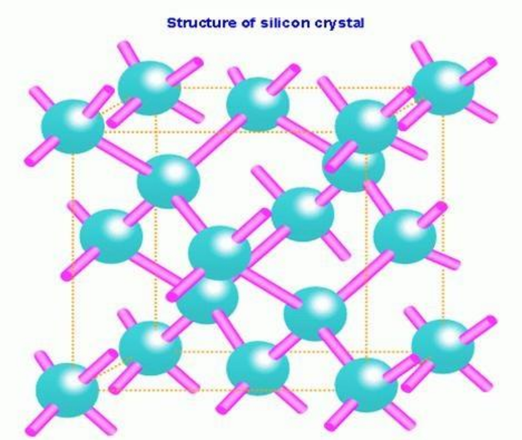
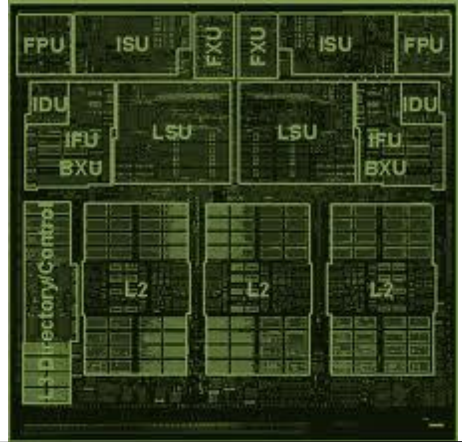
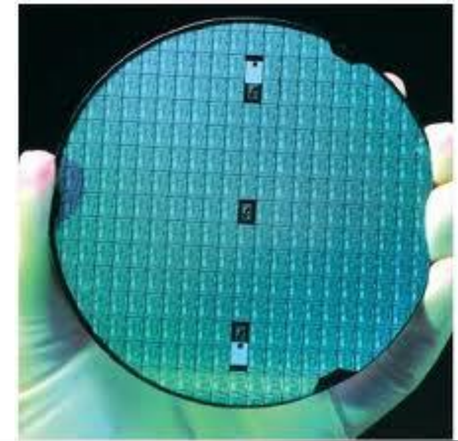






ECE 225

High Speed Digital IC Design

Lecture 3

CMOS Design (Review)

Prof. Kaustav Banerjee
Electrical and Computer Engineering
University of California, Santa Barbara
E-mail: kaustav@ece.ucsb.edu

Static and Dynamic CMOS

- In static circuits at every point in time (except when switching) the output is connected to either GND or V_{DD} via a low resistance path.
  - fan-in of n requires $2n$ (n N-type + n P-type) devices

- Dynamic circuits rely on the temporary storage of signal values on the capacitance of high impedance nodes.
  - requires only $n + 2$ (n+1 N-type and 1 P-type) transistors

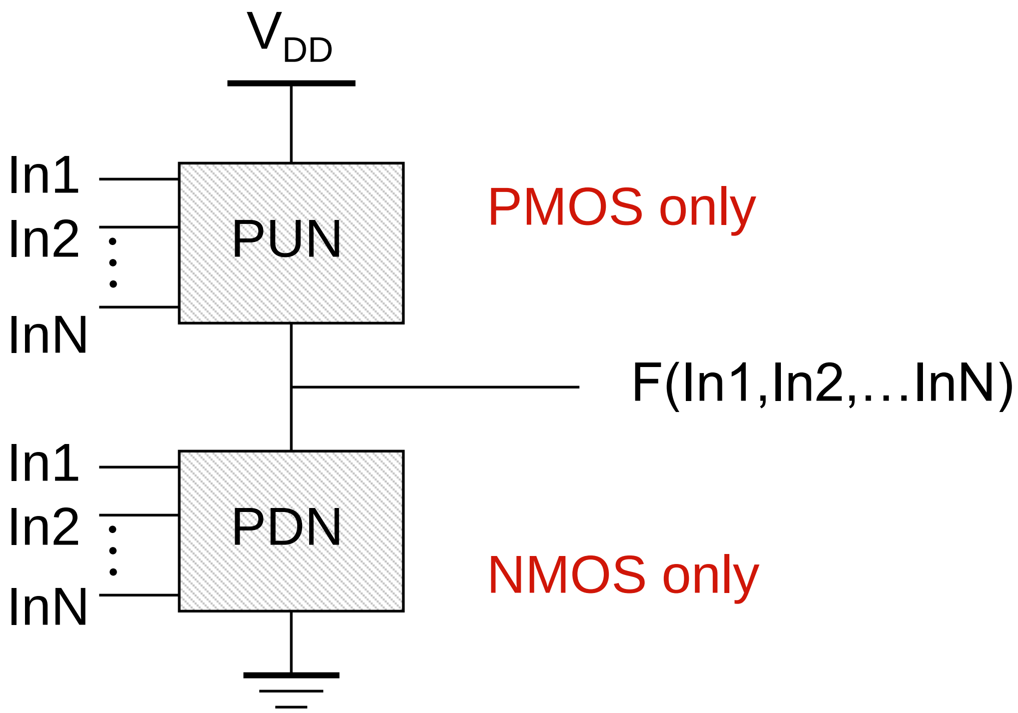
Static and Dynamic CMOS Circuit Families

- Static:
  - Complementary CMOS
    - (robustness, low power, large fan-in expensive in terms of area and performance)
  - Ratioed Logic (pseudo-NMOS, DCVSL)
    - (simple and fast at the expense of reduced NM and static power)
  - Pass-Transistor Logic (Transmission Gate)
  - -attractive for specific circuits: MUX, XOR-dominated logic such as Adders)
- Dynamic:
  - good for fast and complex gates, design process is harder due to parasitic effects, leakage puts an upper limit on the operating frequency of the circuit
    - Domino Logic
    - np-CMOS

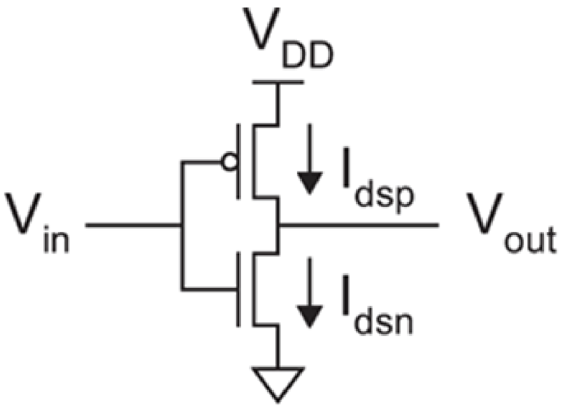
Which style is best?

……depends on ease of design, performance, power, area and robustness.

Static Complementary CMOS



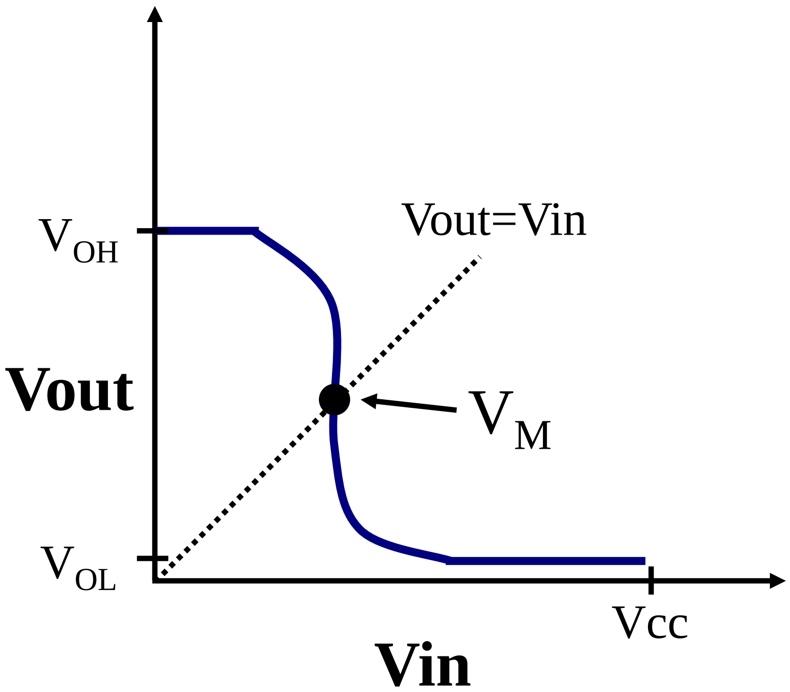
The most widely used gate….



CMOS Inverter

PUN and PDN are dual logic networks

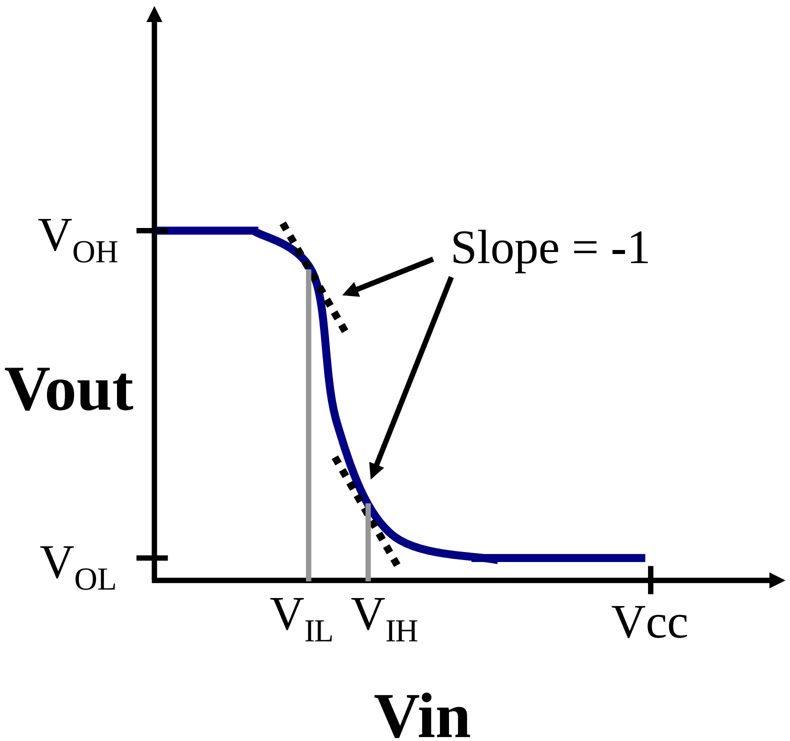
Inverter Threshold



Inverter switching threshold:

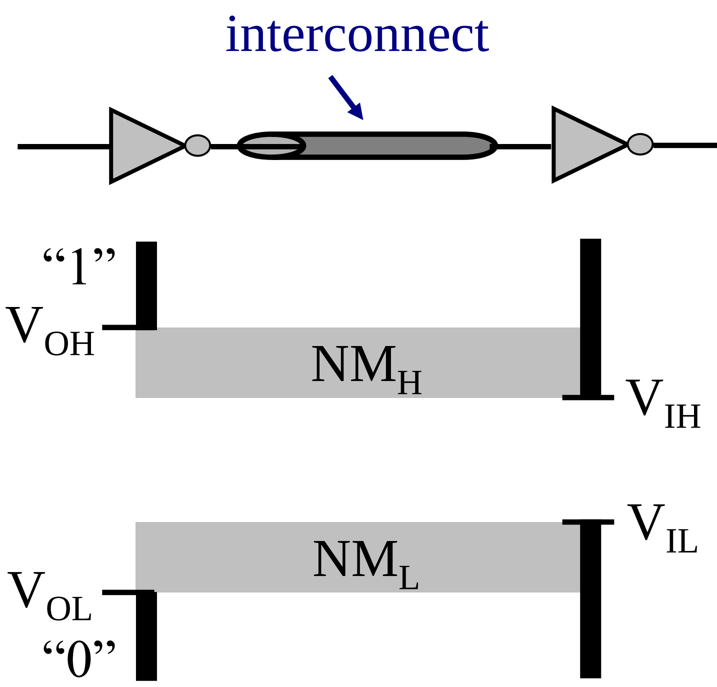
- Point where voltage transfer curve intersects line Vout=Vin
- Represents the point at which the inverter switches state
- Normally, $V_M \approx$ Vcc/2

Noise Margins



- V_{IL} and V_{IH} measure effect of input voltage on inverter output
- V_{IL} = largest input voltage recognized as logic '0'
- V_{IH} = smallest input voltage recognized as logic '1'
- Defined as point on VTC where slope = -1

Noise Margin (cont)

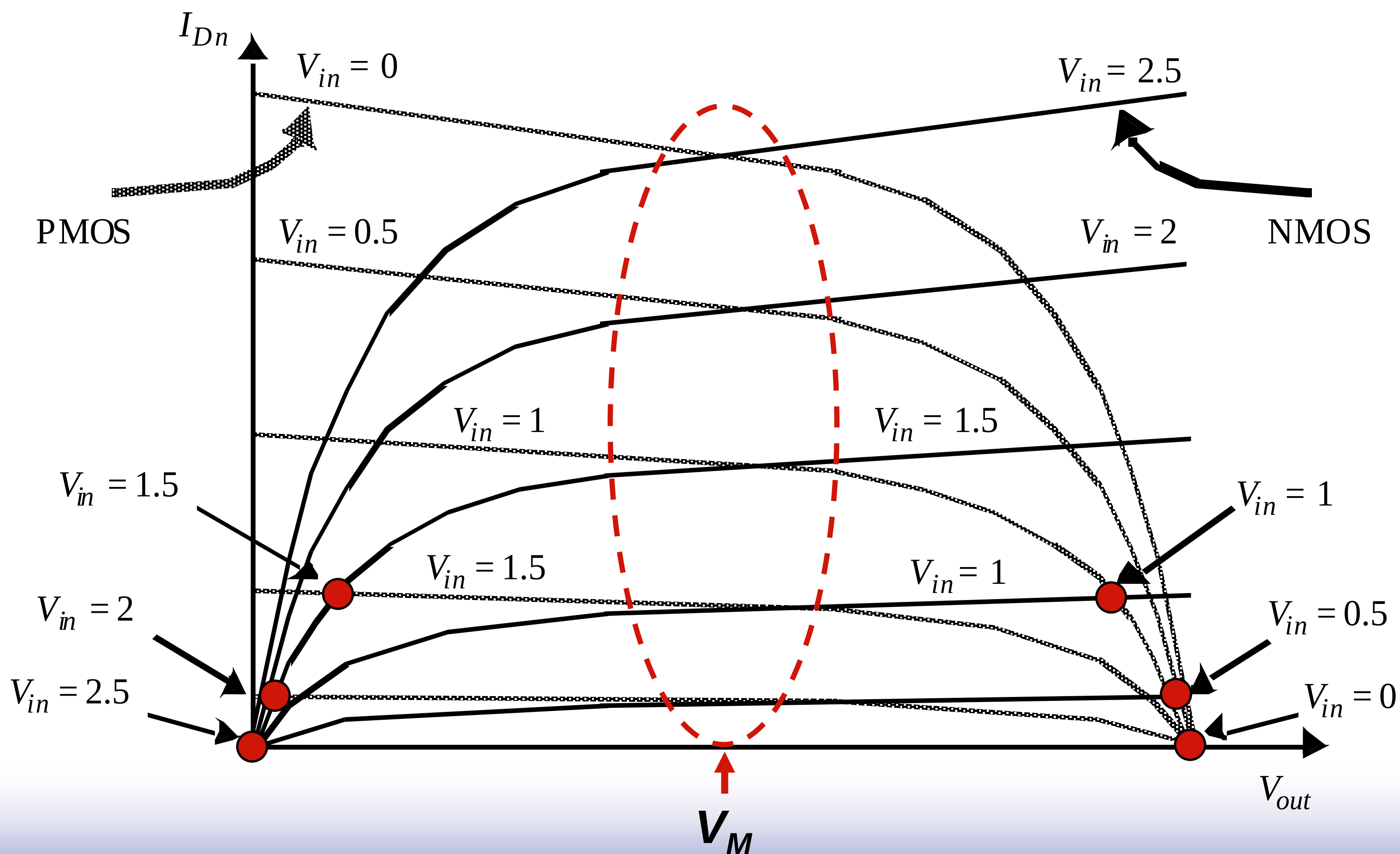


Ideally, noise margin should be as large as possible

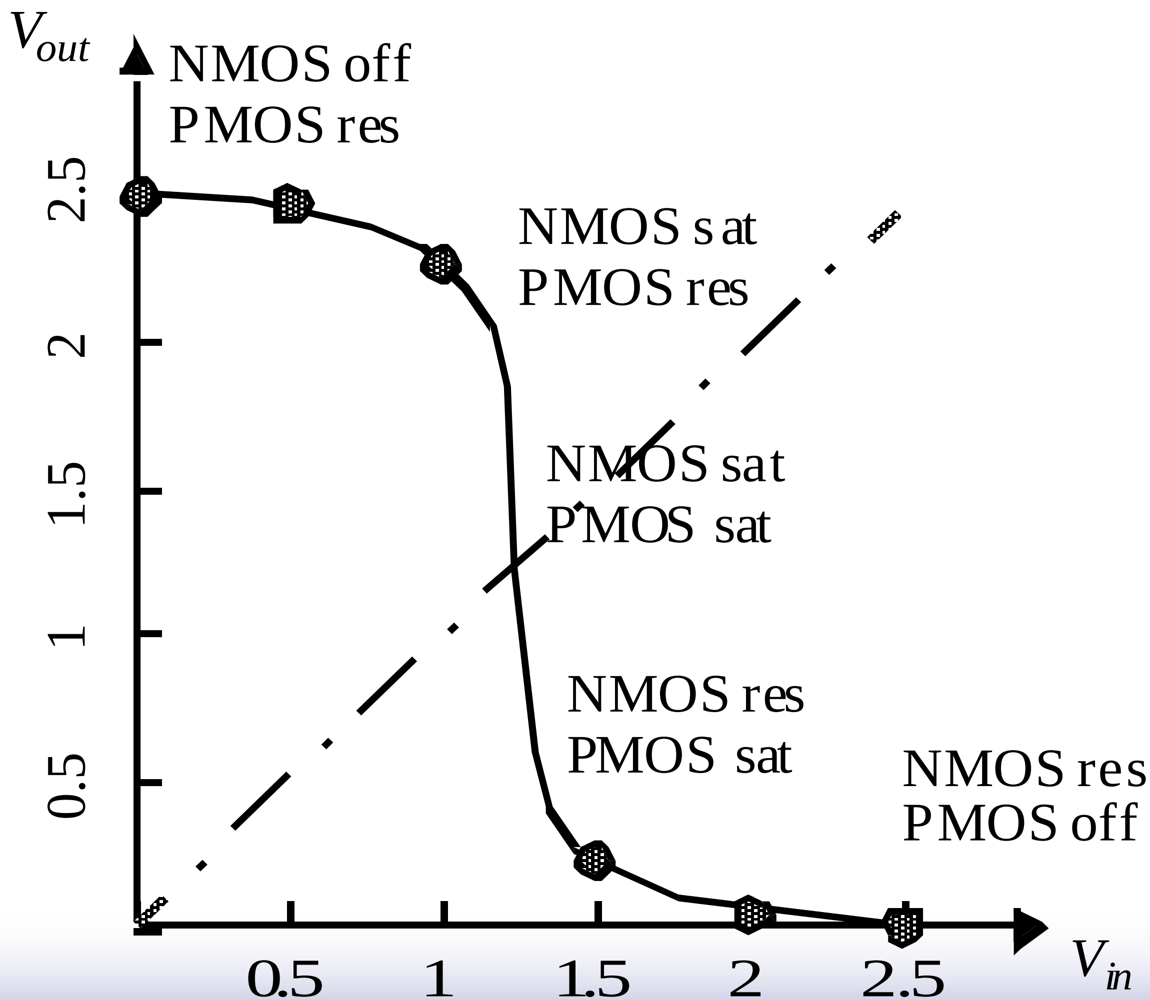
- Noise margin is a measure of the *robustness* of an inverter
  - $N_{ML} = V_{IL} - V_{OL}$
  - $N_{MH} = V_{OH} - V_{IH}$
- Models a chain of inverters. Example:
  - First inverter output is V_{OH}
  - Second inverter recognizes input $> V_{IH}$ as logic '1'
  - Difference V_{OH}-V_{IH} is "safety zone" for noise

CMOS Inverter Load Characteristics

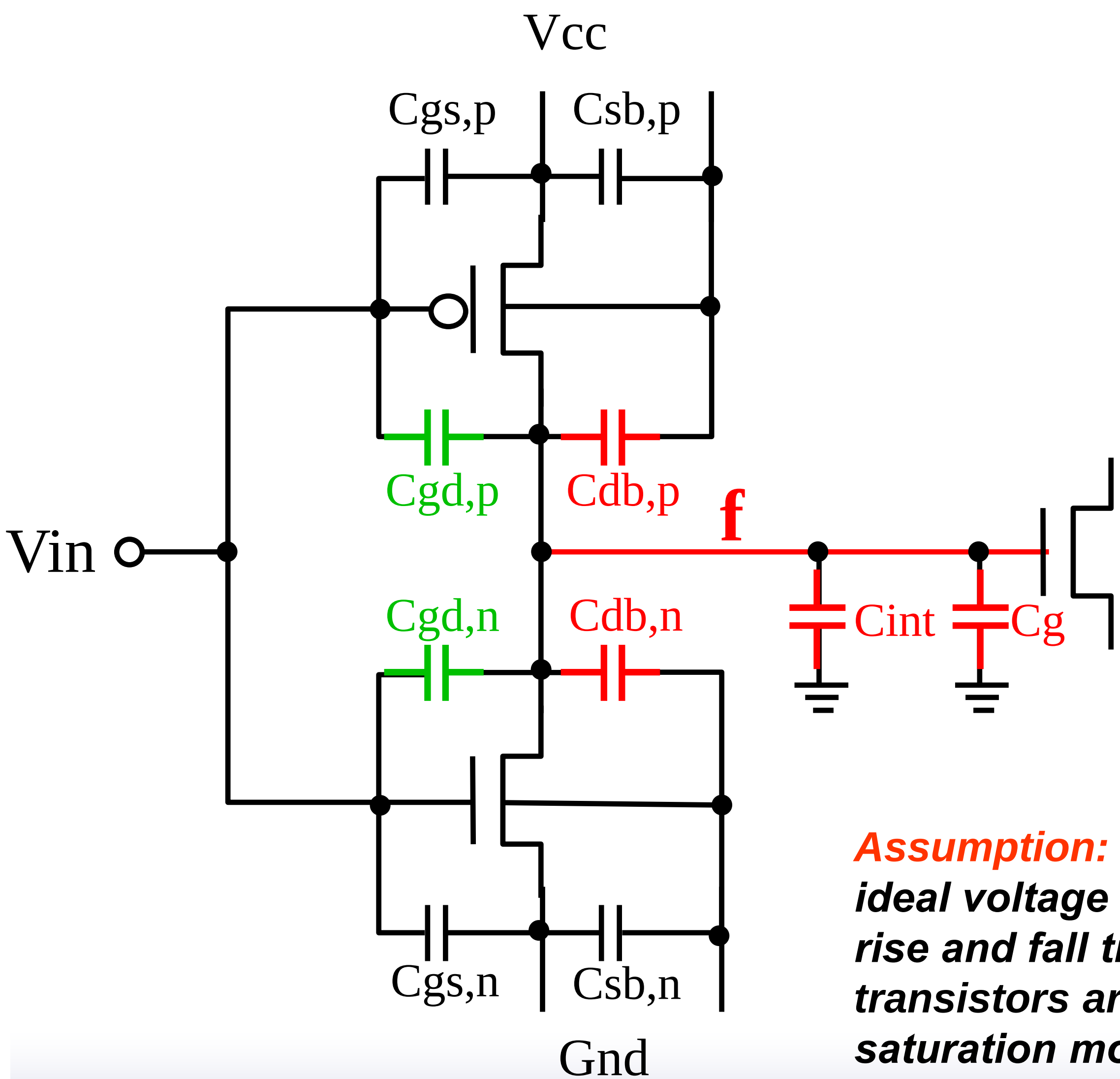
B. For a DC operating point to be valid, currents through NMOS and PMOS must be equal (for a given V_{in}), hence find the points of intersection.



CMOS Inverter VTC



CMOS inverter capacitances



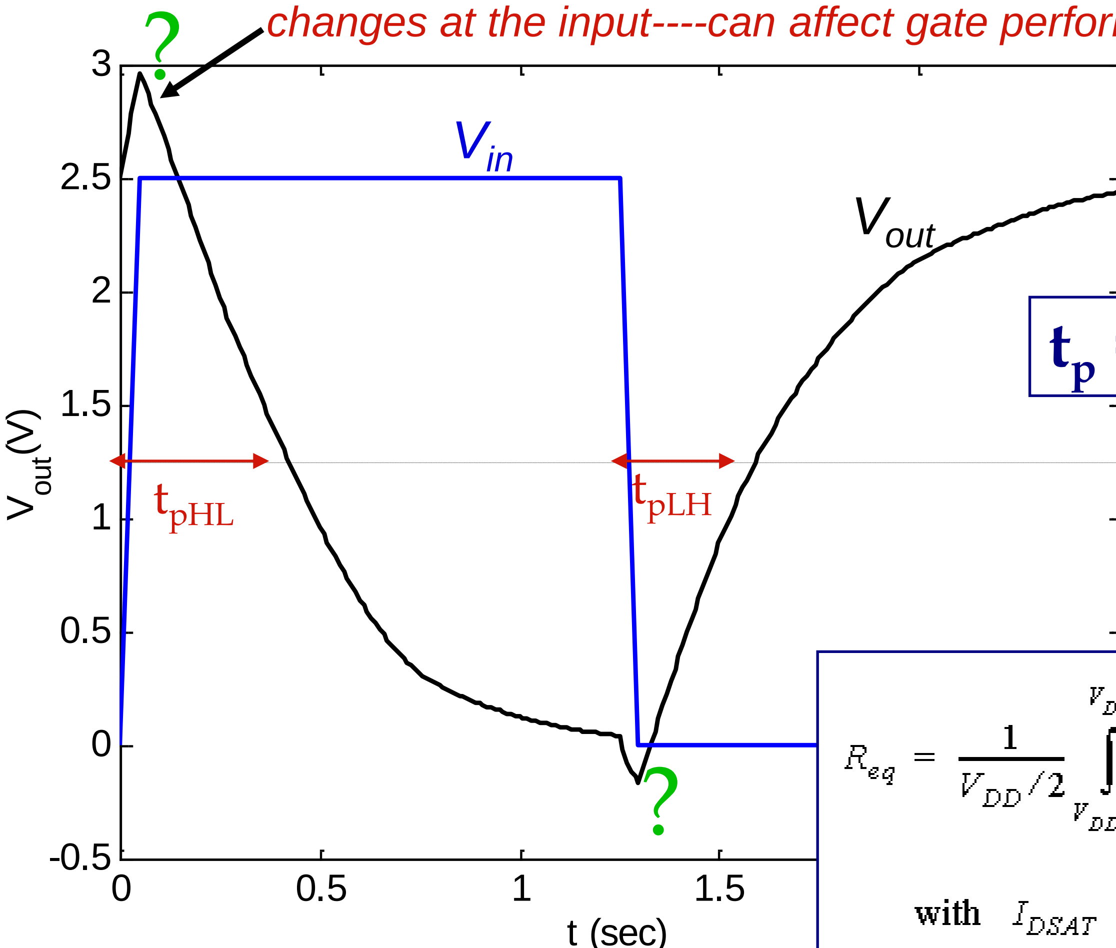
Cap on Node f:

- Junction cap: $C_{db,p}$ and $C_{db,n}$
- Gate (overlap) capacitance $C_{gd,p}$ and $C_{gd,n}$ (beware of Miller effect)
- Interconnect cap: C_{int}
- Receiver gate cap: C_g

Assumption:* *Vin is driven by an ideal voltage source….with zero rise and fall times….hence the transistors are either in cut-off or saturation mode…hence, no channel capacitance

Transient Response

Due to C_{gd} of transistors: directly couples voltage at input to output before the transistors can even start to react to changes at the input----can affect gate performance



Symmetric inverter has $t_{pHL}=t_{pLH}$

$$\mathbf{t_p = 0.69\, C_L\, (R_{eqn}+R_{eqp})/2}$$

$$R_{eq} = \frac{1}{V_{DD}/2}\int_{V_{DD}/2}^{V_{DD}} \frac{V}{I_{DSAT}(1+\lambda V)}\,dV \approx \frac{3}{4}\frac{V_{DD}}{I_{DSAT}}\left(1-\frac{7}{9}\lambda V_{DD}\right)$$

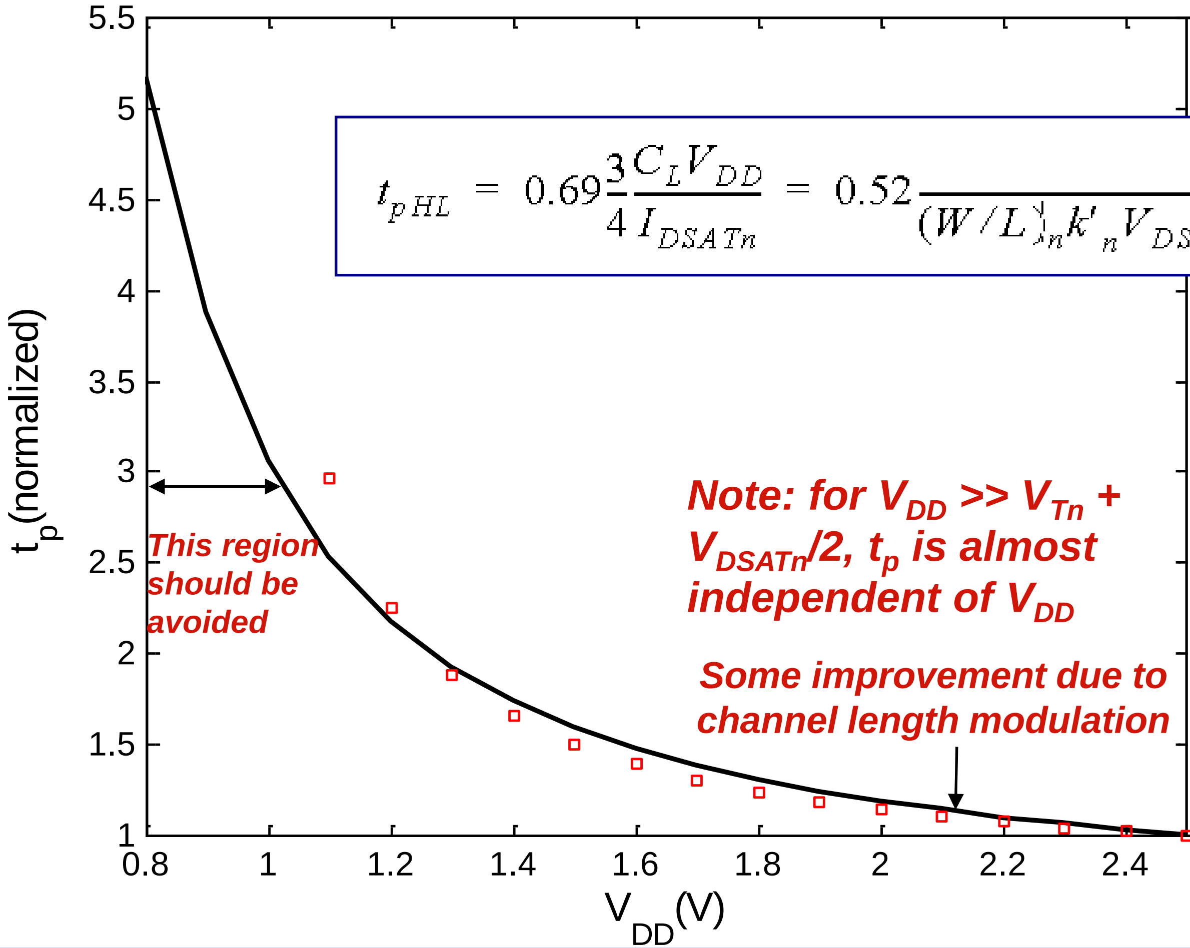
$$\textbf{with}\quad I_{DSAT} = k'\frac{W}{L}\left((V_{DD}-V_T)V_{DSAT}-\frac{V_{DSAT}^2}{2}\right)$$

Delay as a function of V_{DD}

Same as the ON resistance of a transistor….

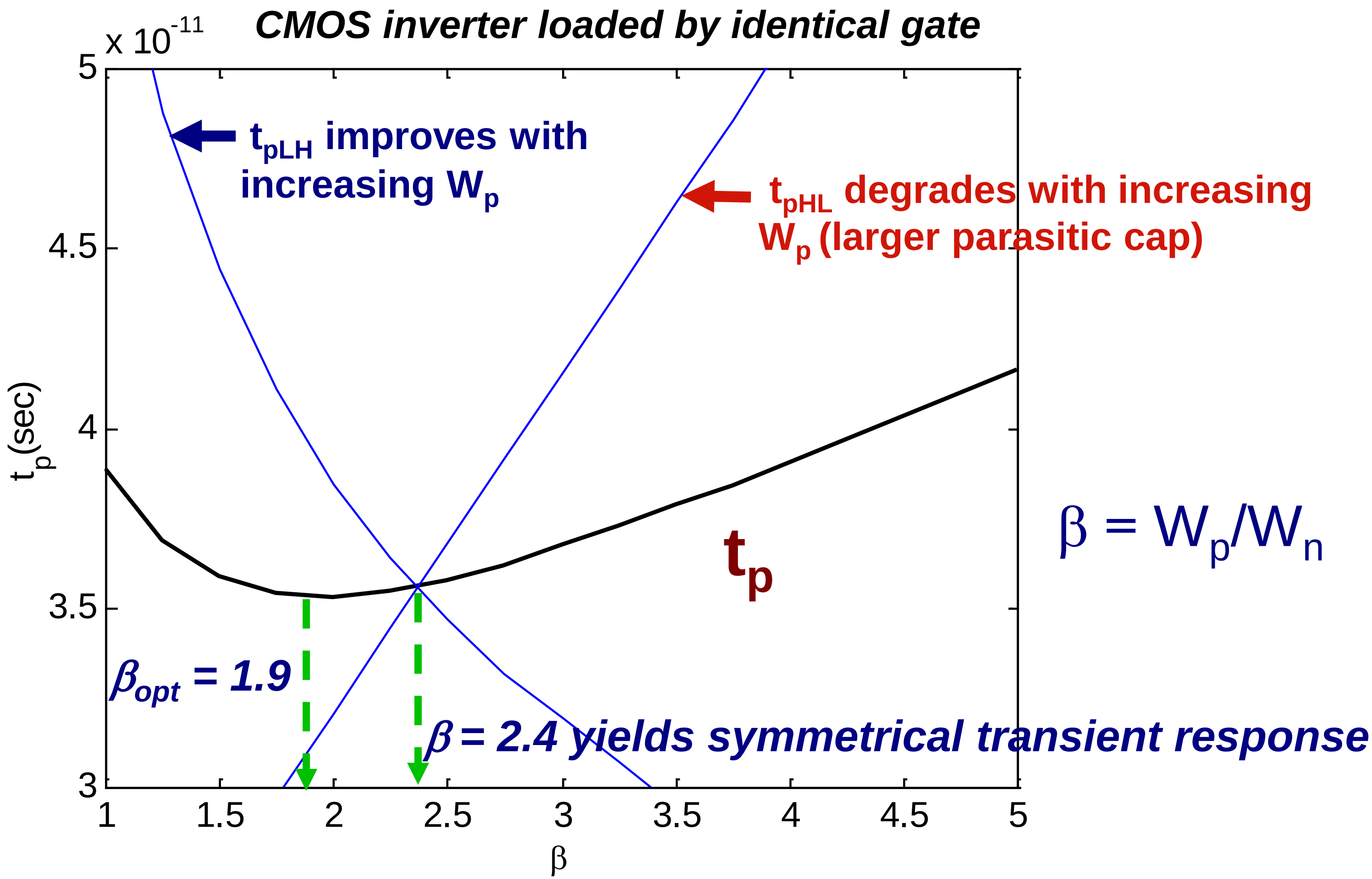
Trade off energy dissipation vs performance…..

$$t_{pHL} = 0.69\frac{3}{4}\frac{C_L V_{DD}}{I_{DSATn}} = 0.52\frac{C_L V_{DD}}{(W/L)_n k'_n V_{DSATn}(V_{DD} - V_{Tn} - V_{DSATn}/2)}$$



Reliability concerns at high V_{DD}….

NMOS/PMOS ratio



If symmetry and noise margins are not of prime concern, inverter delay can be reduced by reducing the width of PMOS….

Designing Combinational Logic Circuits

Equivalent Inverter

- CMOS gates: many paths to Vcc and Gnd
  - Multiple values for V_M, V_{IL}, V_{OL}, etc
  - Different delays for each input combination
- Equivalent inverter
  - Represent each gate as an inverter with appropriate device width
  - Include only transistors which are on or switching
  - Calculate V_M, delays, etc using inverter equations

Transistor Sizing

- Sizing for switching threshold
  - All inputs switch together
- Sizing for delay
  - Find worst-case input combination
- Find equivalent inverter, use inverter analysis to set device sizes

Transistor Sizing

What sizing will lead to a 2:1 equivalent inverter?

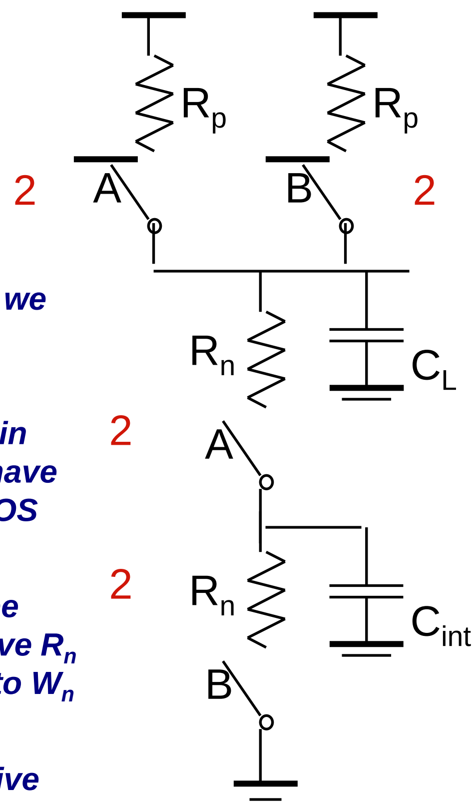
For effective $R_p = R_n$, we

need $W_p = 2W_n$

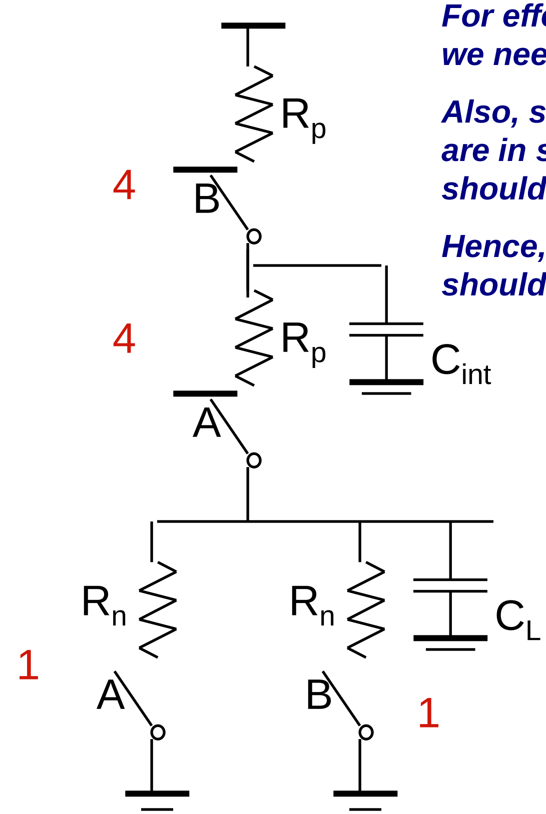
Since two NMOS are in series, each should have $R_n/2$, hence each NMOS size, $W_n = 2$

Each PMOS should be such that R_p = effective R_n (which corresponds to W_n =1)

Hence, W_p = 2 (effective W_n) = 2



2-input NAND



For effective $R_p = R_n$, we need $W_p = 2W_n$

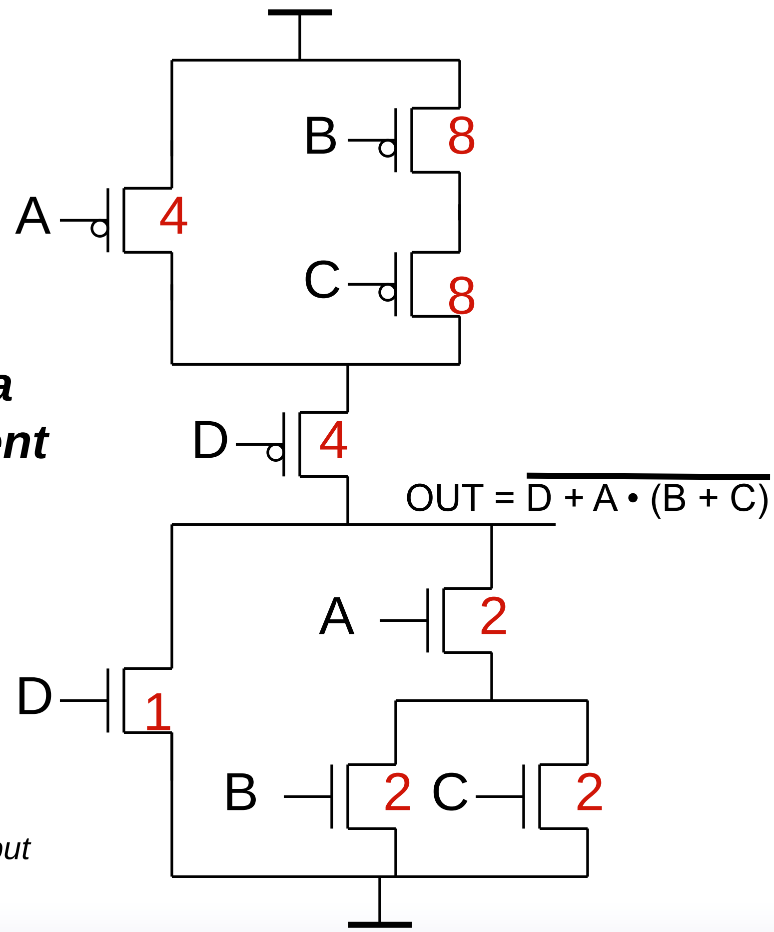
Also, since two PMOS are in series, each should have $R_p/2$

Hence, W_p of each should be $4W_n$

Avoid stacking PMOS transistors in series….

2-input NOR

Transistor Sizing a Complex CMOS Gate



What sizing will lead to a 2:1 equivalent inverter?

Note: A= 3 and B=C=D=6 will also give a 2:1 inverter but that will give higher output parasitic capacitance (due to larger D)

1. Start with a transistor in the PDN, that is (preferably) isolated (i.e., it can pull down the output node on its own)----D in this case
2. Assign a minimum size to this transistor (W=1 for D)
3. Now identify other paths in the PDN that can also discharge the output node: AB and AC in this case.
4. Size A, B, and C such that: each path will be electrically equivalent to the path through D (i.e., with same resistance), hence A, B and C should have W=2
5. Repeat the same for the PUN, keeping in mind that i) effective resistance of any path must equal the effective resistance of any path in the PDN, and ii) PMOS transistors will have to be sized (twice as large in this case) appropriately.

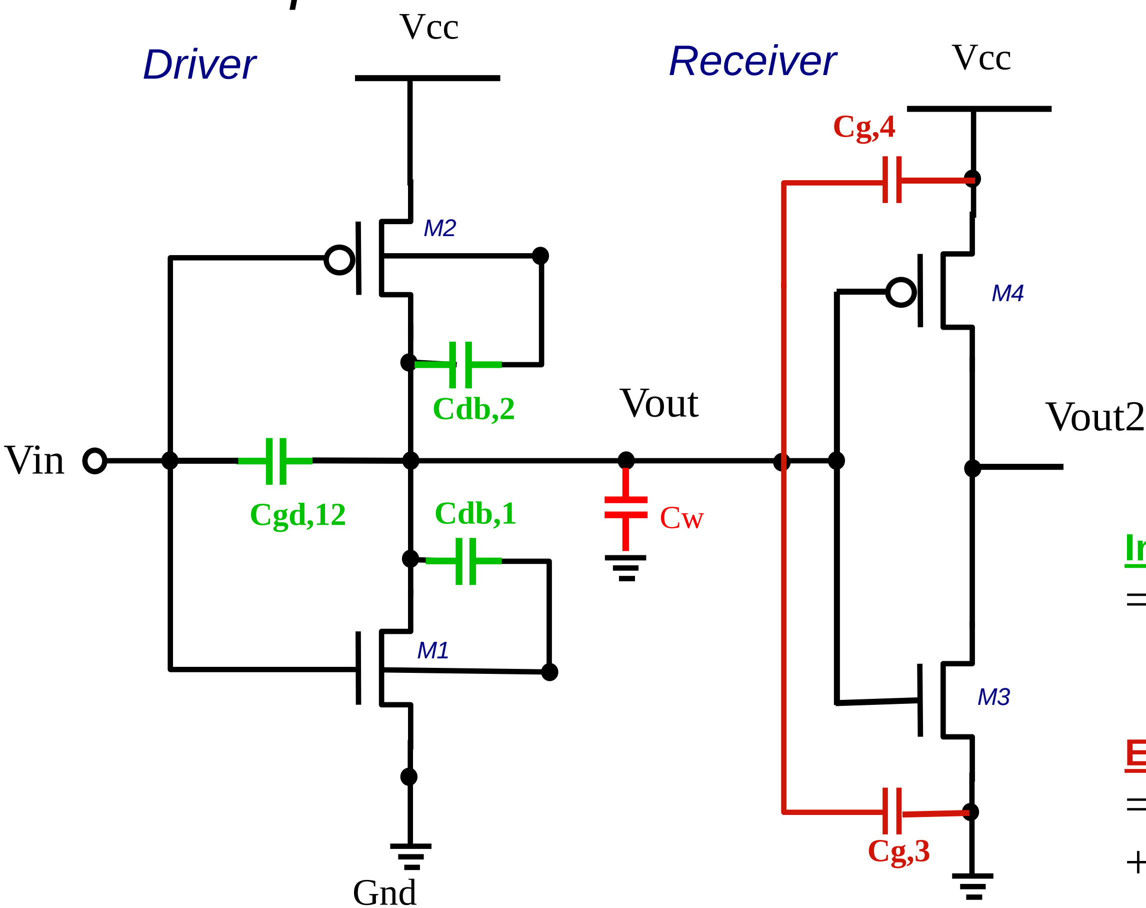
Note: here we ignored internal caps.

CMOS gate design: summary

- Designing a CMOS gate:
  - Find pulldown NMOS network from logic function or by inspection
  - Find pullup PMOS network
    - By inspection
    - Using logic function
    - Using dual network approach
  - Size transistors using equivalent inverter
    - Find worst-case pullup and pulldown paths
    - Size to meet rise/fall or threshold requirements

Inverter Chain Sizing

Load capacitances



$$C_L = C_{int} + C_{ext}$$

Internal Caps of Driver (Cint):

= Junction caps: $C_{db,12}$ +

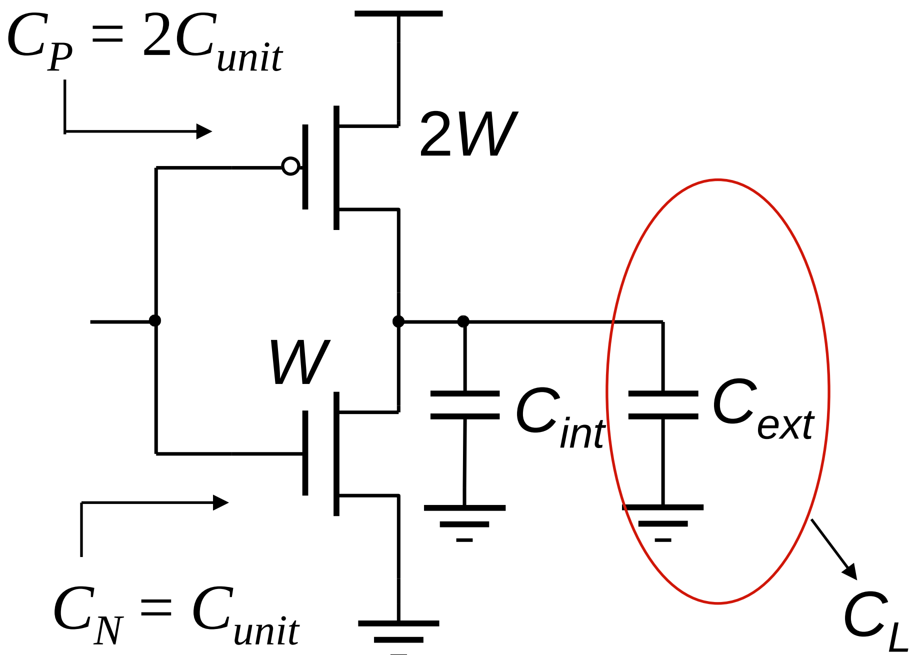
Gate caps: $C_{gd,12}$ (including Miller Caps.)

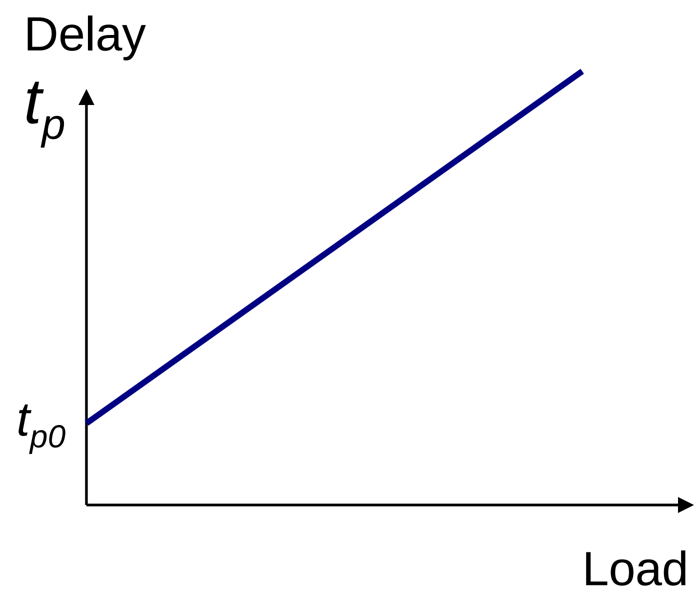
External Caps (Cext):

= Interconnect cap: C_w

+ Receiver gate caps: $C_{g,43}$

Inverter with Load





Delay (t_p) = $kR_W(C_{int} + C_{ext}) = kR_W C_{int} + kR_W C_{ext} = \underbrace{kR_W C_{int}}_{t_{p0}\ (intrinsic\ delay)}(1 + C_{ext}/C_{int})$

Intrinsic delay of CMOS inverter

Let R_{eq} be the equivalent resistance of the gate (inverter), then delay (t_p) is defined as:

$$t_p = 0.69\,R_{eq}\left(C_{int} + C_{ext}\right)$$

$$= 0.69\,R_{eq}\,C_{int}\left(1 + \frac{C_{ext}}{C_{int}}\right)$$

$$= t_{p0}\left(1 + \frac{C_{ext}}{C_{int}}\right)$$

t_{p0} is the intrinsic delay

Impact of sizing on gate delay

Let S be the sizing factor

R_{ref} *be the resistance of a reference gate (usually a minimum size gate)*

C_{iref} *be the internal capacitance of the reference gate*

$$C_{\text{int}} = S\,C_{iref}\;, \quad R_{eq} = \frac{R_{ref}}{S}$$

$$t_p = 0.69\left(\frac{R_{ref}}{S}\right)\left(S\,C_{iref}\right)\left(1+\frac{C_{ext}}{SC_{iref}}\right)$$

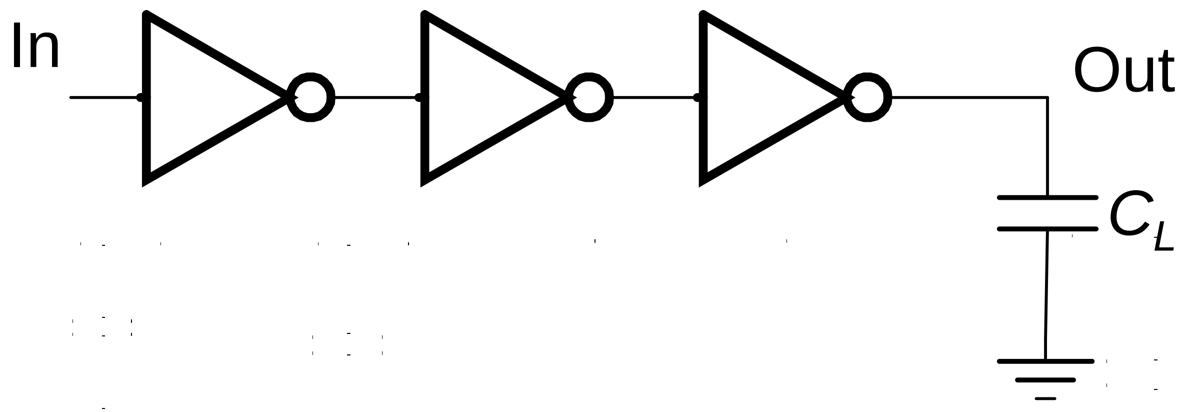
$$= 0.69\,R_{ref}\;C_{iref}\left(1+\frac{C_{ext}}{SC_{iref}}\right)$$

$$= t_{p0}\left(1+\frac{C_{ext}}{SC_{iref}}\right)$$

Hence:

1. *Intrinsic delay is independent of gate sizing, and is determined only by technology and inverter layout*
2. *If S is made very large, gate delay approaches the intrinsic value but increases the area significantly*

Inverter Chain



If C_L is given:
- How many stages are needed to minimize the delay?
- How to size the inverters?

May need some additional constraints.

Delay Formula: inverter chain

$$Delay \sim R_{eq}\left(C_{int} + C_{ext}\right)$$

$$t_p = 0.69 R_{eq} C_{int}\left(1 + C_{ext} / \gamma C_{gin}\right) = t_{p0}\left(1 + f / \gamma\right)$$

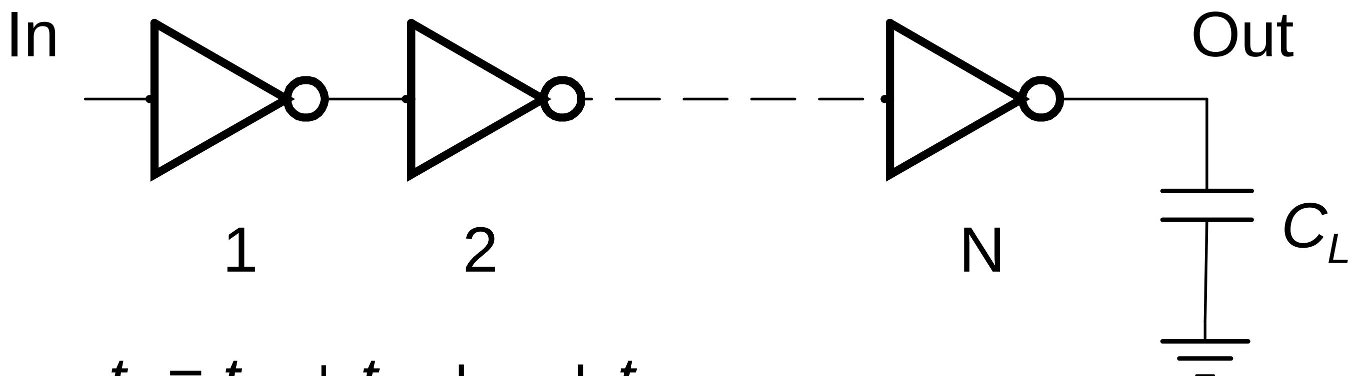
t_{p0}

relates the input gate cap. and the intrinsic output cap. of the inverter…

Let $C_{int} = \gamma C_{gin}$ with $\gamma \approx 1$

$f = C_{ext}/C_{gin}$ - effective fanout

Apply to Inverter Chain



$$t_p = t_{p1} + t_{p2} + \ldots + t_{pN}$$

$$t_{p,j} = t_{p0}\left(1 + \frac{C_{gin,j+1}}{\gamma C_{gin,j}}\right)$$

$$t_p = \sum_{j=1}^{N} t_{p,j} = t_{p0}\sum_{j=1}^{N}\left(1 + \frac{C_{gin,j+1}}{\gamma C_{gin,j}}\right),\ C_{gin,N+1} = C_L$$

Optimal Tapering for Given N

Delay equation has N - 1 unknowns, $C_{g,2}$ …. $C_{g,N}$

Minimize the delay, find N - 1 partial derivatives and equate them to zero, or $\left(\partial t_p / \partial C_{g,j}\right) = 0$

Result: $C_{g,j+1}/C_{g,j} = C_{g,j}/C_{g,j-1}$ With j = 2,…..,N

Size of each stage is the geometric mean of two neighbors

$$C_{g,j} = \sqrt{C_{g,j-1}C_{g,j+1}}$$

- each stage has the same effective fanout ($f_j = f = C_{ext}/C_{g,j}$)
- hence, each stage has the same delay: $t_p = t_{p0}\,(1 + f/\gamma)$

Optimum Delay and Number of Stages

When each stage is sized by *f* and has same eff. fanout *f*:

$$\frac{C_L}{C_{g,N}} = \frac{C_{g,N}}{C_{g,N-1}} = \cdots\cdots\cdots = \frac{C_{g,2}}{C_{g,1}} = f$$

$$Hence,\ \ f^N = C_L / C_{g,1} = F$$

F is the overall effective fanout of the circuit

Effective fanout of each stage: $f = \sqrt[N]{F}$

If C_L and $C_{g,1}$ are known….

Minimum path delay:

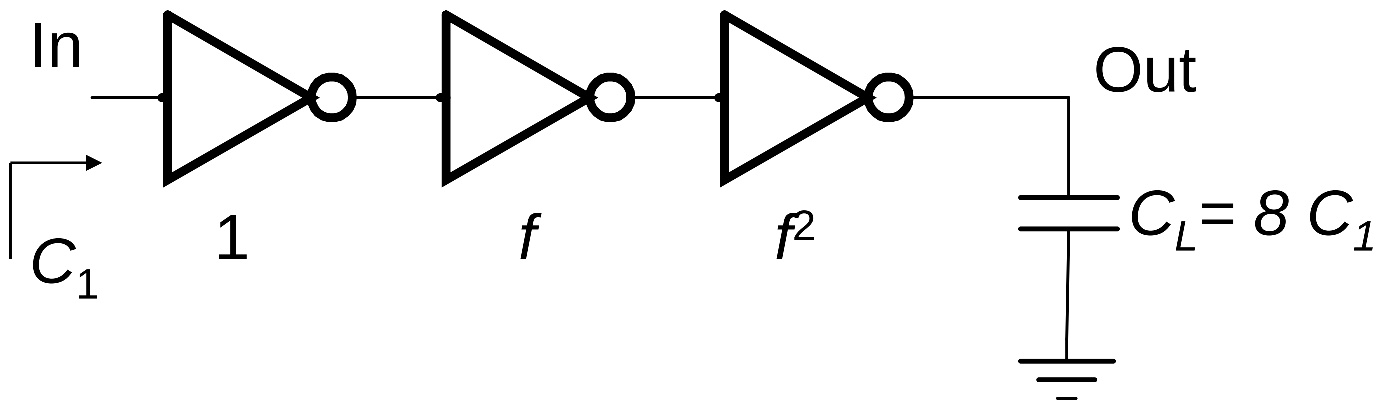
$$t_p = N t_{p0}\left(1 + \sqrt[N]{F} / \gamma\right)$$

If N is too large, intrinsic delay of stages dominate, while if N is small, effective fanout of each stage (f) is large and the second term dominates

How to choose N?

Example

If N is given….



C_L/C_1 has to be evenly distributed across $N = 3$ stages:

$$f = \sqrt[3]{8} = 2$$

Optimum Number of Stages

For a given load, C_L and given input capacitance C_{in}
Find optimal sizing f

$$C_L = F \cdot C_{in} = f^N C_{in} \quad with \quad N = \frac{\ln F}{\ln f}$$

$$t_p = N t_{p0}\left(F^{1/N}/\gamma + 1\right) = \frac{t_{p0} \ln F}{\gamma}\left(\frac{f}{\ln f} + \frac{\gamma}{\ln f}\right)$$

$$\frac{\partial t_p}{\partial f} = \frac{t_{p0} \ln F}{\gamma} \cdot \frac{\ln f - 1 - \gamma / f}{\ln^2 f} = 0$$

If self-loading is ignored….

For $\gamma = 0$, f = e = 2.718, N = ln$F$

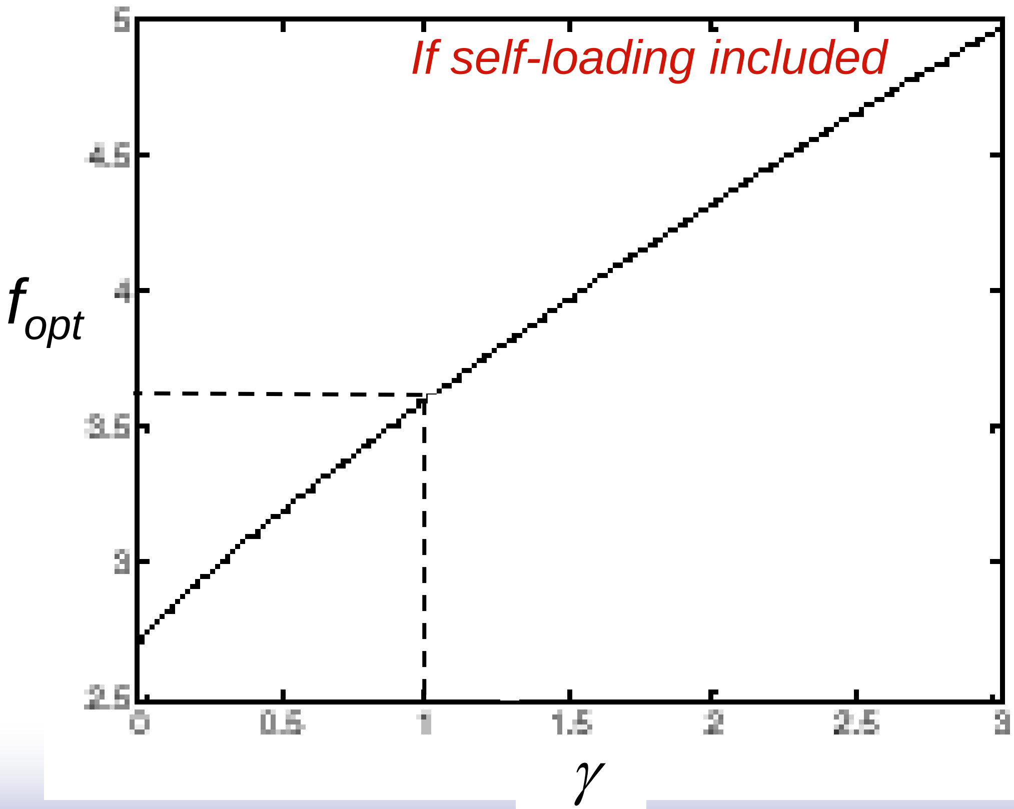
Otherwise….

$$f = \exp\left(1 + \gamma / f\right)$$

Optimum Effective Fanout f

Optimum f for given process defined by γ

$$f = exp\left(1 + \gamma / f\right)$$

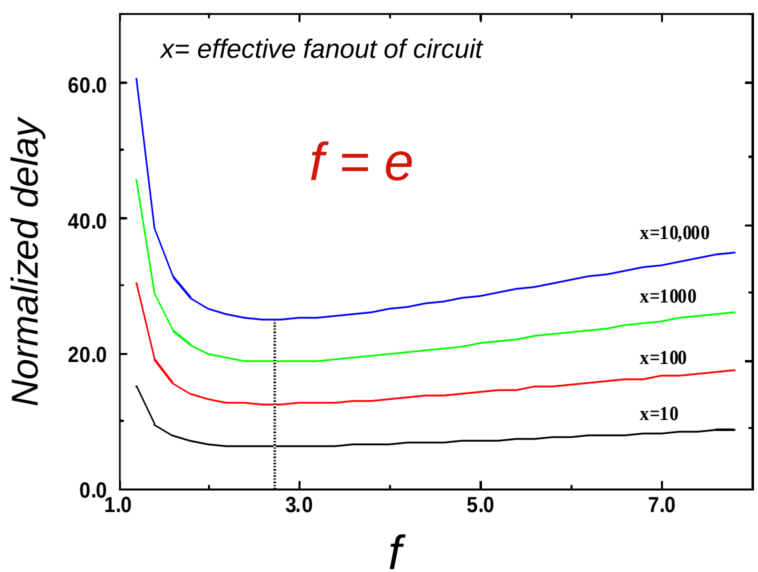


Optimum tapering factor:

f_{opt} = 3.6
for γ=1 (typical case)

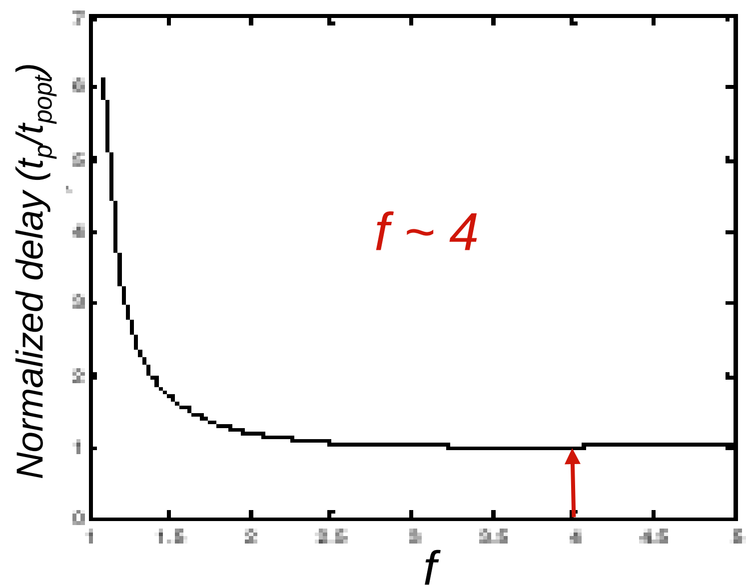
Impact of Self-Loading on tp

No Self-Loading, γ=0



Optimal number of stages, N= ln(F)

With Self-Loading γ=1



If $f<f_{opt}$ (too many stages) will result in delay to increase

Normalized delay function of F

$$t_p = N t_{p0} \left(1 + \sqrt[N]{F} / \gamma\right)$$

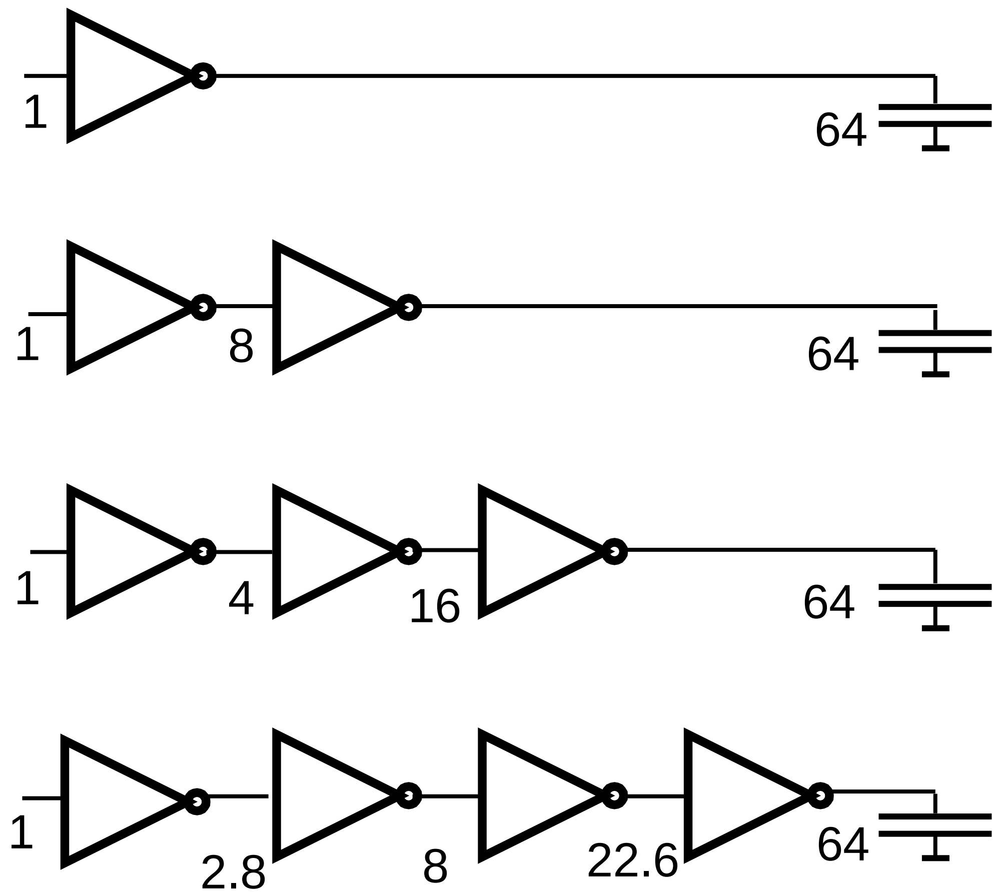
t_{popt}/t_{p0} for $\gamma=1$

F	Unbuffered	Two Stage	Inverter Chain
10	11	8.3	8.3
100	101	22	16.5
1000	1001	65	24.8
10,000	10,001	202	33.1

As F increases, the differences between the unbuffered case (or two-stage buffer case) and the case of inverter chain increases…..

Buffer Design

$$f = F^{1/N}$$



N	f	t_p
1	64	65
2	8	18
3	4	15
4	2.8	15.3

Sizing Logic Paths for Speed

- Frequently, input capacitance of a logic path is constrained
- Logic also has to drive some capacitance
- Example: ALU load in an Intel's microprocessor is 0.5pF
- How do we size the ALU datapath to achieve maximum speed?
- We have already solved this for the inverter chain – can we generalize it for any type of logic?

Minimizing Delay in Complex Logic Networks

$$Delay = t_{p0}\left(1 + \frac{f}{\gamma}\right) \; (inverter)$$

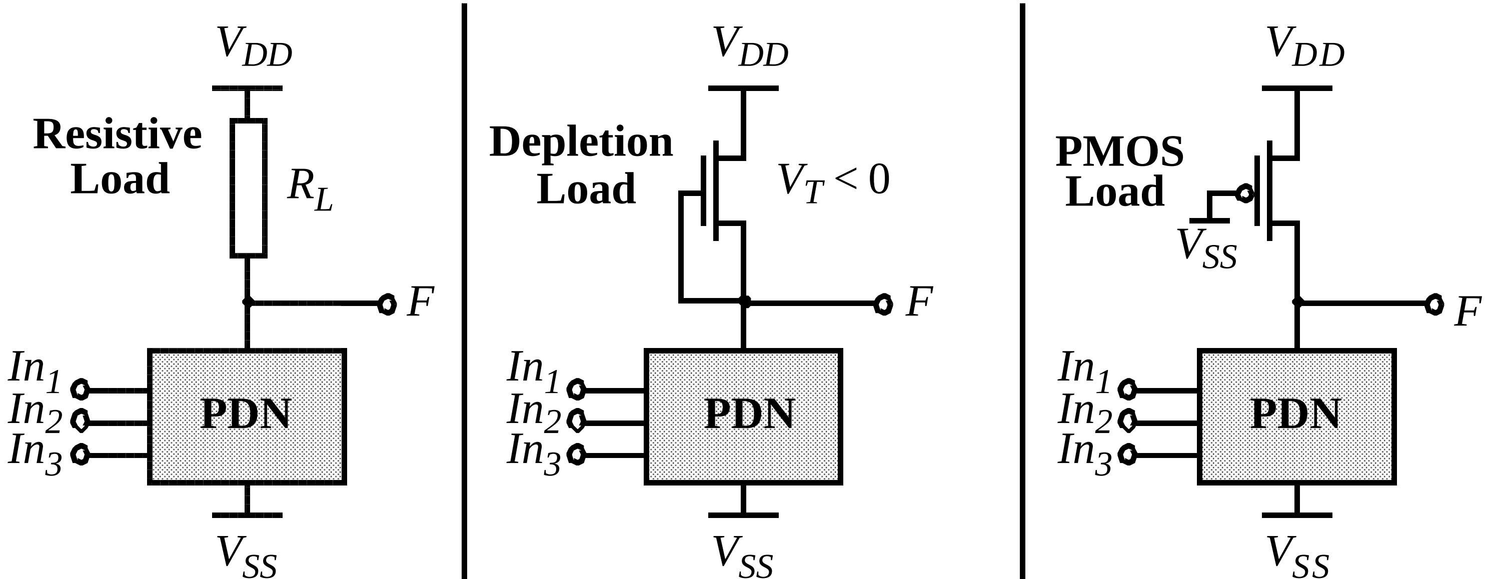
$$= t_{p0}\left(p + \frac{g \cdot f}{\gamma}\right) (Complex\ gate)$$

Everything Normalized w.r.t an inverter:
g_{inv} =1, p_{inv} = 1

f – effective fanout *(ratio of external load and input cap. of gate)*
p – ratio of intrinsic delays of complex gate and inverter
(value increases with complexity of gate)
g – logical effort: how much more input capacitance is presented by the complex gate to deliver the same output current as an inverter (depends only on circuit topology)

Ratioed Logic

Need N+1 transistors vs 2N for complementary CMOS



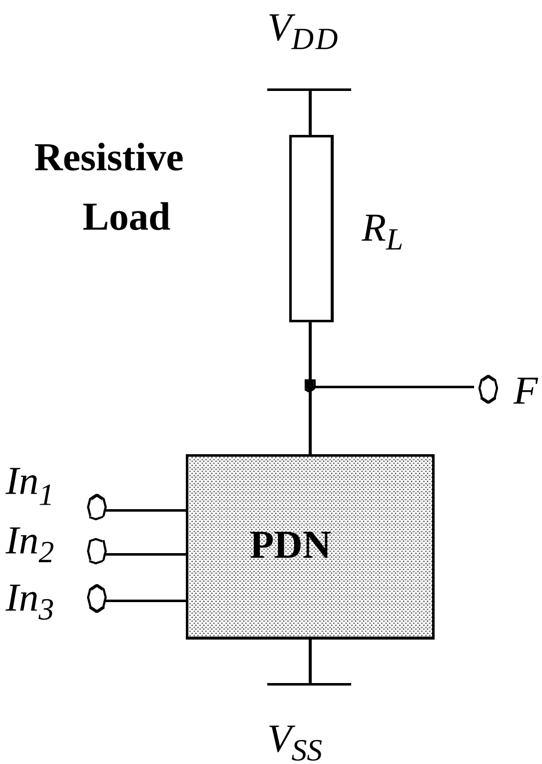
(a) resistive load (b) depletion load NMOS (c) pseudo-NMOS

Note: a depletion mode NMOS is normally ON…an n-type channel connects the source and drain and a negative gate bias is needed to turn it off…..

Goal: to reduce the number of devices over complementary CMOS

….and gets rid of (almost) the PMOS devices….

Ratioed Logic: Resistive Load

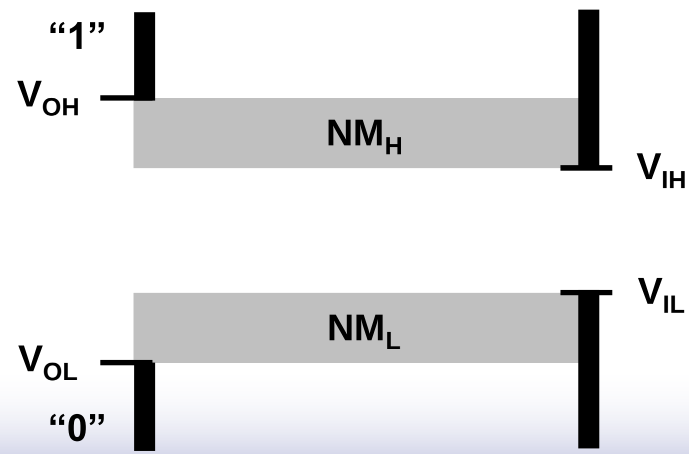


- N transistors + Load
- $V_{OH} = V_{DD}$
- $V_{OL} = \dfrac{R_{PN}}{R_{PN} + R_L}$
- Assymetrical response
- Static power consumption
- $t_{pL} = 0.69\ R_L C_L$

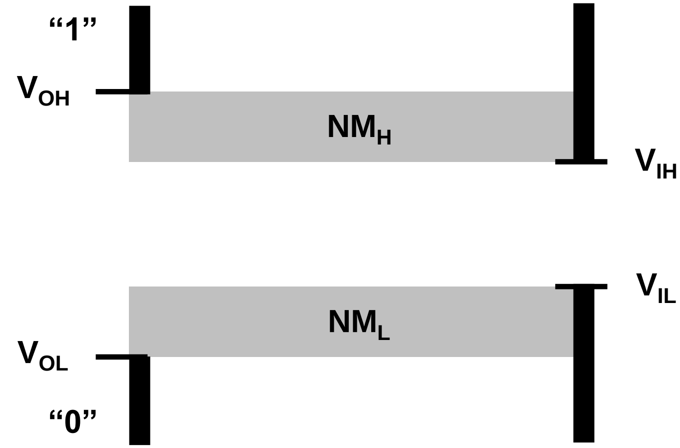
Recall a voltage divider circuit….

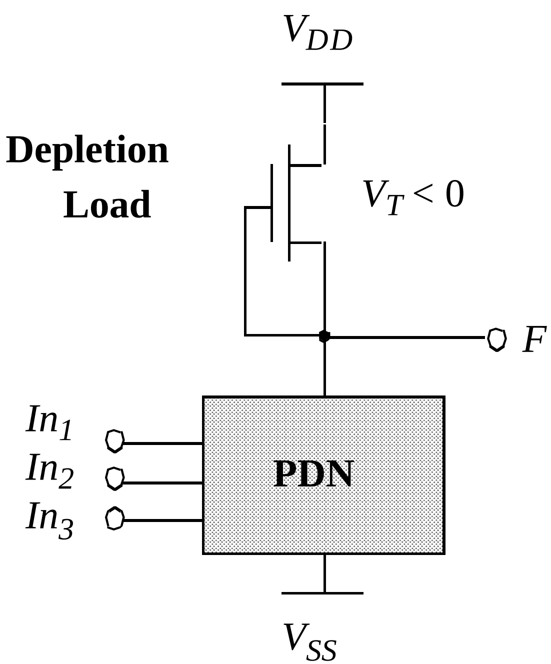
Ideally V_{OL} should be as small as possible. Hence, R_L should be large…

Only R_L involved in t_{plh}, while both R_L and R_{PN} involved in t_{phl}….

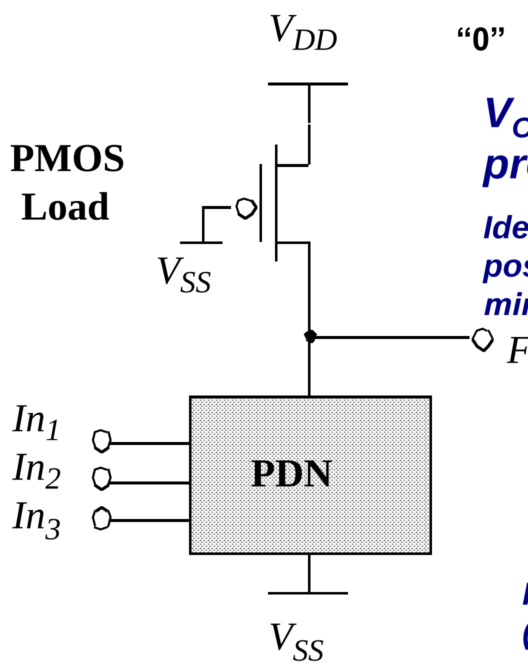


Active Loads





depletion load NMOS



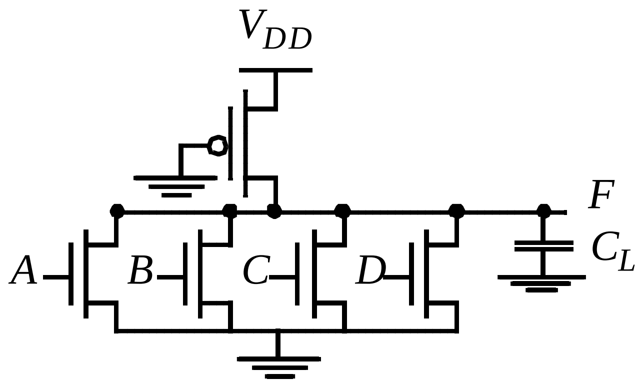
pseudo-NMOS

$V_{OH}=V_{DD}$ (assuming V_{OL} from previous stage $<V_{tn}$)

Ideally V_{OL} should be as small as possible. Hence, PMOS device should be minimum sized…

However, since V_{OL} is not 0 V, (since PUN is always ON) contention between PMOS and the PDN lowers the NM and results in static power dissipation.

Pseudo-NMOS



$V_{OH} = V_{DD}$ (similar to complementary CMOS)

To Find V_{OL}:

$$k_n\left((V_{DD} - V_{Tn})V_{OL} - \frac{V_{OL}^2}{2}\right) + k_p\left((-V_{DD}-V_{Tp})\, V_{DSATp} - V^2_{DSATp}/2\right) = 0$$

Note: NMOS in linear mode, since ideally the output=0 V (V_{ds}=V_{OL} < V_{gs}-V_{tn})

Note: PMOS in saturation mode

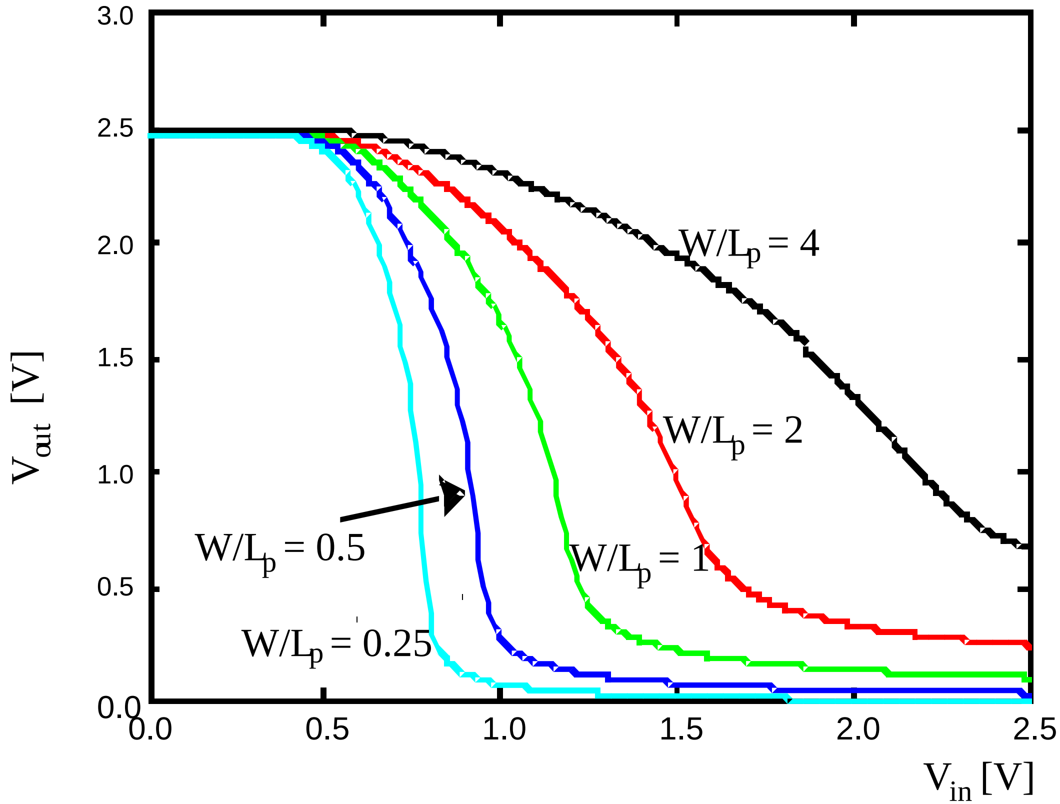
$$V_{OL} = \mu_p/\mu_n\, W_p/W_n\, V_{DSATp}$$

Assuming V_{OL} is small relative to gate drive, (V_{DD}-V_T), and V_{Tp}=V_{Tn}

SMALLER AREA & LOAD <u>BUT</u> STATIC POWER DISSIPATION!!!

Pseudo-NMOS VTC

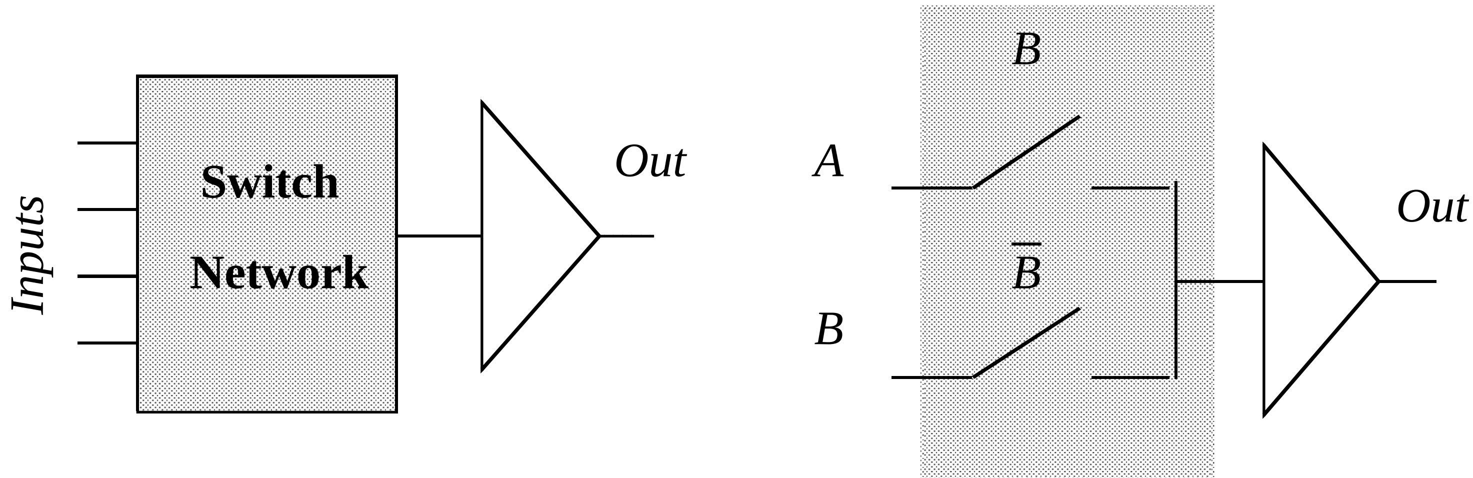
Sizing of the load device can be used to trade off parameters such as NM, delay, and power……



$$V_{OL} = \frac{R_{PN}}{R_{PN} + R_L}$$

A larger pull-up device improves performance but increases static power and degrades NM by increasing V_{OL}

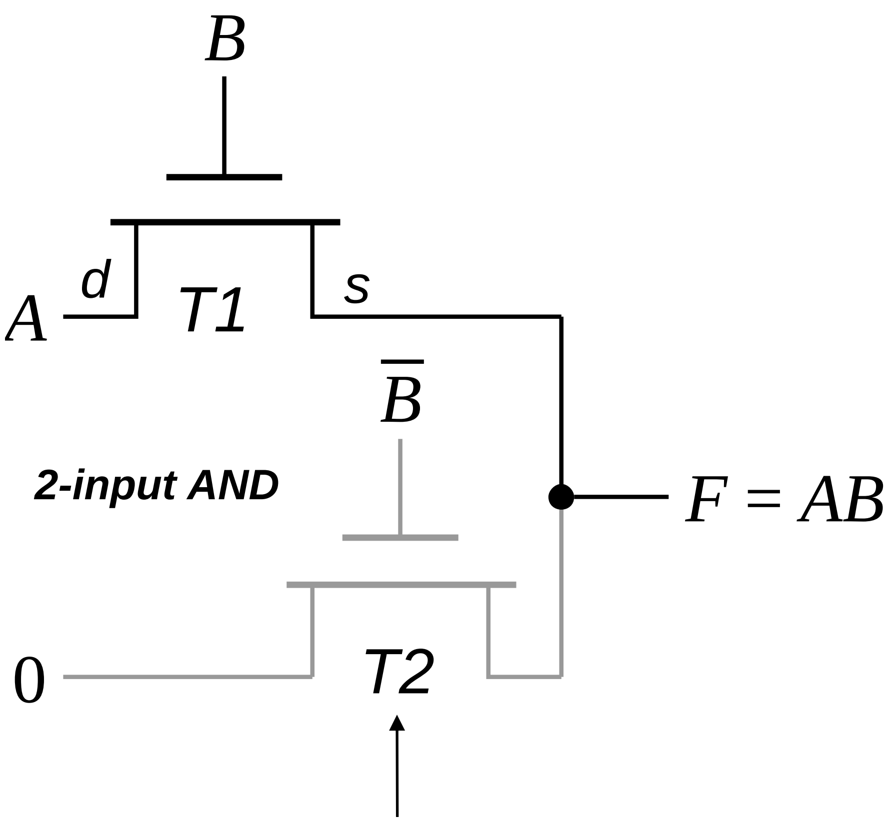
Pass-Transistor Logic



- **N transistors**
- **No static consumption**

Allows primary inputs to drive gate terminals as well as source-drain terminals

Example: AND Gate



2-input AND

Ensures that gate is static: provides low impedance path when B=0

If B=1 *then T1 is ON and T2 is OFF*

Then A=F*, i.e., if A=1, F=1 and if A=0, F=0*

When B=0*, T2 is ON and passes a Zero*

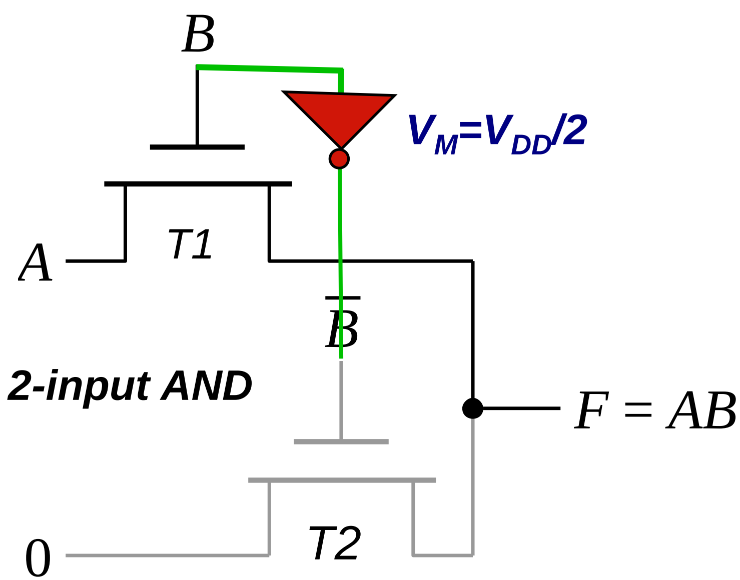
Need fewer transistors*: 4 to implement the AND: lower cap.*

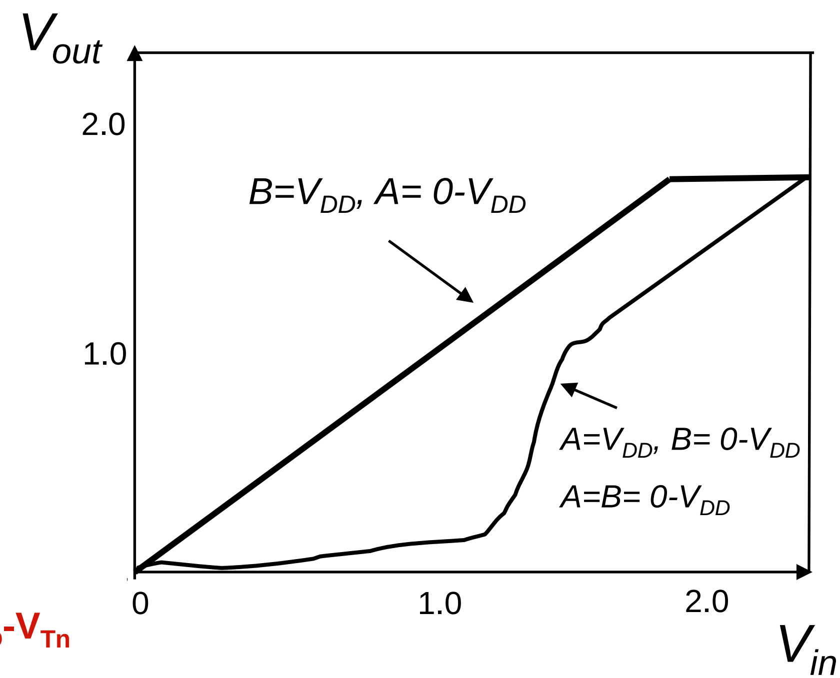
Need 6 to implement in static CMOS (4 for NAND and 2 for INV)

Note: F will charge only up to V_{DD}- V_{tn}

Also, V_{Tn} will be a function of V_F (increase due to RBB)

VTC of Pass-Transistor AND Gate





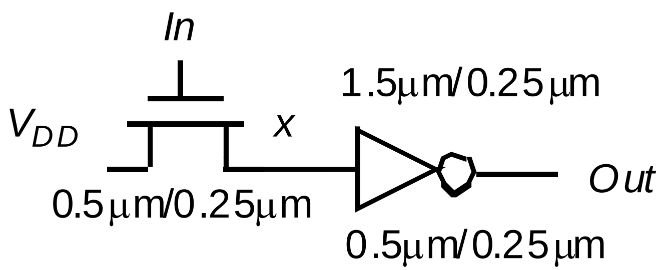
When $B=V_{DD}$, T1 is ON until the input reaches $V_{DD}-V_{Tn}$

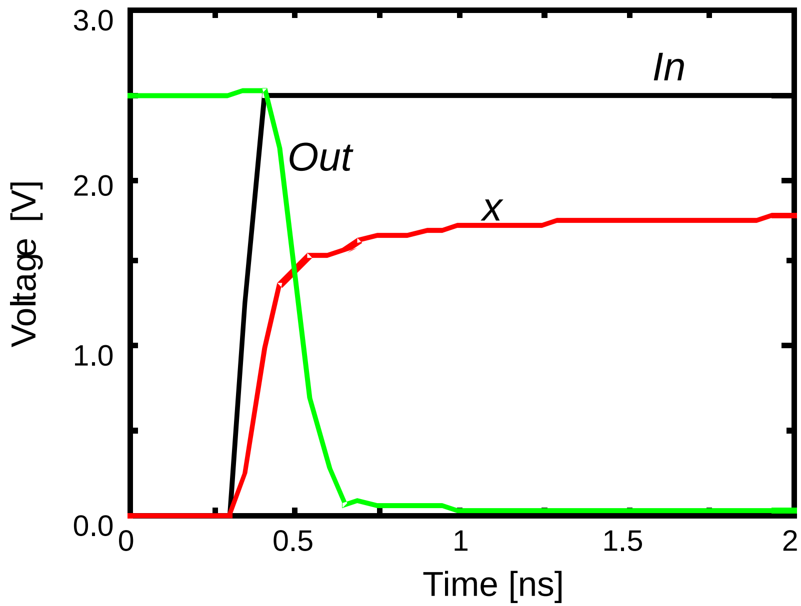
When $A=V_{DD}$, and B makes a transition from 0 to 1, T2 is turned on until $V_{DD}/2$ and Output =0. Once T2 is turned off, output follows the input B minus a threshold drop.

VTC of Pass Transistor Logic is data dependent

NMOS-Only Logic

***Voltage Drop Issue*:**

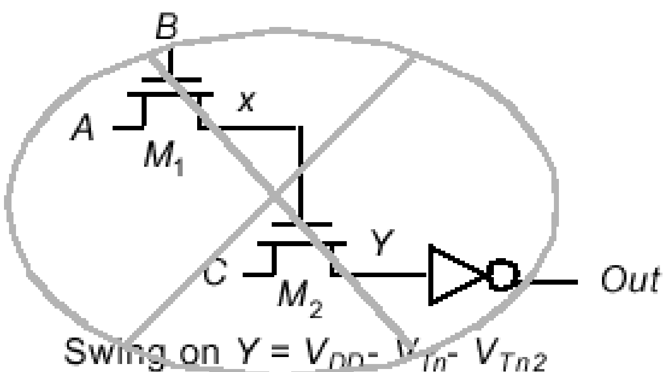




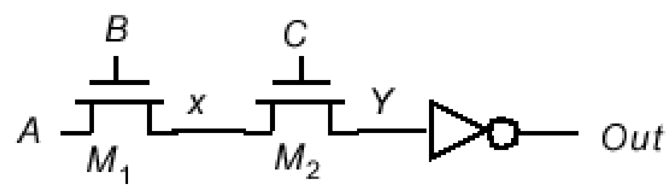
$$V_x = V_{dd} - V_{Tn}(V_x)$$

Hence, pass transistor gates cannot be cascaded by connecting the output of a pass gate to the gate input of another pass transistor. They can only be cascaded in series….

Cascading Pass Transistors

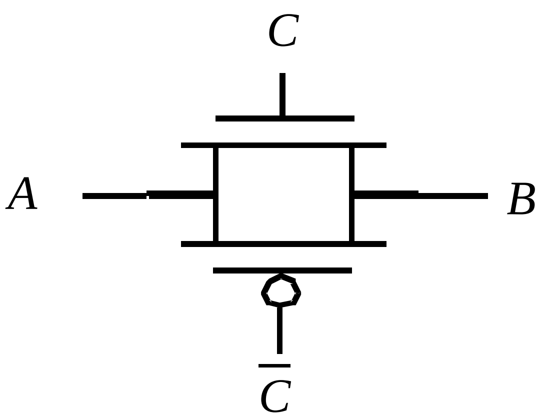


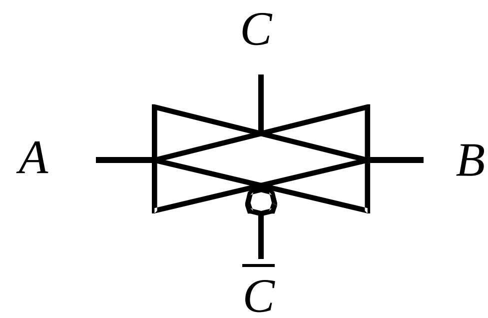
Swing on $Y = V_{DD} - V_{Tn} - V_{Tn2}$
(a)



Swing on $Y = V_{DD} - V_{Tn1}$
(b)

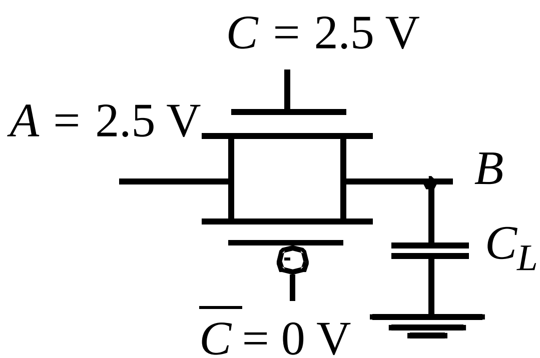
Transmission Gate





Acts like a bidirectional switch controlled by the gate signal C

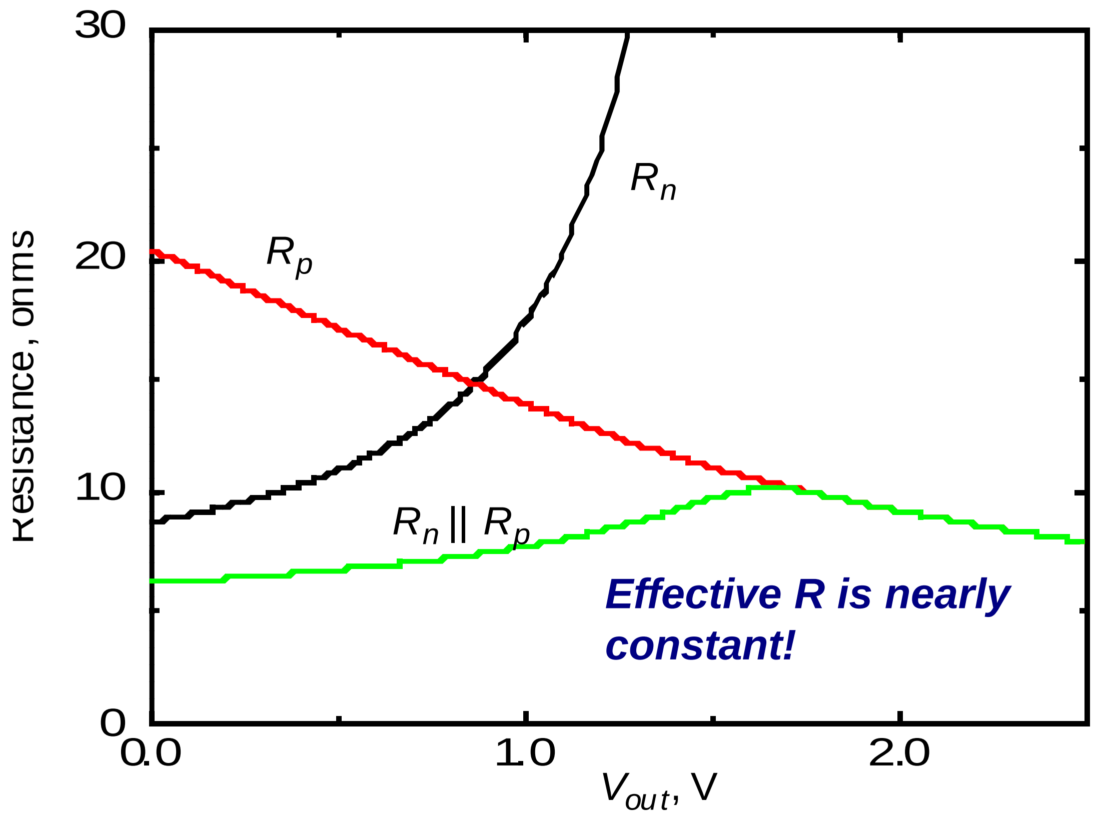
When C=1, both MOSFETS are ON allowing the signal to pass through the gate (A=B, if C=1)

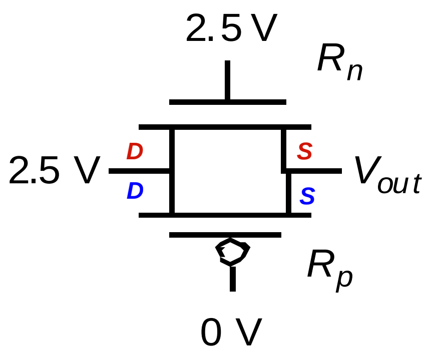


Because of the PMOS, C_L charges to Vdd

Because of NMOS C_L discharges to 0

Resistance of Transmission Gate





$$R_n = (V_{DD}-V_{out})/I_{dn}$$

$$R_p = (V_{out}-V_{DD})/I_{dp}$$

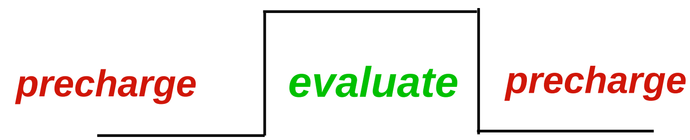
When Vout is low, NMOS is working, hence Rn dominates the equivalent resistance….similarly Rp dominates when Vout is high…..

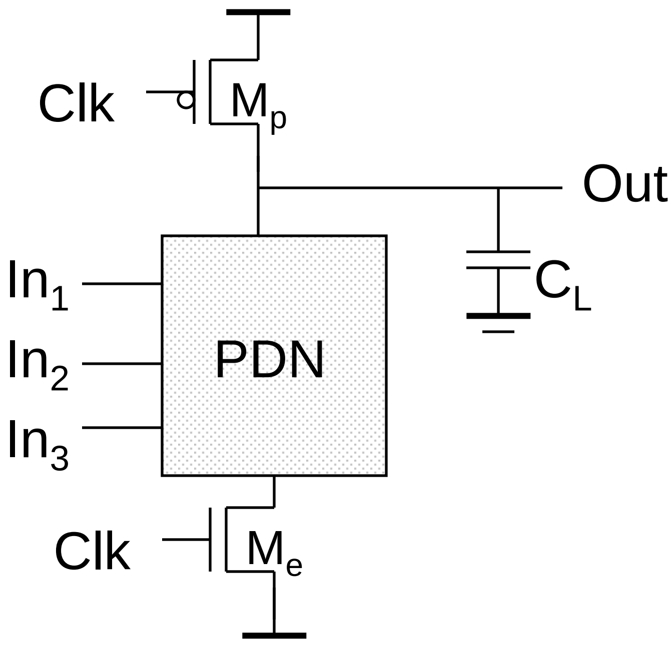
Dynamic CMOS

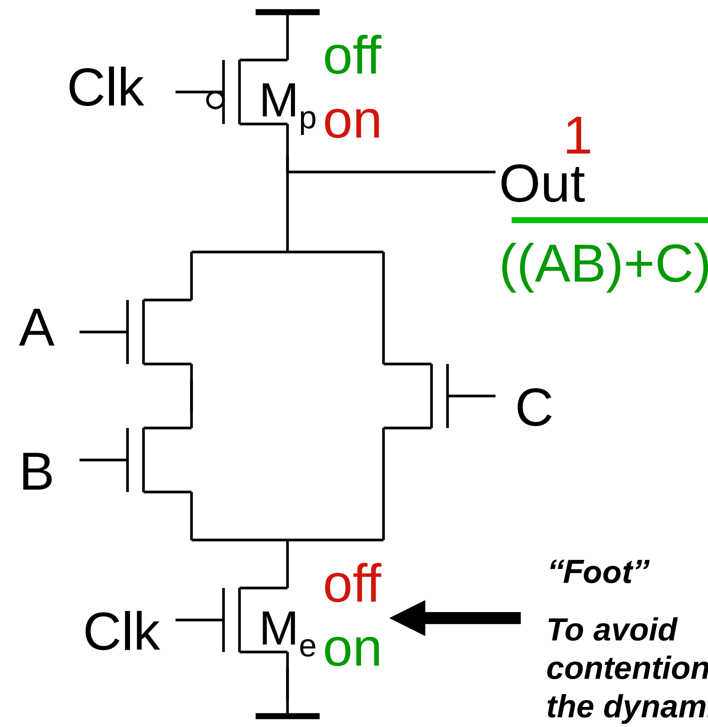
- In static circuits at every point in time (except when switching) the output is connected to either GND or V_{DD} via a low resistance path.
  - fan-in of n requires $2n$ (n N-type + n P-type) devices (for static CMOS)

- Dynamic circuits rely on the temporary storage of signal values on the capacitance of high impedance nodes.
  - requires only $n + 2$ (n+1 N-type and 1 P-type) transistors

Dynamic Gate







"Foot"

To avoid contention at the dynamic node (Out)….

Two phase operation

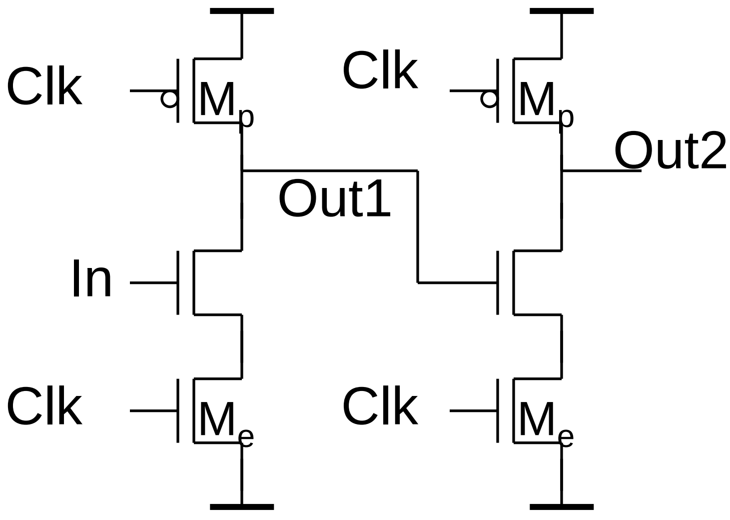
Precharge (Clk = 0)

Evaluate (Clk = 1)

$$Out = \overline{CLK} + \overline{(AB)+C} \,.\, CLK$$

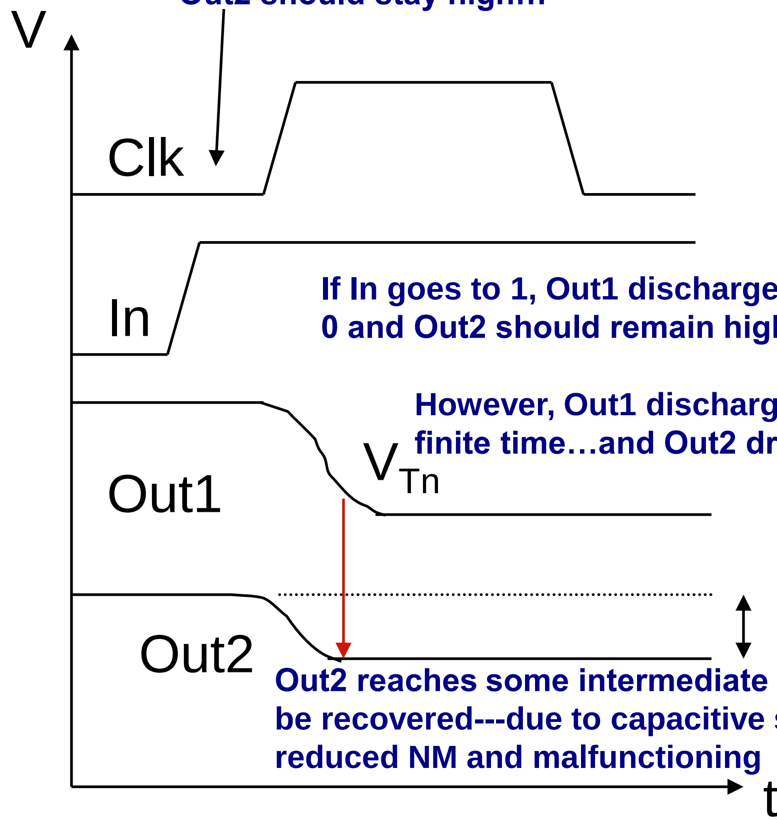
Cascading Dynamic Gates

Simple cascading doesn't work…



As long as Out1>V_M (~V_{Tn}) of the second inverter, Out2 will decrease leading to reduced NMs

During precharge both Out1 and Out2 should stay high…



If In goes to 1, Out1 discharges to 0 and Out2 should remain high…

However, Out1 discharges to 0 within a finite time…and Out2 drops…

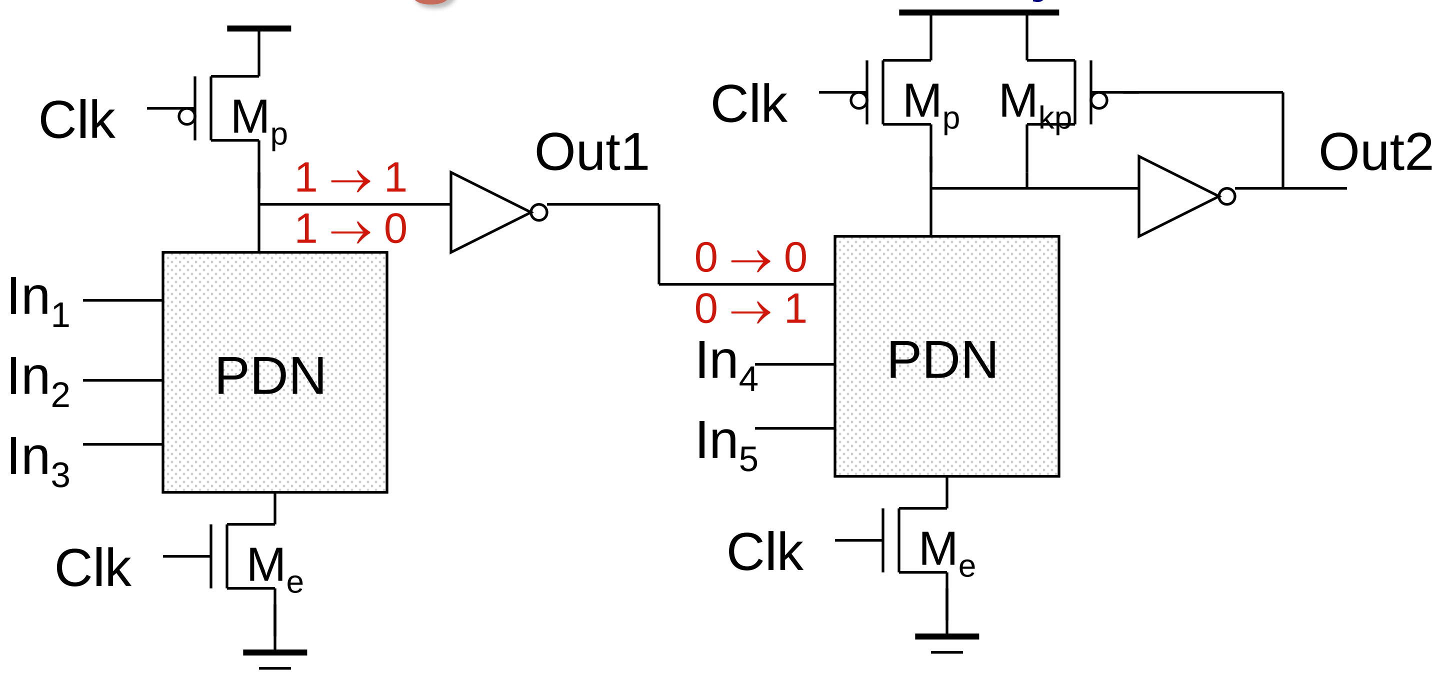
Out2 reaches some intermediate voltage…can't be recovered---due to capacitive storage---- reduced NM and malfunctioning

Solution: Set all inputs to 0 during precharge

For correct operation only 0 $\rightarrow$ 1 transitions should be allowed at inputs!

Domino Logic

An n-type dynamic logic followed by a static inverter…



All inputs (are outputs of other Domino gates) are set to 0 at the end of precharge phase

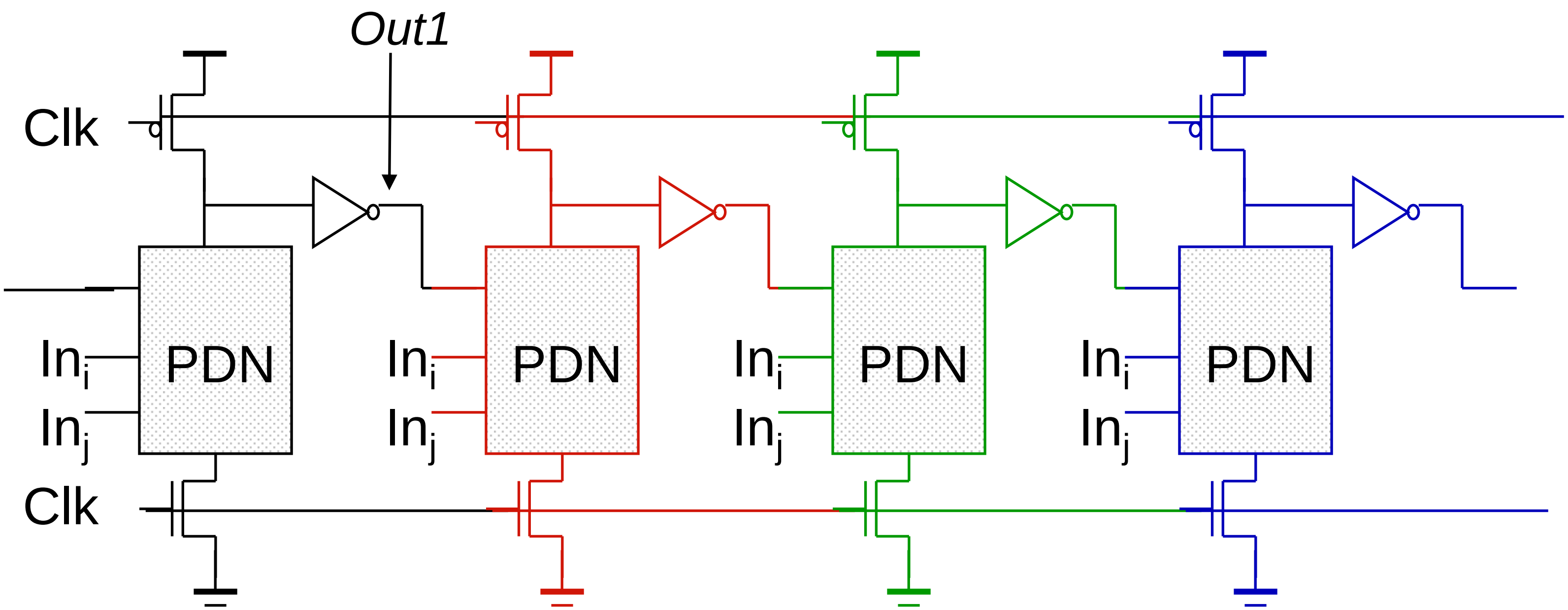
Only 0 to 1 transition at the inputs during evaluation phase: during evaluation, dynamic gate conditionally discharges and the output of the inverter makes a conditional transition from 0 to 1.

The static inverter reduces the capacitance of the dynamic output node by separating internal and load capacitances…..it also increases the NM (due to the low-impedance output)

The inverter can also be used to drive a keeper device to combat leakage and charge redistribution.

Why Domino?

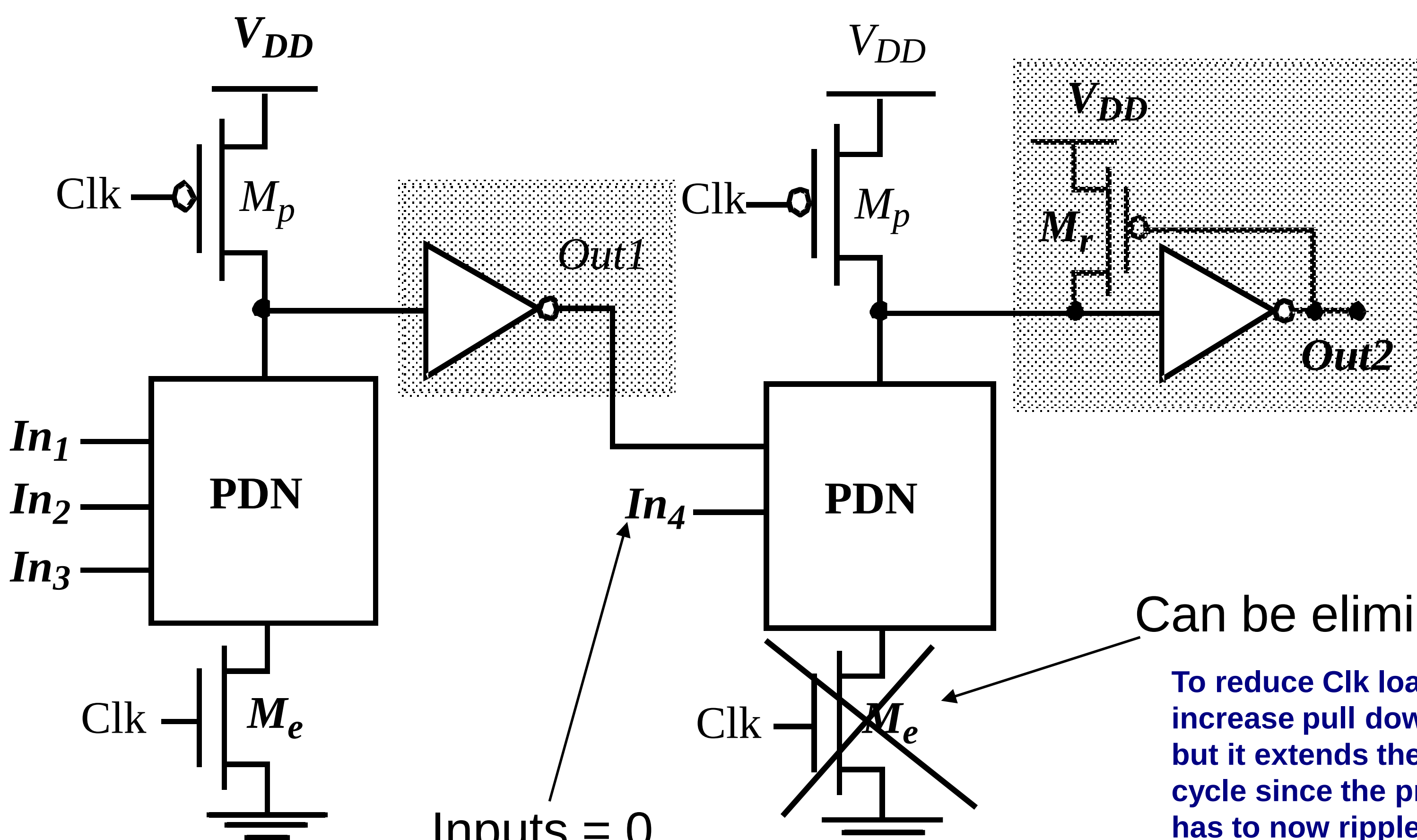
A Domino chain



Precharge: all inputs=0

Evaluation: Output of domino1 either stays at 0 or makes a transition from 0 to 1, affecting the second gate. This effect might ripple through the whole chain…***like a line of falling dominos!***

Designing with Domino Logic



Can be eliminated?

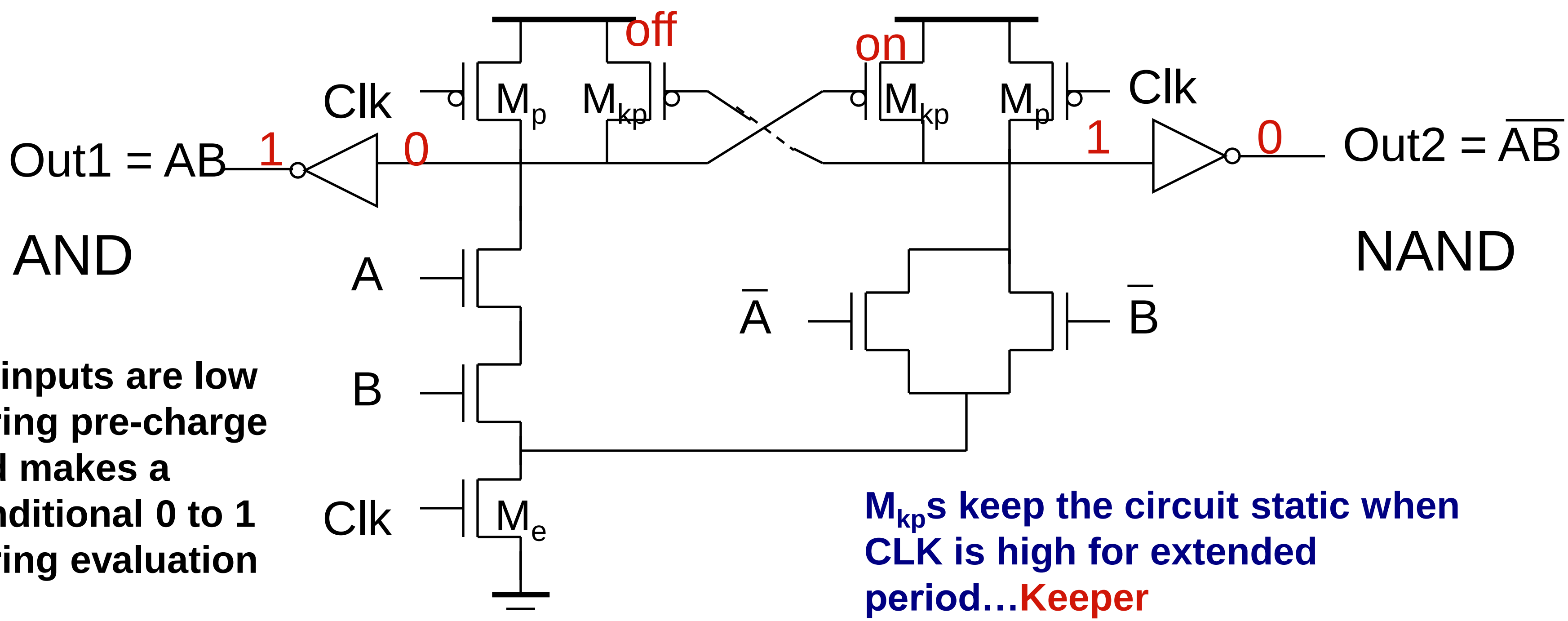
To reduce Clk load and increase pull down drive--- but it extends the precharge cycle since the precharge has to now ripple through the logic network as well.

Inputs = 0 during precharge

Differential (Dual Rail) Domino

Overcomes the non-inverting property of Domino Logic: used commercially in several microprocessors

Uses a pre-charged load….



All inputs are low during pre-charge and makes a conditional 0 to 1 during evaluation

M_{kp}s keep the circuit static when CLK is high for extended period…Keeper

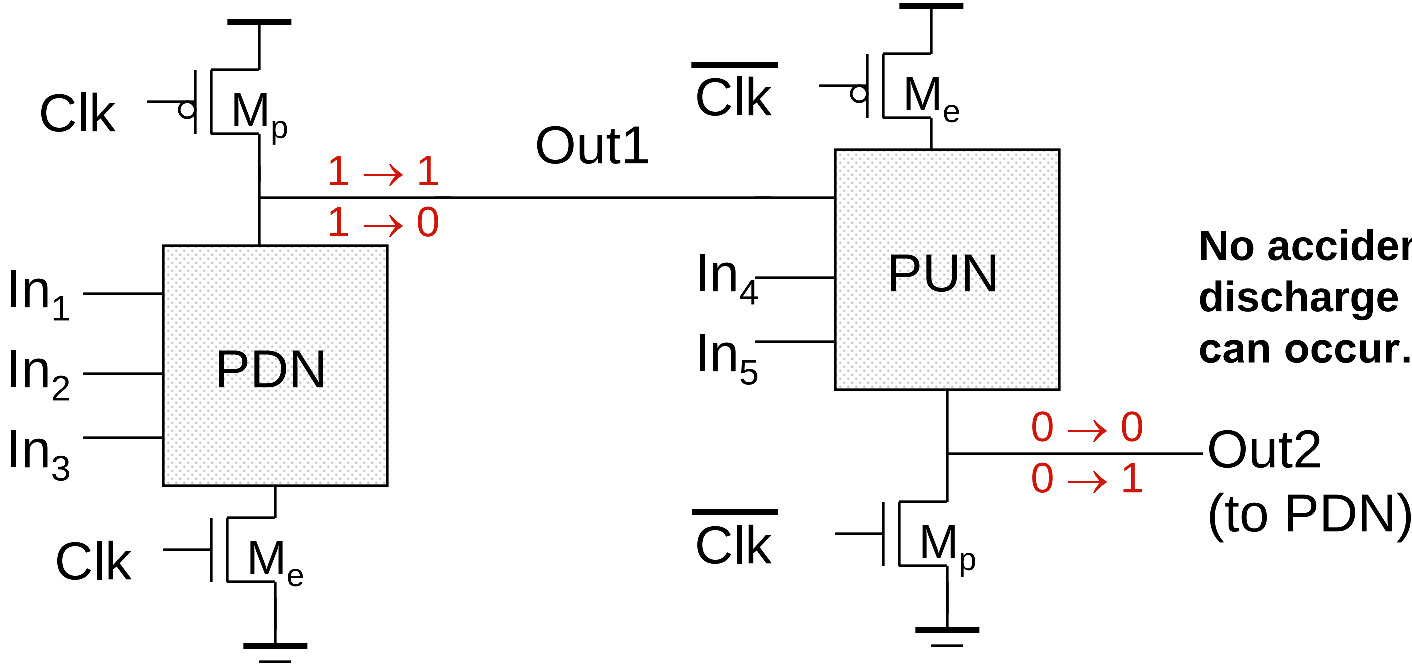
Possible to implement any arbitrary function….but comes at the expense of increased power since a transition is guaranteed every CLK cycle irrespective of the input values….either Out1 or Out2 must make a 0 to 1 transition.

np-CMOS

Alternative to cascading dynamic gates….uses n-type and p-type dynamic logic

Exploits duality between n-tree and p-tree logic gates to eliminate cascading problem

No extra inverter at the outputs….unless output of n-tree (p-tree) needs to be connected to another n-tree (p-tree) gates



No accidental discharge of Out2 can occur….

Only $0 \rightarrow 1$ transitions allowed at inputs of PDN

Only $1 \rightarrow 0$ transitions allowed at inputs of PUN

Drawback: p-tree gates are slower than the n-tree gates…needs proper skewing of PMOS….area penalty

No buffers---so dynamic nodes need to be routed between gates