banding technique in brief. **(2 M)**

(C) Discuss the functional implementation coverage and explain the 'Multiple state machine coverage' in brief? **(2 M)**

Q2:

(i) Using deterministic verification scheme on the calculator 2, draw and label the waveforms indicating the execution of operations for the following commands including the reset cycle at start:

Port#	DelayN	Cmd	Operand1	Operand2	Tag	Result	Response*/
Port1	0	ADD	"11111111"X	"11111111"X	00	"22222222"X	Good
Port1	0	SUB	"22222222"X	"20202020"X	01	"02020202"X	Good
Port1	0	ADD	"33333333"X	"33333333"X	10	"66666666"X	Good
Port1	0	SUB	"44444444"X	"40404040"X	11	"04040404"X	Good

(ii) Draw the state diagram stimulus generation algorithm for a single port. **(4 M)**

Q3: What are advantages of 'transaction model' based test-bench over 'golden vector' based test-bench also list its disadvantages? **(2 M)**

Q4: Explain generic scoreboard components in brief. Also list the drawback of using programming language interface (PLI) as compared with separate C++ test bench in the verification process. **(3 M)**

Q5:

(A) A system Verilog declaration are given as:

bit [3:0] [7:0] abc [3]; bit [31:0] xyz = 32'ha822_abcd; abc [1]= xyz;

(a) For the above packed array declaration find the value of abc [1] [3] and abc [1] [1] [6]. (b) Now declare a variable pqr to observe the 32 bit data as pqr [7] to pqr [0] each of 4 bit width. If pqr =abc [1], find pqr [3] ? **(2 M)**

(B) Comment on the following 'alias' statement?

```
reg clock;
reg clk;
alias clk=clock;
```

(1 M)

Q6: Write the output of following system verilog module after simulation?

```
module abc;
initial
begin
bit [15:0] assoc [byte], idx=1;
repeat (16) begin
assoc [idx] = idx; idx = idx<<1;
end
foreach (assoc[i])
$display ( "“assoc [%h]=%h”",i,assoc[i]);
end
endmodule
```

(3 M)

Q7: For the following system verilog program write the display after execution. **(3 M)**

```
module abc  ();
initial
begin
int j=2; int y[$]; int xm[$]={5,6,7,6}; int x[$]={1,6,9};
x.insert(1,j); x.insert(3,5); x.insert(4,6); x.insert(5,7); x.insert(6,6); $display ("%p",x);
x.delete(2); $display ("%p",x);  x.push_front (7); $display ("%p",x); j=x.pop_back;
$display ("%p",x); y=x.unique(); $display ("%p",y);
end
endmodule
```

Q8:

(A) Write a System Verilog program using task such that a *single call* of same task should take different int type ‘a’ and ‘b’ values and print results as “ a is value , b is value, sum is value” for the following initial block of same program. **(3 M)**

```
initial
begin
a=5; b=5; #10 a=10; #10 b=15; #10 a = 20; #10 b=25; #10 a = 30; #10 b=35; #10 $finish;
end
```

(B) Write a system Verilog program which declare two handles t1, t2 for the Transaction class. (i) The t1 points to the first allocated Transaction object (ii) then both t1 and t2 both points to the first allocated Transaction object (iii) then t1 points to second allocated Transaction object and t2 keep pointing to the first allocated Transaction object. **(2 M)**

----------**********----------