UNIT-V SPECIFICATION USING VERILOG HDL

1. **What are identifiers?**

   Identifiers are names of modules, variables and other objects that we can reference in the design. Identifiers consists of upper and lower case letters, digits 0 through 9, the underscore character(_) and the dollar sign($). It must be a single group of characters.

   Examples: A014, a ,b, in_o, s_out

2. **What are the value sets in Verilog?**

   Verilog supports four levels for the values needed to describe hardware referred to as value sets.

   Value levels Condition in hardware circuits

   - 0 Logic zero, false
   - condition 1 Logic one,
   - true condition X

     Unknown logic value

   Z High impedance, floating state

3. **Give the different arithmetic operators?**

   Operator symbol Operation performed Number of operands

   - Multiply Two
   - / Divide Two
   - \+ Add Two
   - Subtract Two
   - % Modulus Two
   - ** Power (exponent) Two

4. **Give the different bitwise operators.**

   Operator symbol Operation performed Number of operands

   - ~ Bitwise negation
   -

One & Bitwise and

Two

- Bitwise or Two
- ^ Bitwise xor Two
- ^~ or ~^ Bitwise xnor Two
- ~& Bitwise nand Two
- ~| Bitwise nor Two

5. **What are gate primitives?**

Verilog supports basic logic gates as predefined primitives. Primitive logic function keyword provide the basics for structural modeling at gate level. These primitives are instantiated like modules except that they are predefined in verilog and do not need amodule definition. The important operations are and, nand, or,xor, xnor, and buf(non- inverting drive buffer).

6. **Give the two blocks in behavioral modeling.**

An initial block executes once in the simulation and is used to set up initial conditions and step-by-step data flow An always block executes in a loop and repeats during the simulation.

7. **What are the types of conditional statements?**

No else statement

a. Syntax : if ([expression]) true – statement;

One else statement

b. Syntax : if ([expression]) true – statement; else false-statement;

Nested if-else-if

c. Syntax : if ([expression1]) true statement 1;else if ([expression2]) true- statement 2;

d. else if ([expression3]) true-statement 3;else default-statement;

The [expression] is evaluated. If it is true (1 or a non-zero value) true-statement is

e. executed. If it is false (zero) or ambiguous (x), the false-statement is

executed.

8. **Name the types of ports in**

Verilog Types of port

Keyword

- Input port Input
- Output port Output
- Bidirectional port

inout

- What are the types of procedural assignments?
- Blocking assignment
- Non-blocking assignment

9. **Give the different bitwise operators**

+ (addition)
- (subtraction)
* (multiplication)
/ (division)
% (modulus)

10. **What does synthesis mean ?**

Synthesis is a step of mapping the RTL files to convert it to the technology specific files.

11. **What is metastability and list the steps to prevent it ?**

Metastability is an unknown state (neither zero or one).

Metastability happens for the design violating setup or hole time requirements. Prevention steps

- By using synchronizers.
- Using faster flip-flops.

12. What are concatenation and replication operation in verilog and give example

The concatenation operator combines two or more operands to form a larger vector. The replication operator makes multiple copies of an item.

Example

```
wire [1:0] a, b;     wire [2:0] x;       wire [3:0] y, Z;
assign x = {1'b0, a}; // x[2]=0,  x[1]=a[1],  x[0]=a[0]
assign y = {a, b};   /* y[3]=a[1],  y[2]=a[0],  y[1]=b[1],
y[0]=b[0] */

assign {cout, y} = x + Z;  // Concatenation of a result
```

Replication

```
wire [1:0] a, b;     wire [4:0] x;
assign x = {2{1'b0}, a};  // Equivalent to  x = {0,0,a }
assign y = {2{a}, 3{b}};   //Equivalent to  y = {a,a,b,b}
```

13. What is the difference between module and instance? [AUC May 2011]

Modules are building blocks of verilog designs.

Modules are instantiated inside other modules, and each instantiation creates a unique object from the template.

14. Write the process involved in VLSI design flow.

Design specification

Behavioral description RTL description

Functional verification and testing Logic synthesis

Gate level netlist

Logical verification and testing

Floor planning and automatic place and route Physical layout

15. What is verilog HDL ?

Verilog HDL is a hardware description language that can be used to model a digital system at many levels of abstraction ranging from the algorithmic level to switch level.

16. What is gate level modeling ?

verilog all basic gates are available as ready modules called primitives. Each primitive defined in terms of its inputs and outputs are called gate level modeling.

17. What is data flow modelliing ?

Data flow modeling provides the means of describing combinational circuits by their function rather by their gate structure.

18. What is behavioral modeling ?

Behavioral modeling represents digital circuits at a functional and algorithmic level.

19. What id test bench ?

Test bench is a virtual environment used to verify the correctness of a design or model.

20. What is gate delay ?

The signal propagation delay from any gate input to the gate output is called the gate delay.

21. What are the three types of gate delays?

Rise delay , Fall delay , Turn –off delay.

22. What is rise delay ?

When the output gate terminal have the transition to a 1 from another value is called rise delay.

23. What is fall delay ?

When the output gate terminal have the transition to a 0 from another value is called Fall delay.

24. What is turn –off delay?

When the output gate terminal have the transition to the high impedance value from another value is called turn -off delay.

25. What is switch level modeling ?

Designing module such as MOS transistor ,CMOS transistor is called switch level modeling.

26. Give some primitives of switch levelmodeling.

nmos , pmos, cmos, pullup, pulldown ,tran ,tranif

27. What is transport delay ?

Transport delay is the delay caused by the wires connecting the gates.

28. What is subprogram overloading ?

Sub program is a collective name for functions, procedures and operators.

29. Write the verilog module for a half adder.

```
module ha (a,b,s,c);
  input a,b;
```

```
output     s,c;
    xor
    (s,a,b);
and(c,a,b);
    endmodu
    le
```