Professor Fearing EECS150/Problem Set 8 Solution Fall 2013

Due at 12 pm, Thu. Nov. 7 (homework box under stairs)

(This problem set may be done in a group of maximum 2 students, with 1 unique writeup to be turned in per group.)

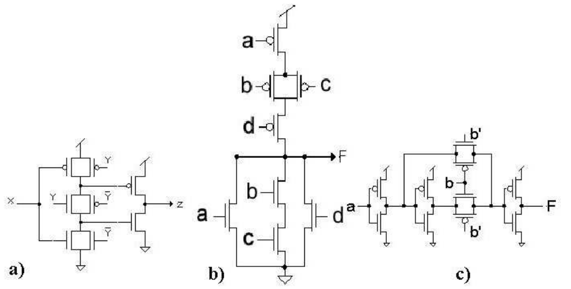
1. (25 pts) CMOS.

a,b,c. Draw its common symbol or state the equivalent Boolean expression for the circuits to the right.

a. If $y = 1$ then $z = x$ else z is high impedance. This is a tristate buffer.

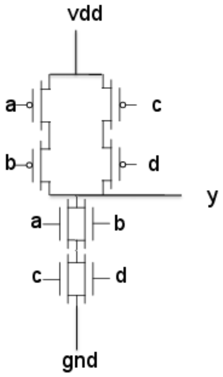
b. $F = \overline{a + \ b \cdot c \ + d}$

c. $F = A$ XOR B.

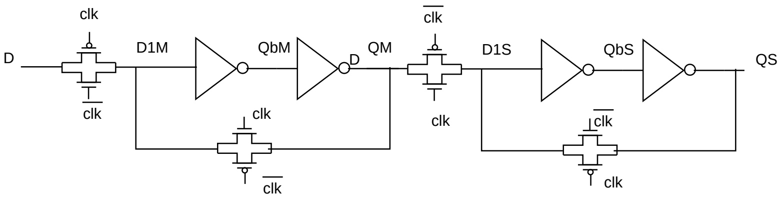


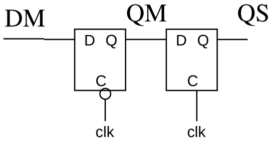
d.

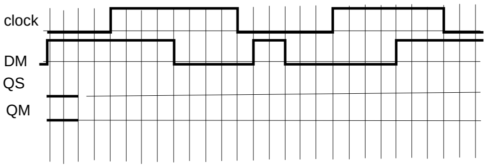
Implement the function $y = \overline{(a+b)(c+d)}$ using CMOS transistors with the minimum number of transistors.



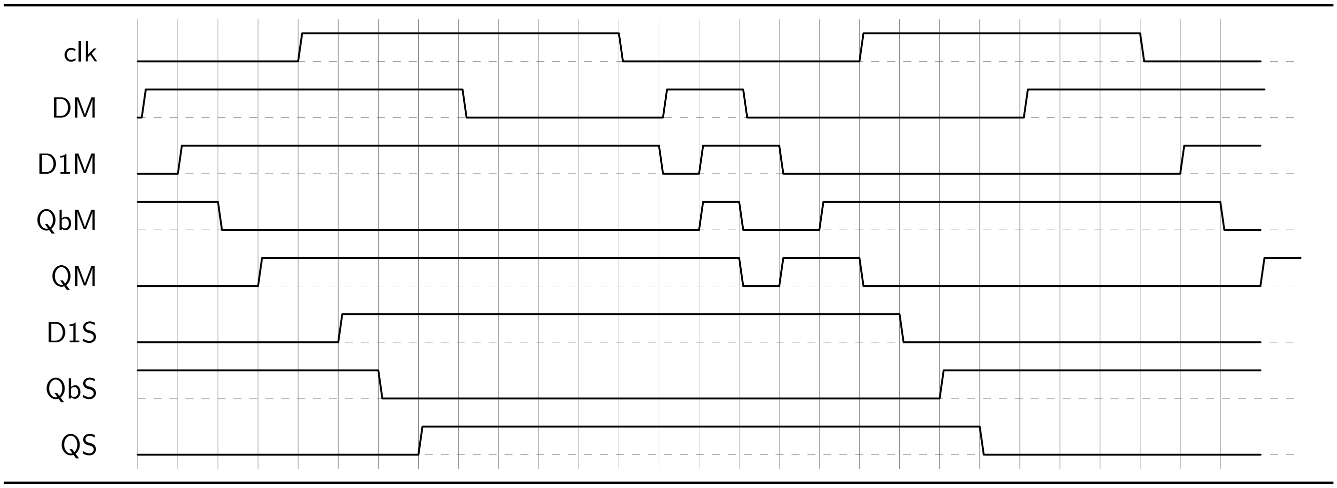
2. D- master-slave FF Timing (40 pts)



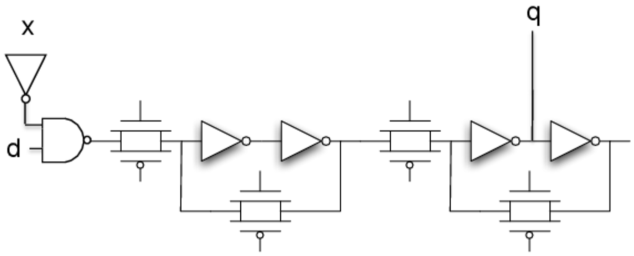




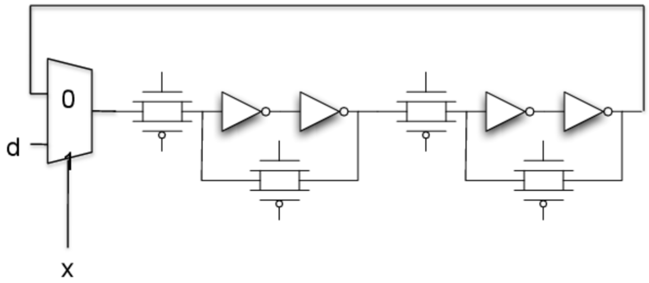
a. Draw a timing diagram for the D master-slave FF circuit below, assuming unit delay through the inverters and transmission gates, and showing signals $clk, DM, D1M, QbM, QM, D1S, QbS, QS$.



b. Modify the D-master-slave FF circuit below, using minimum number of gates, to add a synchronous reset.

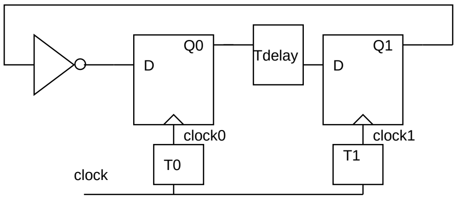


c. Modify the D-master-slave FF circuit below, using minimum number of gates, to add a clock enable.

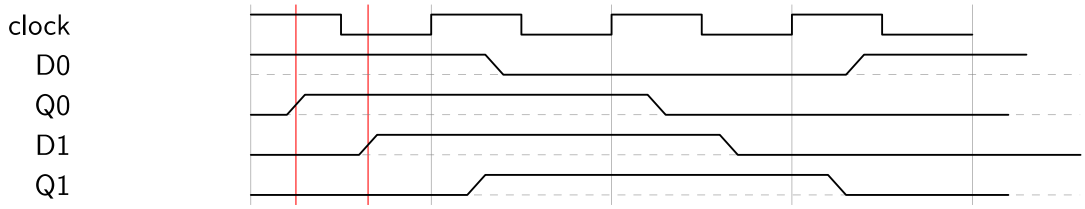


3. Clock Skew (35 pts)

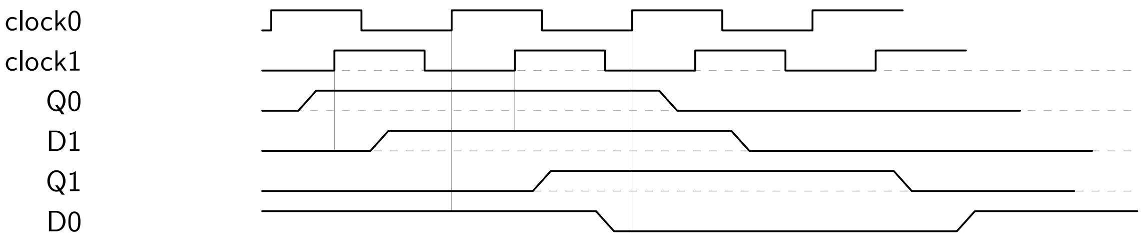
For this problem consider the parameters $T_{setup} = 1.1$ ns min, $T_{hold} = 0$ ns, $T_{CKO} < 1.3$ ns, $T_{delay} = 20$ ns and inverter plus interconnect propagation 4 ns. For each question below, justify your answer with a timing diagram.



a. With delay $T_0 = T_1 = 0$ ns, what is the maximum clock speed?
Min period $= T_{CKO} + T_{delay} + T_{setup} = 1.3 + 20 + 1.1 = 22.4$ ns or 44.64 MHz.



b. What is the maximum $T_1 - T_0$ for proper operation?



From the timing diagram we need to check setup and hold times for both flip flops.

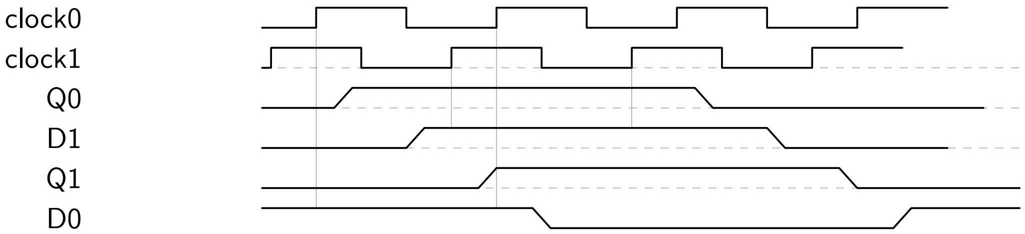
Referring to rising edge of clock0 FF#0:
To avoid setup violation on D0, without clock 0 delay we must have
$T_{CKO1} + T_{inv} + T_{setup} < T_{period}$.
The delayed clock1 reduces available setup time on FF #0:
$(T_1 - T_0) + T_{CKO1} + T_{inv} + T_{setup} < T_{period}$.
Since clock 1 is delayed with respect to clock 0, FF#0 will not have a hold time violation, as with any delay D0 is changing after the rising edge of clock0, and $t_{hold} = 0ns$.

Referring to rising edge of clock1 FF#1:
To avoid setup violation on D1, without clock 1 delay we must have
$T_{CKO0} + T_{delay} + T_{setup} < T_{period}$.
The delayed clock1 actually gives more setup time:
$T_{CKO0} + T_{delay} + T_{setup} - (T_1 - T_0) < T_{period}$.

To avoid hold violation, without clock 1 delay we must have
$T_{CKO0min} + T_{delay} > T_{hold}$.
The delayed clock1 reduces hold time:
$T_{CKO0min} + T_{delay} - (T_1 - T_0) > T_{hold}$.
With $t_{hold} = 0$ ns, $T_1 - T_0$ must be less than $T_{CKO0min} + 20$ ns.

$(T_1 - T_0) = \min(T_{period} - T_{CKO1} - T_{inv} - T_{setup}, T_{CKO0min} + T_{delay})$. If T_{period} is assumed to be 22.4 ns from part a, then $T_1 - T_0$ must be less than 16 ns.

c. What is the minimum $T_1 - T_0$ for proper operation? (i.e. $T_1 - T_0 < 0$).



From the timing diagram we need to check setup and hold times for both flip flops.

Referring to rising edge of clock0 FF#0:
To avoid setup violation on D0, without clock 0 delay we must have
$T_{CKO1} + T_{inv} + T_{setup} < T_{period}$.
The delayed clock0 increases available setup time on FF #0:
$T_{CKO1} + T_{inv} + T_{setup} + (T_1 - T_0) < T_{period}$.
To avoid hold violation, without clock 0 delay we must have
$T_{CKO1min} + T_{inv} > T_{hold}$.
The delayed clock0 reduces hold time:
$T_{CKO1min} + T_{inv} + (T_1 - T_0) > T_{hold}$.
With $t_{hold} = 0$ ns, $T_1 - T_0$ must be greater than than -(1.3 + 4) = -5.3 ns. $T_{CKOmin1}$ must be less than 1.3 ns (but was not specified), so strictly $T_1 - T_0$ must be greater than -4 ns. Note that otherwise, the "next next" state value might be clocked into FF#0.

Referring to rising edge of clock1 FF#1:
To avoid setup violation on D1, without clock 0 delay we must have
$T_{CKO0} + T_{delay} + T_{setup} < T_{period}$.
The delayed clock0 reduces available setup time:
$T_{CKO0} + T_{delay} + T_{setup} - (T_1 - T_0) < T_{period}$.

Since clock 0 is delayed with respect to clock 1, FF#1 will not have a hold time violation, as D1 is changing after the rising edge of clock1, and $t_{hold} = 0ns$.

$(T_1 - T_0) = \max(T_{CKO0} + T_{delay} + T_{setup} - T_{period}, -(T_{CKO1min} + T_{inv}))$. If T_{period} is assumed to be 22.4 ns from part a, then since delaying clock0 reduces available setup time of FF#1, minimum $T_1 - T_0$ is zero. At slower clock speeds, minimum $T_1 - T_0$ is -4 ns.

d. In Xilinx Virtex5 FPGA designs, how can you prevent indeterminate delays on the clock line due to random routing?

Make sure that the clock is routed over a global clock routing network. This is handled in the FPGA_TOP_ML505.v file by the command:

```
BUFG clk_buf_50M (.I(clk_50M_pre),.O(clk_50M));
```