

# *Verilog II*

Dr. Paul D. Franzon

## Outline

1. Test fixture, simulation and synthesis of count.v

## References

1. Franzon/Smith, Chapter 2-6
2. Sutherland, Quick Reference Guide

## Test Fixture

### Used to verify correctness.

```
module test_fixture;
reg          clock;
reg          latch, dec;
reg          [3:0] in;
wire zero;

initial      //following block executed only once
begin
    $dumpfile("count.vcd"); // save waveforms in this file
    $dumpvars; // saves all waveforms
    clock = 0;
    latch = 0;
    dec = 0;
    in = 4'b0010;
    #15 latch = 1; // wait 15 ns
    #10 latch = 0; // wait 10 ns
    #10 dec = 1;
    #100
    $finish; //finished with simulation
end

always #5 clock = ~clock; // 10ns clock

// instantiate modules -- call this counter u1
counter u1(clock, in, latch, dec, clear, zero);
endmodule /*test_fixture*/
```

## *Test Fixture Features*

---

```
initial //following block executed only once
begin
...
end
```

is a procedural block that is executed once at the start of the simulation.

---

```
$dumpfile("count.vcd"); // save waveforms in this file
$dumpvars; // saves all waveforms
```

opens the database that stores the waveforms and specifies what signals to save.

---

#15

are delays in terms of pre-defined units. (Here ns, but it is not always ns).

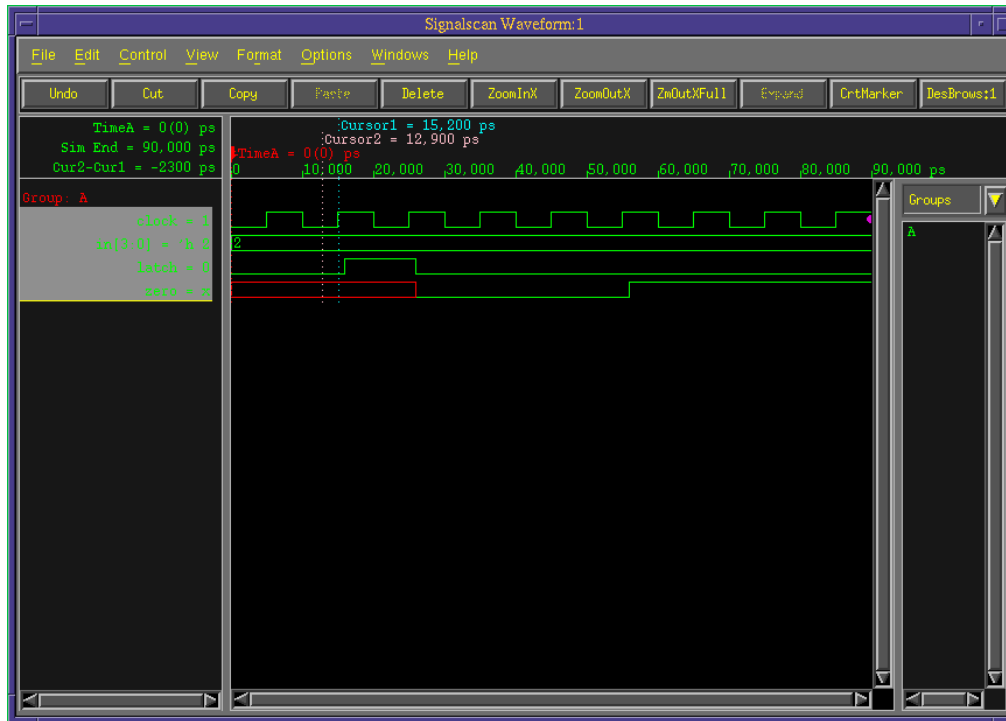
---



## Simulation

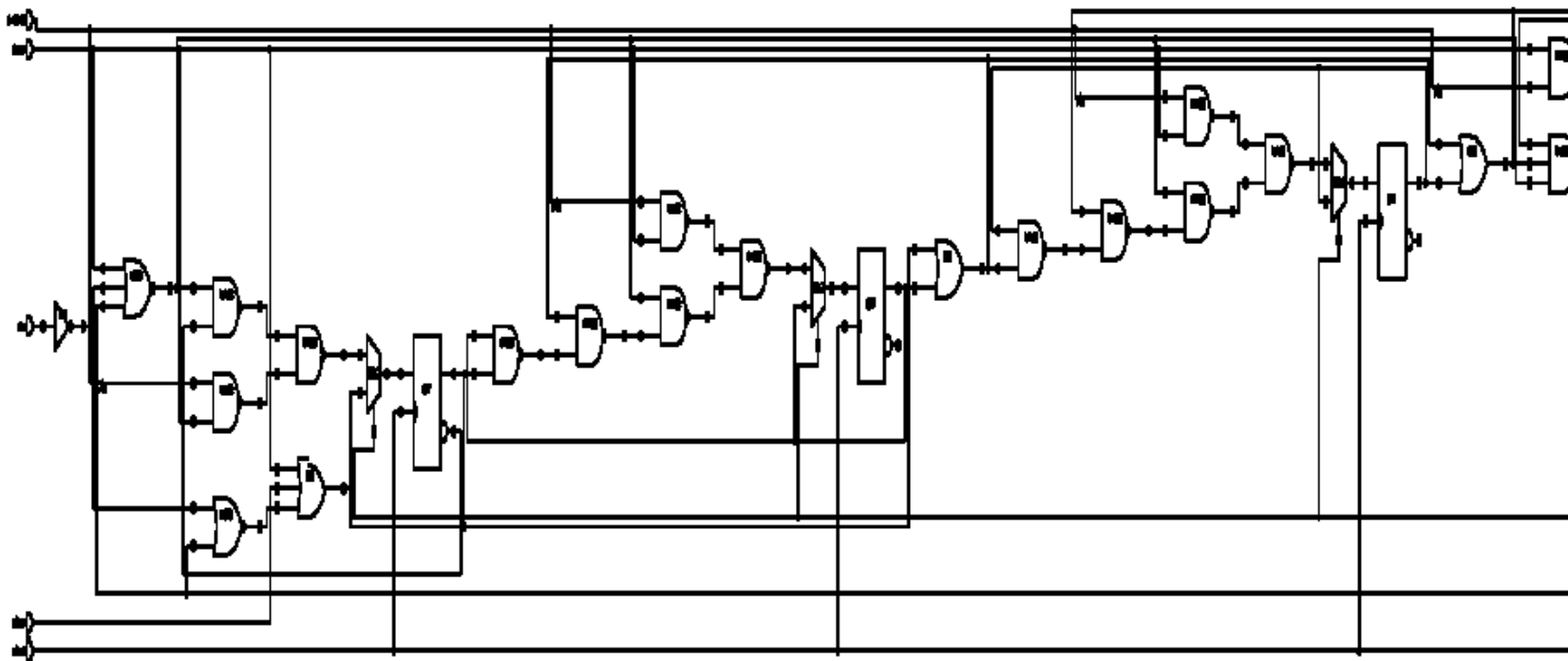
Simulate using Verilog XL and use waveform viewer... (Lab 1)

- > verilog test.v count.v
- > signalscan count.vcd &



## Synthesis

Use Synopsys Design Compiler to create a gate level design “netlist”



## *More Sophisticated Test Fixture*

Don't always need waveform viewer to determine correctness...

```
initial
begin
    clock = 0; latch = 0; dec = 0;
    in = 4'd2;
    #16 latch = 1;
    #10 latch = 0; // down 1 clock later
    #10 dec = 1;
    #28 if (zero == 1'b1) $display ("Check 1 passed")
    else $display ("Error: Check 1 FAILED");
end
```

## *... Test Fixture*

What is this test fixture doing?

What are the potential advantages of this approach?

## *Summary*

What is the purpose of a test fixture?

What is the output of synthesis?

How can you make a test fixture “self checking”?