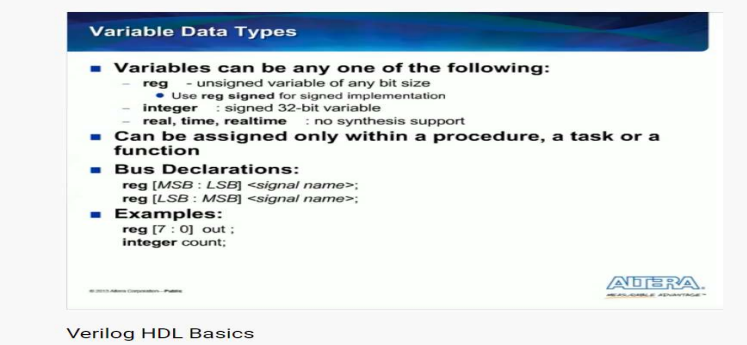
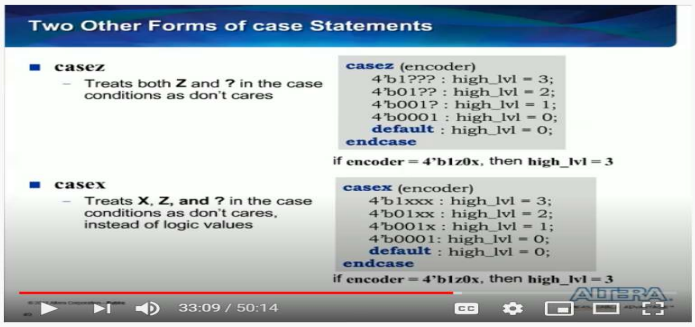
DAILY ASSESSMENT FORMAT

Date:	**05/06/2020**	**Name:**	**Sachin Krishna Moger**
Course:	**DIGITAL DESIGN USING HDL**	**USN:**	**4AL17EC103**
Topic:	**Verilog Tutorials and practice programs** **Building/ Demo projects using FPGA**	**Semester & Section:**	**6-B**
Github Repository:	**Sachin-Courses**		

FORENOON SESSION DETAILS



Verilog HDL Basics



Verilog HDL Basics

Implement a verilog module to count number of 0's in a 16 bit number in compiler.

```
module counter (clk, reset, count);
   input clk, reset;
   output [14:0] count;
   reg [14:0] count;
```

```
    always @(negedge clk or posedge reset )
        if (reset)
            count <= 4'boooo;
        else
             count <= count + 1'b1;
endmodule
```

Testbench:

```
module TEST;
  reg clk;
  reg reset;
  wire [14:0] count;

  counter one(clk, reset, count);
  always #10 clk <= ~clk;

  initial begin
    clk   = 0;
    reset = 1;
    $monitor( "%t: %b%b", $time,  count, clk);
    #1   reset = 0;
    #2560 $finish;
  end
endmodule
```