EE 330
Lecture 43

Digital Circuits

- Logic Effort
- Elmore Delay
- Power Dissipation

EE 330
Lecture 42

Digital Circuits

- Propagation Delay with Arbitrary Gate Sizing
- Optimally driving large capacitive loads
- Logic Effort

Exam Schedule

Exam 1	Friday Sept 24
Exam 2	Friday Oct 22
Exam 3	Friday Nov 19
Final	Tues Dec 14 12:00 p.m.

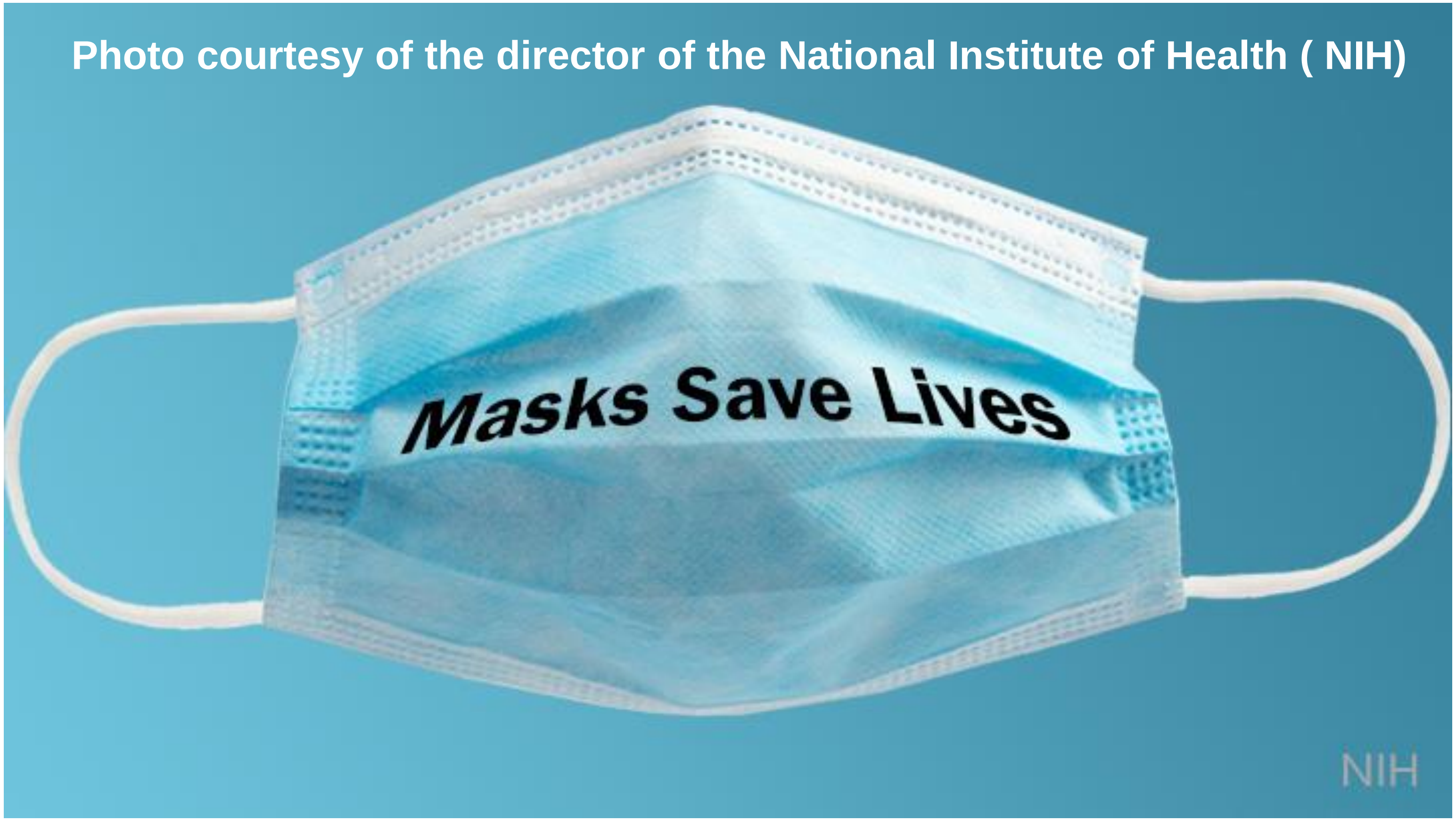


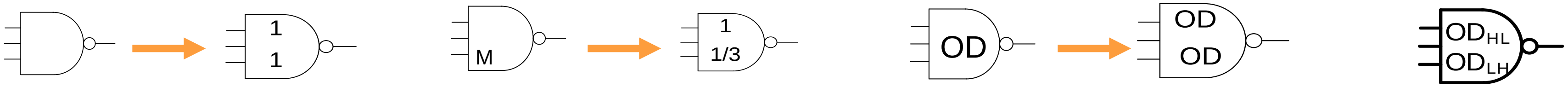
Photo courtesy of the director of the National Institute of Health (NIH)

As a courtesy to fellow classmates, TAs, and the instructor

Wearing of masks during lectures and in the laboratories for this course would be appreciated irrespective of vaccination status

Review from Last Time

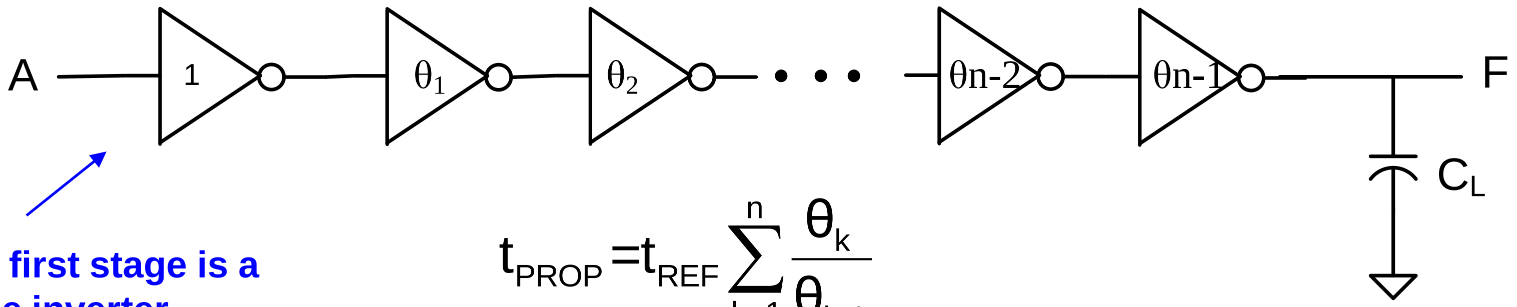
Summary: Propagation Delay in Multiple-Levels of Logic with Stage Loading



	Equal Rise/Fall	Equal Rise/Fall (with OD)	Minimum Sized	
C_{IN}/C_{REF}				
Inverter	1	OD	1/2	O
NOR	$\frac{3k+1}{4}$	$\frac{3k+1}{4} \bullet OD$	1/2	C
NAND	$\frac{3+k}{4}$	$\frac{3+k}{4} \bullet OD$	1/2	k •
Overdrive				
Inverter				
HL	1	OD	1	
LH	1	OD	1/3	
NOR				
HL	1	OD	1	
LH	1	OD	1/(3k)	
NAND				
HL	1	OD	1/k	
LH	1	OD	1/3	

Review from Last Time

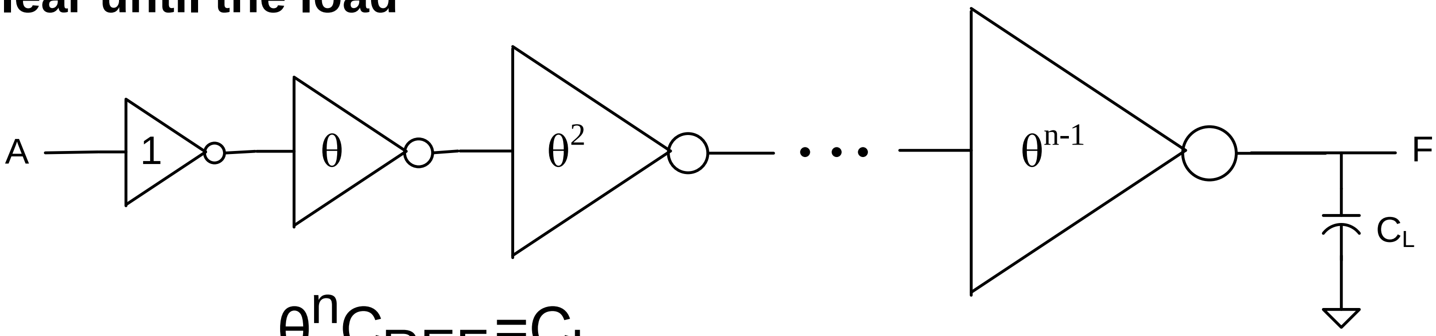
Optimal Driving of Capacitive Loads



Assume first stage is a reference inverter

$$t_{PROP} = t_{REF}\sum_{k=1}^{n}\frac{\theta_k}{\theta_{k-1}}$$

Order reduction strategy : Assume overdrive of stages increases by the same factor clear until the load



$$\theta^n C_{REF} = C_L$$

This becomes a 2-parameter optimization (minimization) problem !

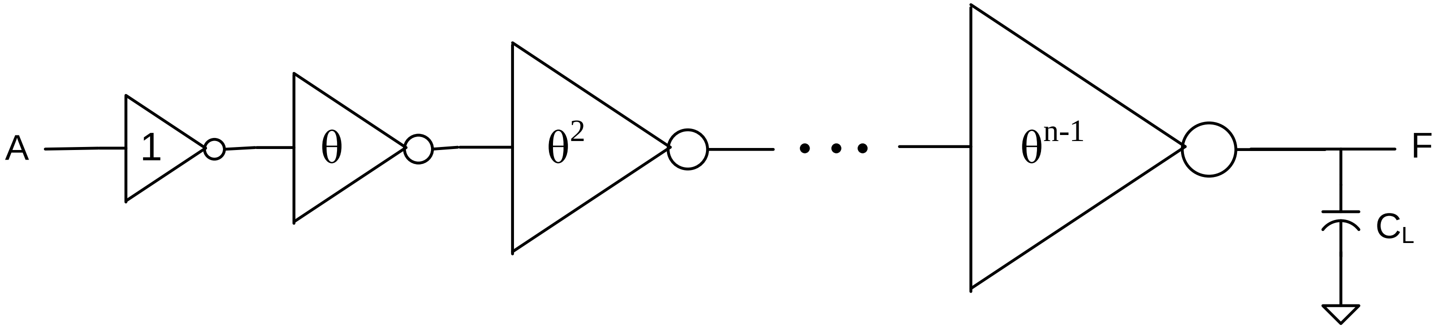
Unknown parameters: $\{\theta, n\}$

One constraint : $\theta^n C_{REF} = C_L$

One degree of freedom

Review from Last Time

Optimal Driving of Capacitive Loads



$$\theta^n C_{REF} = C_L$$

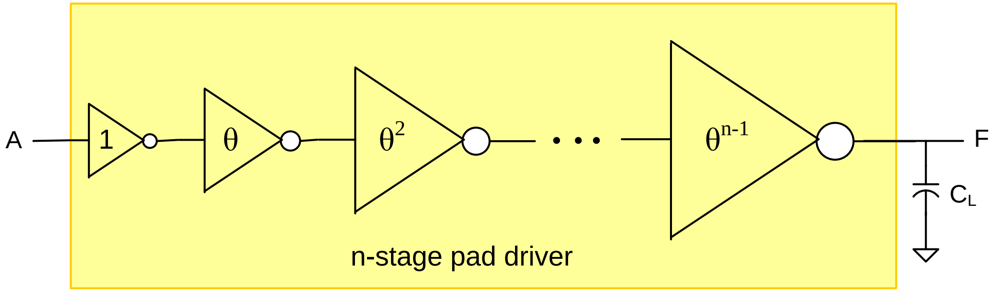
$$\theta_{OPT} = e$$

$$n_{OPT} = \ln\left(\frac{C_L}{C_{REF}}\right) = \ln(FI_L)$$

$$t_{PROP} = t_{REF}\frac{\theta}{\ln(\theta)}\left[\ln\frac{C_L}{C_{REF}}\right] \qquad t_{PROP} = t_{REF}e\left[\ln\frac{C_L}{C_{REF}}\right] = n\theta t_{REF}$$

Review from Last Time

Optimal Driving of Capacitive Loads



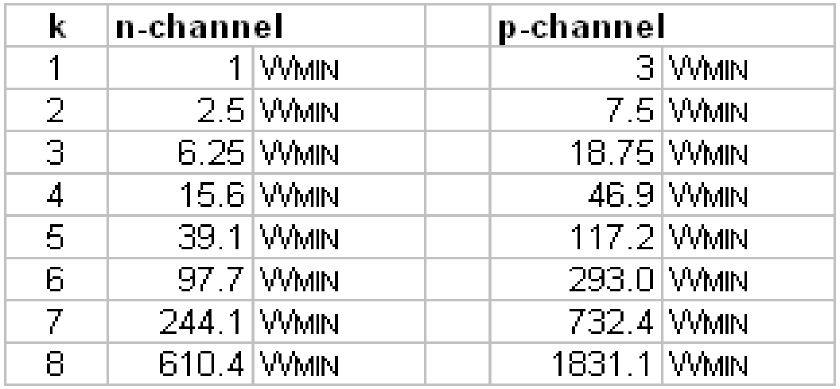
Example: Design a pad driver for driving a load capacitance of 10pF with equal rise/fall times, determine t_{PROP} for the pad driver, and compare this with the propagation delay for driving the pad with a minimum-sized reference inverter.

In 0.5u proc t_{REF}=20ps, C_{REF}=4fF,R_{PDREF}=2.5K

For $\theta = 2.5$, n=8 $W_{REF}=W_{MIN}$

$W_{nk}=2.5^{k-1}\bullet W_{REF}$, $W_{pk}=3\bullet 2.5^{k-1}\bullet W_{REF}$

$L_n=L_p=L_{MIN}$

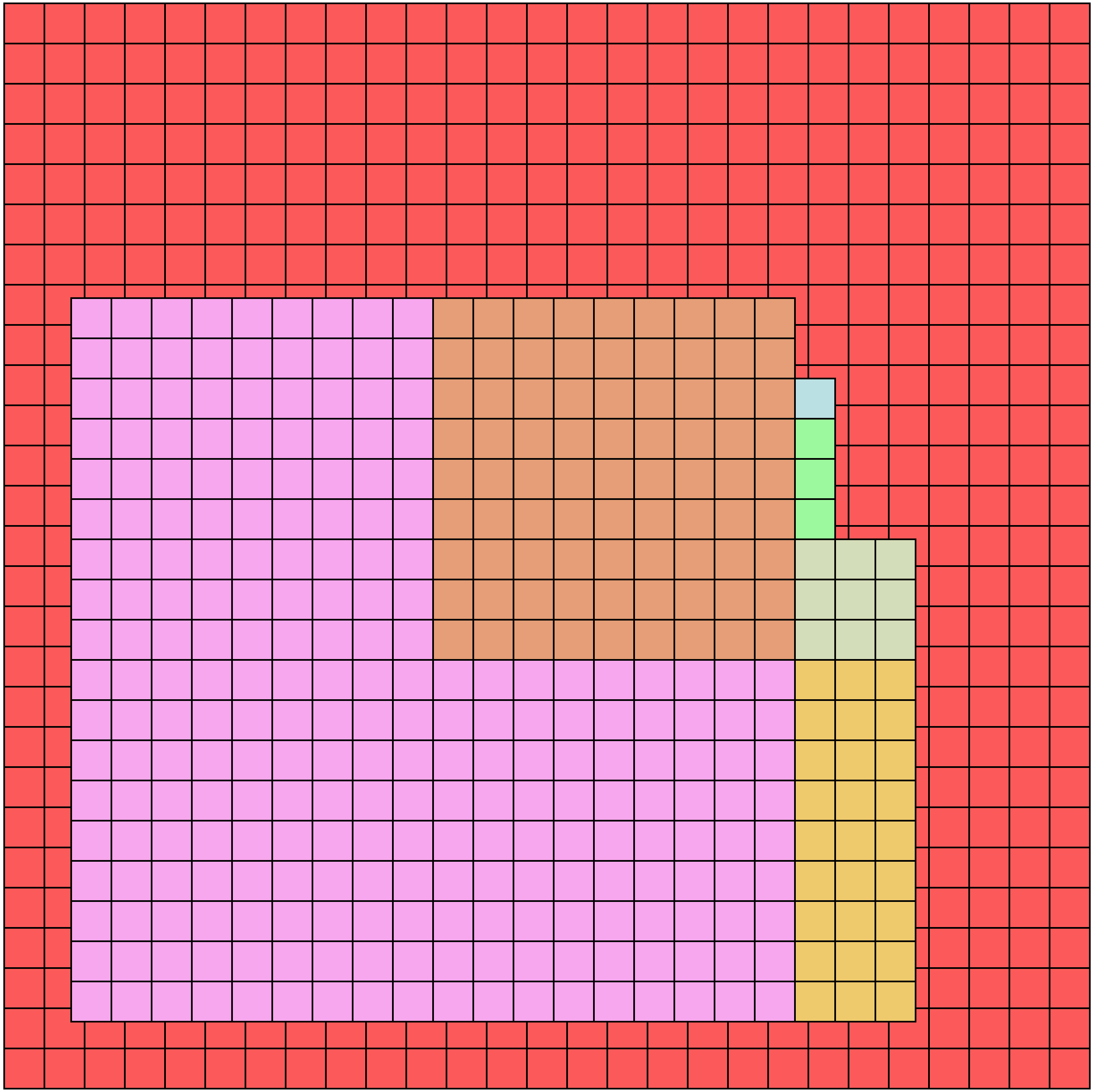


k	n-channel			p-channel	
1	1	WMIN		3	WMIN
2	2.5	WMIN		7.5	WMIN
3	6.25	WMIN		18.75	WMIN
4	15.6	WMIN		46.9	WMIN
5	39.1	WMIN		117.2	WMIN
6	97.7	WMIN		293.0	WMIN
7	244.1	WMIN		732.4	WMIN
8	610.4	WMIN		1831.1	WMIN

Note devices in last stage are very large !

Review from Last Time

Area of Last Stage Larger than that of all previous stages combined!



Digital Circuit Design

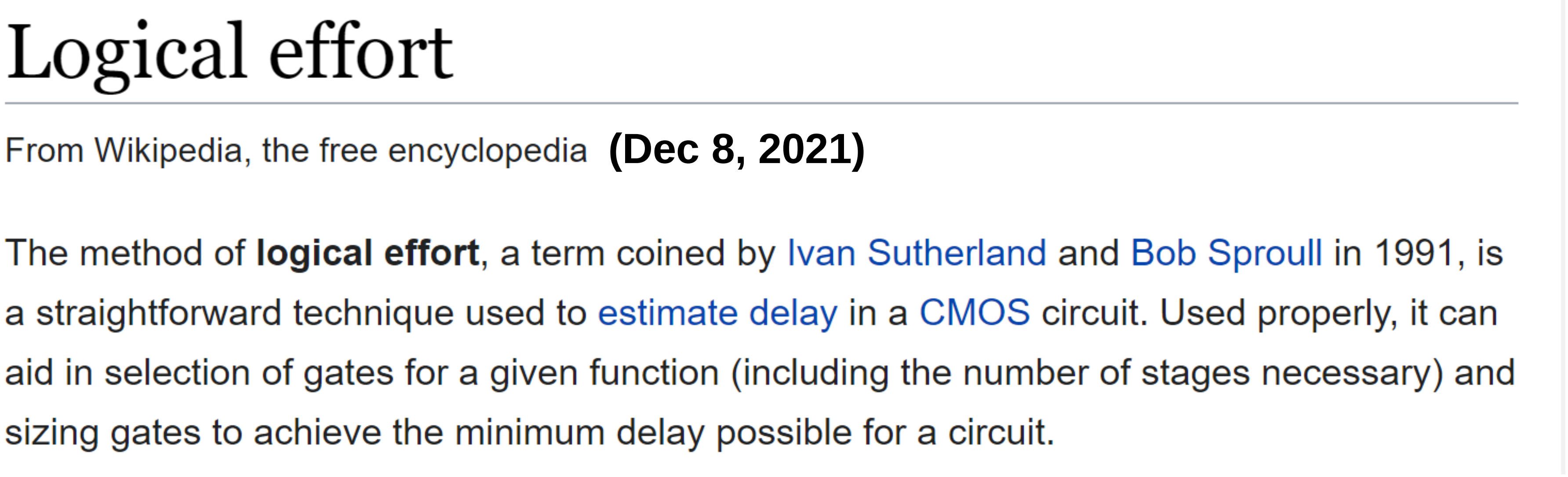
- Hierarchical Design
- Basic Logic Gates
- Properties of Logic Families
- Characterization of CMOS Inverter
- Static CMOS Logic Gates
  - Ratio Logic
- Propagation Delay
  - Simple analytical models
    - FI/OD
    - Logical Effort
  - Elmore Delay
- Sizing of Gates
  - The Reference Inverter
- Propagation Delay with Multiple Levels of Logic
- Optimal driving of Large Capacitive Loads
- Power Dissipation in Logic Circuits
- Other Logic Styles
- Array Logic
- Ring Oscillators

done

partial

Propagation Delay in “Logic Effort” approach

(Discussed in Chapter 4 of Text but definitions are not rigorous)



Logical effort

From Wikipedia, the free encyclopedia **(Dec 8, 2021)**

The method of **logical effort**, a term coined by Ivan Sutherland and Bob Sproull in 1991, is a straightforward technique used to estimate delay in a CMOS circuit. Used properly, it can aid in selection of gates for a given function (including the number of stages necessary) and sizing gates to achieve the minimum delay possible for a circuit.

Propagation Delay in “Logic Effort” approach

(Discussed in Chapter 4 of Text but definitions are not rigorous)

Propagation delay for equal rise/fall gates was derived to be

$$t_{PROP}=t_{REF}\sum_{k=1}^{n}\frac{F_{I(k+1)}}{OD_k}$$

Delay calculations with “logical effort” approach

Logical effort delay approach:

$$t_{PROP}=t_{REF}\sum_{k=1}^{n}f_k$$

(t_{REF} scaling factor not explicitly stated in W_H textbook. As defined in W_H, f_k is dimensionless)

where f_k is the “effort delay” of stage k

$$f_k=g_kh_k$$

g_k=logical effort

h_k=electrical effort

Propagation Delay in “Logic Effort” approach

$$t_{PROP} = t_{REF} \sum_{k=1}^{n} f_k \qquad f_k = g_k h_k$$

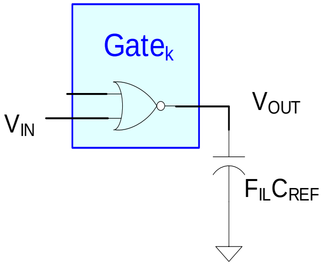
f_k = “effort delay” of stage k

g_k=logical effort

h_k=electrical effort

Logic Effort is the ratio of the input capacitance of a gate to the input capacitance of an inverter that can deliver the same output current

Electrical Effort is the ratio of the gate load capacitance to the input capacitance of a gate

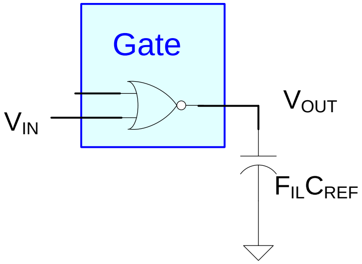


Propagation Delay in “Logic Effort” approach

$$\mathbf{t_{PROP}=t_{REF}\sum_{k=1}^{n} f_k} \qquad \mathbf{f_k=g_k h_k}$$

Logic Effort (g) is the ratio of the input capacitance of a gate to the input capacitance of an inverter that can deliver the same output current

Electrical Effort (h) is the ratio of the gate load capacitance to the input capacitance of a gate



$$g_k = \frac{C_{IN_k}}{C_{REF} \bullet OD_k} \qquad h_k = \frac{C_{REF} \bullet FI_{k+1}}{C_{IN_k}}$$

Propagation Delay in “Logic Effort” approach

$$\mathbf{t_{PROP}=t_{REF}\sum_{k=1}^{n} f_k} \qquad \mathbf{f_k=g_k h_k}$$

$$g_k=\frac{C_{IN_k}}{C_{REF}\bullet OD_k} \qquad h_k=\frac{C_{REF}\bullet F_{I(k+1)}}{C_{IN_k}}$$

$$f_k=\left(\frac{\cancel{C_{IN_k}}}{\cancel{C_{REF}}\bullet OD_k}\right)\left(\frac{\cancel{C_{REF}}\bullet F_{I(k+1)}}{\cancel{C_{IN_k}}}\right)$$

$$f_k=\frac{F_{I(k+1)}}{OD_k}$$

$$\mathbf{t_{PROP}=t_{REF}\sum_{k=1}^{n} f_k = t_{REF}\sum_{k=1}^{n} g_k h_k = t_{REF}\sum_{k=1}^{n}\frac{F_{I(k+1)}}{OD_k}}$$

Propagation Delay in “Logic Effort” approach

$$t_{PROP}=t_{REF}\sum_{k=1}^{n} f_k = t_{REF}\sum_{k=1}^{n} g_k h_k = t_{REF}\sum_{k=1}^{n}\frac{F_{I(k+1)}}{OD_k}$$

- **Note this expression is identical to what we have derived previously**

(t_{REF} scaling factor not included in W_H text)

- **Probably more tedious to use the “Logical Effort” approach**
- **Extensions to asymmetric overdrive factors may not be trivial**
- **Extensions to include parasitics may be tedious as well**
- **Logical Effort is widely used throughout the industry**

Digital Circuit Design

- Hierarchical Design
- Basic Logic Gates
- Properties of Logic Families
- Characterization of CMOS Inverter
- Static CMOS Logic Gates
  - Ratio Logic
- Propagation Delay
  - Simple analytical models
    - FI/OD
    - Logical Effort
  - Elmore Delay
- Sizing of Gates

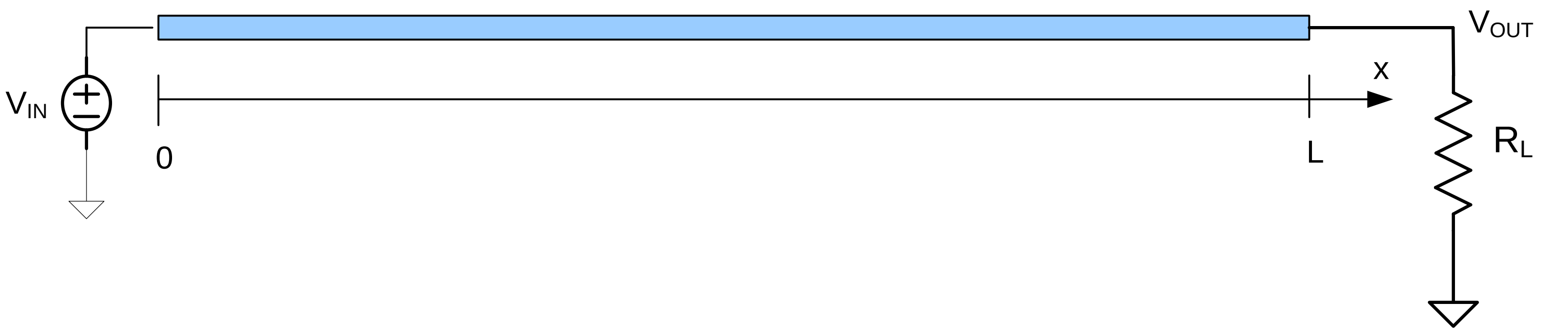
  - The Reference Inverter

- Propagation Delay with Multiple Levels of Logic
- Optimal driving of Large Capacitive Loads
- Power Dissipation in Logic Circuits
- Other Logic Styles
- Array Logic
- Ring Oscillators

done

partial

Elmore Delay Calculations



- **Interconnects have a distributed resistance and a distributed capacitance**

  - Often modeled as resistance/unit length and capacitance per unit length

- **These delay the propagation of the signal**

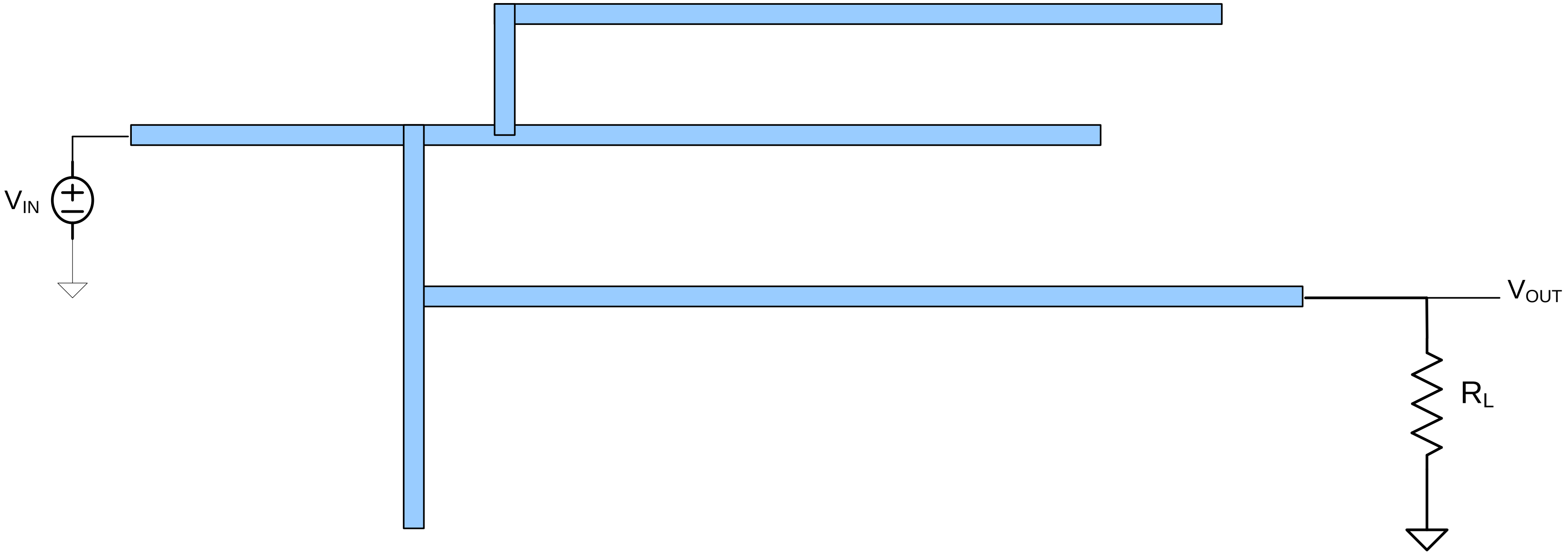
- **Effectively a transmission line**

  - analysis is really complicated

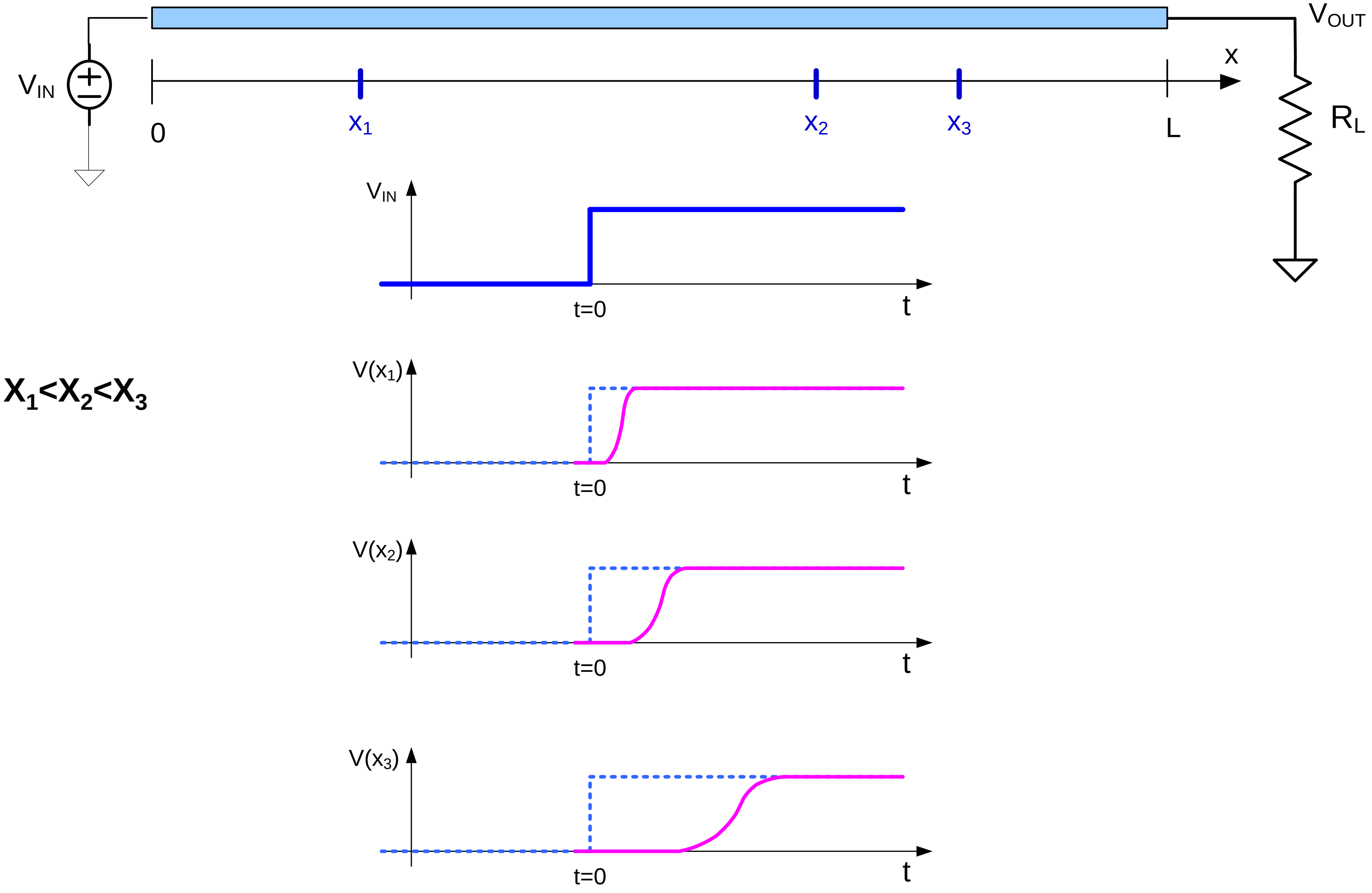
- **Can have much more complicated geometries**

Elmore Delay Calculations

Can have much more complicated geometries

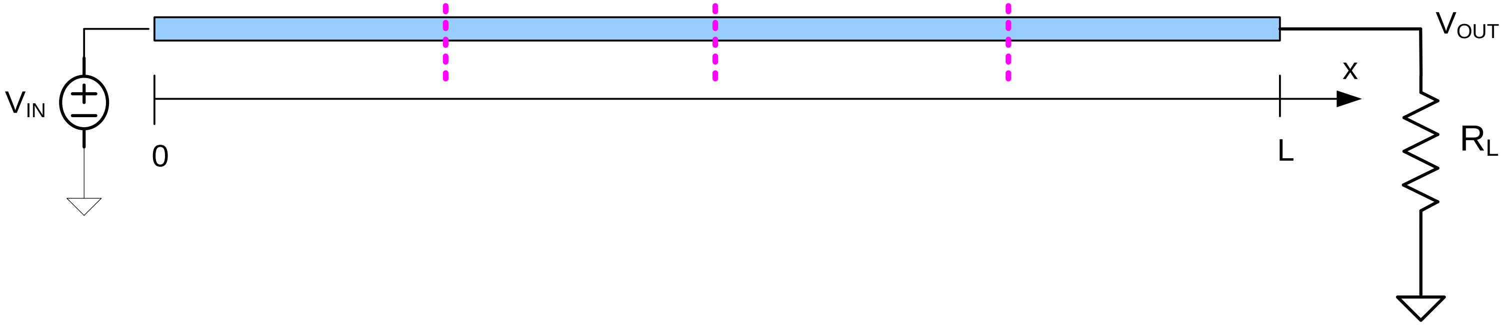


Elmore Delay Calculations

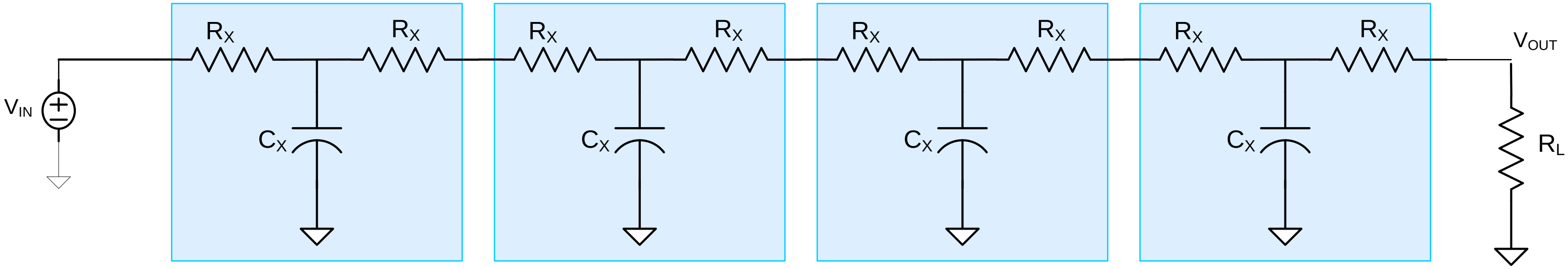


For **$X_1<X_2<X_3$**

Elmore Delay Calculations



A lumped element model of transmission line (with “T” elements)



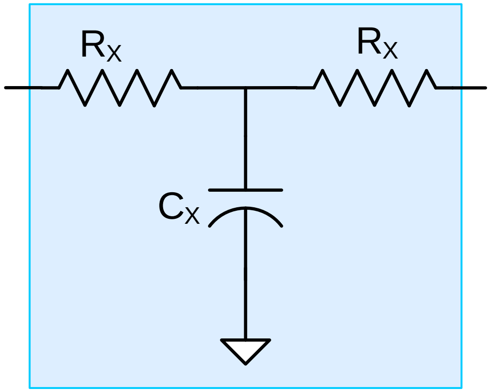
Even this lumped model is 4-th order and a closed-form solution is very tedious !

Can use “L” or other lumped segments as well (with small number some perform better than others)

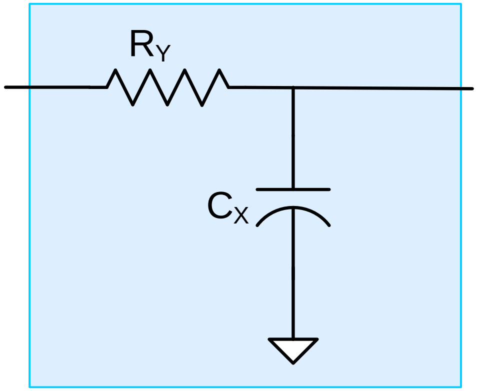
Need a quick (and reasonably good) approximation to the delay of a delay line !!

Did anyone actually analyze a circuit like this in EE 201?

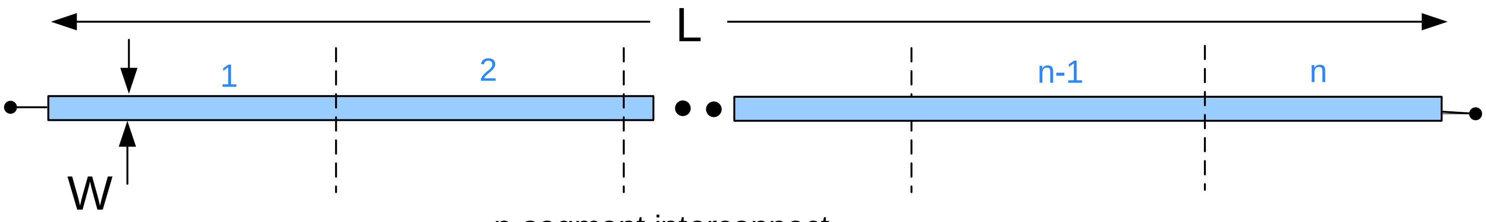
Elmore Delay Calculations



T-Model



L-Model



n-segment interconnect

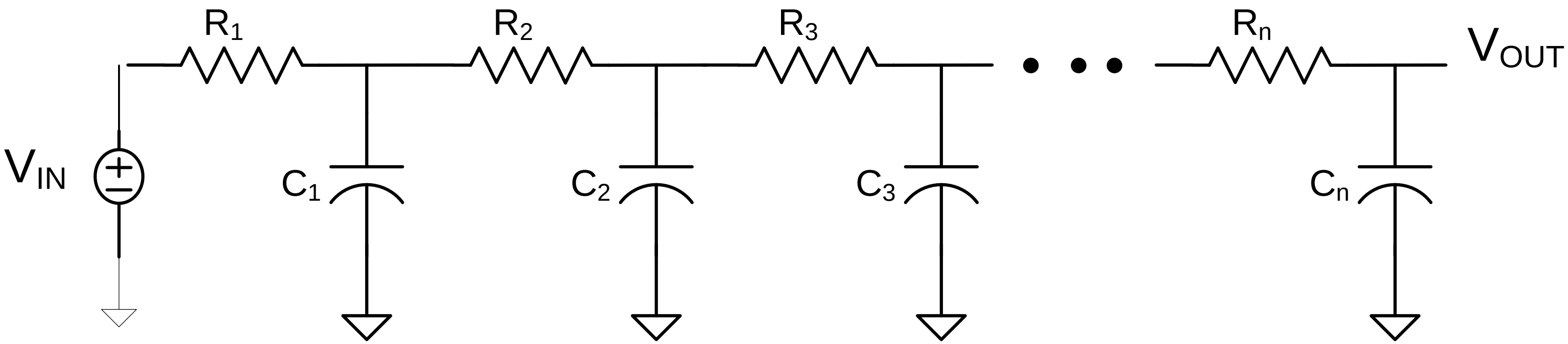
$$R_X = \frac{R_{\square}\frac{L}{W}}{2n}$$

$$C_X = \frac{C_{Density}WL}{n}$$

$$R_Y = \frac{R_{\square}\frac{L}{W}}{n}$$

$$C_X = \frac{C_{Density}WL}{n}$$

Elmore Delay Calculations



Elmore delay: $$t_{PD} = \sum_{i=1}^{n} \left(C_i \sum_{j=1}^{i} R_j \right)$$

- **It can be shown that this is a reasonably good approximation to the actual delay**
  - **- provided sufficient number of stages are used**
  - **- number does not need to be very large**
- **Numbering is critical (resistors and capacitors numbered from input to output)**
- **As stated, only applies to this specific structure**

Elmore Delay Calculations

Elmore delay:

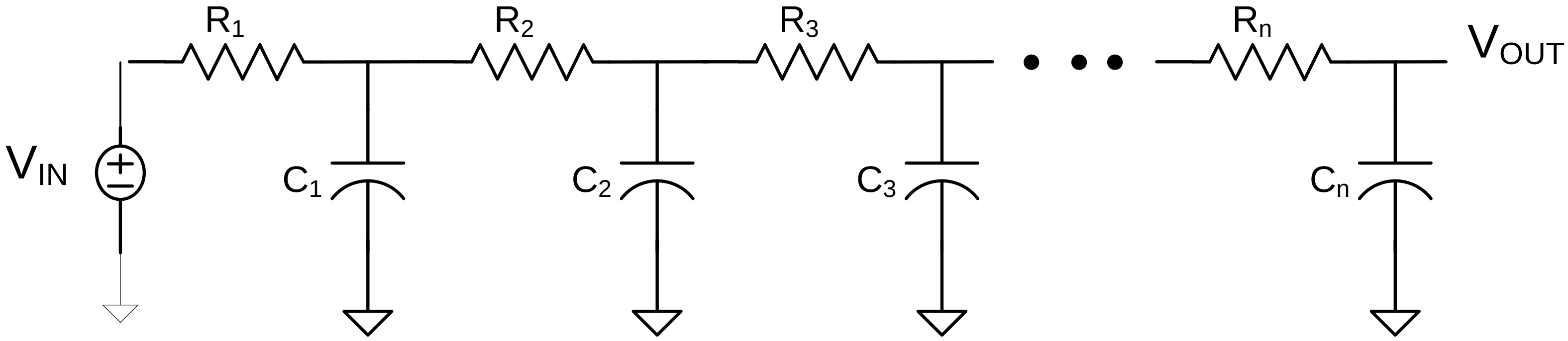
$$t_{PD}=\sum_{i=1}^{n}\left(C_i\sum_{j=1}^{i}R_j\right)$$

- **Note error in text on Page 161 of first edition of WH**

$$\xcancel{t_{pd}=\sum_{i}R_{n-i}C_i=\sum_{i=1}^{N}C_i\sum_{j=i}^{i}R_j}$$

- **Not detailed definition on Page 150 of second edition of WH**

Elmore Delay Calculations



$$t_{PD} = \sum_{i=1}^{n} \left(C_i \sum_{j=1}^{i} R_j \right)$$

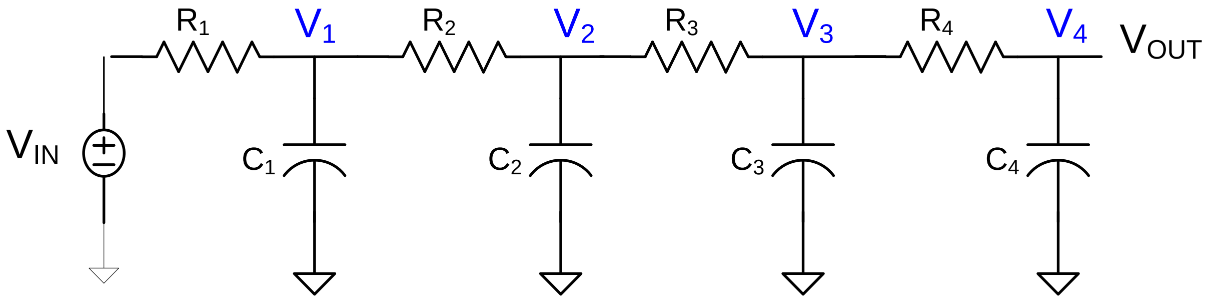
From Wikipedia (Dec 8 2021):

Elmore delay[1] is a simple approximation to the delay through an RC network in an electronic system. It is often used in applications such as logic synthesis, delay calculation, static timing analysis, placement and routing, since it is simple to compute (especially in tree structured networks, which are the vast majority of signal nets within ICs) and is reasonably accurate. Even where it is not accurate, it is usually faithful, in the sense that reducing the Elmore delay will almost always reduce the true delay, so it is still useful in optimization.

[1] W.C. Elmore. *The Transient Analysis of Damped Linear Networks with Particular Regard to Wideband Amplifiers*. J. Applied Physics, vol. 19(1), 1948.

Elmore Delay Calculations

Example:

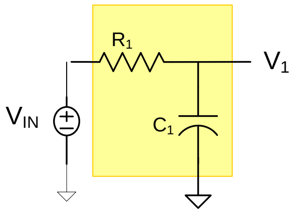


Elmore delay:

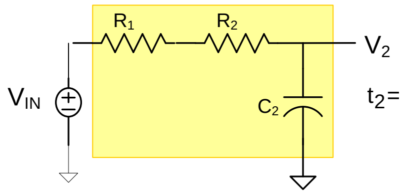
$$t_{PD}=\sum_{i=1}^{4}\left(C_i\sum_{j=1}^{i}R_j\right)$$

$$t_{PD}=\sum_{i=1}^{4}(t_i)$$

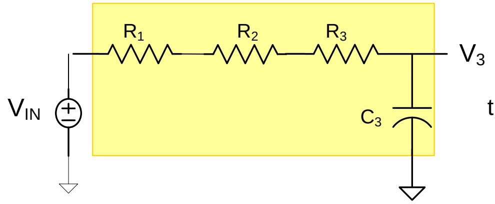
where $t_i=C_i\sum_{j=1}^{i}R_j \qquad j=1,2,3,4$



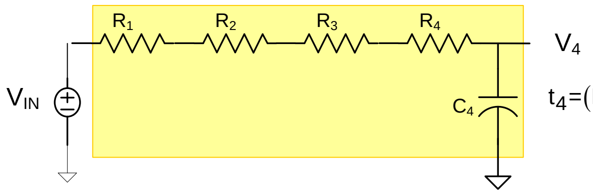
$t_1=R_1C_1$



$t_2=(R_1+R_2)C_2$



$t_3=(R_1+R_2+R_3)C_3$



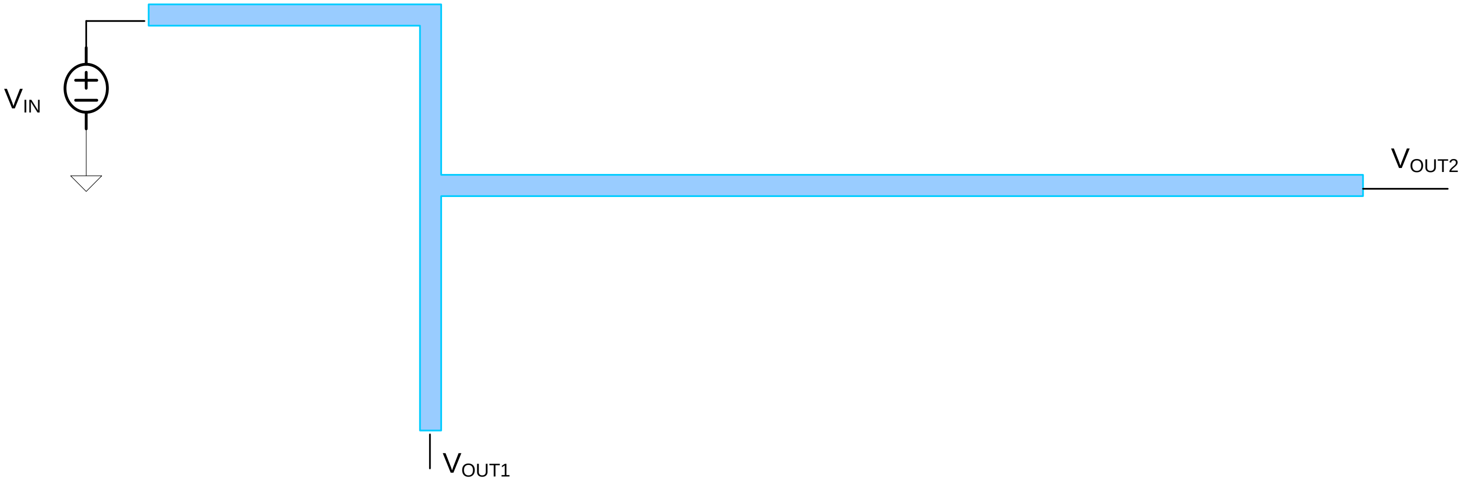
$t_4=(R_1+R_2+R_3+R_4)C_4$

What is really happening?

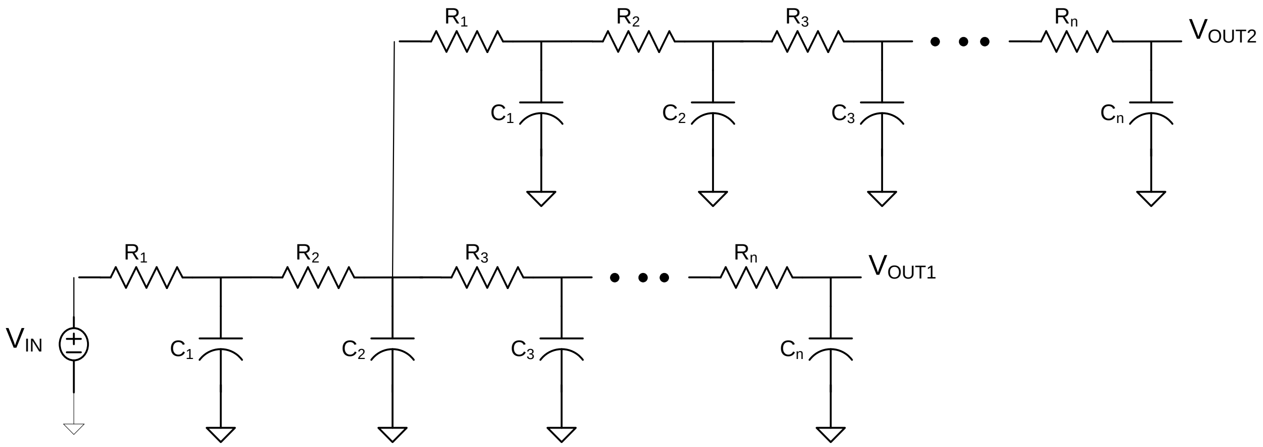
- **Creating 4 first-order circuits**
- **Delay to V_1, V_2, V_3 and V_4 calculated separately by considering capacitors one at a time and assuming others are 0**

Elmore Delay Calculations

Extensions:



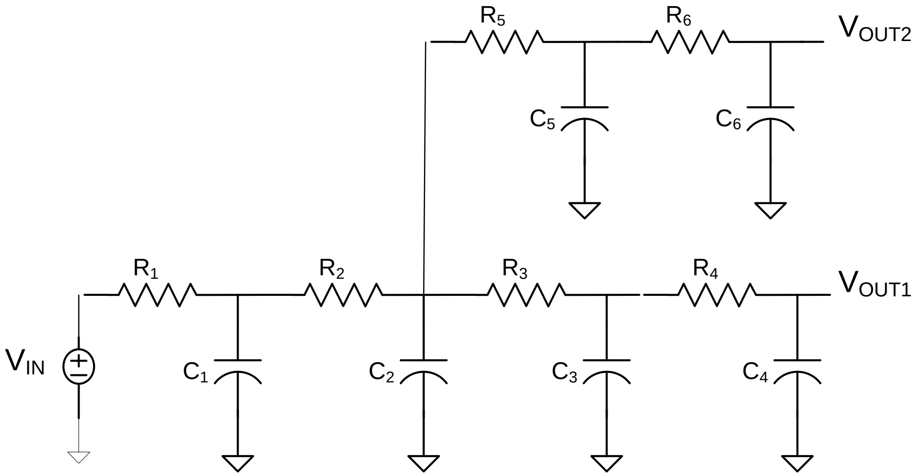
Lumped Network Model:



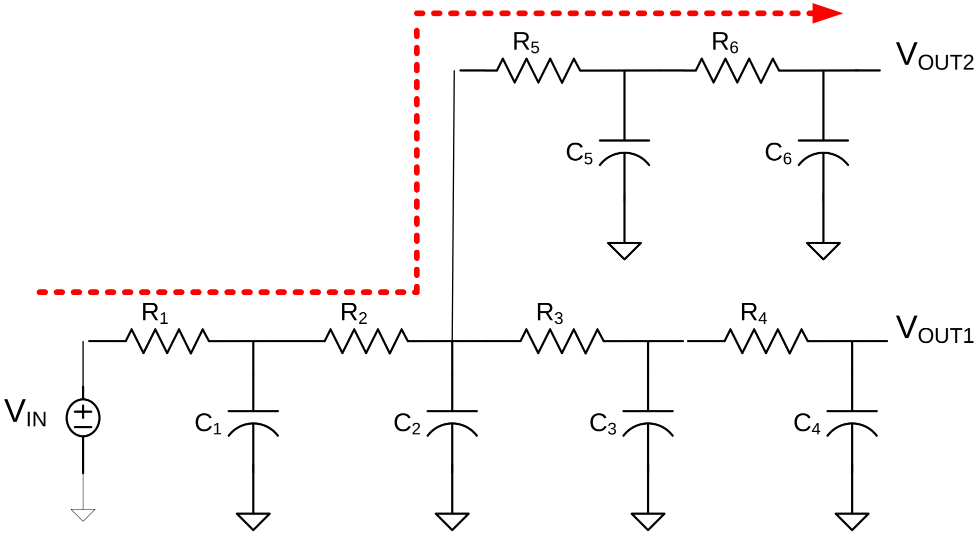
Elmore Delay Calculations

Extensions:

1. **Create a lumped element model**



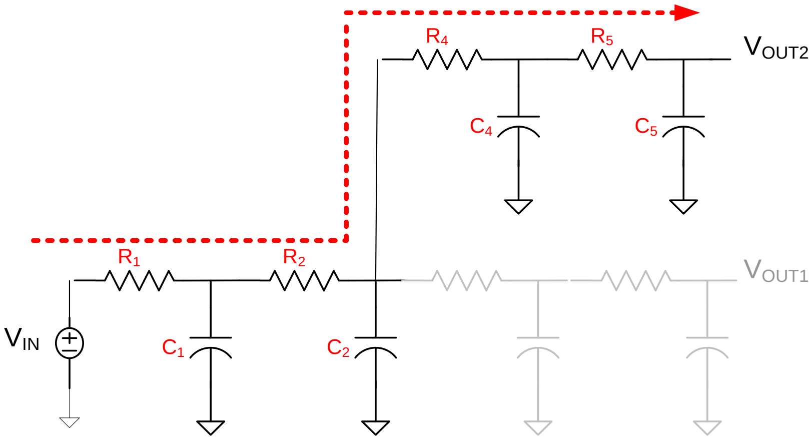
2. **Identify te a path from input to output**



Elmore Delay Calculations

Extensions:

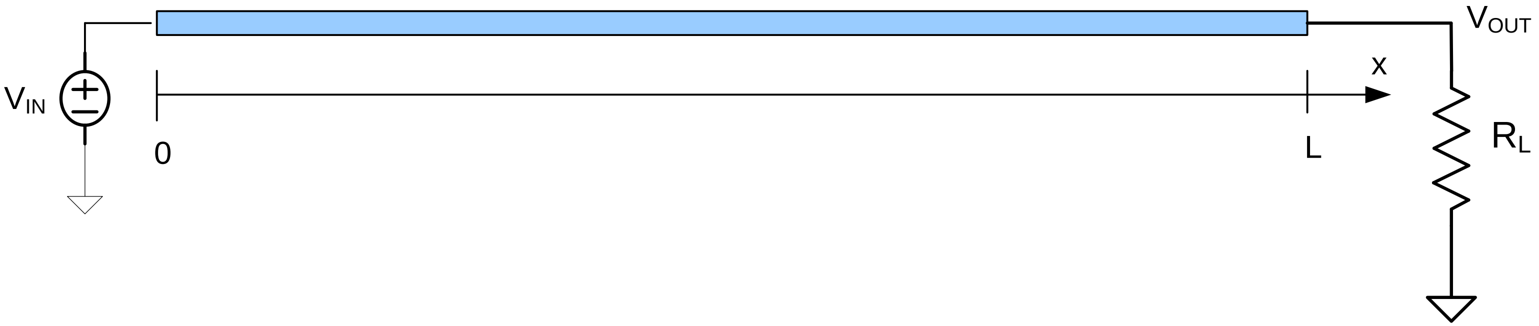
3. Renumber elements along path from input to output and neglect off-path elen



4. Use Elmore Delay equation for elements on this RC network

$$t_{PD}=\sum_{i=1}^{4}\left(C_i\sum_{j=1}^{i}R_j\right)$$

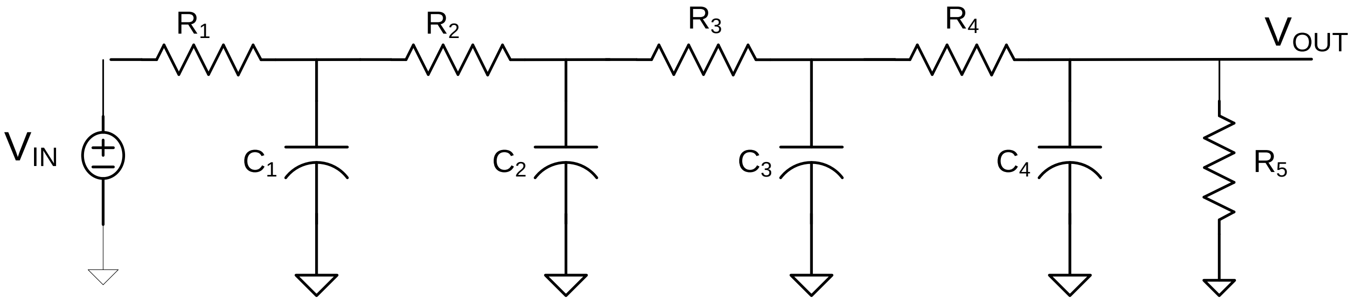
Elmore Delay Calculations



How is a resistive load handled?

Elmore Delay Calculations

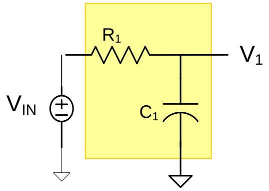
Example with resistive load:



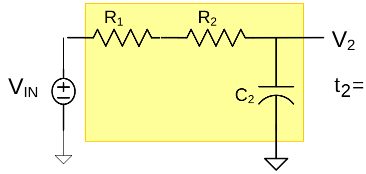
Elmore delay:

$$t_{PD}=\sum_{i=1}^{4}\left(C_i\sum_{j=1}^{i}R_j\right)$$

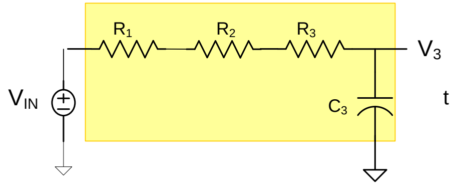
where



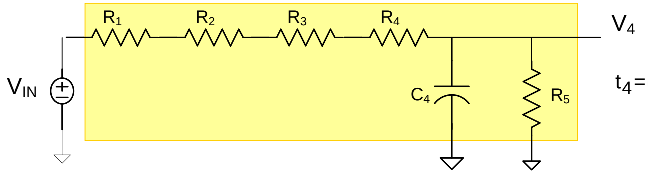
$t_1=R_1C_1$



$t_2=(R_1+R_2)C_2$



$t_3=(R_1+R_2+R_3)C_3$



$t_4=((R_1+R_2+R_3+R_4)//R_5)C_4$

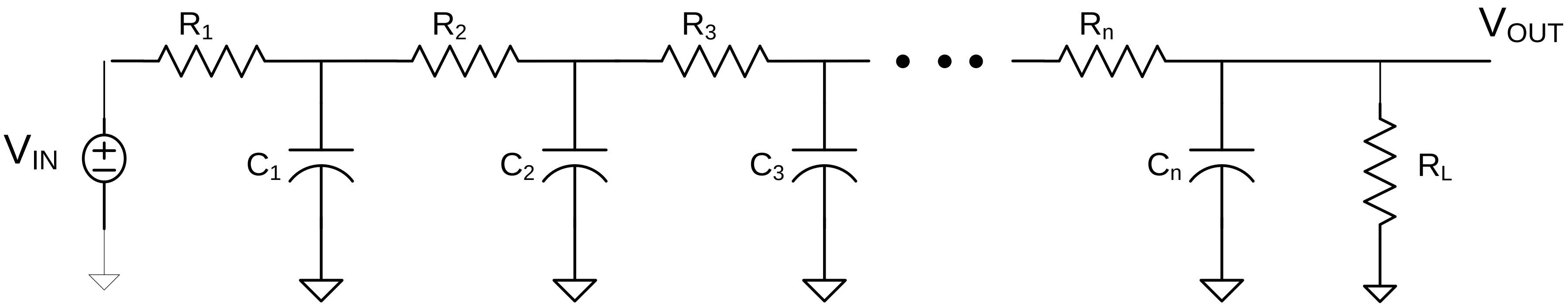
$$t_{PD}=\sum_{i=1}^{4}(t_i)$$

$$t_i=C_i\sum_{j=1}^{i}R_j \qquad j=1,2,3$$

$$t_4=C_4\left(\left[\sum_{j=1}^{4}R_j\right]//R_5\right)$$

Elmore Delay Calculations

With resistive load:

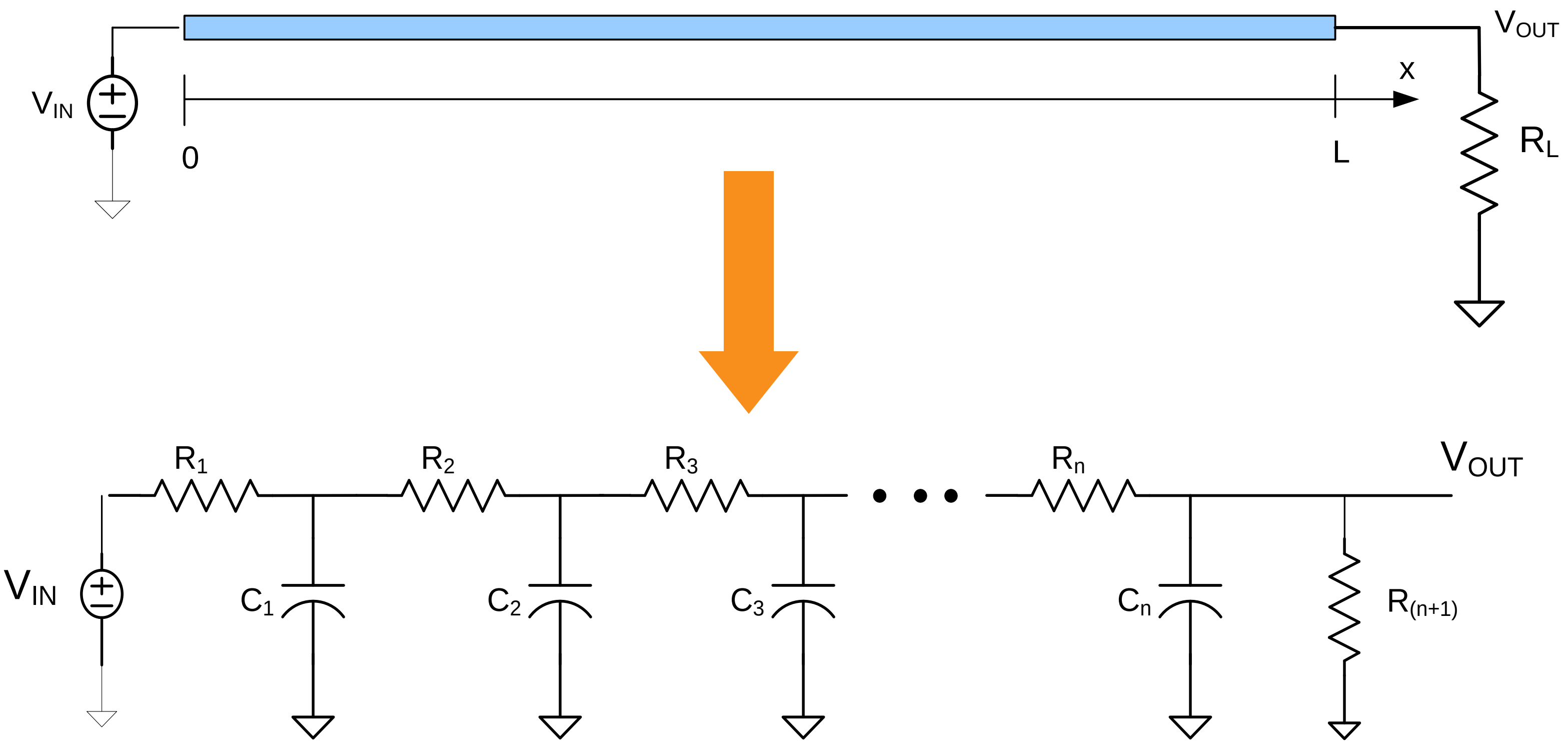


Simple Elmore delay:

$$t_{PD}=\sum_{i=1}^{n-1}\left(C_i\sum_{j=1}^{i}R_j\right)+C_n\left(\left(\sum_{j=1}^{n}R_j\right)//R_L\right)$$

Actually, R_L affects all of the delays and a modestly better but modestly more complicated delay model is often used

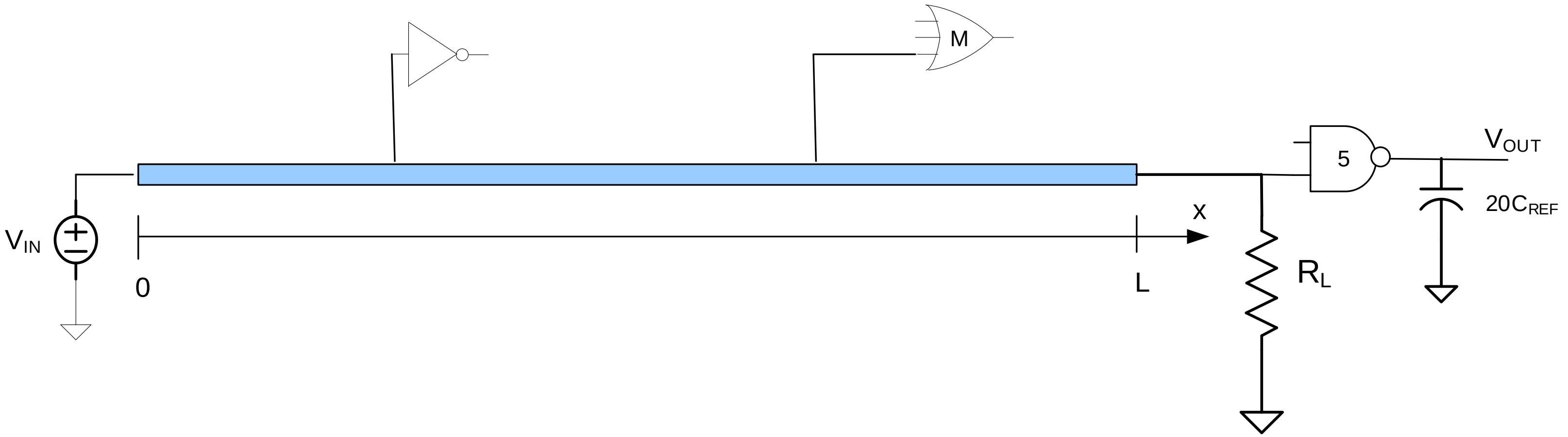
Elmore Delay Calculations



How are the number of stages chosen?

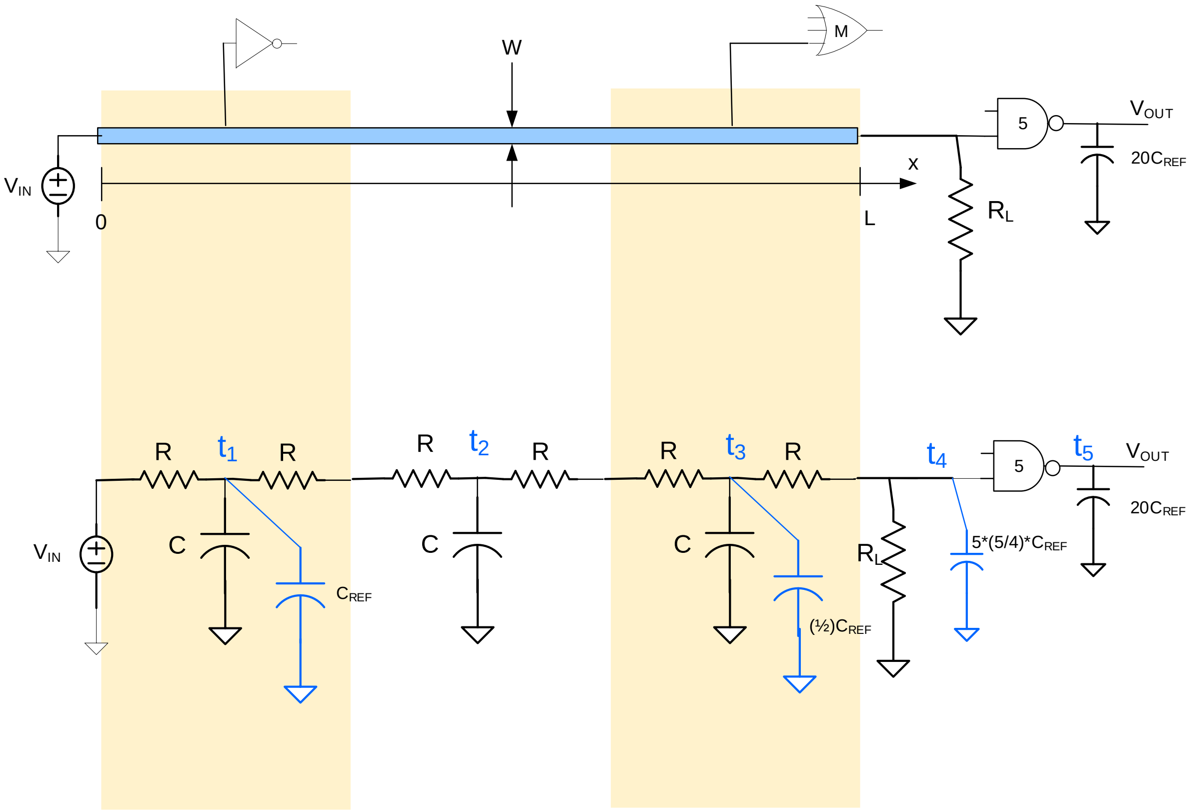
- **For hand analysis, keep number of stages small (maybe 3 or 4 for simple delay line) if possible)**
- **If “faithfulness” is important, should keep the number of stages per unit length constant**

Elmore Delay Calculations



?

Elmore Delay Calculations



$$R = \frac{1}{6} R_{\square} \frac{L}{W}$$

$$C = \frac{1}{3} C_D$$

$$t_1 = R\left(C + C_{REF}\right)$$

$$t_2 = 3RC$$

$$t_3 = 5R\left(C + \frac{1}{2} C_{REF}\right)$$

$$t_4 = [6R\ ///\ R_L]\frac{25}{4} C_{REF}$$

$$t_5 = t_{REF} \frac{20}{5}$$

$$t_{PROP} = \sum_{i=1}^{5} t_i$$

Digital Circuit Design

- Hierarchical Design
- Basic Logic Gates
- Properties of Logic Families
- Characterization of CMOS Inverter
- Static CMOS Logic Gates
  - Ratio Logic
- Propagation Delay
  - Simple analytical models
    - FI/OD
    - Logical Effort
  - Elmore Delay
- Sizing of Gates

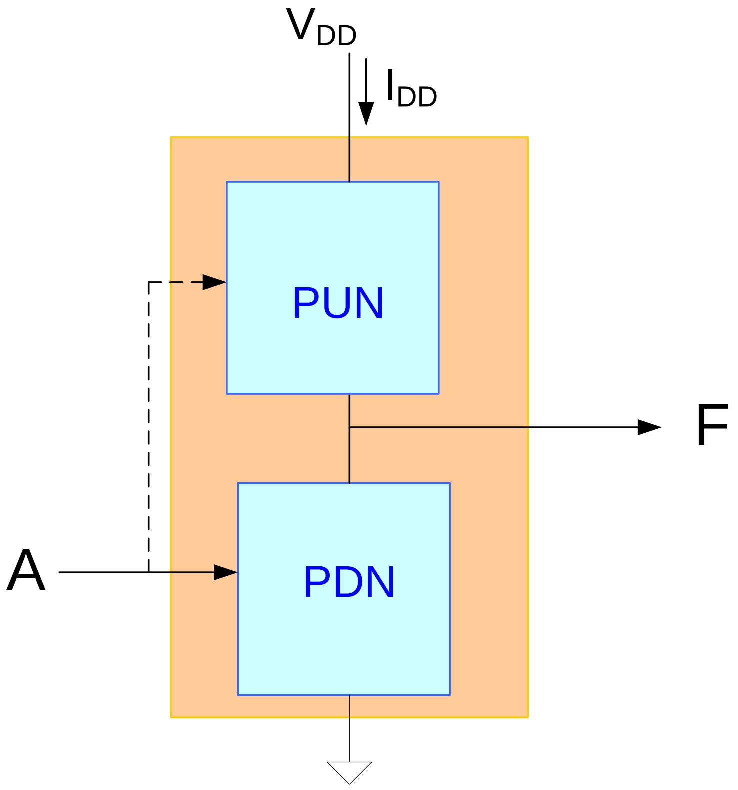
  - The Reference Inverter

- Propagation Delay with Multiple Levels of Logic
- Optimal driving of Large Capacitive Loads
- Power Dissipation in Logic Circuits
- Other Logic Styles
- Array Logic
- Ring Oscillators

done

partial

Power Dissipation in Logic Circuits



Assume current periodic with period T_{CL}

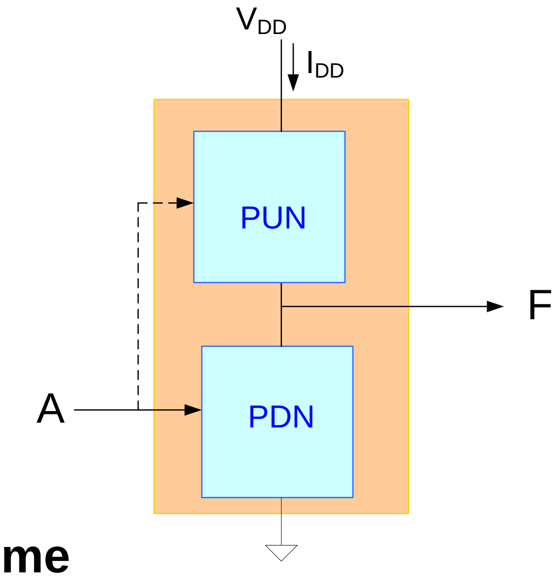
$$P_{AVG,T} = \frac{1}{T_{CL}} \int_{t_1}^{t_1+T_{CL}} V_{DD} I_{DD}(t)\,dt$$

Power Dissipation in Logic Circuits

Types of Power Dissipation

- Static
- Pipe
- Dynamic
- Leakage
    - Gate
    - Diffusion
    - Drain

Static Power Dissipation



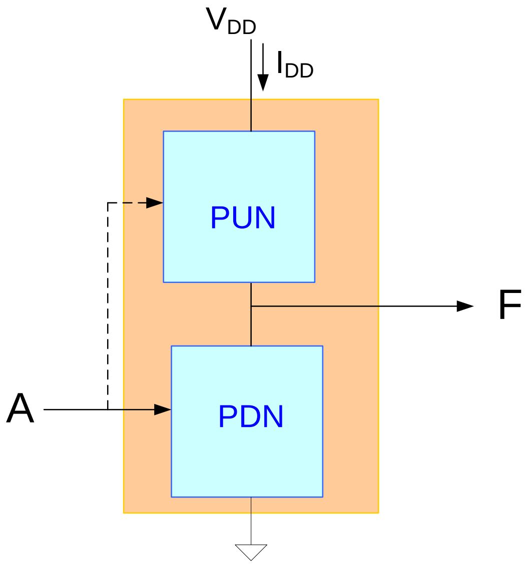
If Boolean output averages H and L 50% of the time

$$P_{STAT,AVG} = \frac{P_H + P_L}{2}$$

$$P_{STAT,AVG} = \frac{V_{DD}(I_{DDH} + I_{DDL})}{2}$$

- **Generally decreases with V_{DD}**
- **I_{DDH}=I_{DDL}=0 for static CMOS gates so P_{STAT}=0**
- **A major source of power dissipation in ratio logic circuits and the major reason CMOS is so widely used**

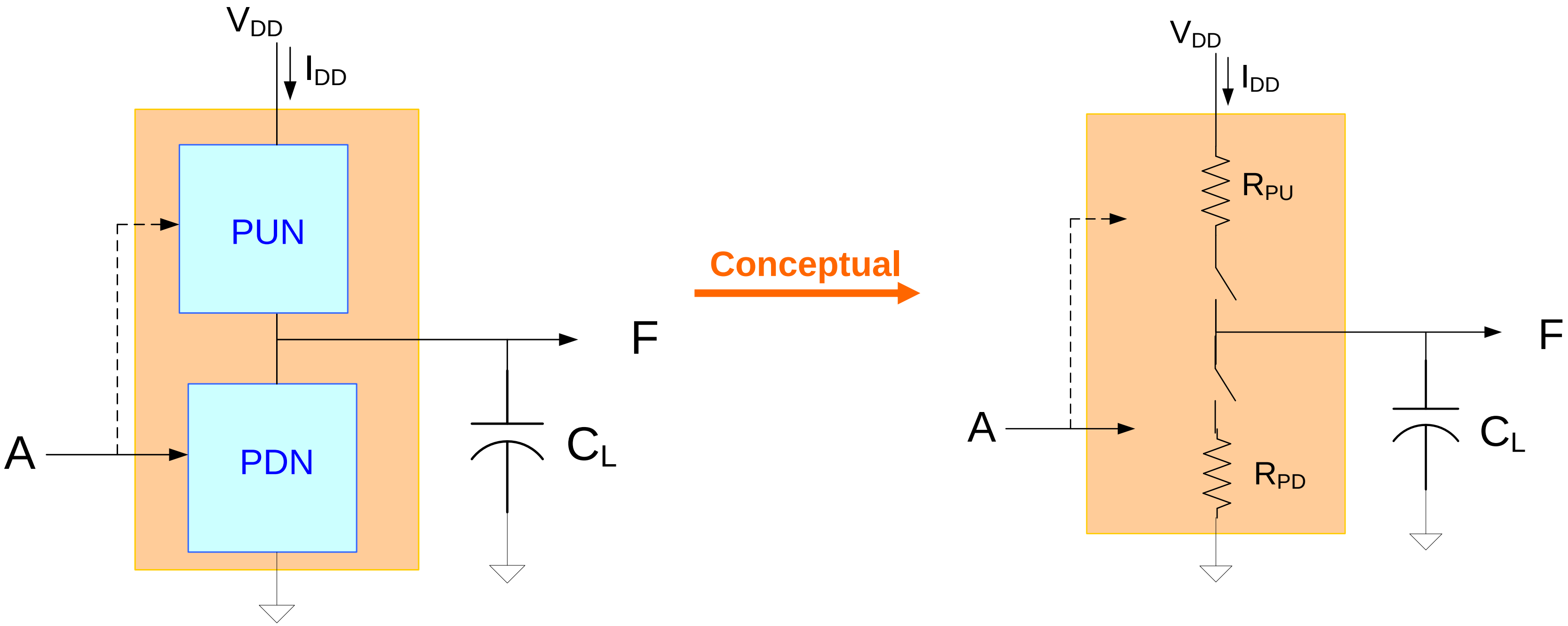
Pipe Power Dissipation



Due to conduction of both PUN and PDN during transitions

- **Can be made small if transitions are fast**
- **Usually negligible in Static CMOS circuits**

Dynamic Power Dissipation



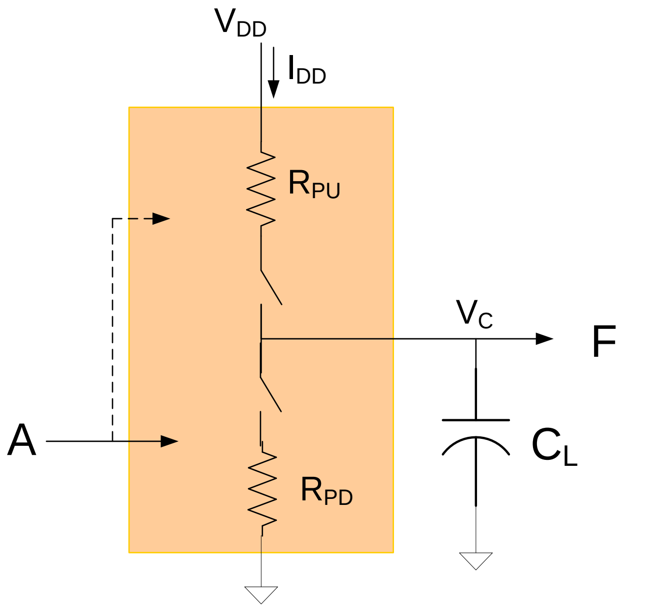
Due to charging and discharging C_L on logic transitions

C_L dissipates no power but PUN and PDN dissipate power during charge and discharge of C_L

C_L includes all gate input capacitances of loads and interconnect capacita

Dynamic Power Dissipation

Energy supplied by V_{DD} when C_L charges



$$\left.\begin{array}{l} E=\int_{t_1}^{\infty} V_{DD}I_{DD}(t)dt \\ I_{DD}=C_L\dfrac{dV_C}{dt} \end{array}\right\}$$

$$E=\int_{t_1}^{\infty} V_{DD}C_L\frac{dV_C}{dt}dt$$

$$E=\int_{V_C=0}^{V_{DD}} V_{DD}C_L dV_C = V_{DD}C_L\int_{V_C=0}^{V_{DD}} dV_C = V_{DD}C_L V_C\Big|_{V_C=0}^{V_{DD}} = V_{DD}^2C_L$$

Energy stored in C_L after C_L is charged to V_{DD}:

$$E=\frac{1}{2}C_L V_{DD}^2$$

Dynamic Power Dissipation

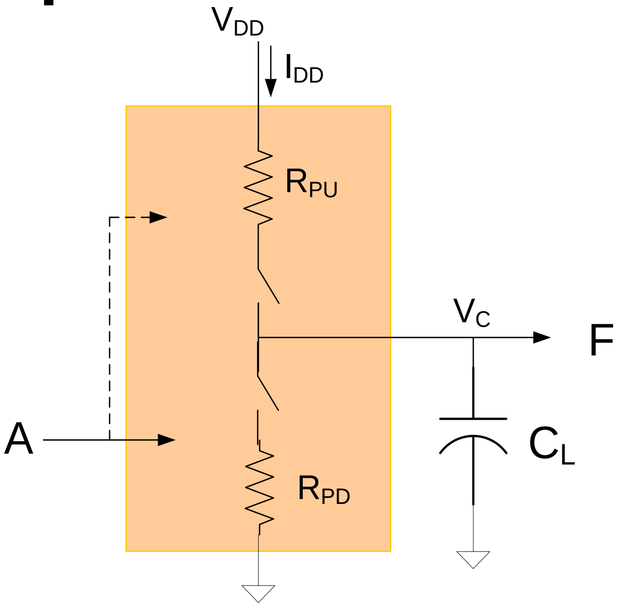
Energy supplied by V_{DD} and dissipated in R_{PU} when C_L charges

$$E_{DIS} = \frac{1}{2} C_L V_{DD}^2$$

Energy stored on C_L after L-H transition

$$E_{STORE} = \frac{1}{2} C_L V_{DD}^2$$

$$E = E_{DIS} + E_{STORE} = C_L V_{DD}^2$$



When the output transitions from H to L, energy stored on C_L is dissipated in PDN

Thus, energy from V_{DD} for one L-H: H-L output transition sequence is

$$E = C_L V_{DD}^2$$

If f is the average transition rate of the output, determine P_{AVG}

Dynamic Power Dissipation

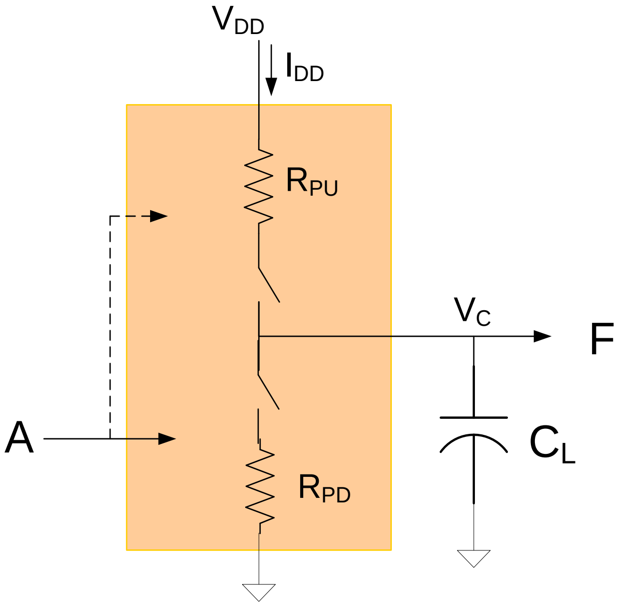
Energy from V_{DD} for one L-H: H-L output transition sequence is

$$E = C_L V_{DD}^2$$

If f is the average transition rate of the output, determine P_{AVG}

$$P_{AVG} = \frac{E}{T} = Ef$$

$$P_{DYN} = f C_L V_{DD}^2$$



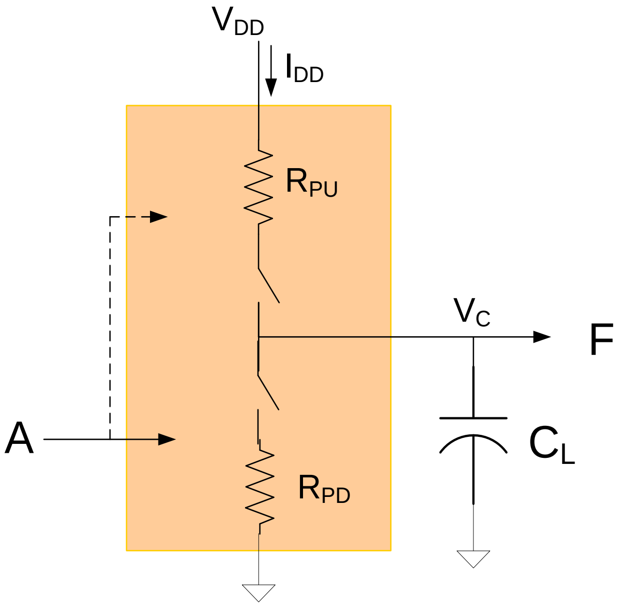
If a gate has a transition duty cycle of 50% with a clock frequency of f_{CL}

$$P_{DYN} = \frac{f_{CL}}{2} C_L V_{DD}^2$$

Note dependent on the square of V_{DD} ! …. Want to make VDD small !!!

Major source of power dissipation in many static CMOS circuits for L_{min}>0.1u

Dynamic Power Dissipation



Energy dissipated with clock signal itself

$$P_{DYN} = f_{CL} C_L V_{DD}^2$$

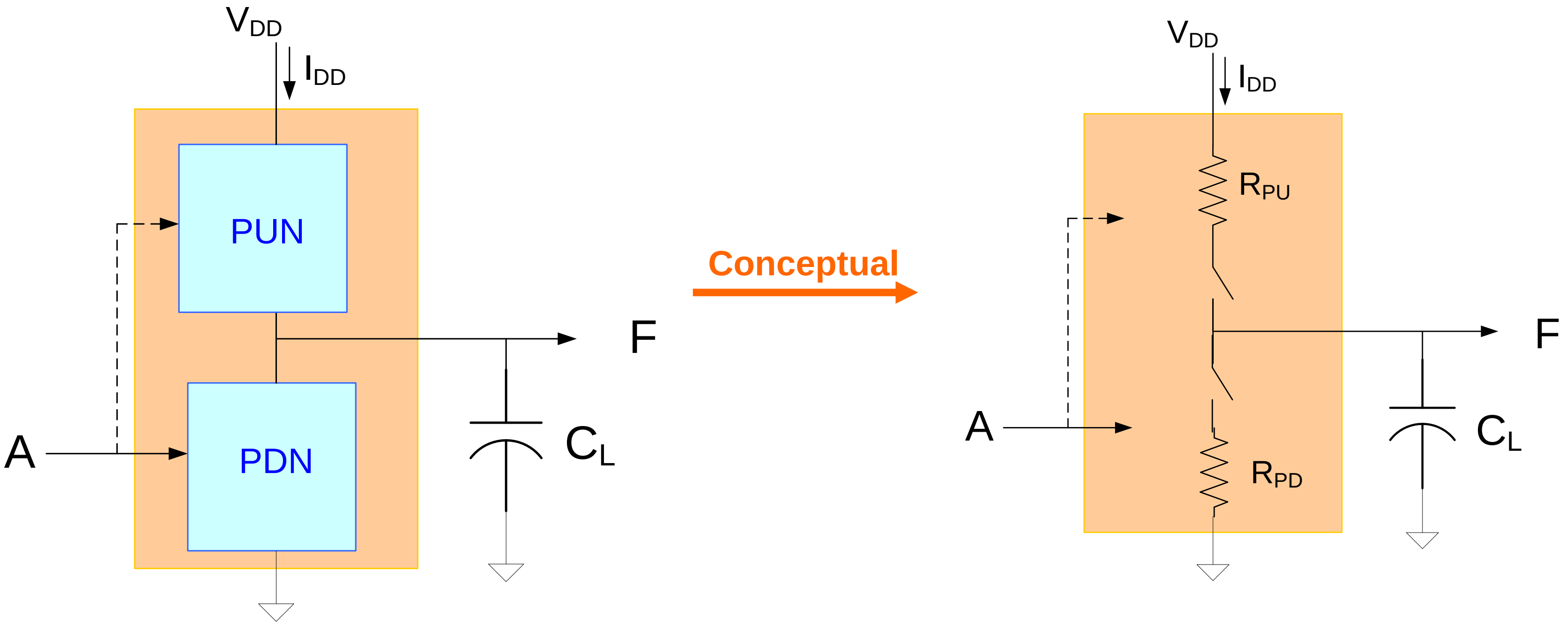
The clock transitions on every clock cycle (i.e. it has a transition duty cycle of 100%)

Clock distribution can cause significant power dissipation

But if a gate has a transition duty cycle of 50% with a clock frequency of f_{CL}

$$P_{DYN} = \frac{f_{CL}}{2} C_L V_{DD}^2$$

Power Dissipation



- **All power is dissipated in pull-up and pull-down devices**

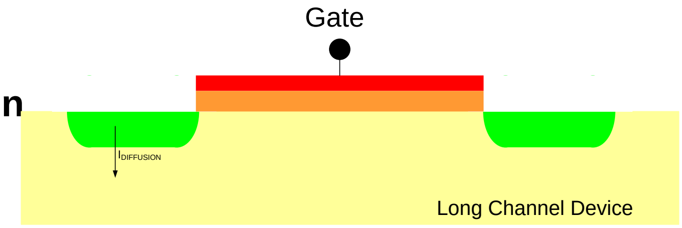
- **C_L dissipates no power but PUN and PDN dissipate power when charging and discharging C_L**

- **Dynamic power dissipation reduced by more (often much more) than a factor of 2 if minimum sizing strategy is used**

Leakage Power Dissipation

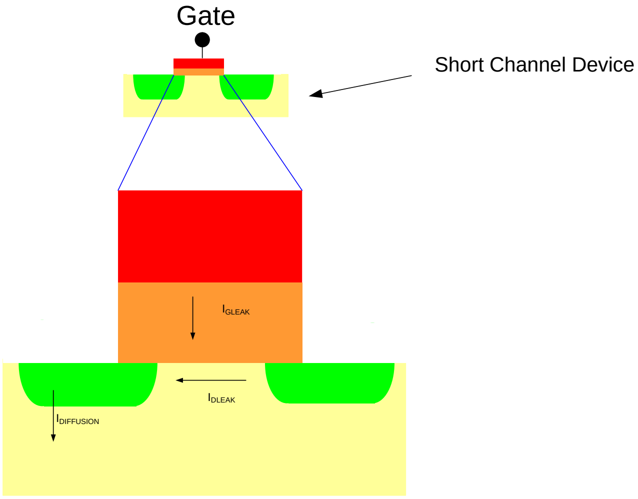
- Gate

- with very thin gate oxides, some gate leakage current flows
- major concern in 60nm and smaller processes
- actually a type of static power dissipation



-Diffusion

- Leakage across a reverse-biased pn junction
- Dependent upon total diffusion area
- May actually be dominant power loss on longer-channel devices
- Actually a type of static power dissipation

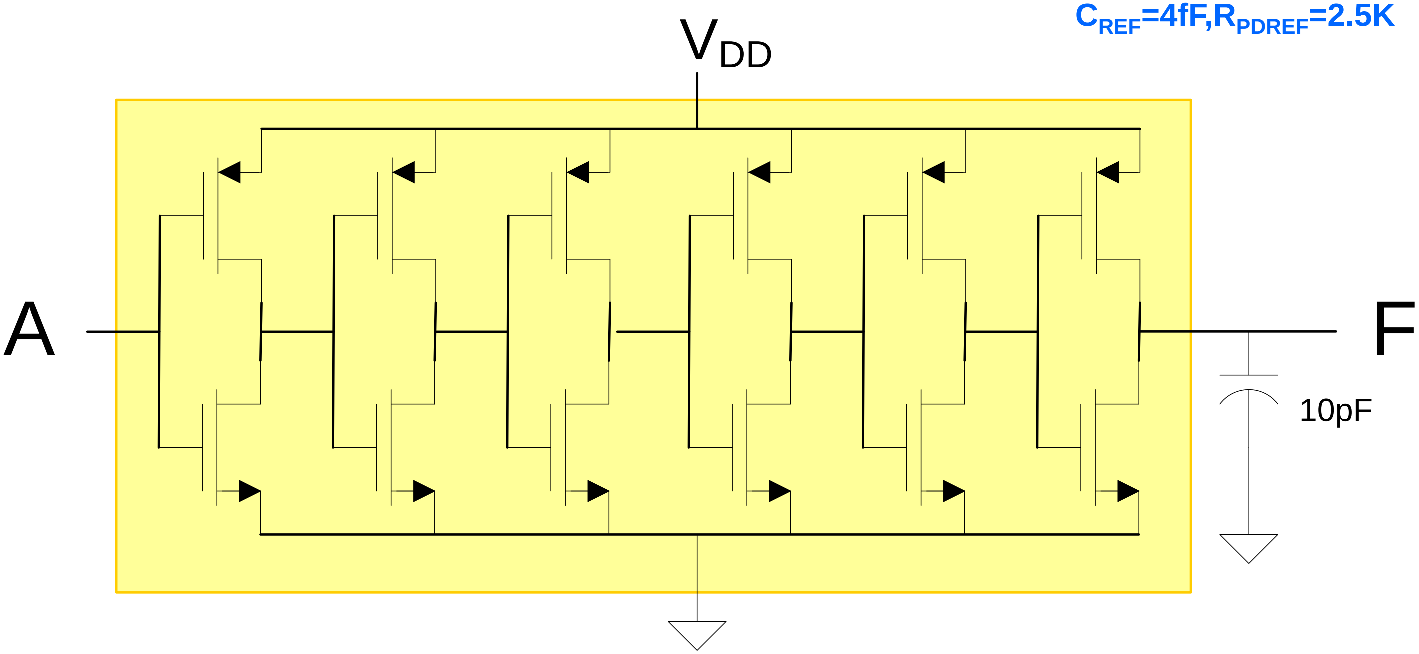


-Drain

- channel current due to small V_{GS}-V_T
- of significant concern only with low V_{DD} processes
- actually a type of static power dissipation

Example: Determine the dynamic power dissipation in the last stage of a 6-stage CMOS pad driver if used to drive a 10pF capacitive load if the system clock is 500MHz and the output changes with 50% of the clock transitions. Assume pad driver with OD of θ=2.5 and V_{DD}=3.5V

In 0.5u proc t_{REF}=20ps, C_{REF}=4fF,R_{PDREF}=2.5K

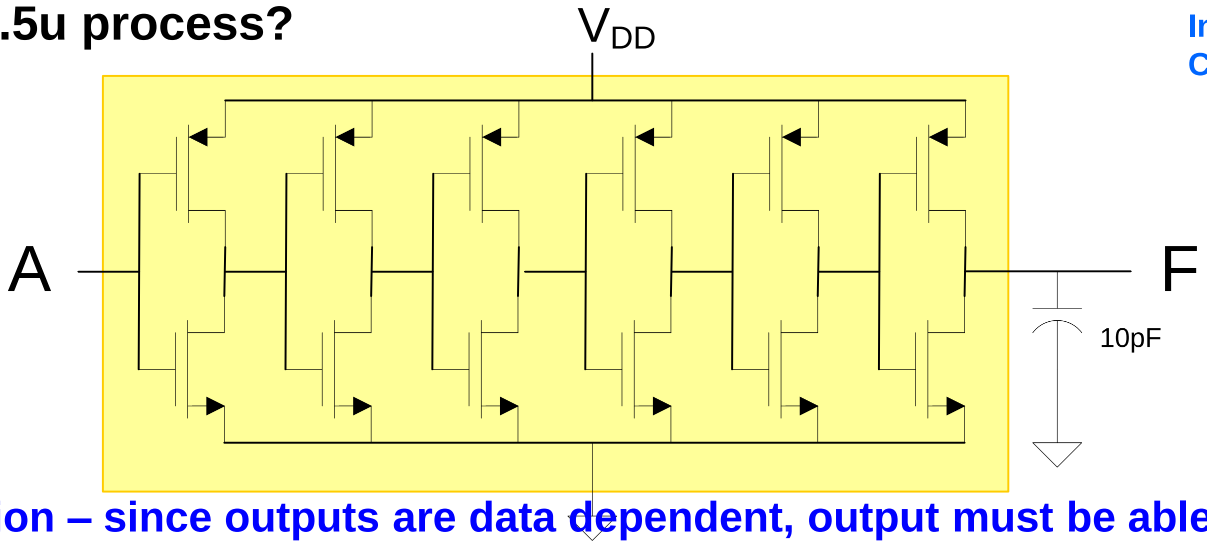


Solution: (assume output changes with 50% of clock transitions)

$$P_{DYN}=\frac{f_{CL}}{2}C_L V_{DD}^2 = \frac{5E8}{2}\bullet 10pF\bullet 3.5^2 = 30.5\ mW$$

Note this solution is independent of the OD and the process

Example: Will the CMOS pad driver actually be able to drive the 10pF load with a system clock of 500MHz as in the previous example in the 0.5u process?



In 0.5u proc t_{REF}=20ps, C_{REF}=4fF,R_{PDREF}=2.5K

Solution – since outputs are data dependent, output must be able to operate 500Mz:

$$t_{CLK} = \frac{1}{500\text{MHz}} = 2\text{nsec}$$

$$\text{FI}_{load} = \frac{10\,pF}{4\,fF} = 2500 \qquad OD_6 = \theta^5 = 98$$

$$t_{PROP} = 5\theta \bullet t_{REF} + \frac{\text{FI}_{load}}{OD_6} t_{REF}$$

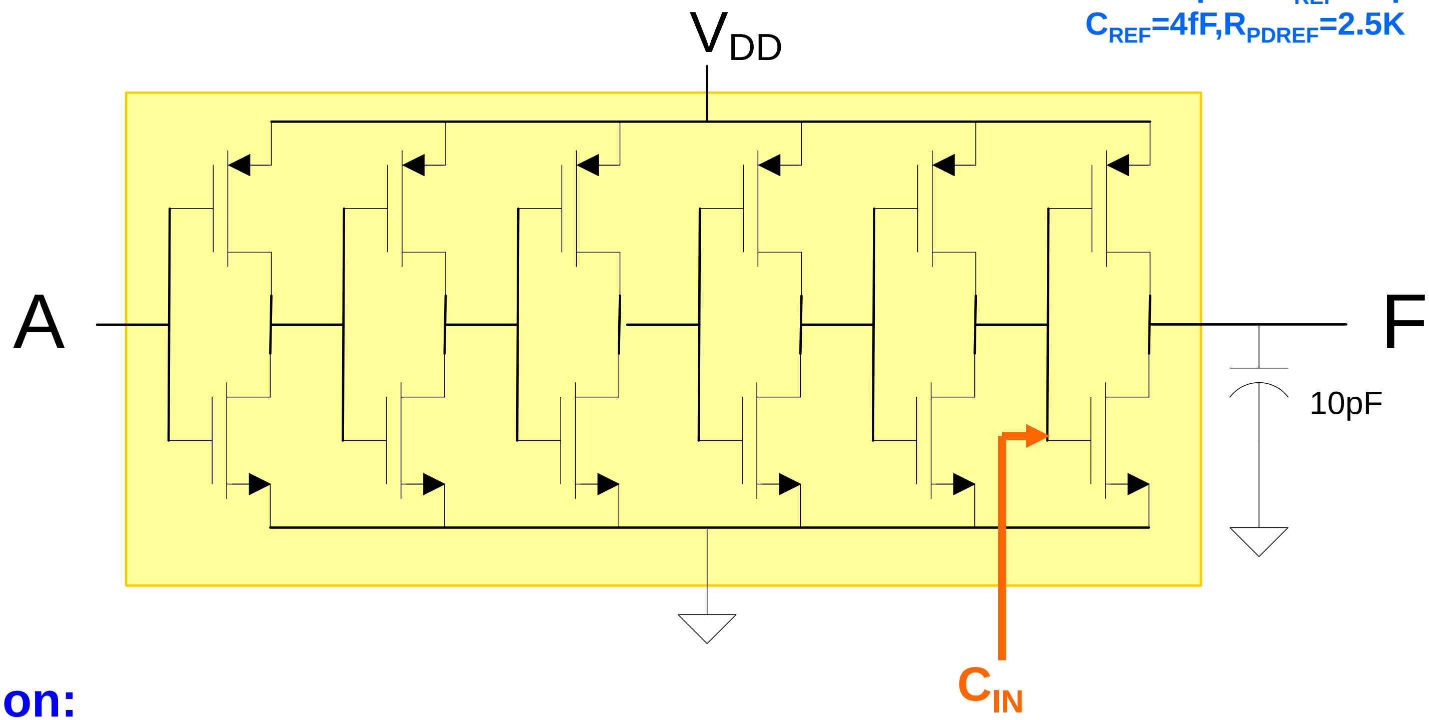
$$\frac{\text{FI}_{load}}{OD_6} = \frac{2500}{98} \cong 25$$

$$t_{prop} = 5 \bullet 2.5 \bullet 20\text{psec} + 25 \bullet 20\text{psec} = (12.5 + 25)20\text{psec} = 0.75\text{nsec}$$

since t_{CLK}>t_{PROP}, this pad driver can drive the 10pF load at 500MHz

Example: Determine the dynamic power dissipation in the next to the last stage of a 6-stage CMOS pad driver if used to drive a 10pF capacitive load if clocked at 500MHz. Assume pad driver with OD of θ=2.5 and V_{DD}=3.5V

In 0.5u proc t_{REF}=20ps, C_{REF}=4fF,R_{PDREF}=2.5K



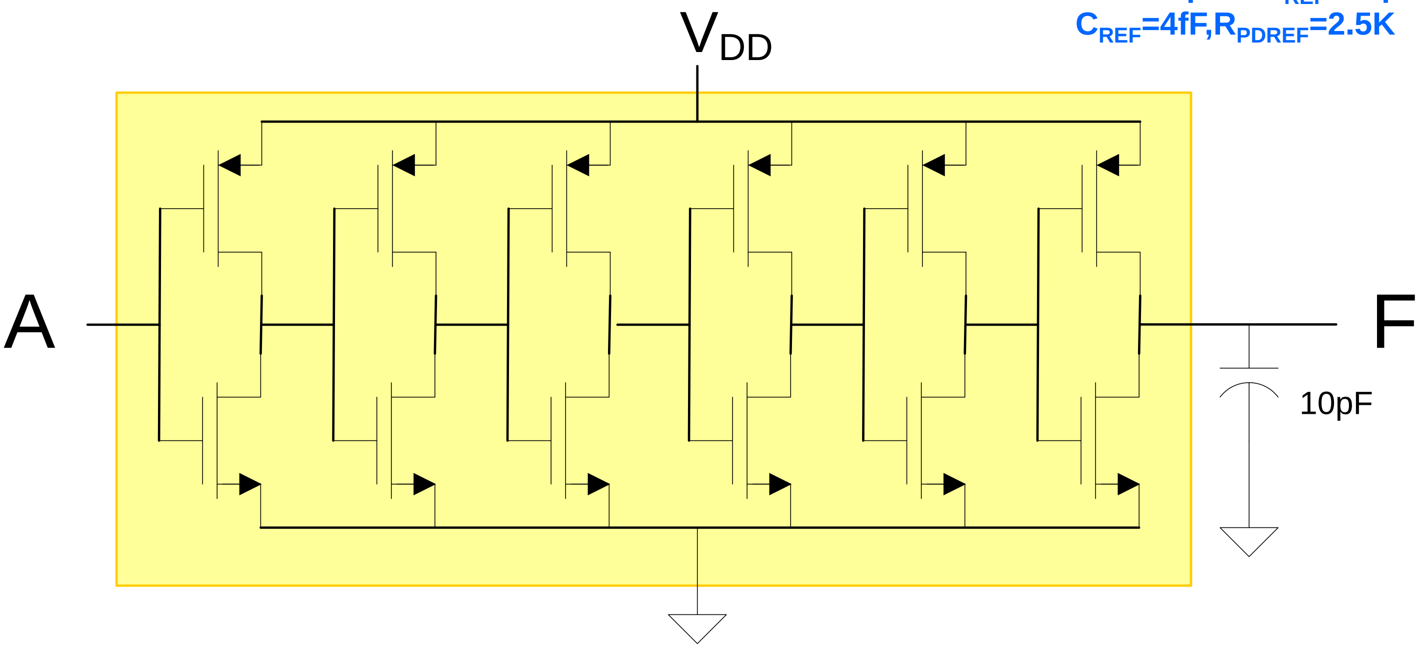
Solution:

$$C_{IN}=\theta^5 C_{REF}=2.5^5 \bullet 4fF=390fF$$

$$P_{DYN}=f_{CL} C_L V_{DD}^2 \quad =5E8 \bullet 390fF \bullet 3.5^2 = 2.4mW$$

Example: Is the 6-stage CMOS pad driver adequate to drive the 10pF capacitive load as fast as possible? Assume pad driver with OD of θ=2.5 and V_{DD}=3.5V

In 0.5u proc t_{REF}=20ps, C_{REF}=4fF,R_{PDREF}=2.5K



Solution:

$$n_{OPT} = \ln\left(\frac{C_L}{C_{REF}}\right) = \ln\left(\frac{10pF}{4fF}\right) = 7.8$$

No – an 8-stage pad driver would drive the load much faster (but is not needed If clocked at only 500MHz)

Example: Determine the power that would be required in the last stage of a CMOS pad driver to drive a 32-bit data bus off-chip if the capacitive load on each line is 2pF. Assume the clock speed is 500MHz and that each bit has an average 50% toggle rate. Assume V_{DD}=3.5V

In 0.5u proc t_{REF}=20ps, C_{REF}=4fF,R_{PDREF}=2.5K

Solution:

$$P_{DYN}=32\bullet\frac{f_{CL}}{2}C_L V_{DD}^2 \quad =32\bullet\frac{5E8}{2}\bullet 2pF\bullet 3.5^2 = 196mW$$

Note: A very large amount of power is required to take a large bus off-chip if bus has a high rate of activity.

Stay Safe and Stay Healthy !

End of Lecture 43