Lecture #22

ANNOUNCEMENTS

- Midterm 2 thurs. april 15, 9:40-11am.
- A-M initials in 10 Evans
- N-Z initials in Sibley auditorium
- Closed book, except for two 8.5 x 11 inch cheat sheets
- Comprehensive, but focuses on HW's 5-9; L's,C's, 1st-order ckts,semiconductor devices, diode ckts, mosfet model, common source amplifier
- Tutebot lab kits distributed this week.

OUTLINE

» Timing diagrams

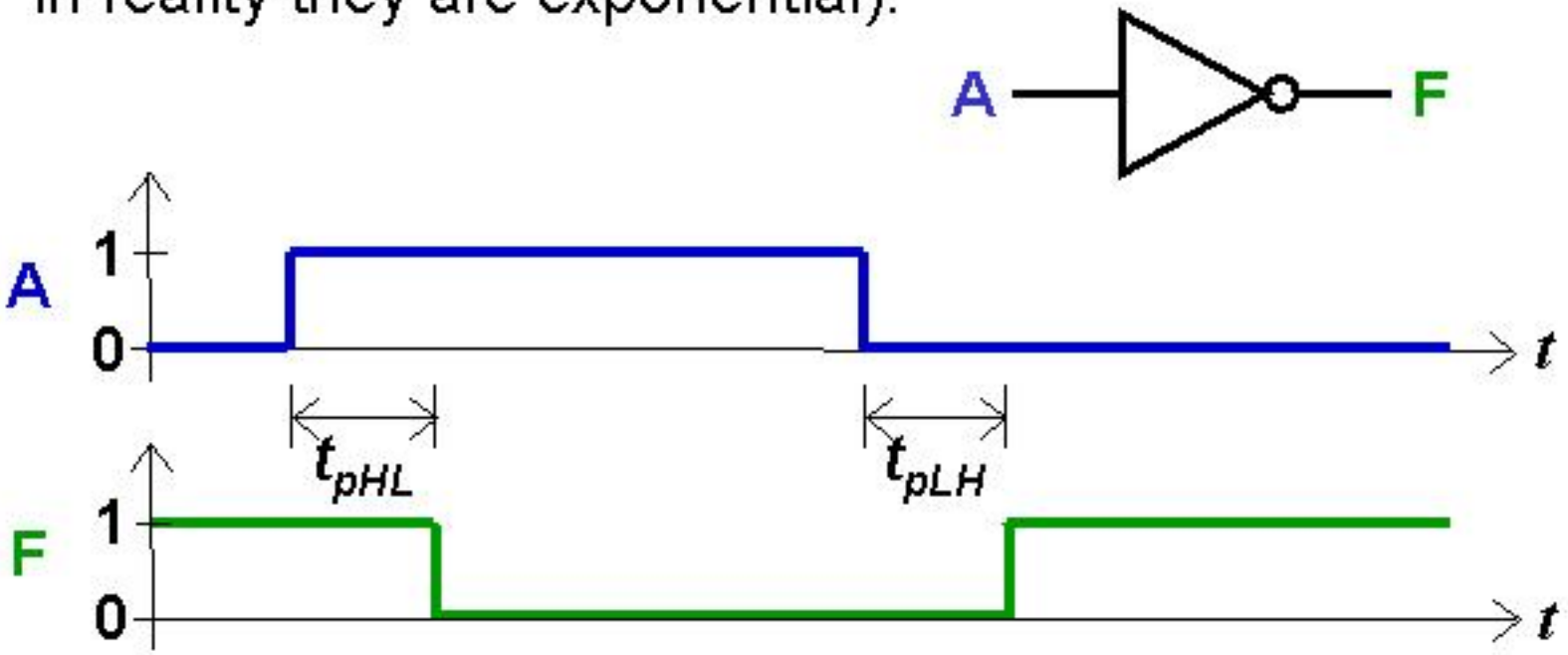
» **Delay Analysis**

Reading (Rabaey *et al.*)

- Chapter 5.4
- Chapter 6.2.1, pp. 260-263

Propagation Delay in Timing Diagrams

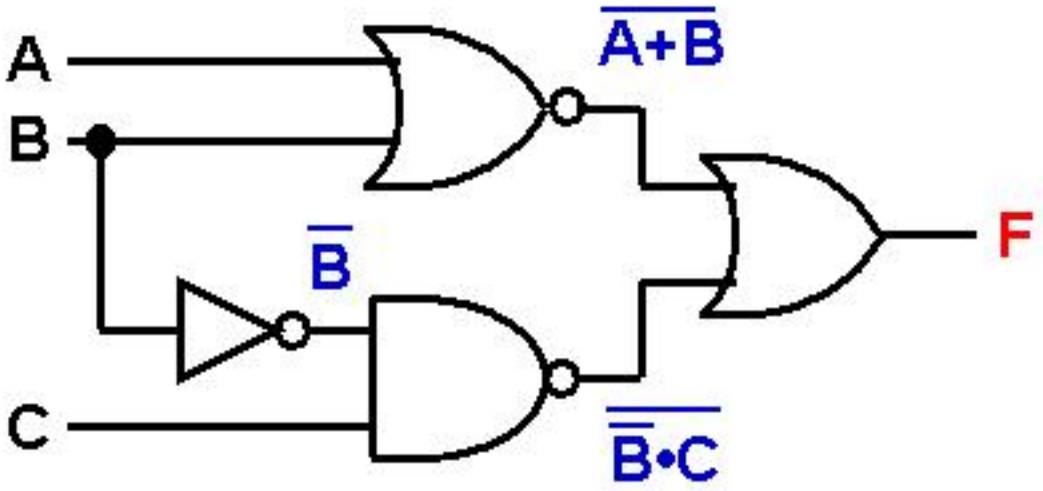
- To simplify the drawing of timing diagrams, we can approximate the signal transitions to be abrupt (though in reality they are exponential).

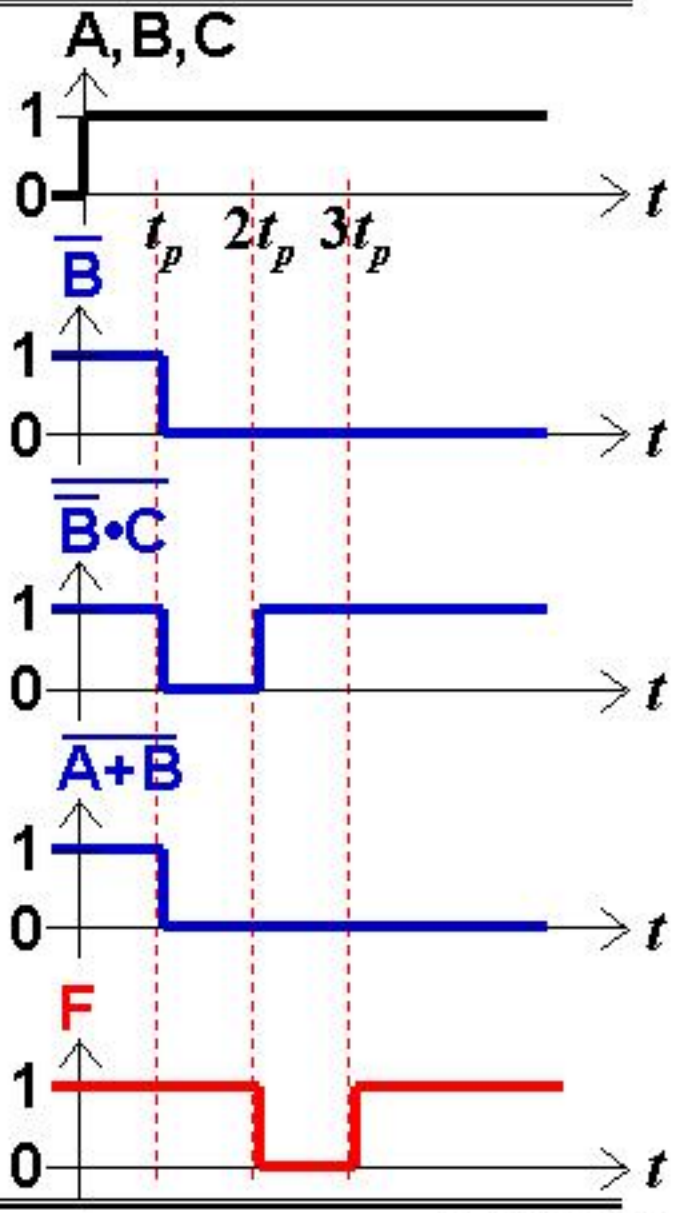


To further simplify timing analysis, we can define the propagation delay as $t_p = (t_{pHL} + t_{pLH})/2$

Glitching Transitions

The propagation delay from one logic gate to the next can cause spurious transitions, called ***glitches***, to occur. (A node can exhibit multiple transitions before settling to the correct logic level.)

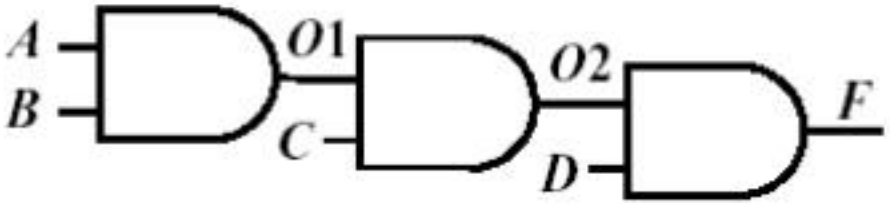




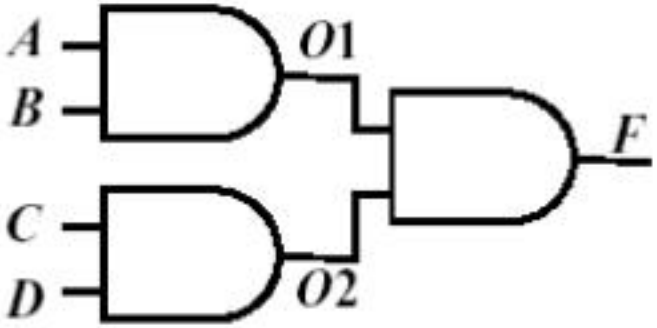
Glitch Reduction

- Spurious transitions can be minimized by balancing signal paths

Example: **F = A•B•C•D**



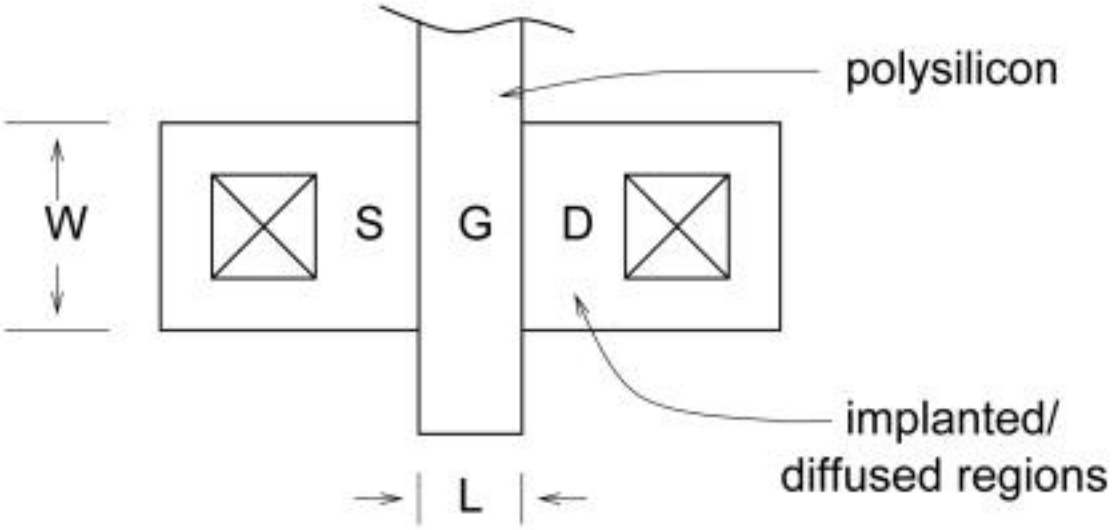
Chain structure



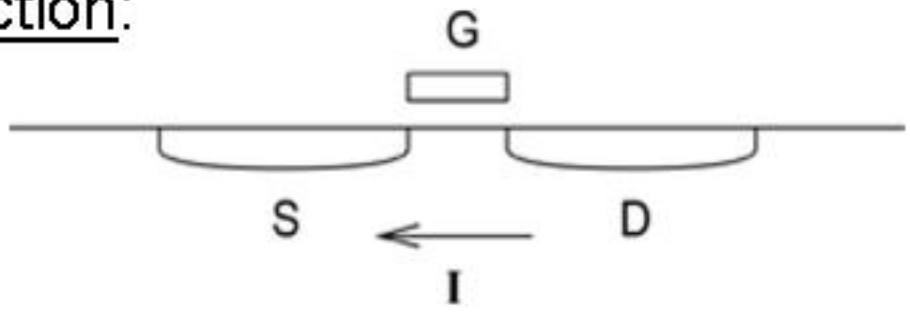
Tree structure

MOSFET Layout and Cross-Section

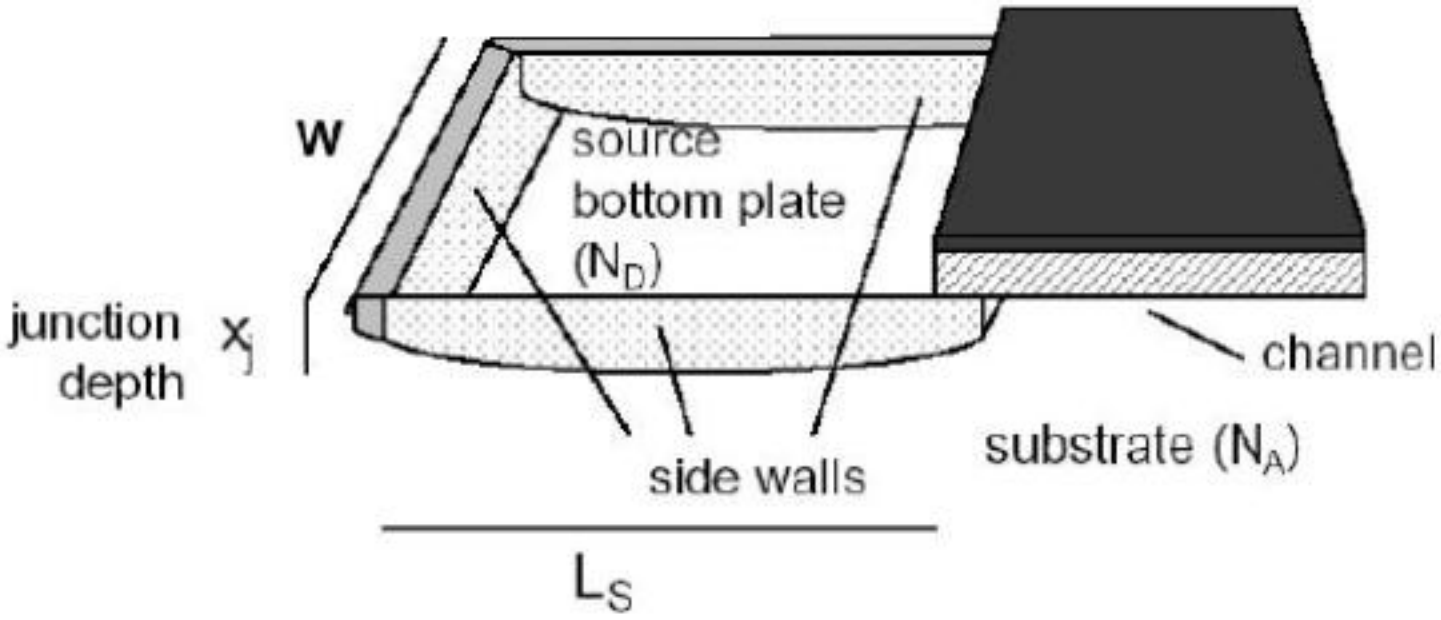
Top View:



Cross Section:



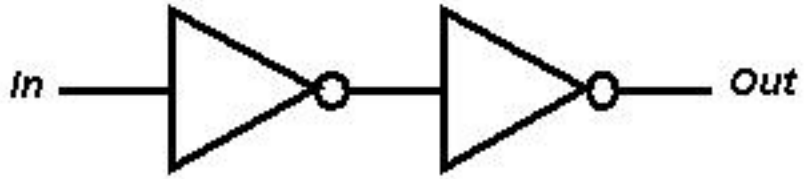
Source and Drain Junction Capacitance

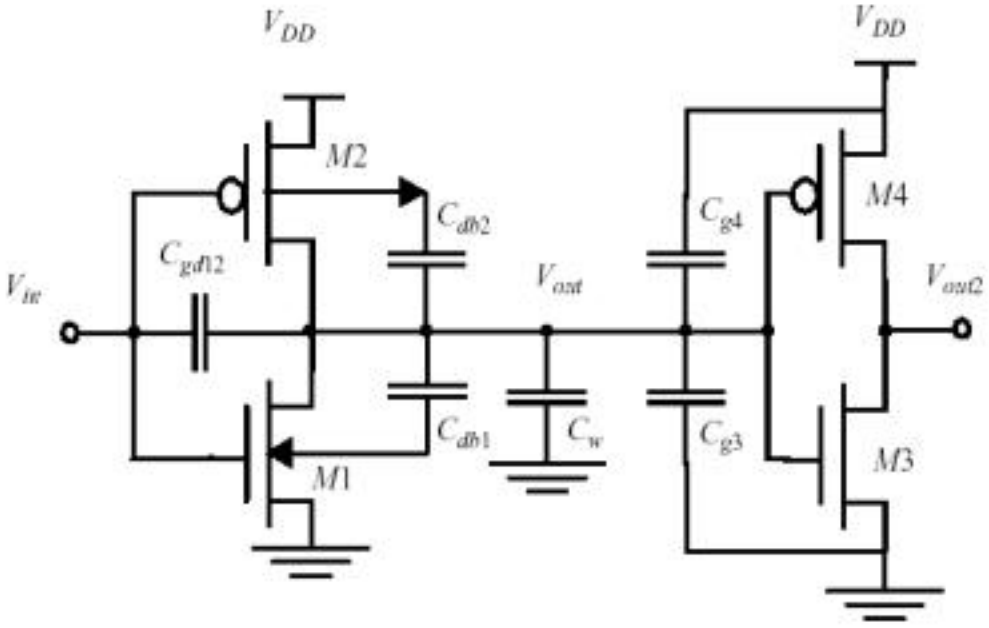


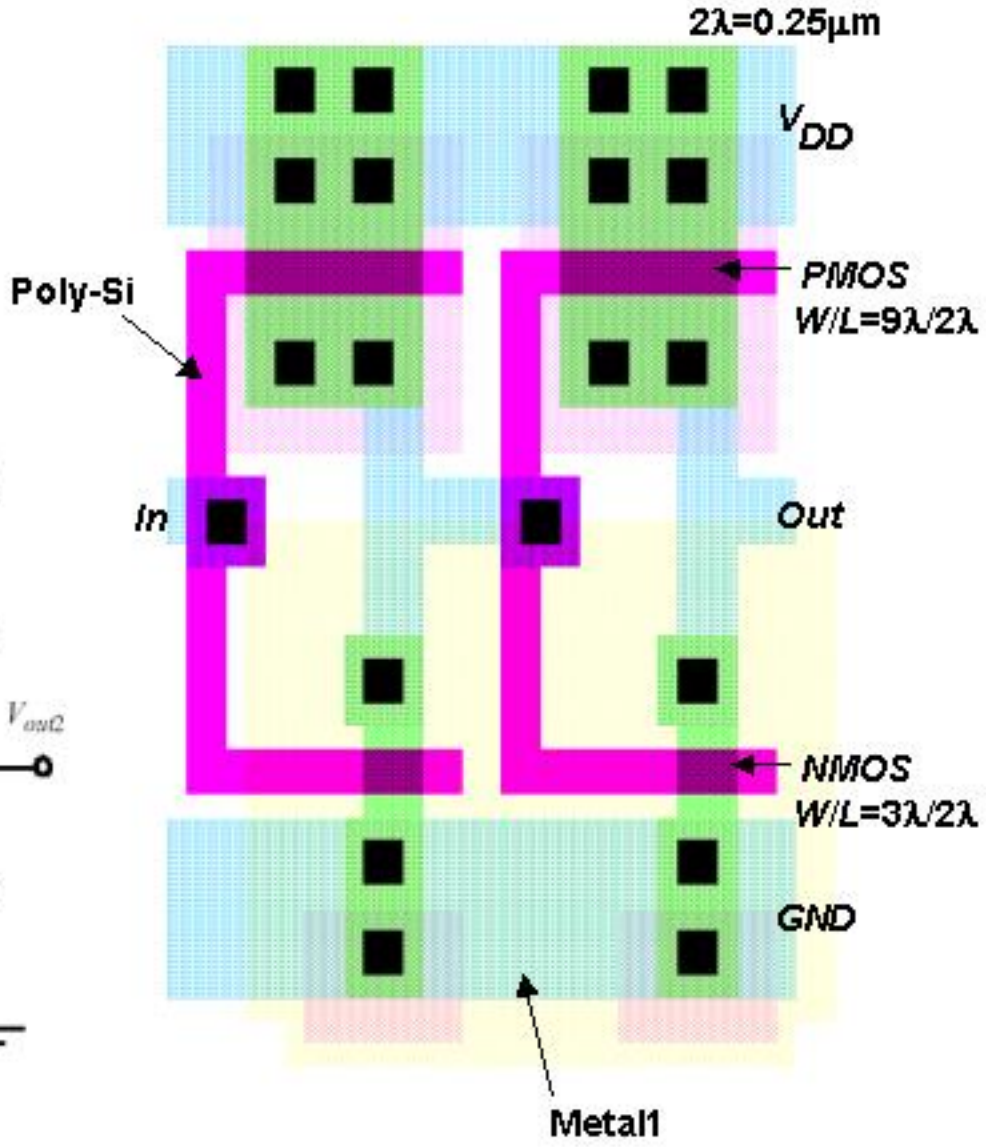
$$C_{source} = C_j \times (\text{AREA}) + C_{jsw} \times (\text{PERIMETER})$$
$$= C_j L_S W + C_{JSW}(2L_S + W)$$

Computing the Output Capacitance

Example 5.4 (pp. 197-203)

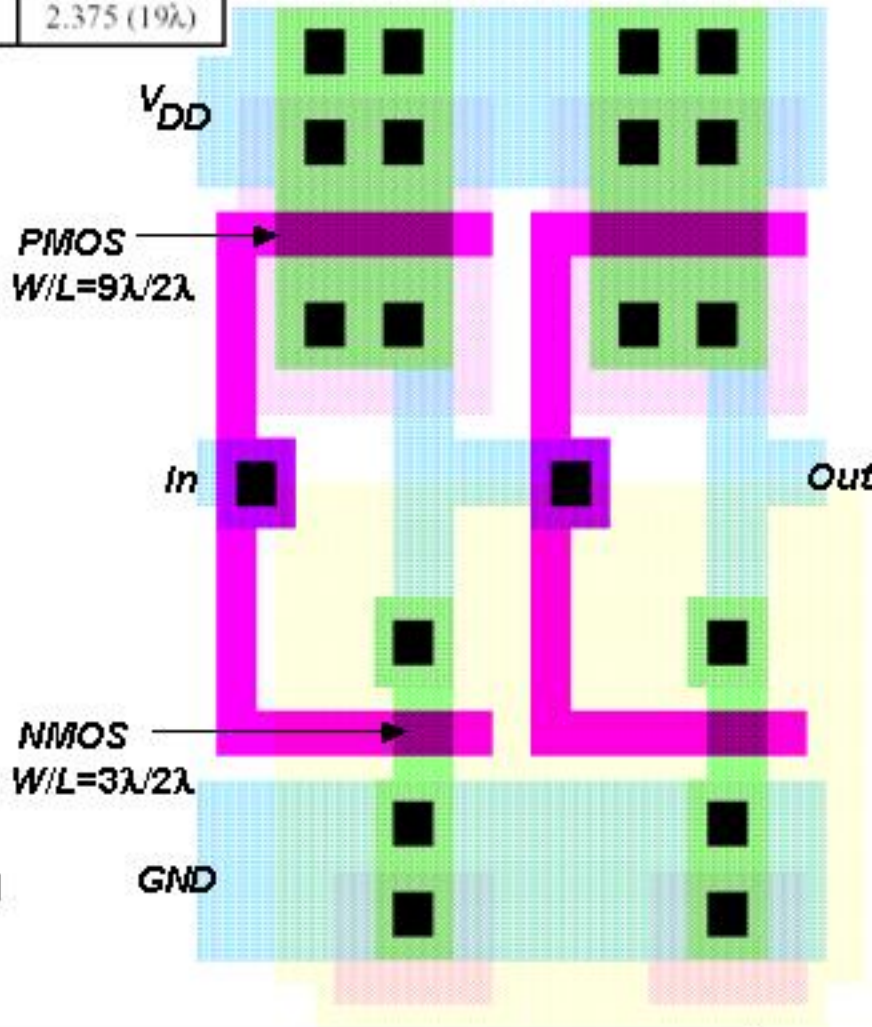






	W/L	AD (μm^2)	PD (μm)	AS (μm^2)	PS (μm)
NMOS	0.375/0.25	0.3 (19 λ^2)	1.875 (15λ)	0.3 (19 λ^2)	1.875 (15λ)
PMOS	1.125/0.25	0.7 (45 λ^2)	2.375 (19λ)	0.7 (45 λ^2)	2.375 (19λ)

$2\lambda = 0.25\mu m$



Capacitances for 0.25µm technology:

Gate capacitances:

- $C_{ox}(NMOS) = C_{ox}(PMOS) = 6\ fF/\mu m^2$

Overlap capacitances:

- $CGDO(NMOS) = C_{on} = 0.31fF/\mu m$
- $CGDO(PMOS) = C_{op} = 0.27fF/\mu m$

Bottom junction capacitances:

- $CJ(NMOS) = K_{eqbpn}C_j = 2\ fF/\mu m^2$
- $CJ(PMOS) = K_{eqbpp}C_j = 1.9\ fF/\mu m^2$

Sidewall junction capacitances:

- $CJSW(NMOS) = K_{eqswn}C_j = 0.28fF/\mu m$
- $CJSW(PMOS) = K_{eqbpp}C_j = 0.22fF/\mu m$

C Term	Expression	Value (fF) H→L	Value (fF) L→H
C_{GD1}	$2\,C_{on}\,W_n$	0.23	0.23
C_{GD2}	$2\,C_{op}\,W_p$	0.61	0.61
C_{DB1}	$K_{eqbpn}AD_nC_j + K_{eqswn}PD_nC_{jsw}$	0.66	0.90
C_{DB2}	$K_{eqbpp}AD_pC_j + K_{eqswp}PD_pC_{jsw}$	1.5	1.15
C_{G3}	$(2\,C_{on})W_n + C_{ox}W_nL_n$	0.76	0.76
C_{G4}	$(2\,C_{op})W_p + C_{ox}W_pL_p$	2.28	2.28
C_w	from extraction	0.12	0.12
C_L	Σ	6.1	6.0

Typical MOSFET Parameter Values

- For a given MOSFET fabrication process technology, the following parameters are known:
  - V_T (~0.5 V)
  - C_{ox} and k' (<0.001 A/V^2)
  - V_{DSAT} (≤ 1 V)
  - λ (≤ 0.1 V^{-1})

Example R_{eq} values for 0.25 μm technology ($W = L$):

V_{DD} (V)	1	1.5	2	2.5
NMOS (kΩ)	35	19	15	13
PMOS (kΩ)	115	55	38	31

Compute propagation delays

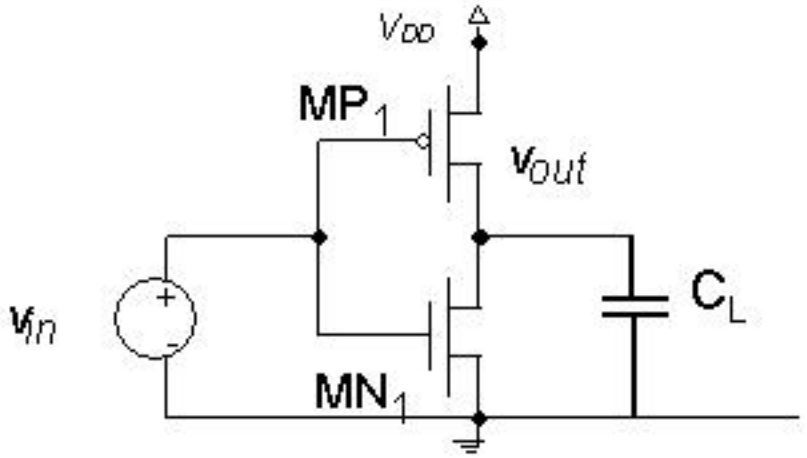
Examples of Propagation Delay

Product	CMOS technology generation	Clock frequency, *f*	Fan-out=4 inverter delay
Pentium II	0.25 μm	600 MHz	~100 ps
Pentium III	0.18 μm	1.8 GHz	~40 ps
Pentium IV	0.13 μm	3.2 GHz	~20 ps

Typical clock periods:

- high-performance μP: ~15 FO4 delays
- PlayStation 2: 60 FO4 delays

STATIC CMOS DRIVING LARGE LOADS



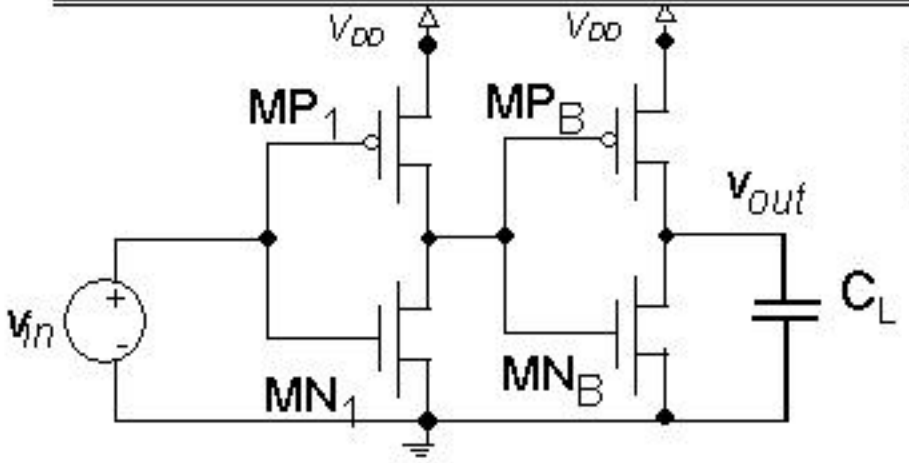
The load, C_L , may be the capacitance of a long line on the chip (e.g. up to 1pF, or may be the load on one of the chip output pins (e.g. up to 50pF).

We have seen that the typical driving resistance R for a minimum sized inverter is in the range of 10 KΩ. A 1 KΩ resistor driving a 50pF load would have a stage delay of 35nsec, huge in comparison to normal stage delays.

Thus we need to use larger devices to drive large capacitive loads, that is greatly increase W/L.

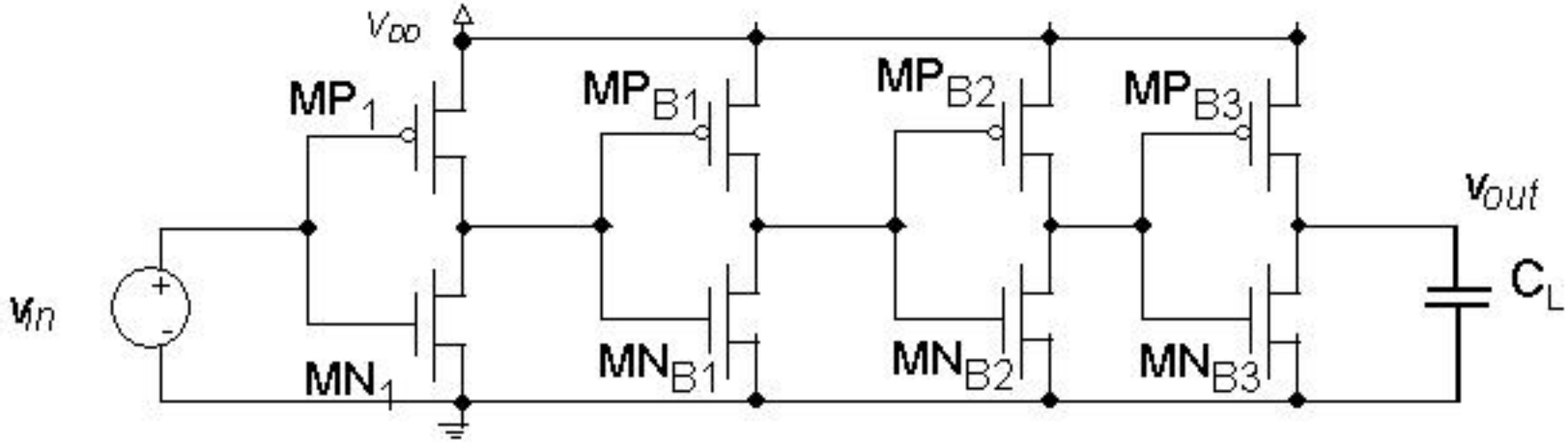
However, increasing W/L of a stage will increase the load it presents to the stage driving it, and we just move the delay problem back one stage.

STATIC CMOS DRIVING LARGE LOADS



PROBLEM: A minimum sized inverter drives a large load, C_L, leading to excessive delay, even with a buffer stage.

PROPOSED SOLUTION: Insert several simple inverter stages with increasing W/L between Inverter 1 and the load C_L. The total delay through the multiple stages will be less than the delay of one single stage driving C_L.



STATIC CMOS DRIVING LARGE LOADS

Example: The 2.5V 0.25μm CMOS inverter driving 50 pF load.

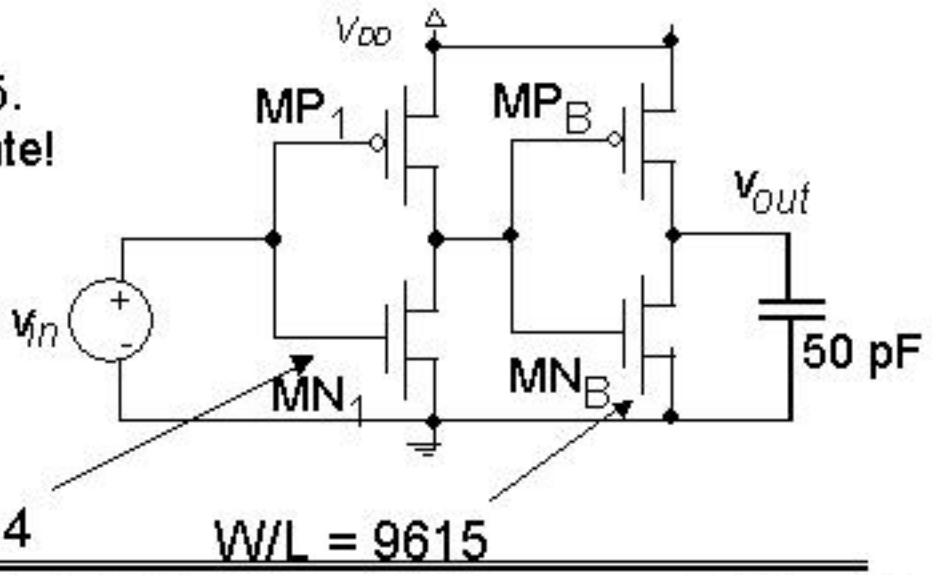
Properties: $W/L|_N = 1/.25$, $W/L|_P = 2/.25$, $V_{DD} = 2.5V$, $V_T = 0.5V$.
$Rn = 13\ K\Omega / 4 = 3.25\ K\Omega$; $Rp = 31\ K\Omega / 8 = 3.75\ K\Omega$
5nm oxide thickness , $C_{ox} = 6.9\ fF/\mu m^2$.
NMOS: $C_{Gp} = W \times L \times C_{ox} = 1.7\ fF$.
PMOS : $C_{Gp} = W \times L \times C_{ox} = 3.4\ fF$. Thus $C_{IN} = 5.2\ fF$

Basic gate delay (0.69RC) is about 10pS.

If we size one inverter to drive the load with this time constant it requires a W/L increase by a factor of 50pF/5.2fF =9615.
So C_{IN}= 50000fF =50pF for the buffer gate!

Thus the gate delay for the first stage is (50000/5.2)X10pS = 96.1nS.

Total delay = 96.1 + .01 = 96.11nS.
TOO LONG and NO IMPROVEMENT!



Note: We are ignoring drain capacitance in these examples.

STATIC CMOS DRIVING LARGE LOADS

Same example with tapered device sizes (geometric series)

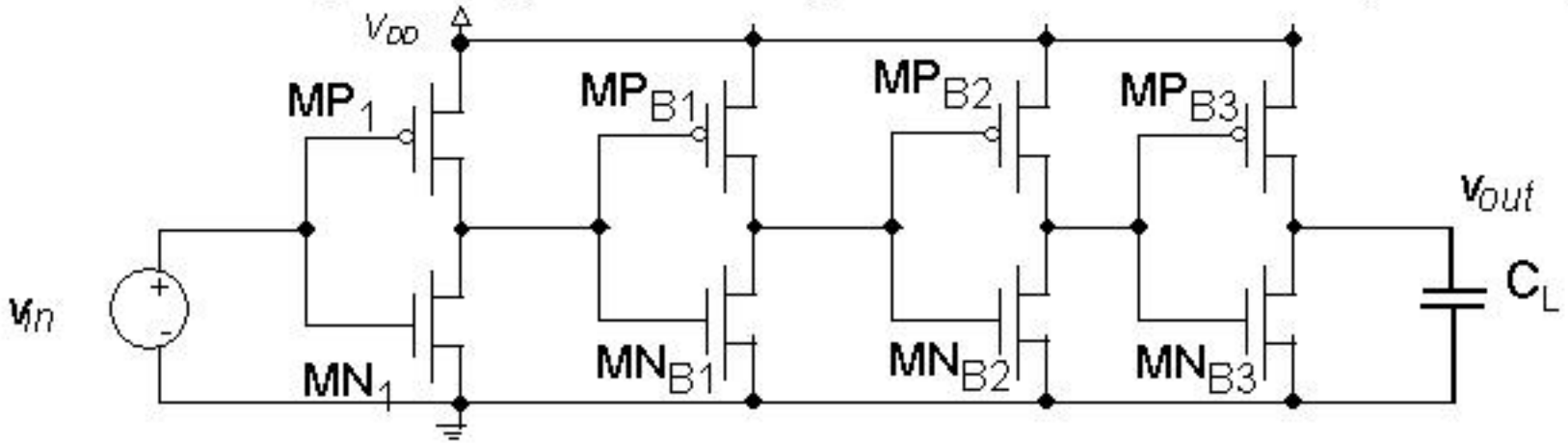
Case 1: Same example, but with buffer devices scaled by factor of 98 (98^2=9615)

Stage 1 load = 98 X 5.2fF, (R= 3.5K)

Stage 2 load = 50 pF , (R = 3.5K /98)

Delay = 98 X 10pS + 96nS/98 =0.98 +0.98 nS ~2nS

Case 2: Now taper through 3 buffer stages with W/L ratios of 9.9 (9.9^4=9615)



4 equal gate delays of 9.9 x 10pS =99pS Total = 4 X .099nS ~0.4nS

Gate delay through 4 gates is much less than through 2!

Note: We are ignoring drain capacitance in these examples.

STATIC CMOS DRIVING LARGE LOADS

Comments

In our example we got better results with 3 buffer stages than 1. 7 buffer stages would do even better.

How many buffer stages are optimum? Well under these simple assumptions (like ignoring drain and wiring capacitance, and operating asynchronously) you can show that the number of buffer stages, N obeys N +1 = ln(R) where R is the ratio of the load capacitance to the capacitance of a minimum sized stage. This formula is not important, but you should remember the concept that buffering with multiple stages usually leads to lower net delay if the load is large.