Q.P. Code: 23890

Time:-3 Hrs **Marks: 80**

N.B. : 1. Question No. ONE is compulsory
2. Solve any THREE out of remaining questions
3. Assume suitable data if required

Q1. Solve the following (Any Four) **20 Marks**

A. Explain the clocking resources and XADC available in VERTEX-7 FPGAs.

B. Justify that we can randomize objects in SystemVerilog?

C. Explain verification directives 'assert' and 'cover' with suitable example of each?

D. Differentiate between PLI and DPI? Which should be used to instantiate 'C' function in SystemVerilog?

E. What is difference between Static and Non-Static fields of a class?

Q2. A. List the benefits and drawbacks of 'directed' testing and 'random' testing. **05 Marks**

B. Write the SystemVerilog code to: **05 Marks**

Declare a 2-state array, my_array, that holds five 12-bit values.

Initialize my_array so that:

1) my_array[0] = 12'h010
2) my_array[1] = 12'h111,
3) my_array[2] = 12'h234,
4) my_array[3] = 12'h0AB;
5) my_array[3] = 12'h9CD;

Traverse my_array and print out bits [5:4] of each 12-bit element

6) Using a for loop
7) Using a foreach loop

C. Explain each and every statement of the following code: **05 Marks**

```
class Packet
    rand bit [31:0] length;
    constraint c_short {length inside {[1:32]};}
    constraint c_long {length inside {[1000:1023]};}
endclass
Packet p:
initial begin
    p=new();
    p.c_short.constraint_mode(0);
    `SV_RAND_CHECK(p.randomize());
    transmit(p);
    p.constraint_mode(0);
    p.c_short.constraint_mode(1);
    `SV_RAND_CHECK(p.randomize());
    transmit(p);
end // initial
```

D. Explain use of $display, $monitor and $assert in Verilog test benches with suitable examples. **05 Marks**

Q3. A. Justify that there are situations in which it is not beneficial to use an interface. **05 Marks**

B. What do you mean by dynamic array and when it is useful? **05 Marks**

C. Explain about the Timeunit, Timeprecision and timescale. **05 Marks**

D. Explain each and every statement of the following code: **05 Marks**

```
class Packet;
    rand bit [31:0] src, dst, data[8];
    randc bit [7:0] kind;
    constraint c {src > 10; src < 15;}
endclass
Packet p;
initial begin
    p=new();
    if (!p.randomize())
    $finish;
    transmit(p);
end
```

Q4. A. With suitable example explain how 'event' is used for synchronization done in SystemVerilog. **05 Marks**

B. With respect to SystemVerilog explain the following with suitable example **05 Marks**

I. Virtual Interface

II. Virtual Class

C. Write the output of the following bounded mailbox **05 Marks**

```
initial begin
  mbx = new(1);
  fork
    for (int i=1; i<4; i++) begin
      $display("Producer: before put(%0d)", i);
      mbx.put(i);
      $display("Producer: after put(%0d)", i);
    end
    repeat(4) begin
      mbx.get(j);
      $display("%0t: Consumer: Working on (%0d)", $time, j);
      #10ns;
      $display("%0t: Consumer: Done Working on (%0d)", $time, j);
    end
  join
end
```

D. For the interface: **05 Marks**

```
interface ram_if(input bit clk);
   bit wr;
   bit [31:0] data_in;
   bit [15:0] add;
   logic [31:0] data_out;
endinterface
```

Add a clocking block that:

1. Is sensitive to the positive edge of clock
2. All I/O are synchronous to the clock
3. Creates a modport for the testbench called master and for the DUT called slave
4. Will be used in the slave

Q5. A. Describe the 'scope' for each 'bound' in the following code: **05 Marks**

```
int bound;
program automatic p;
    int bound;
    class Foo;
        int bound, array[];
        function void print (int bound);
          for (int i=0; i< bound; i++)
            $display("%m: array[%0d]=%0d", i, array[i]);
        endfunction
    endclass
    initial begin
         int bound = $root. bound;
        Foo bar;
        bar = new();
        bar.print(bound)
        bar.print(bar.bound);
    end
endprogram
```

B. Explain various assertion states and possible outcomes of assertions. **05 Marks**

C. What is the concept of randomization and why it is required in design verification? Give detail explanation with suitable example. **10 Marks**

Q6. A. List and explain the various coverage options available in systemverilog. **05 Marks**

B. Justify that it is necessary to get high code coverage as well as high functional coverage during verification. **05 Marks**

C. With suitable example for each, explain the following with respect to SystemVerilog assertions:

a. Property
b. Sequence **05 Marks**

D. Explain how data type matching is achieved between 'C' and SystemVerilog. **05 Marks**