DAY-61
#100DAYSRTL

"System Verilog: Unique & Priority"

"Introduction":-

- Improperly coded Verilog case statements can frequently cause unintended synthesis optimizations or unintended latches. These problems, if not caught in pre-silicon simulations or gate-level simulations, can easily lead to a non-functional chip. The new System Verilog unique and priority keywords are designed to address these coding traps.
- The System Verilog unique and priority modifiers are placed before an if, case, casez, casex.
- With the if…else statement, the System Verilog unique or priority keyword is placed only before the first if but affects all subsequent else if and else statements.

"Unique":-

Unique causes a simulator to add run-time checks that will report a warning if any of the following conditions are true:

1. More than one if/case item matches the case expression
2. No if/case item matches the case expression, and there is no default case/else

Firstly, by adding the System Verilog unique keyword, the designer asserts that only one case item can match at a time. the unique keyword tells the tool that all valid case items have been specified and can be evaluated in parallel. Synthesis is free to optimize the case items that are not listed.

“Code Practising”:-

```
module tb;
 int a=5;
  initial begin
    unique if(a>0) $display("Yes=1");
    else if (a<10) $display("Yes=2");
  end
endmodule
```

“Result”:-

```
Yes=1
Error: Assertion 'unique_if_1' FAILED at time: 0, testbench.sv(4),
```

“Code Practising”:-

```
module tb;
 int a=5;
  initial begin
    unique if(a>0) $display("Yes=1");
    else if (a>10) $display("Yes=2");
  end
endmodule
```

“Result”:-

```
Yes=1
```

“Priority”:-

The priority keyword instructs all tools that support System Verilog that each selection item in a series of decisions must be evaluated in the order in which they are listed, and all legal cases have been listed.

Unique causes a simulator to add run-time checks that will report a warning if any of the following conditions are true:

1. If the case/if expression does not match any of the case/if item expressions, and there is no default case/else

“Code Practising”:-

```
module tb;
 int a=5;
  initial begin
    priority if(a<10) $display("Yes=1");
    else if (a<20) $display("Yes=2");
  end
endmodule
```

“Result”:-

```
Yes=1
```

“Code Practising”:-

```
module tb;
 int a=5;
  initial begin
    priority if(a>10) $display("Yes=1");
    else if (a>20) $display("Yes=2");
  end
endmodule
```

“Result”:-

```
Errors: 0, Warnings: 1
```