EECS151 : Introduction to Digital Design and ICs

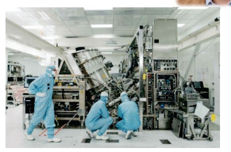
Lecture 14 – Gate Delays

Bora Nikolić

Moore's Law Could Ride EUV for 10 More Years

September 30, 2021, EETimes - ASML plans to introduce new extreme ultraviolet (EUV) lithography equipment that will extend the longevity of Moore's Law for at least ten years, according to executives at the world's only supplier of the tools, which are crucial for the world's most advanced silicon.

Starting in the first half of 2023, the company plans to offer customers equipment that takes EUV numerical aperture (NA) higher to 0.55 NA from the existing 0.33 NA. The company believes that the new equipment will help chip makers reach process nodes well beyond the current threshold (2nm) for at least another 10 years, according to ASML vice president Teun van Gogh, in an interview with EE Times.



EETimes

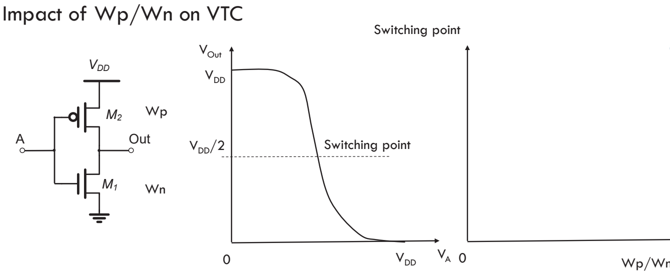
Review

- CMOS allows for convenient switch level abstraction
- CMOS pull-up and pull-down networks are complementary
  - Graph models for CMOS gates
- Transistor sizing affects gate performance

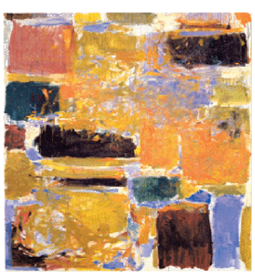
CMOS Sizing

Transistor Sizing

- Impact of Wp/Wn on VTC

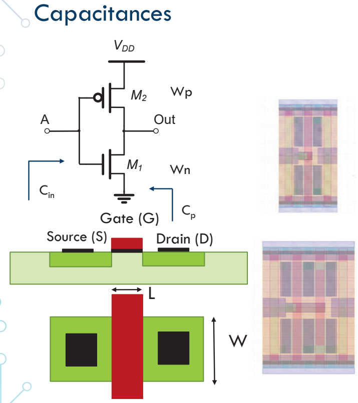


- In the past, Wp > Wn (see Rabaey, 2nd ed)
- In modern processes (finFET), Wp = Wn
- Weak dependence on Wp/Wn



CMOS Delay

Capacitances

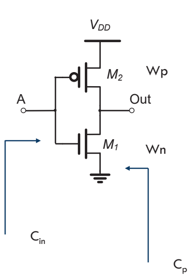


- C_{in} is largely set by the gate cap
  - ~WL
  - 2xW = 2xC_{in}
  - It is non-linear, but we will ignore that
- C_p is largely set by the drain cap
  - ~W (drain area/perimeter)
  - 2xW = 2xC_p

$\boxed{C_p = \gamma C_{in}}$

Gate Sizing

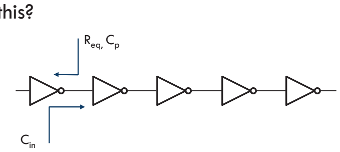
- Doubling the gate size (by doubling Ws):



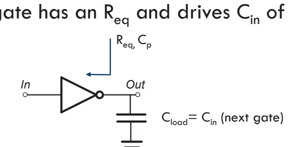
- Doubles C_{in}
- Halves equivalent gate resistance
- Doubles C_p

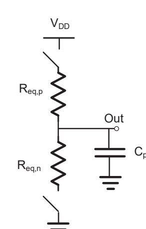
Inverter Delay

- How to time this?



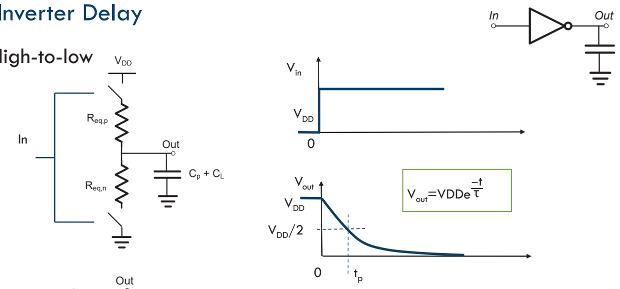
- Each gate has an R_{eq} and drives C_{in} of the next gate

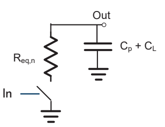




Inverter Delay

- High-to-low

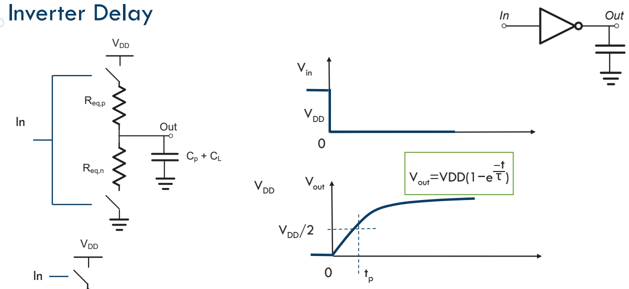


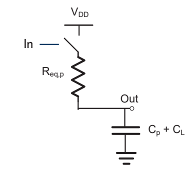


$t_{p,HL} = (\ln 2)\tau = 0.7\ Req_{,n}(C_p+C_L)$

$\tau = Req_{,n}(C_p+C_L)$

Inverter Delay

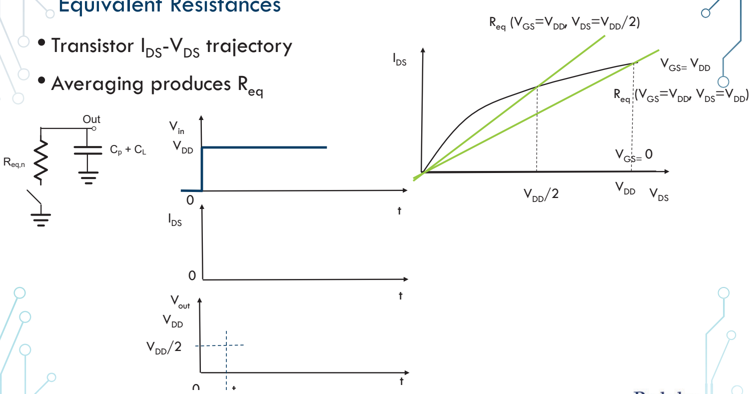




$t_{p,LH} = (\ln 2)\tau = 0.7\ Req_{,p}(C_p+C_L)$

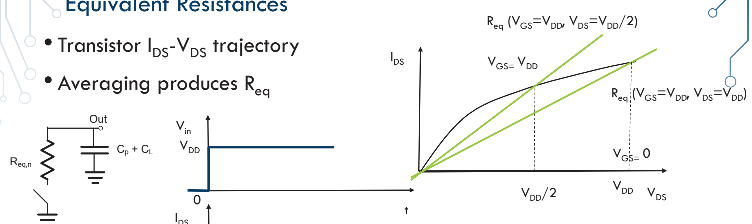
Equivalent Resistances

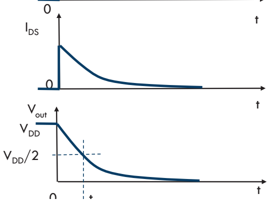
- Transistor I_{DS}-V_{DS} trajectory
- Averaging produces R_{eq}



Equivalent Resistances

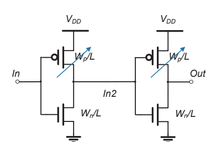
- Transistor I_{DS}-V_{DS} trajectory
- Averaging produces R_{eq}





$R_{eq} = (R_{eq,start} + R_{eq,mid})/2$

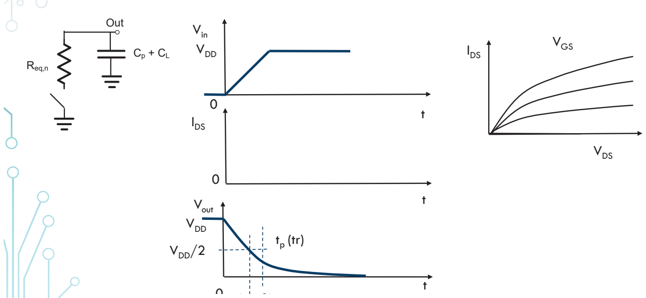
Optimal P/N Sizing



- Increasing Wp:
  - Reduces R_p, increases $C_{in,p}$
  - Reduces $t_{p,LH}$
  - Increases $t_{p,HL}$
- Optimum
  - Wp/Wn = 2 in older technologies, with velocity saturation (like 130nm)
  - Wp/Wn = 1.6 in technologies with strained silicon (e.g. 28nm)
  - Wp/Wn = 1 in finFET technologies

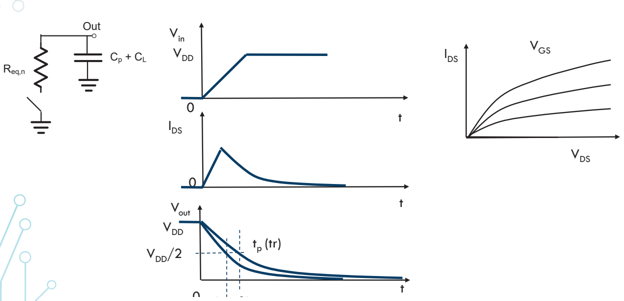
Impact of Rise/Fall times

- Impacts the I_{DS}-V_{DS} trajectory



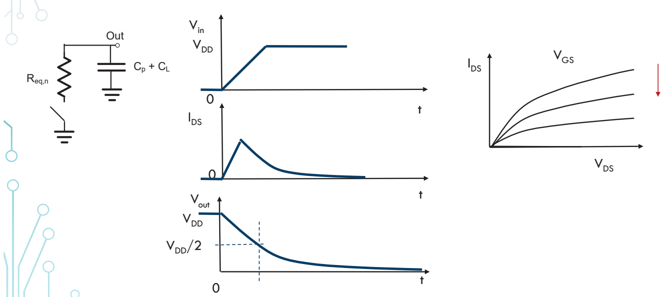
Impact of Rise/Fall times

- Impacts the I_{DS}-V_{DS} trajectory



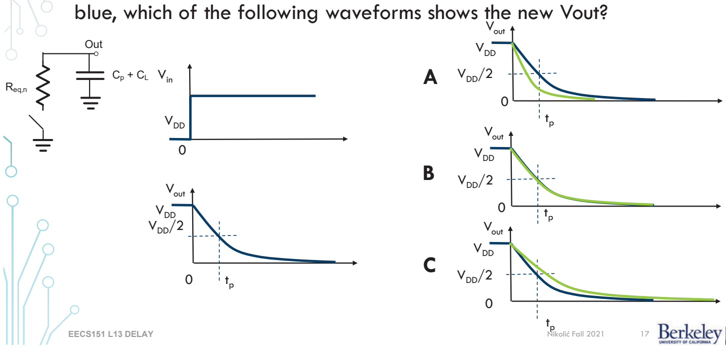
Impact of Supply Voltage

- Lowering VDD, slows down the circuit



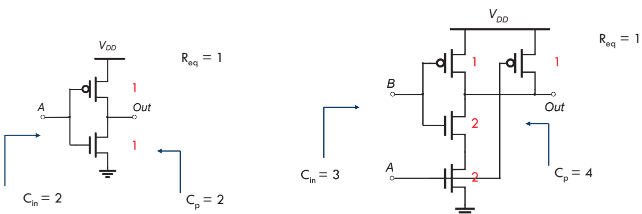
Quiz: Inverter Delay

- If we double the load capacitance, assuming the default Vout shown in blue, which of the following waveforms shows the new Vout?



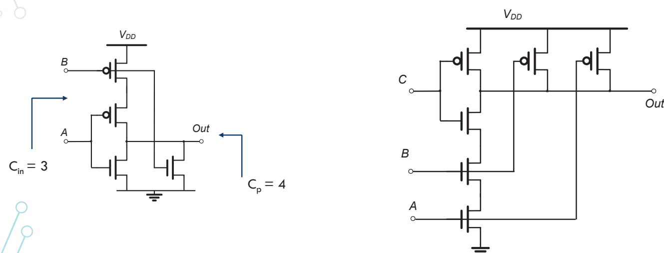
Sizing CMOS Gates

Sizing for equal output resistance

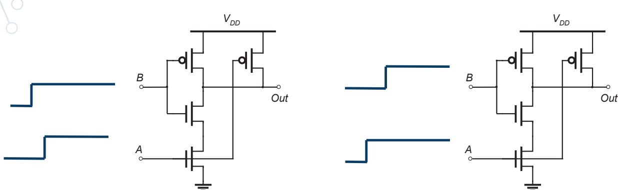


- In velocity-saturated devices Ion of a stack is 2/3 (not a half) of two devices
  - So the correct upsizing factor is 1.5 (not 2)
- We will use 2, as it makes calculations easier

Other Gates, NOR2, NAND3



Stack Ordering

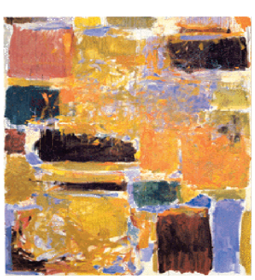


- Critical path goes on top of stack

Administrivia

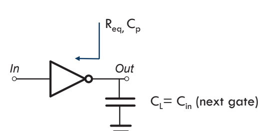
- Homework 5 due this week
- Lab 6 (last) this week
- Projects start next week

- There is still time to apply for undergrad scholarships in SoC design!
- Have you thought about doing undergrad research?



Minimizing Logic Delay

Inverter RC Delay



- $t_p = R_{eq}(C_p + C_L) = Req(\gamma\ Cin + C_L)$
  - $\gamma = 1$ (closer to 1.2 in recent processes)
- $t_p = R_{eq}C_{in}(1+C_L/C_{in}) = \tau_{INV}(1+f)$
  - Propagation delay is proportional to fanout
- Normalized Delay = 1 + f

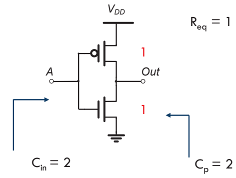
$\boxed{\text{Fanout} = f = C_L/C_{in}}$

$\boxed{t_p = \tau_{INV}(1+f)}$

Generalizing to Arbitrary Gates

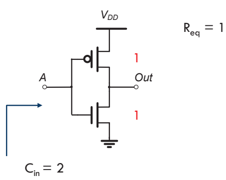
- Delay has two components: $d = f + p$
- f: *effort delay* $= gh$ (a.k.a. stage effort)
  - Again has two components
- g: *logical effort*
  - Measures relative ability of gate to deliver current
  - $g = 1$ for inverter
- h: *electrical effort* $= C_{out} / C_{in}$
  - Ratio of output to input capacitance
  - Sometimes called fanout
- p: parasitic delay
  - Represents delay of gate driving no load
  - Set by internal parasitic capacitance

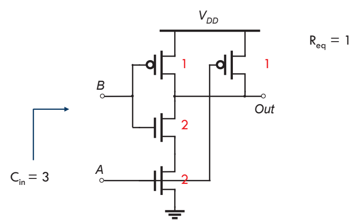
Inverter Delay



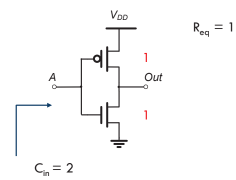
- Parasitic p is the ratio of intrinsic capacitance to an inverter
  - p(inverter) =
- Logical Effort g is the ratio of input capacitance to an inverter
  - g(inverter) =
- Electrical Effort h is the ratio of the load capacitance to the input capacitance
  - h(inverter) =
- Delay = p + f = p + g * h = 1 + f

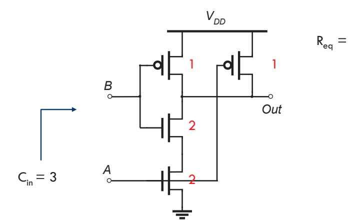
NAND2 Gate





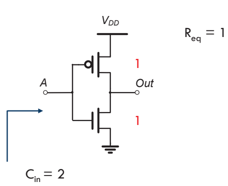
Logical Effort of NAND2 Gate

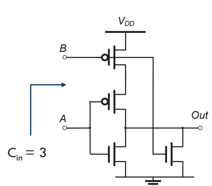




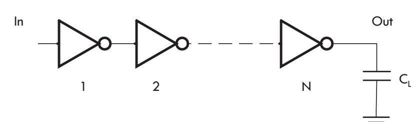
- In velocity-saturated devices Ion of a stack is 2/3 (not a half) of two devices
  - So the correct upsizing factor is 1.5 (not 2)
- We will use 2, as it makes calculations easier

NOR2 Gate



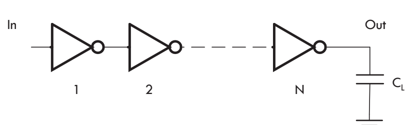


Example: Inverter Chain



Logical Effort:	g =
Electrical Effort:	h =
Parasitic Delay:	p =
Stage Delay:	d =
Total Delay:	d_total =

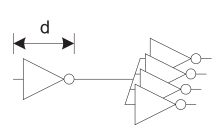
Example: Inverter Chain



Logical Effort:	g = 1
Electrical Effort:	h = 1
Parasitic Delay:	p = 1
Stage Delay:	d = 2
Total Delay:	d_total = 2*N

Example: FO4 Inverter

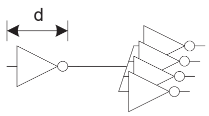
- Estimate the delay of a fanout-of-4 (FO4) inverter



Logical Effort:	g =
Electrical Effort:	h =
Parasitic Delay:	p =
Stage Delay:	d =

Example: FO4 Inverter

- Estimate the delay of a fanout-of-4 (FO4) inverter



Logical Effort: g = 1

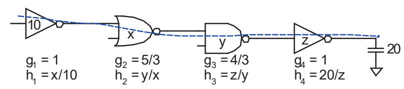
Electrical Effort: h = 4

Parasitic Delay: p = 1

Stage Delay: d = 5

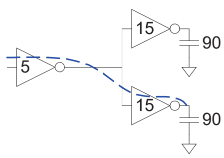
Multi-stage Logic Networks

- Logical effort generalizes to multistage networks
- *Path Logical Effort* $G = \prod g_i$
- *Path Electrical Effort* $H = \dfrac{C_{\text{out-path}}}{C_{\text{in-path}}}$
- *Path Effort* $F = \prod f_i = \prod g_i h_i$



Branching Effect

$$b = \frac{C_{\text{on path}} + C_{\text{off path}}}{C_{\text{on path}}} \qquad B = \prod b_i$$



G = 1

H = 90 / 5 = 18

GH = 18

h_1 = (15 +15) / 5 = 6

h_2 = 90 / 15 = 6

B = 2

F = $g_1 g_2 h_1 h_2$ = 36 = BGH

Designing Fast Circuits

$$D = \sum d_i = D_F + P$$

- Delay is smallest when each stage bears same effort

$$\hat{f} = g_i h_i = F^{\frac{1}{N}}$$

- Thus minimum delay of N stage path is

$$D = NF^{\frac{1}{N}} + P$$

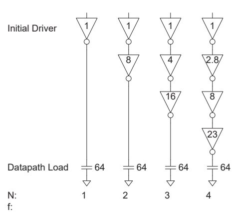
- This is a key result of logical effort
  - Find fastest possible delay
  - Doesn't require calculating gate sizes

Example: Best Number of Stages

- How many stages should a path use?
  - Minimizing number of stages is not always fastest
- Example: drive 64-bit datapath with unit inverter

D = $NF^{1/N}$ + P

= $N(64)^{1/N}$ + N

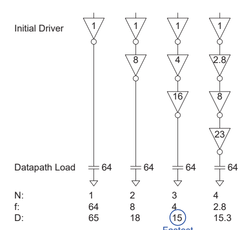


Example: Best Number of Stages

- How many stages should a path use?
  - Minimizing number of stages is not always fastest
- Example: drive 64-bit datapath with unit inverter

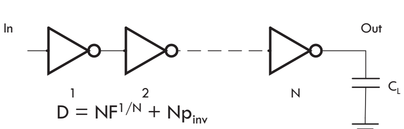
D = $NF^{1/N}$ + P

= $N(64)^{1/N}$ + N



Best Stage Effort

- How many stages should a path use?
  - To drive given capacitance



D = $NF^{1/N}$ + Np_{inv}

- Define best stage effort $\rho = F^{\frac{1}{N}}$
- Neglecting parasitics (p_{inv} = 0), we find ρ = e = 2.718
- For p_{inv} = 1, solve numerically for ρ = 3.59
- Choose 4 – less stages, less energy

Logical Efforts Method

1) Compute path effort $F = GBH$

2) Estimate best number of stages $N = \log_4 F$

3) Sketch path with N stages $D = NF^{\frac{1}{N}} + P$

4) Estimate least delay $\hat{f} = F^{\frac{1}{N}}$

5) Determine best stage effort $C_{in_i} = \dfrac{g_i C_{out_i}}{\hat{f}}$

6) Find gate sizes

Summary

- Delay is a linear function of R and C
- Delay optimization is critical to improve the frequency of the circuit.
- The dimensions of a transistor affect its capacitance and resistance.
- We use RC delay model to describe the delay of a circuit.
- Two delay components:
  - Parasitic delay (p)
  - Effort delay (F)
    - Logical effort (g): intrinsic complexity of the gate
    - Electrical effort (h): load capacitance dependent