ECE 351

Professor Greenwood “somebody who doesn’t know the answer tell me…”

04/16/19

* Nested ternary operations are legal
  + Example:

**Assign** out = s1 ? (s0 ? i3: i2) : (s3 ? i3: i1);

* Concatenated LHS statements are also legal
  + Example:

**assign** {c\_out, sum} = a + b + c\_in;

* + - The upper 1 bit will go to c\_out
    - The lower 4 bits will got o sum
    - Sum is a 4 bit output.
    - C\_out is a 1 bit output.
* Example 6-8
  + Port list: q, qbar, d, clk, clear
  + Instantiation: q, , ~q, clk, clear
    - Note that the second item in the portlist is not used and a “ “ space is used to show it as unused, or unconnected.
* **Behavioral Level of Abstraction**: Highest level of abstraction.
  + Arguably the 1st choice for programming.
    - Coding should be evaluated to determine if the abstraction level should be lowered for better performance output.
  + Testbench first approach: Not covered in the book
  + Not everything in **Behavioral Modeling** is synthesizable.
  + Some non-synthesizable models can be used in the testbench.
* Behavioral level of Abstraction: **Models have 2 structured procedures**
  + Initial statements (Not synthesizable)
  + Always statements
  + No limit on the number of these in a module
  + All of them execute concurrently and begin a time t=0;
  + These cannot be nested.
* Behavioral level of Abstraction:  **Testbench**
  + A program designed to exercise a digital system to verify its functionality.
  + **Not Synthesizable.**
  + **Testbenches** are written in Verilog at the **Behavioral Level**.
  + **Testbenches** don’t verify timing.
  + **Testbenches** generate test patterns to exercise the digital design.
  + **Testbenches c**ollect and output the circuit’s response.
  + Optionally, they are also used to compare outputs with expected outputs.
  + All testbenches will include the following behavioral statement:

**Initial**

#50 $finish

* + This tells the simulator to run the system task “$finish” after 50 time units of delay.
    - This means the simulator will end the testbench after 50t.
* Behavioral level of Abstraction: **Syntax**
  + Example:

**initial** (keyword)

[delay] “Procedural Statement”

* + Procedural Statement:
    - * + assignment (blocking, non-blocking)
        + continuous assignment
        + conditional statement
        + case statement
        + loop
        + event trigger
        + parallel block
        + wait statement
        + disable statement
  + When there are more than 1 procedural statements:

**Initial** (keyword)

**Begin** (keyword)

… [delay] “Procedural Statement”

… [delay] “Procedural Statement”

… [delay] “Procedural Statement”

… …

**End** (keyword)

* + - This is an example of a sequential block.
* Behavioral level of Abstraction: **Blocking Assignments**  (a = b;)
  + LHS updates before next statement executes.
  + Example:
* Behavioral level of Abstraction: Non-**Blocking Assignments** (a <= b;)
  + RHS is recorded
  + LHS updates at the end of an initial/ always procedure
* Behavioral level: In a sequential block, statements **DO EXECUTE SEQUENTIALLY**
  + Example

**Initial**

**Begin**

#5 a = 1’b1;

#25 b = 1’b0;

**End**

* + - The delay’s using “#” are sequential, thus additive.
    - “b” will not be initialized until time unit 30.