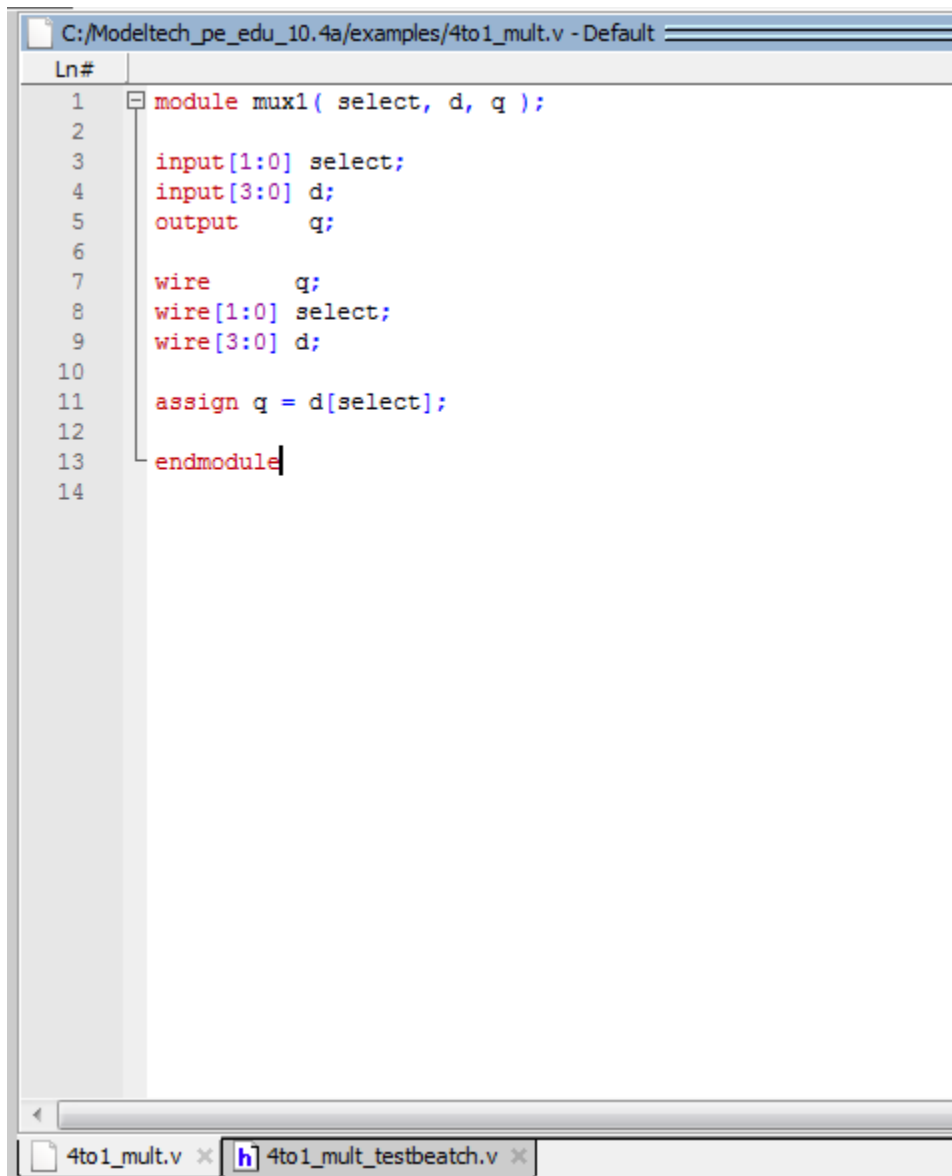


Chirag solanki

15035

Homework assignment 4

Answer 4



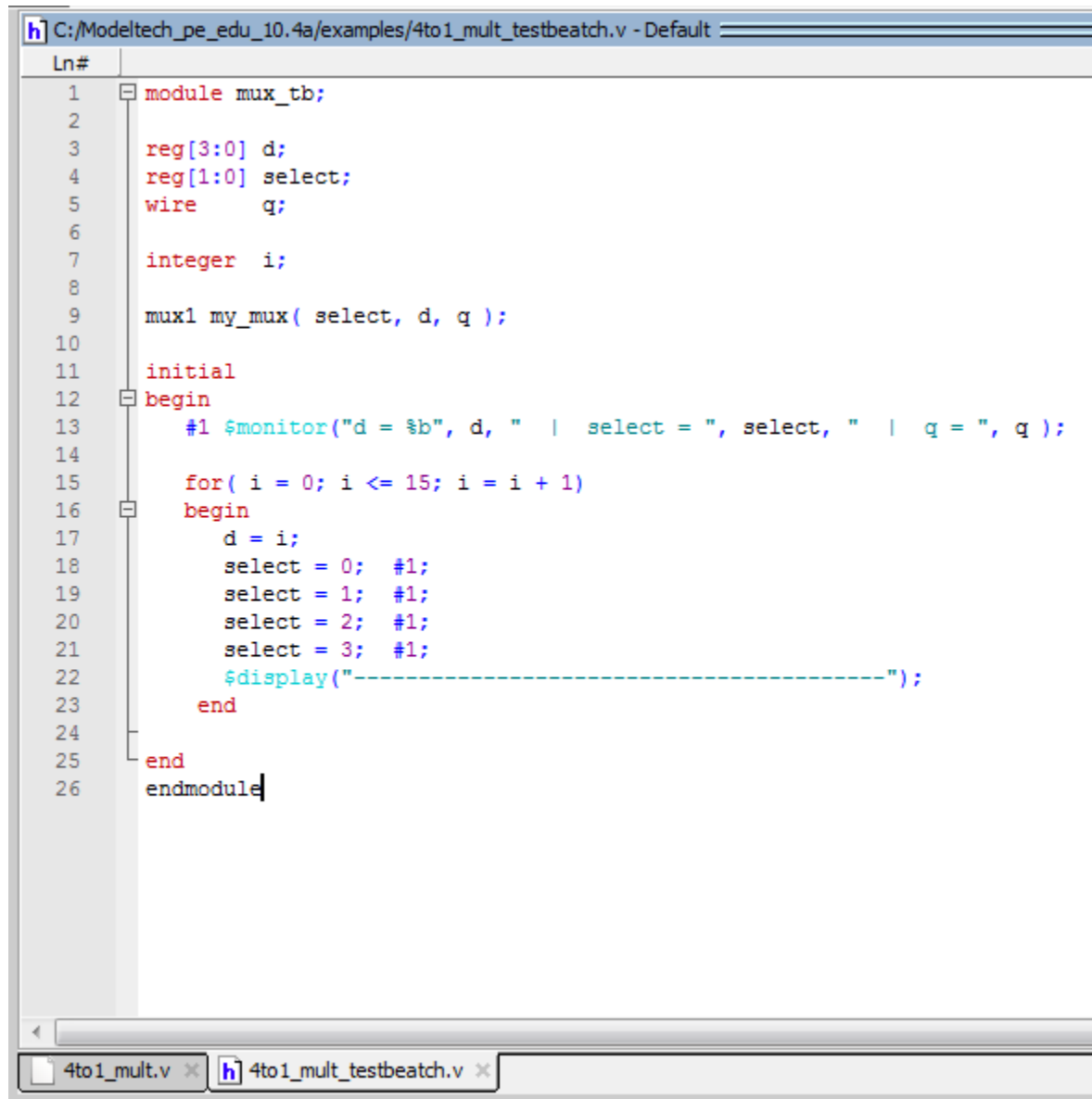
The image shows a screenshot of a Verilog code editor. The title bar indicates the file path: C:/Modeltech_pe_edu_10.4a/examples/4to1_mult.v - Default. The editor displays the following Verilog code:

```
Ln# |  
1  | module mux1( select, d, q );  
2  |  
3  |     input[1:0] select;  
4  |     input[3:0] d;  
5  |     output      q;  
6  |  
7  |     wire        q;  
8  |     wire[1:0] select;  
9  |     wire[3:0] d;  
10 |  
11 |     assign q = d[select];  
12 |  
13 | endmodule  
14 |
```

The code defines a module named `mux1` with two inputs: `select` (1-bit) and `d` (4-bit), and one output: `q` (1-bit). It uses `wire` declarations for `q`, `select`, and `d` inside the module, and an `assign` statement to connect the output `q` to the selected input from `d` based on the `select` signal. The module is closed with `endmodule`.

The bottom of the screenshot shows two tabs: `4to1_mult.v` and `4to1_mult_testbeatch.v`.

Test batch



The screenshot shows a Verilog code editor window with the title bar "C:/Modeltech_pe_edu_10.4a/examples/4to1_mult_testbeatch.v - Default". The code is a testbench for a 4-to-1 multiplexer, named `mux_tb`. It includes a module declaration, register and wire declarations, an integer counter, an instance of a multiplexer module, and an initial block with a loop to test all four select values. The code is as follows:

```
1 module mux_tb;
2
3     reg[3:0] d;
4     reg[1:0] select;
5     wire     q;
6
7     integer i;
8
9     mux1 my_mux( select, d, q );
10
11    initial
12    begin
13        #1 $monitor("d = %b", d, " | select = ", select, " | q = ", q );
14
15        for( i = 0; i <= 15; i = i + 1)
16        begin
17            d = i;
18            select = 0; #1;
19            select = 1; #1;
20            select = 2; #1;
21            select = 3; #1;
22            $display("-----");
23        end
24    end
25 endmodule
```

The editor interface includes a line number column on the left, a main code area, and a tab bar at the bottom showing two open files: `4to1_mult.v` and `4to1_mult_testbeatch.v`.

Answer 5

```
module mux(a,b,c,d,e,sel,out);  
input a,b,c,d,e;  
input[2:0] sel;  
output out;  
reg out  
  
always @(in0 or in1 or in2 or in3 or in4 or in5 or sel)  
  
case (sel)  
  
    3'b000 : out = in0;  
  
    3'b001 : out = in1;  
  
    3'b010 : out = in2;  
  
    3'b011 : out = in3;  
  
    3'b100 : out = in4;  
  
    3'b101 : out = in5;  
  
    default : out = 64'bz;  
  
endcase
```