System Verilog Lab4

After completing this lab, you should be able to:

Implement self-checking testbenches.

Understand how to store expected and actual data.

Understand how to measure test correctness.

Implement compare method.

*Verify DUT behaviour **without** waveform by self-checking mechanism in testbench.*

Implement test passing and failing criteria.

LAB4:

Copy the lab3 code as lab4

Step 1: Implement compare method.

1. Define int matched and mis_matched in Section 1.
2. Implement a function compare in Section 4.
3. Add the below code in Section 4 in testbench.sv

```
function void compare();
  foreach(exp_q[i]) begin
    if(exp_q[i] == dut_out[i])
        matched++;
     else begin
        mis_matched++;
        $display("[Error] Expected = %0d and Received = %0d",exp_q[i],dut_out[i]);
     end
   end
endfunction
```

4. Increment the matched and mis_matched counts.

Step 2:Check the count of matched and mis_matched to print test Passed or Failed.

1. Call the compare() function (Section 6 in testbench.sv)
2. Add the below code into Section 6 in testbench.sv

```
compare(); // Calling the method to compare the matched and mis_matched.

if(mis_matched == 0) begin
    $display("*********************************");
    $display("***********Test Passed***********");
    $display("Matched = %0d Mis_matched = %0d",matched,mis_matched);
    $display("*********************************");
end
else begin
    $display("*********************************");
    $display("***********Test Failed***********");
    $display("Matched = %0d Mis_matched = %0d",matched,mis_matched);
    $display("*********************************");

end
```

Step 3: Validate the output of DUT with the results printed by self-checking mechanism.

1. Check the test passed or failed and debug if there is any failuers.