Northwestern Polytechnic University

EE461 Verilog-HDL

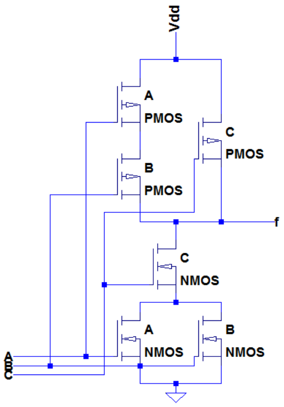
Lab Assignment #3

Due day: 11/18/2016

Instruction:

1. Print your program for each question in "word" file and paste running results in screen shot on it. Printing results or electrical copies from monitor can't be accepted.
2. Please follow the code style rule like programs on handout.
3. Overdue homework submission could not be accepted.
4. Takes academic honesty and integrity seriously (Zero Tolerance of Cheating & Plagiarism)

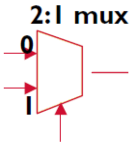
1. Design a static logic gate $f = \overline{(a+b)c}$ in Verilog modeling by PMOS/NMOS devices based on the logic schematic as follows. And then write the testbench to verify the design to cover all input combinations and some x/z.



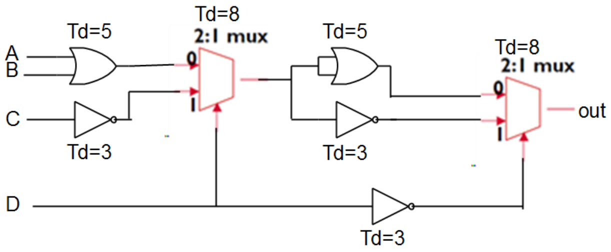
```
module testbench;
        reg     a_r;
        reg     b_r;
        reg     c_r;
        wire    f_w;

        designModule u(
                                .a_i    (a_r),
                                .ab_i   (b_r),
                                .c_i    (c_r),
                                .f_o    (f_w)
        );

        initial begin
                a_r = 1'b0; b_r = 1'b0; c_r = 1'b0;
                #5 a_r = … … ; b_r = … … ; c_r = … … ;
                #5 … …
```

```
                    #5 $monitor(... ...);
                    #5 $finish;
            end
        endmodule
```

2. Design a 2-to-1 mux UDP, and write testbench to assign selection bit to 0, 1 and X/Z to verify you design.



3. Create top module to instantiate mux UDPs from the above primitive and system gates with delay time as below and simulate it by testbench with only several inputs values. What is the delay time of longest path from inputs to output in hand calculation if we don't consider the logic values of selection bits in two mux? If D is either 1 or 0, what is the possible delay time of longest path from inputs to output?

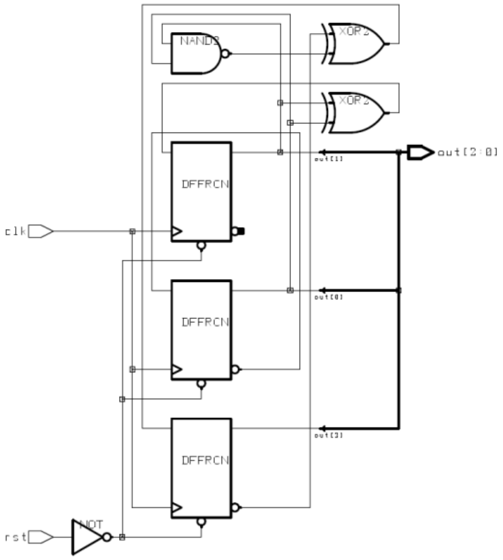


4. Fix the bugs in the following example primitive, and explain why.

```
  #include "basicPrimitive.v"
  primitive example(a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r);
      output[1:0]    a, r;
      input          b, d, e, f, g, h, i, j, k, l, m, n, o, p, q;
      inout          c;

      reg[1:0]       a, r;

      table
      //a: c:   r:   b  d   e  f  g  h  i  j  k   l   m   n   o   p   q
      00: 1: 01:     0  1   0  1  0  1  0  1  0   1   0   1   0   1   0
       1: 1: 11:     0  1   0  1  0  1  0  1  0   1   0   1   0   1   ?
      00: 0: 01:     0  1   0  1  0  1  0  1  0   1   0   1   0   1   0
        ... ...
      endtable

      basicPrimitive u0(a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r);
```

endprimitive

5. Generate UDP of D-Flip Flop with reset, and then build 3 bits counter in top module by instantiating UDP DFF and NOT/NAND/XOR gates in terms of following schematic. After that, write testbench to observe output results.


NAND2
XOR2
XOR2
out[2:0]
out[1]
DFFRCN
clk
out[0]
DFFRCN
out[2]
DFFRCN
rst
NOT