**AIN SHAMS UNIVERSITY**
**FACULTY OF ENGINEERING**

***Department of Computer and Systems Engineering***
***Senior 1, and Senior 2 Level Students***

| | | |
|---|---|---|
| ***2nd Semester, 2023/2024*** | ***Course Code: CSE 313s*** | ***Time allowed: 1 Hr.*** |

## *Digital Verification*

*The Exam Consists of **six** Questions in **six** Pages.* | ***Maximum Marks: 30 Marks*** | ***1 / 6***

**تعليمات هامة**

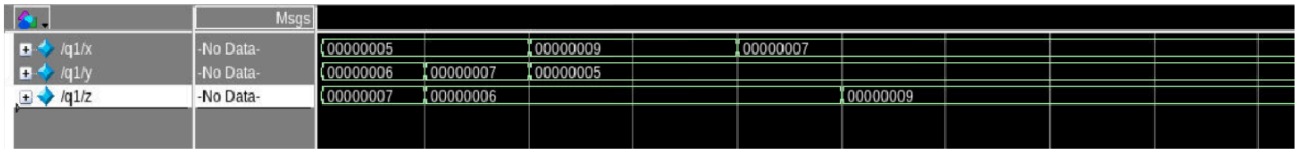
- حيازة التيلفون المحمول مفتوحا داخل لجنة الأمتحان يعتبر حالة غش تستوجب العقاب.
- لا يسمح بدخول سماعة الأذن أو البلوتوث.
- لايسمح بدخول أي كتب أو ملازم أو أوراق داخل اللجنة والمخالفة تعتبر حالة غش.

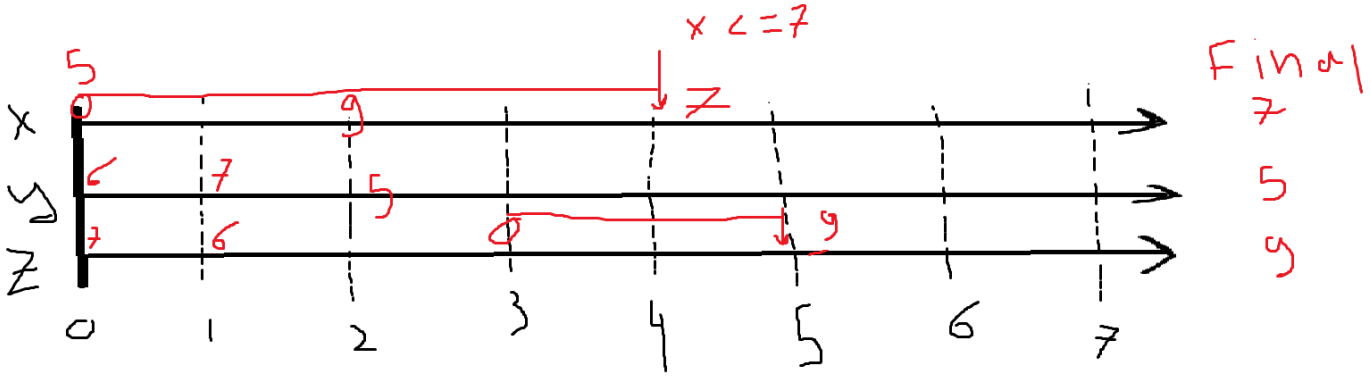
**Question 1**: [5 marks]

Show the value of x, y, and z on the timeline.

```
initial begin
  x=5;y=6;z=7;
  x<=#4z;
  #1;
  z<=y;
  y=y+1;
  #1
  y<=x;
  x=9;
  #1;
  z<=#2x;
end
```

Solution:



**Question 2**: [5 marks]

Calculate the following code coverage items

- statement coverage
- Branch coverage
- toggle coverage for x and y

```
initial begin
  x=1;
  y=0;
end
always @(a or b)begin
  if(a==1)
      x=0;
  else if(b=1)
      y=1;
end
```

```
1 initial begin
2   x=1;
3   y=0;
  end
4 always @(a or b)begin
5   if(a==1)
6     x=0;
7   else if(b=1)
8     y=1;
  end
```

statment covergae:

excuted statments: 1,2,3,4

coverage: 4/8 = 50%

branch coverage:

excuted branches:

coverage: 0/2 = 0%

toggle coverage:

x: 1 yes, 0 no, 1->0 no, 0->1 no

y: 1no, 0 yes, 1->0 no, 0->1 no

coverage 2/8 =25%

**Question 3**: [5 marks]

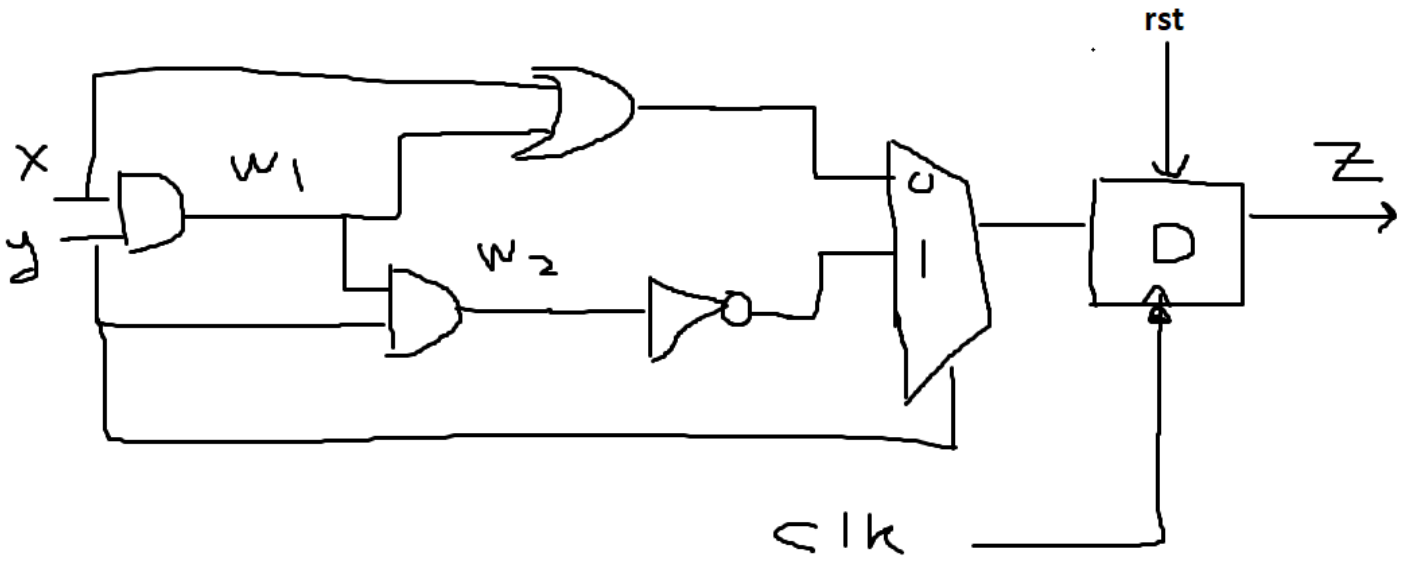
1) What is the difference between static and dynamic verification ?
    a) Static verification is analysing/checking design code without running the code simulation, while dynamic verification involves exercising DUT with different inputs
2) If we have a high functional coverage but low code coverage, what is most likely to be the problem in real life development ?
    a) Verification plan is missing some features
3) What is the difference between theses 3 expressions: **verification, post-silicon validation** and t**esting**
    a) Verification: usually done on design phase using simulation tools/emulators to check design correctness
    Post-silicon validation: Done on test chips usually in design labs to check electric requirements
    Testing: done on production usually at fablabs to check against fabrication faults
4) What is the main disadvantage of pure random testing ?
    a) It requires a lot of time to cover all needed features
5) Hardware acceleration is usually used to speed up verification closure and gain more trust in the design, however it has a couple of drawbacks, what are they ?
    a) High cost
    Limited observability/debugging options

**Question 4**: [5 marks]

Draw the logic diagram for the circuit that represents this module

```
module module_a(input x,y,clk,rst, output reg z);

  wire w1, w2;

  assign w1 = x &y;
  assign w2 = w1 & y;

  always @(posedge clk)
    if(y==1)
      z <= ! w2;
    else
      z <= x || w1;
  always @(posedge rst)
      z <= 0;

endmodule
```

Solution:



**Question 5**: [5 marks]

Write a verification plan for the given binary_counter in the following form.

| Feature | Checkers list | Stimulus | Priority |
|---|---|---|---|
| Reset behaviour | • Count is back to zero with +ve edge of rst<br>• Count stays zero as long as rst=1<br>• Count starts from 0 with -ve ege of rst<br>• Count changes every clk as long as rst=0 | Rst: 0->1->0 | • High<br>• Medium<br>• high<br>• Medium<br>Priorities are always dependent on the actual use case so any different justifiable answer is accepted |
| Clk sensitivity | • Count changes at every +ve edge of clk(while rst=0)<br>• Count stable between clk edges | Clk: Toggling | • High<br>• Medium |
| Output correctness | • Count is the equal to the previous count +1 | Clk,rst: random | • High |

```
module binary_counter(input clk,rst, output reg [3:0] count);

  always @(posedge clk or posedge rst) begin
      if (rst) begin
          count <= 4'b0000;
      end else begin
          count <= count + 1;
      end
  end

endmodule
```

**Question 6**: [5 marks]

1) Write a systemverilog code that declares a dynamic array of integers named “my_arr” that have 8 elements
    a) Integer my_arr [] = new [8];
2) Fill the array with values from 1 to 8 using literals.
    a) my_arr = '{1, 2, 3, 4, 5, 6, 7, 8};
3) Using one of array manipulation functions print number of elements greater than 5
    a) my_arr.sum with ( int’(item > 5) )
4) What is the data structure that will be returned by the function you used in (3)
    a) queue
5) What is the data type of my_arr[9] ?
    a) wire