

**Problem #1: Complementary CMOS logic, Euler paths and logical effort**

For this problem we will use the following function:  $F = (a * b) \oplus [c * (d + e)]$ .  
 (Note:  $\oplus$  stands for an XOR operation). All signals and their complements are available as inputs.

- a) Find a PDN configuration, which implements the function  $Y = A \oplus B$  with four transistors.
- b) Using the result from a), draw the complementary CMOS circuit that implements the function  $F$ . Size it such that it has the same pull-up/pull-down strength as a minimum-sized 2/1 inverter.
- c) Determine the order of the input signals, which allows the largest number of diffusions to be shared. How many separate diffusion regions will you have in your layout? Show the signal graphs used to arrive at your solution.
- d) For the sizing used in part b), find the logical efforts ( $g$ ) for all of the inputs.

**Problem #2: Logic and logical effort**

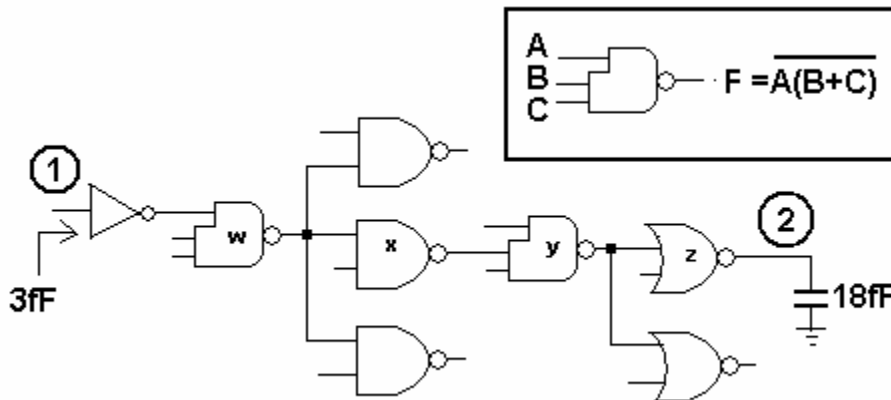


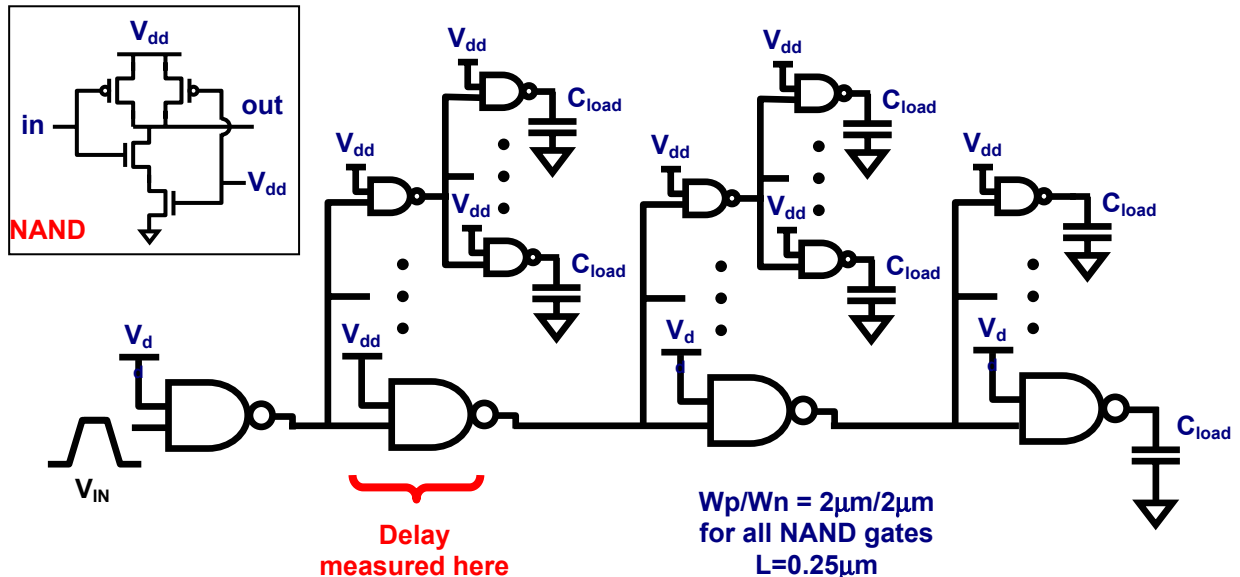
Figure 1 XNORT path.

1. A three-input XNORT gate (see insert above) works like a two-input NOR as long as input A is high; otherwise, the output is stuck high. Implement the XNORT gate in complementary CMOS, and size all transistors such that the worst-case delay is equal to that of a minimum sized 2/1 inverter. Find the logical effort associated with each input.

2. Assuming all input combinations are equally likely, what is the transition activity (probability) of a XNOR gate? Averaged over many cycles, will a XNOR gate typically consume more or less power than a two-input NOR gate, if they both drive equally large output loads? What about a two-input XOR?
3. For the logic path from node (1) to node (2) shown in Figure 1, find the path branching effort, path electrical effort, path logical effort, and total path effort. What is the optimum effort per stage for minimizing delay?
4. Find the input capacitances  $\{w, x, y, z\}$  necessary for each of the gates in the path in order to achieve the optimum effort per stage.

### Problem 3 – Self-loaded NAND Delay

Since finding the self-loaded inverter delay in the homework #5 was so easy for you, you decide to move on to a NAND gate. Again, you use a 4-stage chain... but you have to change the connections a bit to get the three stages flipping around correctly:



- a) Use HSPICE to find the average propagation delay for a NAND gate in this process for a fanout of 1, 2, 3, and 4. Plot the propagation delay as a function of the fanout.
- b) What is the self-loaded delay of a NAND gate?
- c) What is the slope of the best-fit line through your data points (additional delay per fanout)? This slope is related to the number obtained from logical effort calculations when you divide by the slope for an inverter (last homework)... compare  $\text{slope}_{\text{NAND}}/\text{slope}_{\text{INV}}$  to the theoretical value from logical effort.

- d) From your answers to b) and c), find the  $C_d/C_g$ . Compute  $C_g$  and  $C_d$ .
- e) Compute  $R_{eq}$ .

**Problem #4: An introductory adder problem**

- a) Show that the expression for S in Eq. 11.6 is equivalent to the one in Eq. 11.1
- b) Convince yourself that the circuit in Fig. 11.4 implements Eq. 11.6. Size the transistors such that each stage (carry and sum) has the same pull-up/pull-down strength as a minimum-sized 2/1 inverter. Assume the inverters are sized 2/1. Find the logical effort for each input (consider the carry and sum stages separately).
- c) Now consider the mirror adder in Fig. 11.6. Make sure that you understand what it does. Size the transistors such that each stage (carry and sum) has the same pull-up/pull-down strength as a minimum-sized 2/1 inverter. Find the logical effort for each input (consider the carry and sum stages separately). How does it compare to the architecture in Fig. 11.4?