

*) HDL → • Hardware Descriptive Language .

• Specialized computer language

• Describe structure & behaviour of electronic circuits ,

*) Properties :-

• Includes notion of time (includes real time hardware delays)

• supports concurrency (2 blocks execute simultaneously)

*) When we talk about HDL directly or indirectly we talk about hardware .

• In early 70s, when HDL wasn't there, pen ~~and~~ paper and drafting table described digital circuit .

Verilog

- Verilog is hardware descriptive language . (HDL) Standardized as IEEE 1364-2001 .
- Describe low level hardware
- Syntax similar to C language .
- Founded in 1983
- Standardized in 1995 .

Verilog

- Have notion of time.
- Helps to design & describe digital system.
- .v extension
- Hardware Descriptive Language

C

- No notion of time.
- Helps to build operating system, compilers, interpreters etc
- .c extension
- General purpose programming language.

VHDL

- VHDL stands for ~~VHSIC~~ (Very high speed integrated circuit) Hardware Descriptive Language
- Model structure & behaviour of digital system at multiple abstraction levels.
- Strongly typed & deterministic

Verilog

- Weakly typed
- Based on C language
- Case Sensitive
- Simple data type
- Newer
- less complex

VHDL

- Strongly typed
- Based on Pascal & ~~Ada~~ Ada language
- Not case sensitive.
- Complex data type
- Older
- More Complex

Verilog abstraction

- switch level
- Gate level
- Data Flow level
- Behavioural level

Switch level

- Module can be implemented in terms of switches.
- nmos and pmos are used as switches.

\$ dumpfile → System task specifies
name of dump file

- VCD file (dump.vcd) contains information about value changes on specified variables
- Provide info about other post processing tools.
- \$ dumpvars (level, module name)
→ Dumps all variables in design.sv (without arguments)

\$ finish → tell simulator to terminate current simulation.

- exits simulation → Passes control back to operating system.

\$ stop → suspends simulation, puts simulator in interactive mode

Switch level primitive gates - pmos,
nmos

◦ user defined primitive (UDP) → Combinational or sequential

• 2 datatype →

◦ net → physical connection bet
(wire) structural elements

• variable (register) → abstract data storage
(reg, integer) element

◦ Gateway Design Automation in
(1983 → developed)

◦ IEEE standard in 1995.

◦ initial

begin

A = 1'b0;

B = 1'b0;

#? A = 1'b0; B = 1'b1;

#? A = 1'b1; B = 1'b0;

#? A = 1'b1; B = 1'b1;

end

register → data storage elements
(unsigned)

Integer → general purpose
(signed)
(integer)

real .

initial variable - name = 1
\$ time

time a ;
initial a = \$ time ;

blocking assignments

initial begin
executed in order they are

end specified in sequential block .

Non blocking

Allow scheduling of assignments

without blocking creation
of statements

- Using \times^2
- Used as relational operator in expression
- Assignment operator in non blocking assignment

wire	reg
• Cannot store value	• Can't store value
• Net datatype	• Variable
• Value determined by its driver such as gate, module.	• Used in procedural block such as initial block
• Signed	• unsigned

Display gets executed when (display immediate value) control of program reaches it
 Monitor gets executed parameter changes.

$$a \equiv \#10\ b \text{ (intra)}$$

$$\#10\ a = b \text{ (inter)}$$

timescale $1\text{ns}/1\text{ps} \rightarrow$

Unit of time is 1ns ,
Accuracy upto 1ps .

- Case Z \rightarrow Z as don't care
- Case X \rightarrow both n and Z as don't care
- Case \rightarrow considers n and Z as it is.
- When variable asynchronously holds its previous value, signal will infer a latch.
 (When net is not assigned to a known value)

always @ (a or b);

~~begin~~

out = a+b+c

end

① Binary to Gray

~~110~~

1011 to Gray

1011
↓ + + + + ↓
1110

Gray to Binary

1110
↓ ⊕ ↓ ⊕ ↓
101101

Binary to XS-3

$$(11101)_2 = (?)_{3 \times 3}$$

$$\begin{array}{c} \downarrow \\ (29)_{10} \\ \downarrow \quad \downarrow \\ +3 \quad +3 \end{array}$$

$$\begin{array}{r} 5 \quad 1 \ 2 \\ \hline \end{array}$$

1 bit

4 bit

$$0101 \quad 1100$$

$$\hline (0101 \ 1100)$$

$\begin{array}{r} 1000 \\ 16 \quad 8 \quad 4 \quad 2 \end{array}$

17

Notes:



① XS-3 to Binary

$$(0101 \quad 1100)_{XS-3} = (?)_3$$

$\downarrow \qquad \downarrow$

$$\begin{array}{r} 5 \\ -3 \\ \hline 2 \end{array} \qquad \begin{array}{r} 12 \\ -3 \\ \hline 9 \end{array})_{10}$$