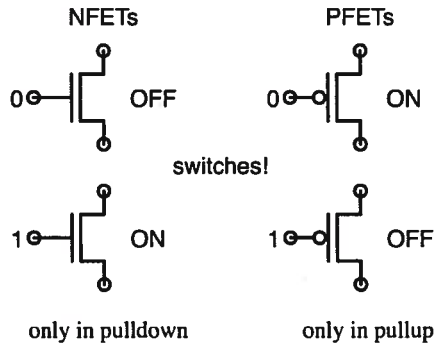
Computation Structures

CMOS Technology Worksheet

Concept Inventory:

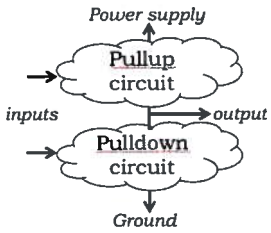
- PFET, NFET: voltage controlled switches
- CMOS composition rules: complementary pullup and pulldown
- CMOS gates are naturally inverting
- t_{PD} and t_{CD} timing specifications
- Lenient gates

Notes:



We want *complementary* pullup and pulldown logic, i.e., the pulldown should be "on" when the pullup is "off" and vice versa.

pullup	pulldown	F(inputs)
on	off	driven "1"
off	on	driven "0"
on	on	driven "X"
off	off	no connection



CMOS gates are naturally inverting:

- Rising input (0 to 1): NFETs turn on, PFETs turn off; if output changes, it falls (1 to 0)
- Falling input (1 to 0): NFETs turn off, PFETs turn on; if output changes, it rises (0 to 1)

Timing:

- t_{PD} (propagation delay): how long after inputs are stable and valid until outputs are stable and valid = max over all paths from input to output (sum of component t_{PD} along path)
  - t_{PD} specification is an upper bound on all measured propagation delays
- t_{CD} (contamination delay): how long output stays valid after inputs go invalid = min over all paths from input to output (sum of component t_{CD} along path)
  - t_{CD} specification is a lower bound on all measured contamination delays

Lenient gate:

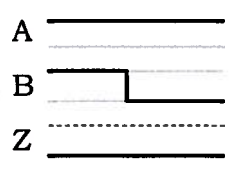
- If a subset of a lenient gate's inputs is suffice to guarantee an specific output value (i.e., the values of the other inputs don't matter in this case), then the output will remain valid and stable by transitions on the irrelevant inputs.
- CMOS gates are naturally lenient

NOR:

A	B	Z
0	0	1
0	1	0
1	0	0
1	1	0

Lenient NOR:

A	B	Z
0	0	1
X	1	0
1	X	0



Problem 1.

(A) Which of the above CMOS pulldown circuits would implement F if the corresponding complementary pullup circuit was also provided? For each pulldown, select Yes if it is a valid pulldown for F, and No if it is not a valid pulldown for F.

F is Ø when
- A=1 and
- D=1 and
- either B=1 or C=1

PD1 (Yes/No): YES

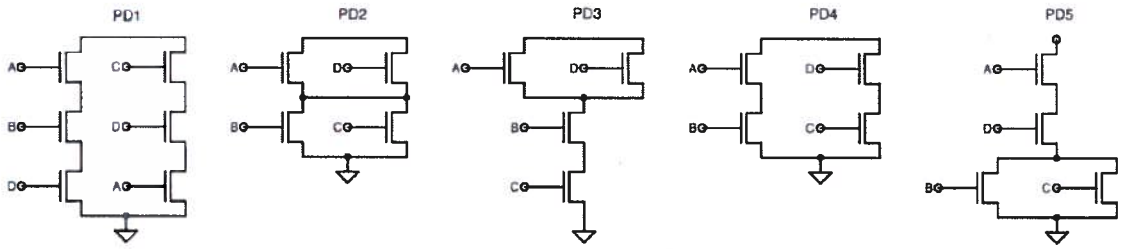
PD2 (Yes/No): NO

PD3 (Yes/No): NO

PD4 (Yes/No): NO

PD5 (Yes/No): YES

A	B	C	D	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0



(B) Are all the implementations you selected for part (A) lenient?

single CMOS gate is always lenient. **All lenient (Yes/No):** YES

Problem 2.

(A) A single CMOS gate, consisting of an output node connected to a single PFET-based pullup circuit and a single NFET-based pulldown circuit (as described in lecture) computes F(A, B, C, D). It is observed that F(1, 0, 1, 0) = 1. What can you say about the following values?

nfet off, pfet on ⇒ no change

(circle one) F(0, 0, 1, 0) = : 0 ... 1 ... (can't say)

(circle one) F(1, 1, 1, 0) = : 0 ... 1 ... (can't say)

all nfets on

(circle one) F(1, 1, 1, 1) = : 0 ... 1 ... (can't say)

(B) The Boolean function F(A,B,C) can be implemented using a *single* CMOS gate operating as a combinational device that obeys the static discipline. It's known that F(1,1,0) = 1 and F(0,1,1) = 0. What can be determined about the value of F in the following cases? Please circle one of "0", "1" or "Can't tell".

if F(1,1,0)=1
then F(?,?,0)=1

(circle one) F(1,0,0) = 0 ... (1) ... Can't tell

(circle one) F(1,0,1) = 0 ... 1 ... (Can't tell)

(circle one) F(1,1,1) = (0) ... 1 ... Can't tell

(C) A single CMOS gate, consisting of an output node connected to a single pullup circuit containing one or more PFETs and a single pulldown circuit containing one or more NFETs (as described in lecture), computes $F(A,B)$. F has the property that for all A, $F(A,0) = \overline{F(A,1)}$. What can you say about the value of $F(1,0)$?

$F(1,0) = \overline{F(1,1)} = \overline{0} = 1$
↑ all NFETs on

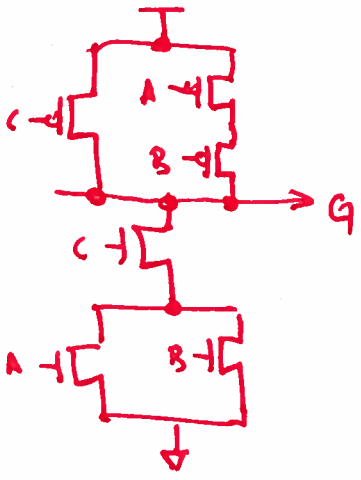
(circle one): F(1,0) = (1) ... 0 ... can't tell

Problem 3.

For each of the functions F and G, if the function can be implemented using a **single CMOS gate**, please draw the corresponding single CMOS gate. If it cannot be implemented using a single CMOS gate, then write NONE. **For full credit use a minimum number of FETs.**

A	B	C	F	G
0	0	0	1	1
0	0	1	1	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	1
1	0	1	0	0
1	1	0	0	1
1	1	1	1	0

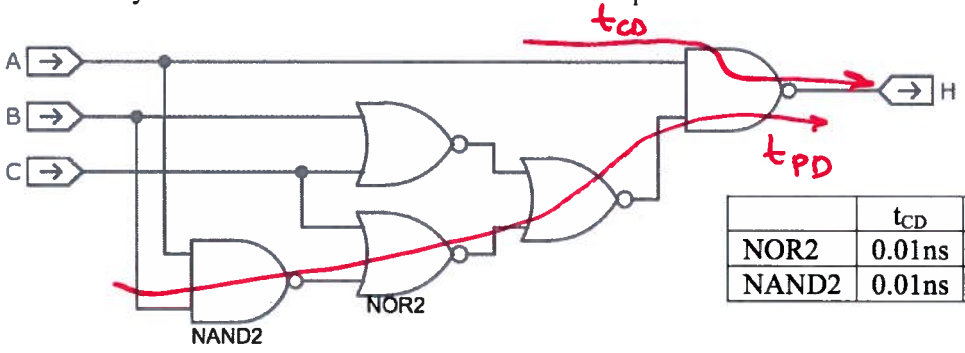
G is Φ when
- C is 1 and
- either A=1 or B=1

Draw CMOS implementation of F(A,B,C) below or write NONE if F cannot be implemented as single CMOS gate.	**Draw CMOS implementation of G(A,B,C) below or write NONE if G cannot be implemented as single CMOS gate.**
F(1,1,1) = 1 ⇒ not a CMOS gate since all NFETs on should produce a zero.	

Problem 4.

Consider the Boolean function that has the truth table shown to the right; a possible implementation as a combinational circuit is shown in the schematic below. You may assume that the NOR2 and NAND2 components are combinational.

A	B	C	H
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



	t_{CD}	t_{PD}
NOR2	0.01ns	0.05ns
NAND2	0.01ns	0.03ns

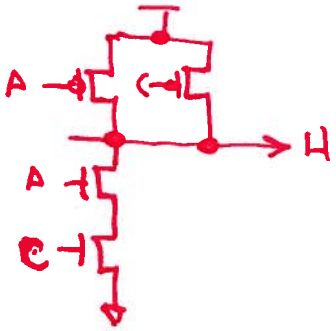
(A) Using the timing specifications shown above for NOR2 and NAND2, compute the contamination and propagation delay for the implementation of H shown above.

timing for H (ns): t_{CD}= 0.01 t_{PD}= 0.16

(B) Can H be implemented as a single CMOS gate (only PFETs in the pullup circuit, only NFETs in the pulldown circuit)? If so draw the MOSFET schematic for H to the right, otherwise write "NO".

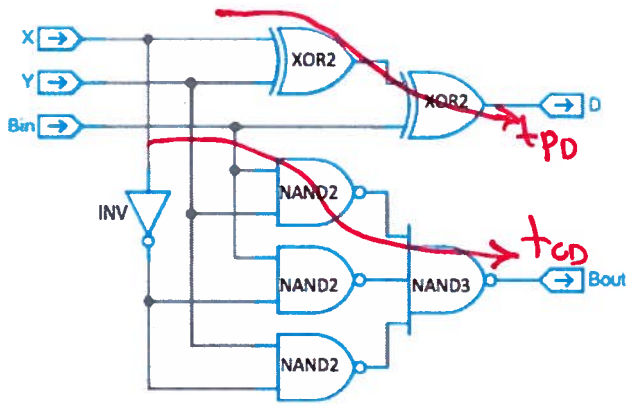
Draw schematic or write "NO"

H=0 when A=1 and C=1



Problem 5.

A gate-level schematic is shown below. Using the t_{CD} and t_{PD} information for the gate components shown in the table below, compute t_{CD} and t_{PD} for the circuit.



Compute timing specs:

t_{CD} = 0.5 ns

t_{PD} = 5 ns

Gate	t_{CD}	t_{PD}
INV	0.1ns	1.0ns
NAND2	0.2ns	1.5ns
NAND3	0.3ns	1.8ns
XOR2	0.6ns	2.5ns

Problem 6.

A minority gate has three inputs (call them A, B, C) and one output (call it Y). The output will be 0 if two or more of the inputs are 1, and 1 if two or more of the inputs are 0.

In the space below, draw the *pulldown* circuit for a single CMOS gate that implements the minority function, using the minimum number of NFETs. You needn't draw the *pullup* circuit.

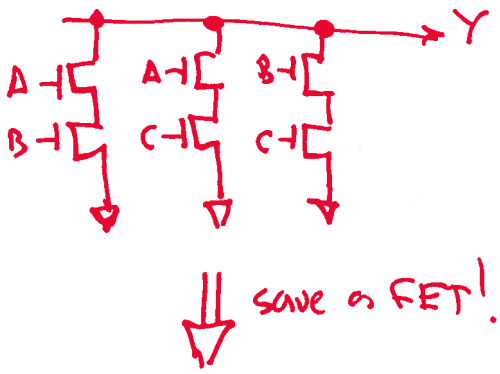
If you're convinced that the function cannot be implemented as a single CMOS gate, give a brief, convincing explanation.

Can it be implemented as single CMOS gate? Circle one: (YES) can't tell NO

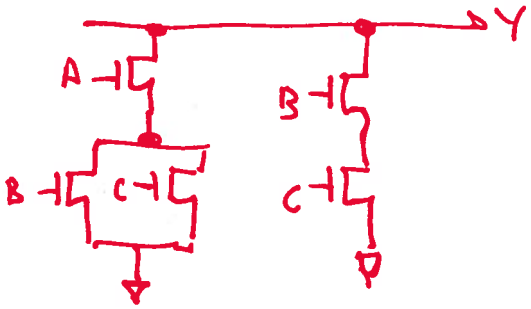
A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Y is 0 when

- A=1 and B=1, or
- A=1 and C=1 or
- B=1 and C=1

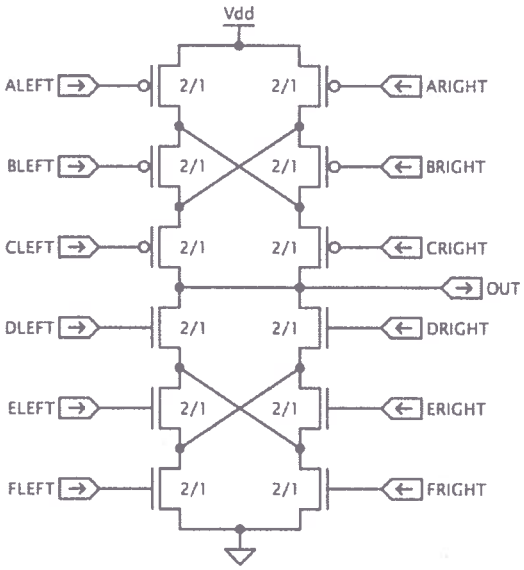


save a FET!



Problem 7.

In his bid for the Lemelson Prize, Ben Bitdiddle has invented the "flexible gate," a single CMOS gate that implements different functions depending how its inputs are wired up. The FlexGate® (see figure at right) uses 6 PFETs in its pullup circuit and 6 NFETs in its pulldown circuit..



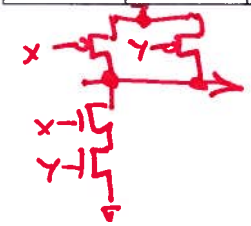
Each of the FlexGate's twelve inputs can connected to an input signal (X, Y, …), GND (logical "0") or VDD (logical "1"). To show off its versatility, Ben has asked you to show how to hook up the inputs so the FlexGate computes several different functions whose Boolean equations are given below. Associated with each equation is a table with 12 entries; in each cell of the table please write an input name, GND or VDD as appropriate. Note that there may be several possible implementations for each of the three functions – any correct answer will be acceptable. Hint: there should be an entry in each cell, i.e., a connection should be specified each input!

If the desired function cannot be implemented, please draw a big "X" through the table.

Note : other answers possible! **Fill in tables below or mark with "X"**

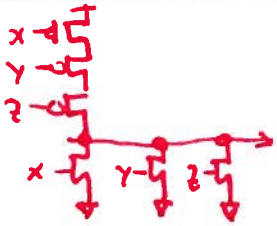
$$OUT = \overline{X \cdot Y}$$

input	LEFT	RIGHT
A	GND	GND
B	VDD	VDD
C	X	Y
D	X	GND
E	GND	GND
F	GND	Y



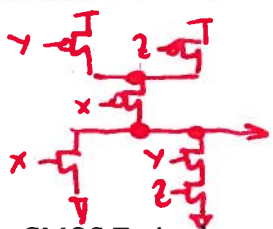
$$OUT = \overline{X + Y + Z}$$

input	LEFT	RIGHT
A	Z	VDD
B	Y	VDD
C	X	VDD
D	X	VDD
E	GND	Y
F	Z	VDD



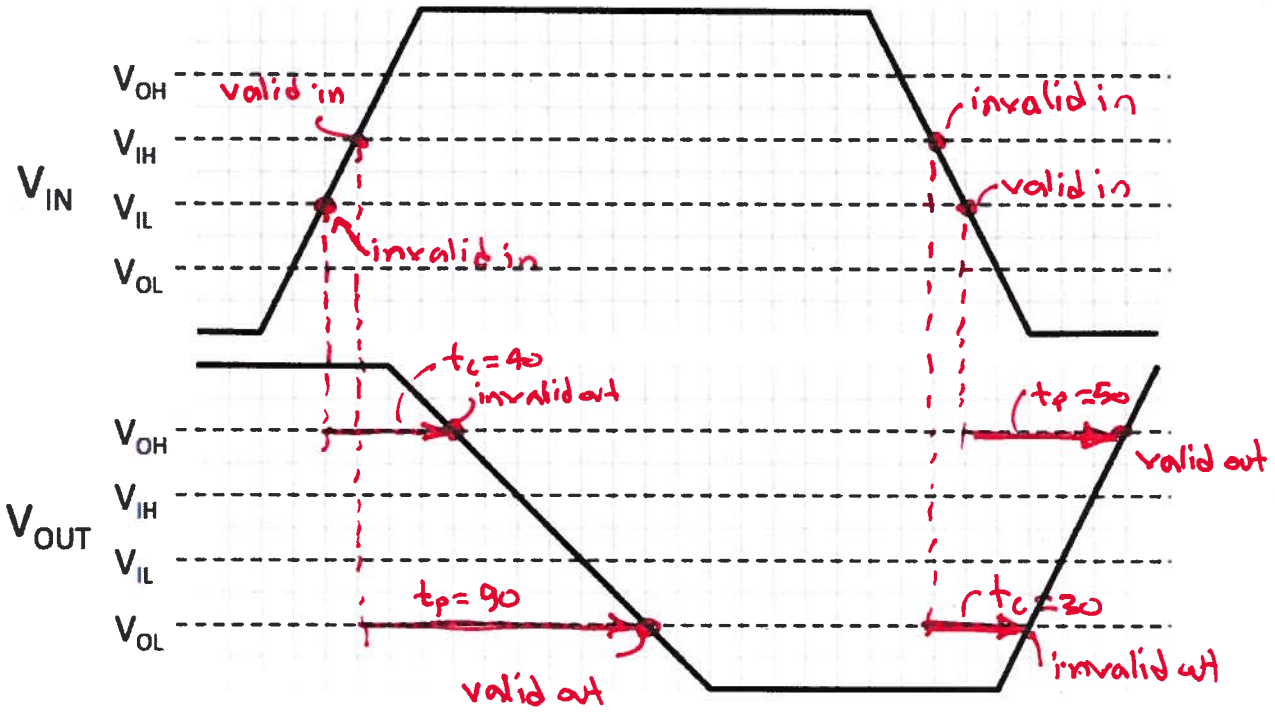
$$OUT = \overline{X + Y \cdot Z}$$

input	LEFT	RIGHT
A	Y	Z
B	GND	VDD
C	X	VDD
D	X	Y
E	GND	Z
F	GND	VDD



Problem 8.

The response of a combinational gate to a test input waveform is shown below. Each horizontal division of the plot represents 10 ps.



(A) Based on the figure below, what is an appropriate choice for the contamination delay of the gate?

$$\min(t_c) = \min(40, 30) = 30\text{ps} = t_{CD}$$

t_{CD} is lower bound on all t_c

(B) Based on the figure below, what is an appropriate choice for the propagation delay of the gate?

$$\max(t_p) = \max(90, 50) = 90\text{ps} = t_{PD}$$

t_{PD} is upper bound on all t_p.

6.004 Computation Structures
Spring 2017