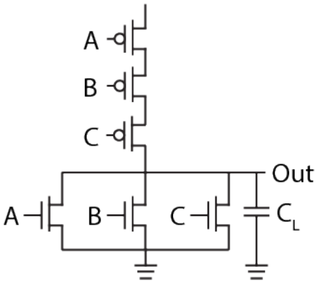
EECS 151/251A Homework 7

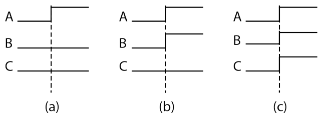
Due Friday, November 1st, 2019

Problem 1: Gate delays [5 pts]

Below is a CMOS implementation of a 3 input NOR gate. Each NMOS transistor is sized to have an on resistance of 3R while each PMOS transistor is sized to have an on resistance of R. For simplicity, assume the only capacitor in the circuit is the one explicitly shown. Initially all inputs are 1.



1. For the The inputs transition as shown in (a). Draw an RC equivalent of the circuit after t0. Calculate the propagation delay in terms of R and CL. Repeat the same for the input transitions shown in (b) and (c). You may assume that the input transitions are ideal.



2. On the same plot, sketch the waveform of the output for all three cases. Annotate the waveforms. (Note: Your waveforms do not have to be extremely accurate, but you need to clearly show the basic shape of the waveform as well as how they compare to each other.)

Problem 2: Complex Gate Sizing and Simplification [5 pts]

1. Design a complex CMOS logic gate that implements the function below, without simplifying the expression (in practice you would never build a gate without simplifying!). You can assume you have access to both regular and inverted versions of your inputs. Remember to size the transistors for equivalent worst case pull-up and pull-down delays.

$$f = \overline{A \cdot \overline{B} \cdot \overline{C} + A \cdot \overline{B} \cdot C + \overline{A} \cdot \overline{B} \cdot \overline{C} + \overline{A} \cdot \overline{B} \cdot C + B \cdot \overline{C}}$$

2. Using boolean algebra rules, simplify the expression from (1) as far as possible.

3. Design a CMOS logic gate to implement the reduced function from (2). Size the transistors as usual.

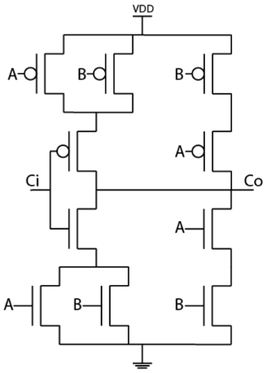
4. Compare the total transistor area of each implementation. Note: Gate area can be estimated as the sum of all transistor widths.

Problem 3: Complex Gate Sizing and Simplification [5 pts]

1. Implement the following logic function using a complementary pull-up and pull-down network. Remember to properly size the gates for equivalent worst-case pull-up and pull-down delays.

$$C_o = \overline{AB + AC_i + BC_i}$$

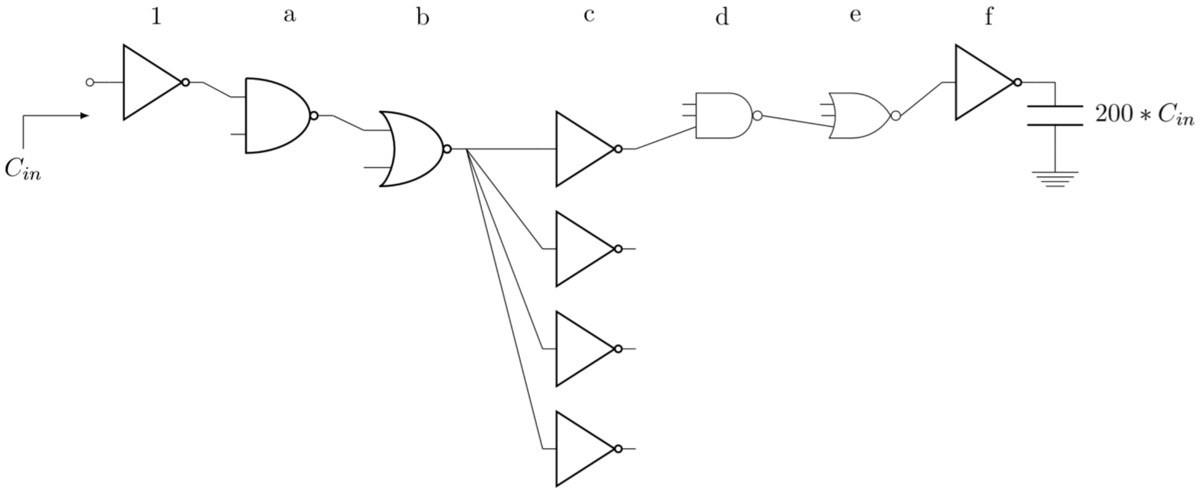
2. A friend proposes the following implementation of the function in subquestion (1). Does this perform the same function as the gate from (1)? If so, what advantage does it have over your implementation in (1)? If not, what is the function it implements?



3. Is the gate shown in (2) a static gate? Is it a complementary CMOS gate? Explain your answer.

Problem 4: Logical Effort [5 pts]

Size the gates in the chain below for minimum delay. As usual, all gates are sized so that the pull-up and pull-down resistance match that of the reference inverter.


1
a
b
c
d
e
f
C_{in}
$200 * C_{in}$