CHAPTER 5

Combinational Circuit Design

Outline

1. **Static CMOS**
2. Ratioed Circuits
3. Cascode Voltage Switch Logic
4. Dynamic Circuits
5. Pass-Transistor Circuits
6. Circuit Pitfalls

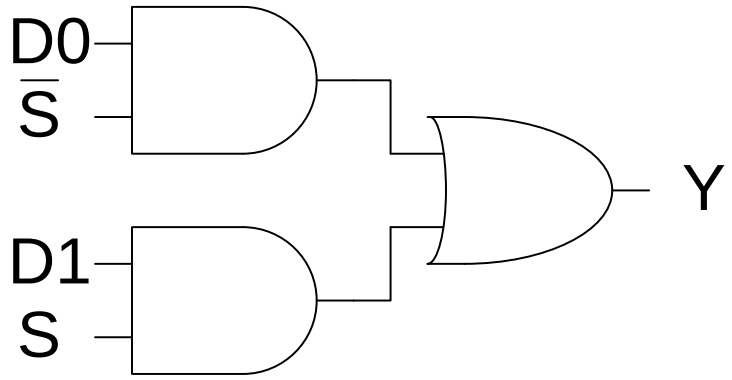
Static CMOS

- Bubble Pushing
- Compound Gates
- Logical Effort Example
- Input Ordering
- Asymmetric Gates
- Skewed Gates
- Best P/N ratio

Example 1

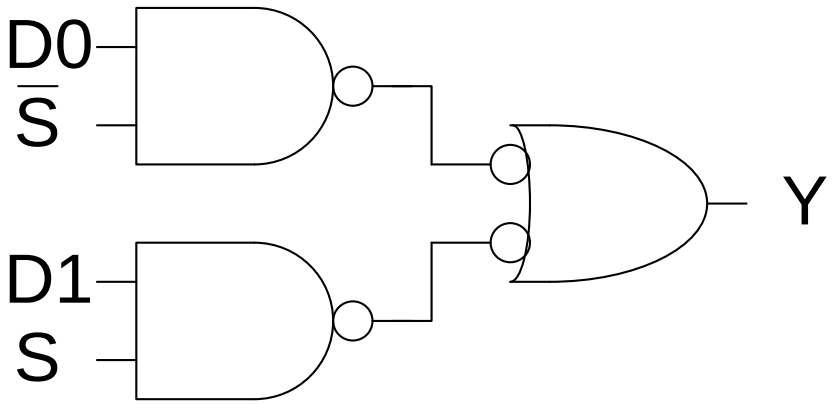

```
module mux(input  s, d0, d1,
           output y);

   assign y = s ? d1 : d0;
endmodule
```

1) Sketch a design using AND, OR, and NOT gates.



Example 2

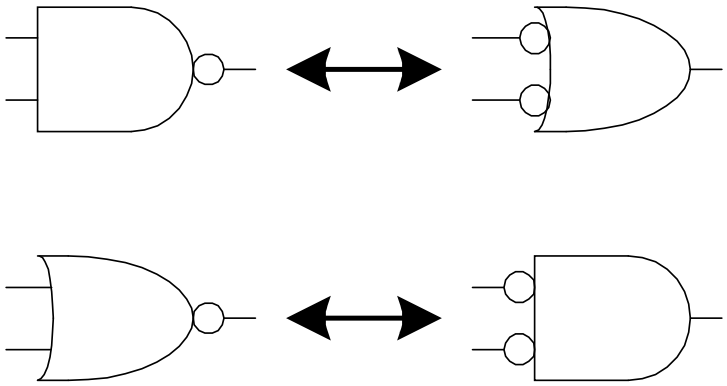
2) Sketch a design using NAND, NOR, and NOT gates. Assume ~S is available.



Bubble Pushing

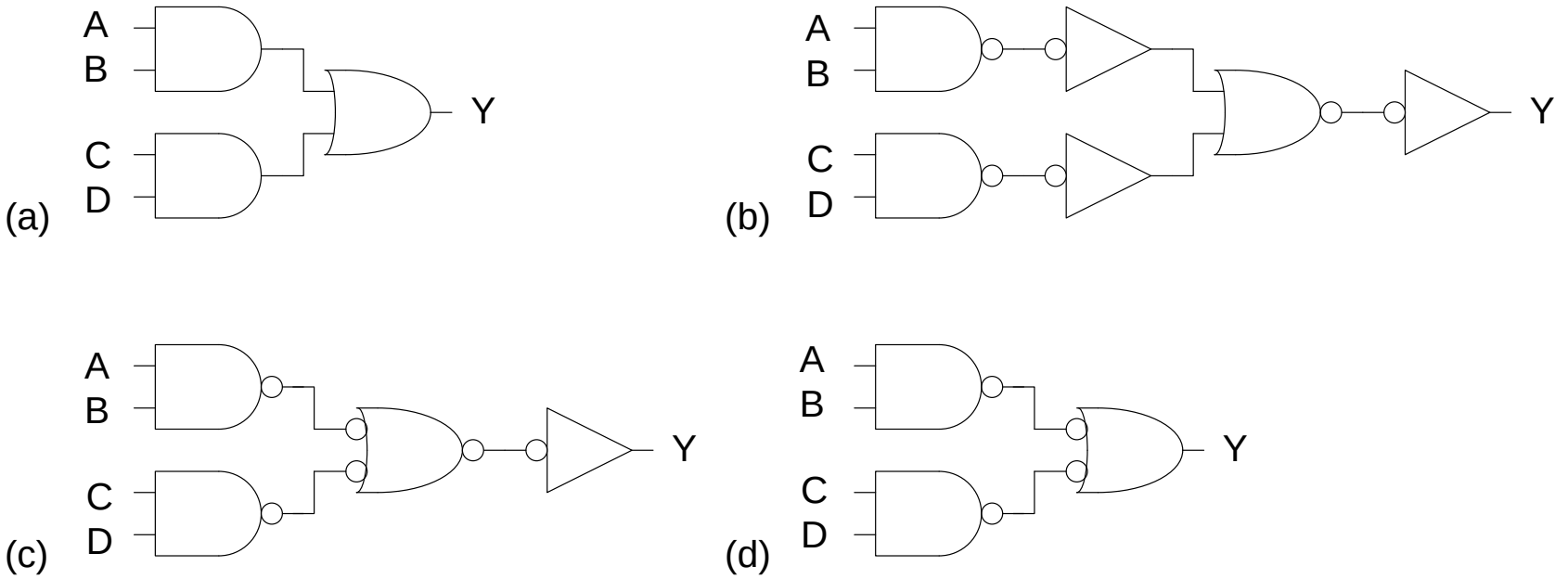
- Start with network of AND / OR gates
- Convert to NAND / NOR + inverters
- Push bubbles around to simplify logic
  - Remember DeMorgan's Law

$$\overline{A \cdot B} = \overline{A} + \overline{B} \qquad \overline{A + B} = \overline{A} \cdot \overline{B}$$



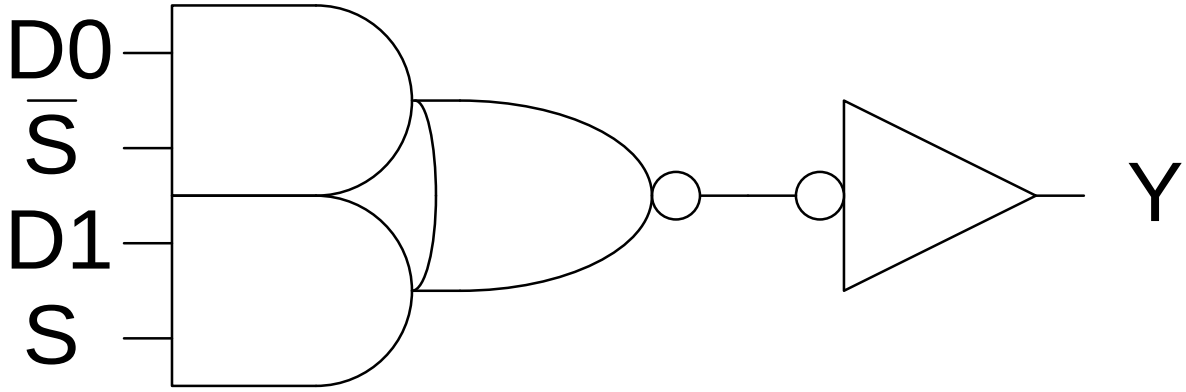
Bubble Pushing

- Y = AB + CD



Example 3

3) Sketch a design using one compound gate and one NOT gate. Assume ~S is available.

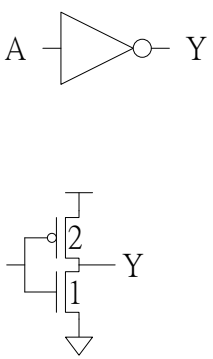


Compound Gates

- Logical Effort of compound gates

unit inverter

$$Y = \overline{A}$$

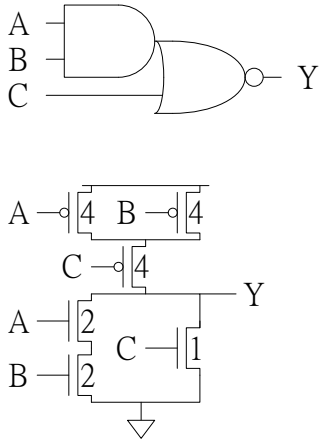


$g_A = 3/3$

$p = 3/3$

AOI21

$$Y = \overline{A \bullet B + C}$$



$g_A = 6/3$

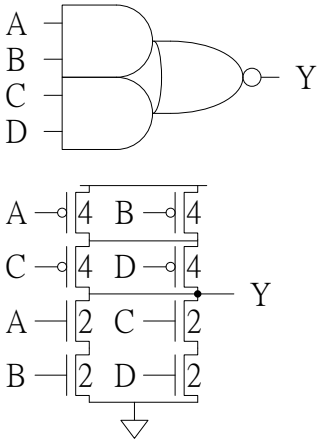
$g_B = 6/3$

$g_C = 5/3$

$p = 7/3$

AOI22

$$Y = \overline{A \bullet B + C \bullet D}$$



$g_A = 6/3$

$g_B = 6/3$

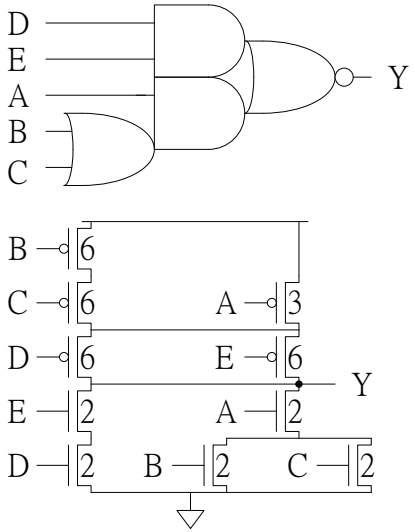
$g_C = 6/3$

$g_D = 6/3$

$p = 12/3$

Complex AOI

$$Y = \overline{A \bullet (B + C) + D \bullet E}$$



$g_A = 5/3$

$g_B = 8/3$

$g_C = 8/3$

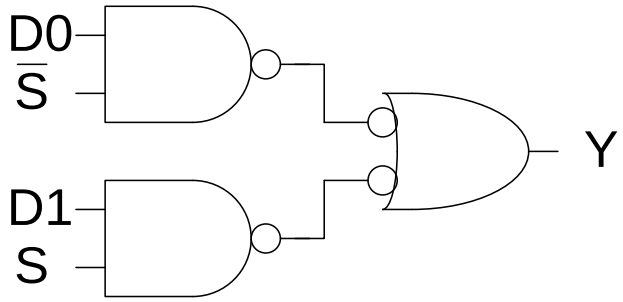
$g_D = 8/3$

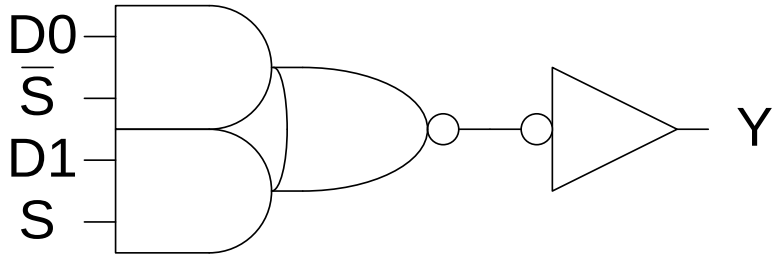
$g_E = 8/3$

$p = 16/3$

Example 4

- The multiplexer has a maximum input capacitance of 16 units on each input. It must drive a load of 160 units. Estimate the delay of the NAND and compound gate designs.





H = 160 / 16 = 10
B = 1
N = 2

NAND Solution

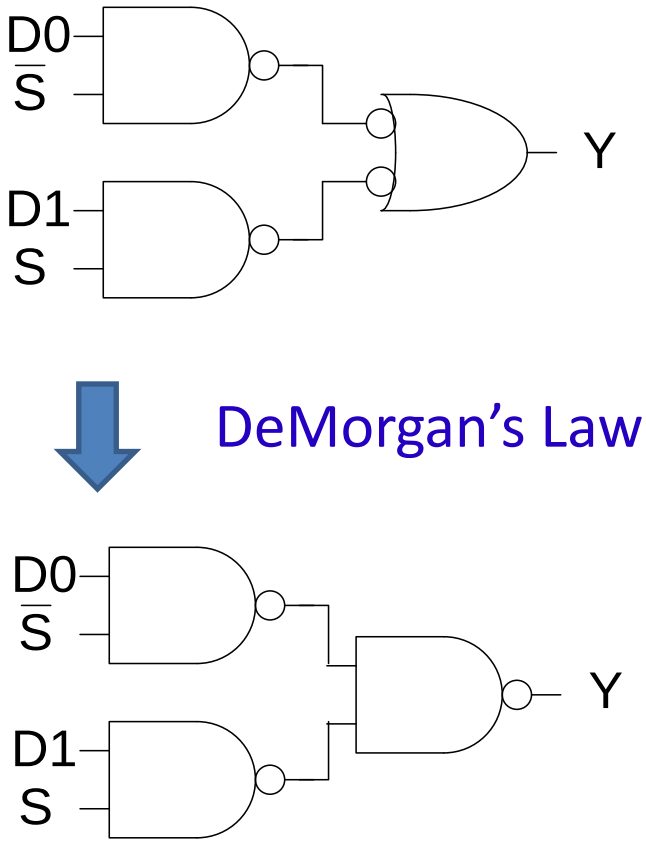
$$P = 2 + 2 = 4$$

$$G = (4/3) \bullet (4/3) = 16/9$$

$$F = GBH = 160/9$$

$$\hat{f} = \sqrt[N]{F} = 4.2$$

$$D = N\hat{f} + P = 12.4\tau$$



Compound Solution

$$P = 4 + 1 = 5$$

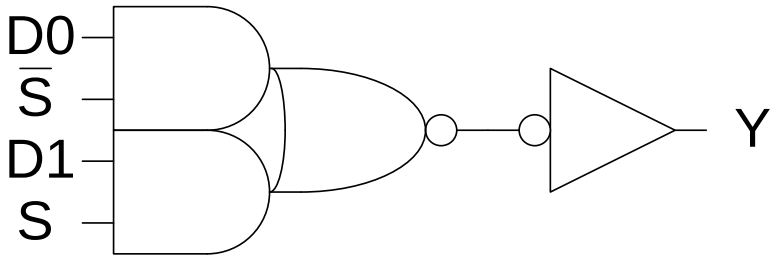
$$G = (6/3) \bullet (1) = 2$$

$$F = GBH = 20$$

$$\hat{f} = \sqrt[N]{F} = 4.5$$

$$D = N\hat{f} + P = 14\tau$$

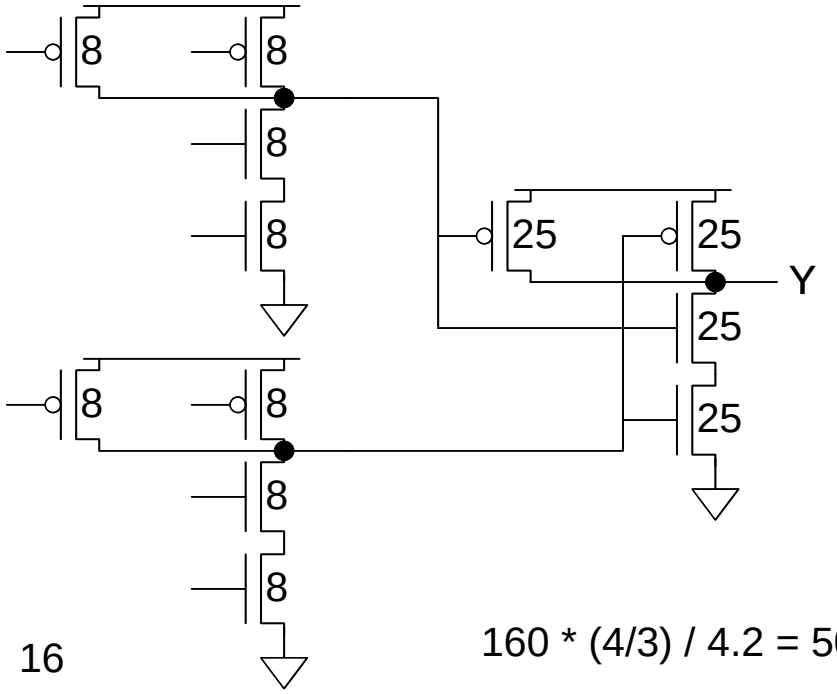
AOI22 + INV



Example 5

- Annotate your designs with transistor sizes that achieve this delay.

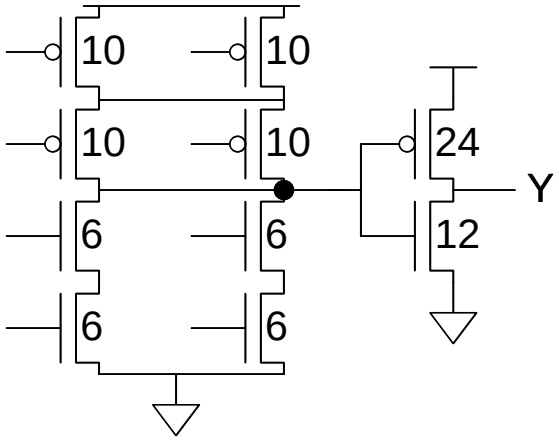
NAND solution



16

160 * (4/3) / 4.2 = 50

Compound solution

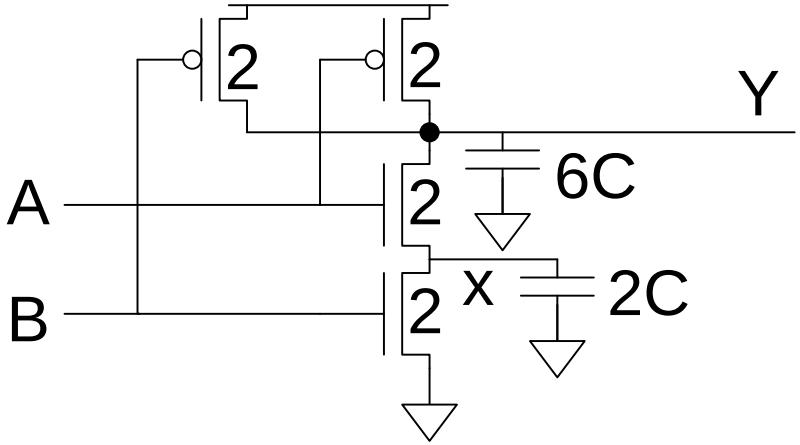


16

160 * 1 / 4.5 = 36

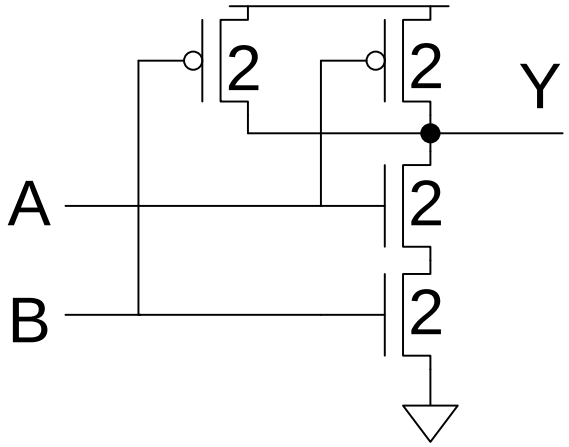
Input Order

- Our parasitic delay model was too simple
  - Calculate parasitic delay for Y falling
    - If A arrives latest? 2τ
    - If B arrives latest? 2.33τ



Inner & Outer Inputs

- *Outer* input is closest to rail (B)
- *Inner* input is closest to output (A)



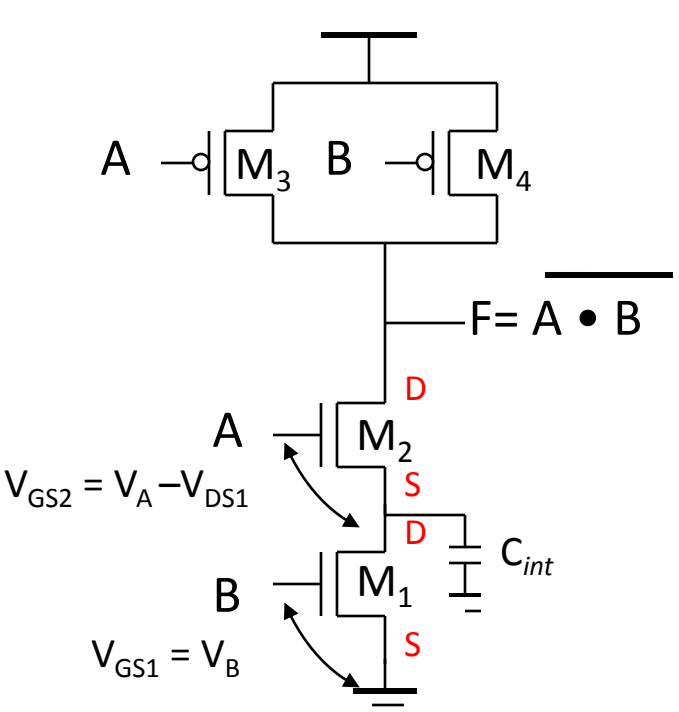
- If input arrival time is known
  - Connect latest input to inner terminal

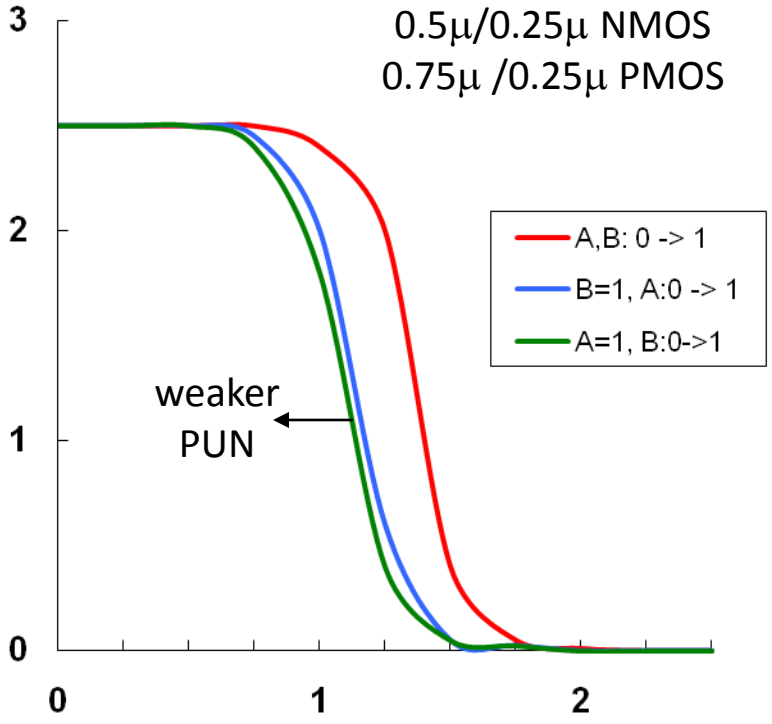
VTC is Data-Dependent

- The threshold voltage of M_2 is higher than M_1 due to body effect (γ)

$$V_{Tn1} = V_{Tn0} \qquad V_{Tn2} = V_{Tn0} + \gamma(\sqrt{(|2\phi_F| + V_{int})} - \sqrt{|2\phi_F|})$$

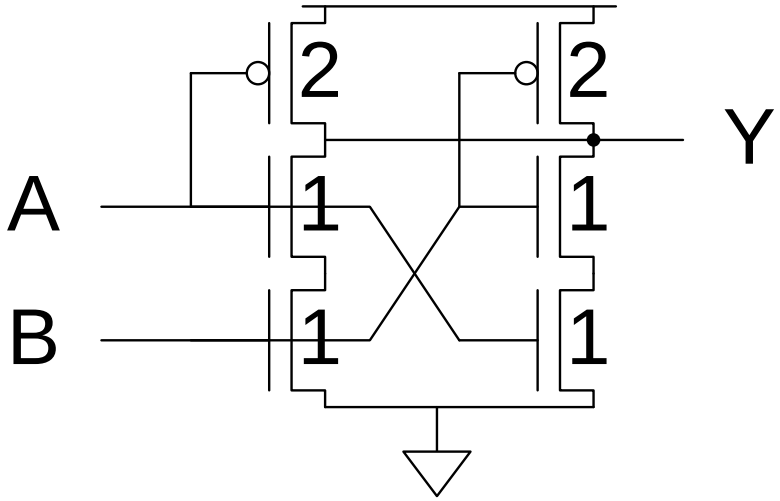
since V_{SB} of M_2 is not zero (when $V_B = 0$) due to the presence of C_{int}





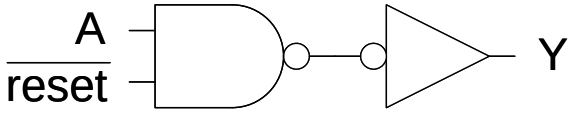
Symmetric Gates

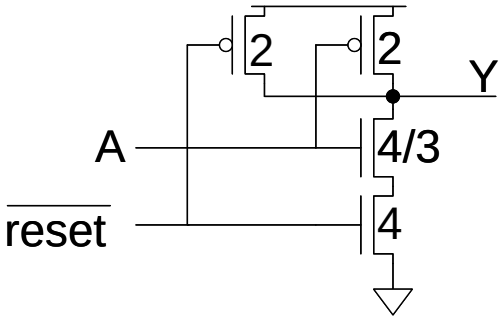
- Inputs can be made perfectly symmetric



Asymmetric Gates

- Asymmetric gates favor one input over another
- Ex: suppose input A of a NAND gate is most critical
  - Use smaller transistor on A (less capacitance)
  - Boost size of noncritical input
  - So total resistance is same
  - $g_A = 10/9$
  - $g_B = 2$
  - $g_{total} = g_A + g_B = 28/9$

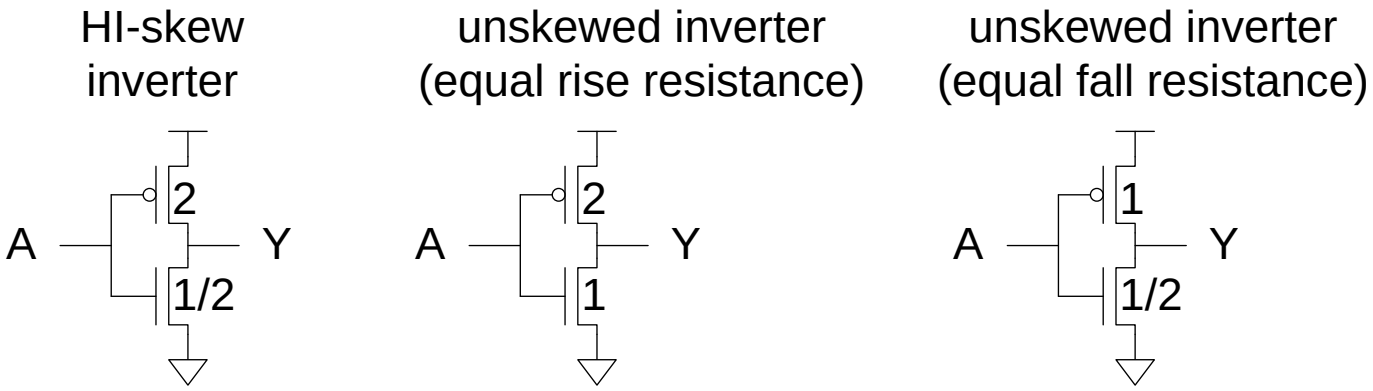




- Asymmetric gate approaches g = 1 on critical input
- But total logical effort goes up

Skewed Gates

- Skewed gates favor one edge over another
- Ex: suppose rising output of inverter is most critical
  - Downsize noncritical nMOS transistor

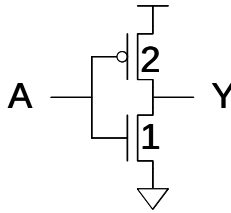
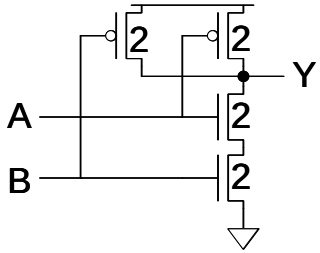
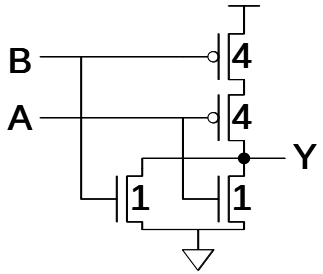
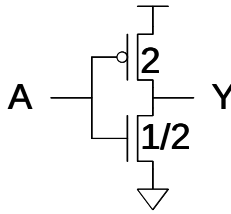
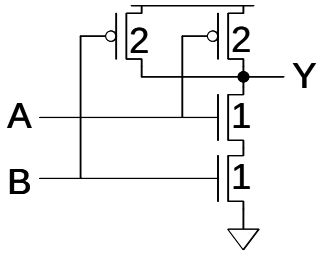
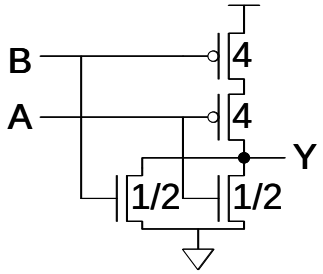
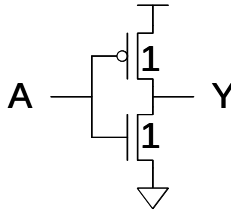
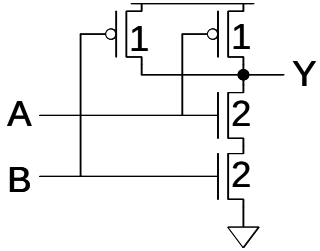
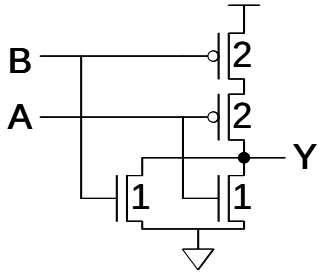


- Calculate logical effort by comparing to unskewed inverter with same effective R on that edge.
  - $g_u = 2.5 / 3 = 5/6$
  - $g_d = 2.5 / 1.5 = 5/3$

HI- and LO-Skew

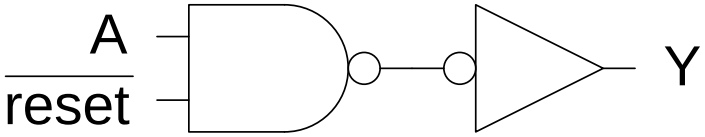
- Def: Logical effort of a skewed gate for a particular transition is the ratio of the input capacitance of that gate to the input capacitance of an unskewed inverter delivering the same output current for the same transition.
- Skewed gates reduce size of noncritical transistors
  - HI-skew gates favor rising output (small nMOS)
  - LO-skew gates favor falling output (small pMOS)
- Logical effort is smaller for favored direction
- But larger for the other direction

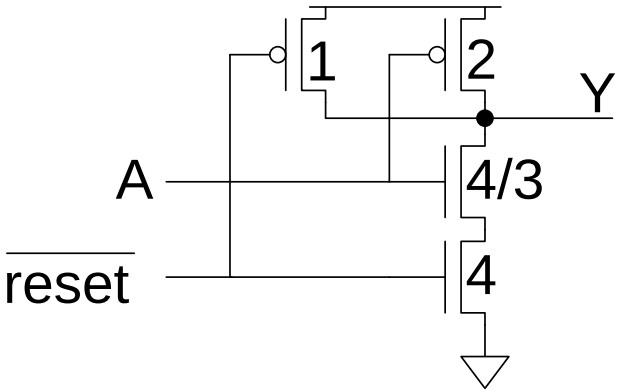
Catalog of Skewed Gates

	Inverter	NAND2	NOR2
unskewed	 $g_u = 1$ $g_d = 1$ $g_{avg} = 1$	 $g_u = 4/3$ $g_d = 4/3$ $g_{avg} = 4/3$	 $g_u = 5/3$ $g_d = 5/3$ $g_{avg} = 5/3$
HI-skew	 $g_u = 5/6$ $g_d = 5/3$ $g_{avg} = 5/4$	 $g_u = 1$ $g_d = 2$ $g_{avg} = 3/2$	 $g_u = 3/2$ $g_d = 3$ $g_{avg} = 9/4$
LO-skew	 $g_u = 4/3$ $g_d = 2/3$ $g_{avg} = 1$	 $g_u = 2$ $g_d = 1$ $g_{avg} = 3/2$	 $g_u = 2$ $g_d = 1$ $g_{avg} = 3/2$

Asymmetric Skew

- Combine asymmetric and skewed gates
  - Downsize noncritical transistor on unimportant input
  - Reduces parasitic delay for critical input

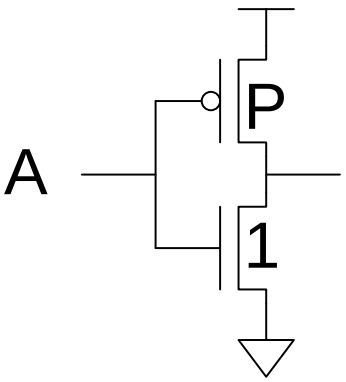




10/9	100/99	~1
10	100	∞

Best P/N Ratio

- We have selected P/N ratio for unit rise and fall resistance (μ = 2-3 for an inverter).
- Alternative: choose ratio for least average delay
- Ex: inverter
  - Delay driving identical inverter
  - $t_{pdf} = (P+1)$
  - $t_{pdr} = (P+1)(\mu/P)$
  - $t_{pd} = (P+1)(1+\mu/P)/2 = (P + 1 + \mu + \mu/P)/2$
  - Differentiate t_{pd} w.r.t. P
  - Least delay for P = $\sqrt{\mu}$

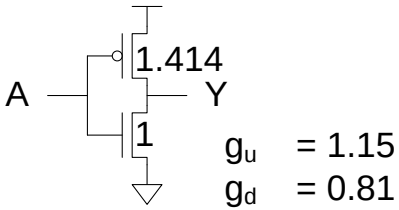


P/N Ratios

- In general, best P/N ratio is sqrt of that giving equal delay.
  - Only improves average delay slightly for inverters
  - But significantly decreases area and power

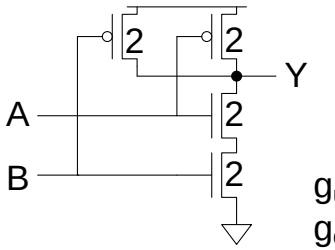
fastest P/N ratio

Inverter



g_u = 1.15
g_d = 0.81
g_{avg} = 0.98

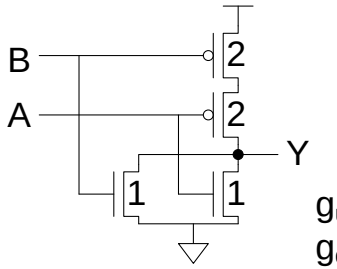
NAND2



?

g_u = 4/3
g_d = 4/3
g_{avg} = 4/3

NOR2



?

g_u = 2
g_d = 1
g_{avg} = 3/2

Observations

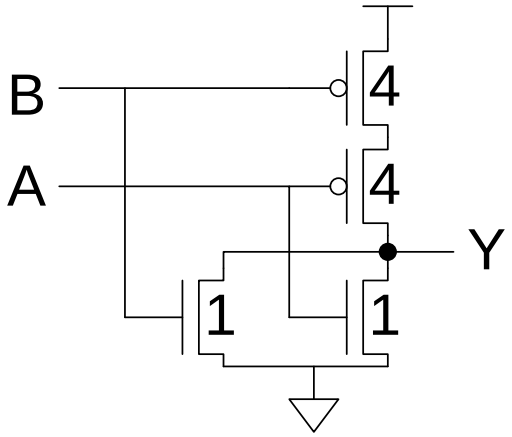
- For speed:
  - NAND vs. NOR
  - Many simple stages vs. fewer high fan-in stages
  - Latest-arriving input
- For area and power:
  - Many simple stages vs. fewer high fan-in stages

Outline

1. Static CMOS
2. **Ratioed Circuits**
3. Cascode Voltage Switch Logic
4. Dynamic Circuits
5. Pass-Transistor Circuits
6. Circuit Pitfalls

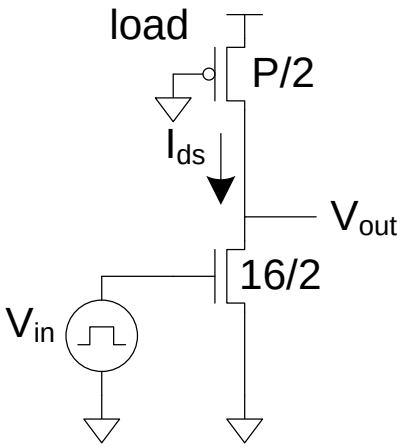
Introduction

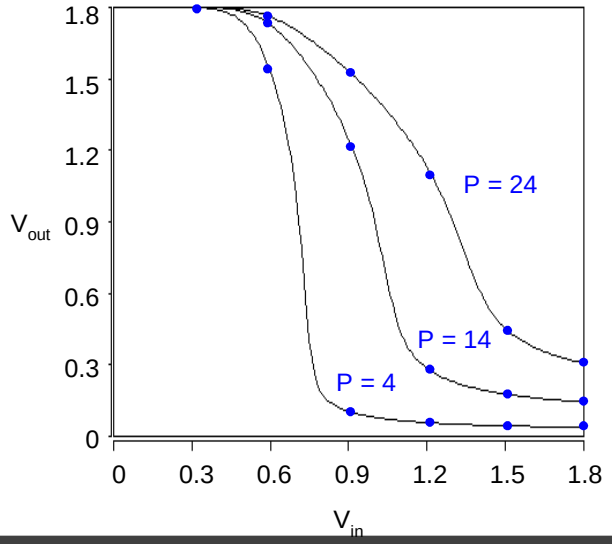
- What makes a circuit fast?
  - I = C dV/dt -> $t_{pd} \propto (C/I)\,\Delta V$
  - low capacitance
  - high current
  - small swing
- Logical effort is proportional to C
- pMOS are the enemy!
  - High capacitance for a given current
- Can we take the pMOS capacitance off the input?
- Various circuit families try to do this…



Pseudo-nMOS

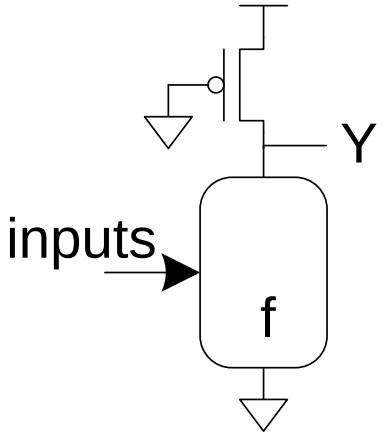
- In the old days, nMOS processes had no pMOS
  - Instead, use pull-up transistor that is always ON
- In CMOS, use a pMOS that is always ON
  - *Ratio* issue, Make pMOS about 1/3~1/6 effective strength of pulldown network



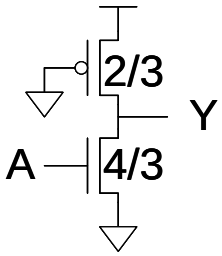


Pseudo-nMOS Gates

- Design for unit current on output to compare with unit inverter.
- Choose pMOS size between 1/3 ~ 1/6 the effective width (pick 1/3)

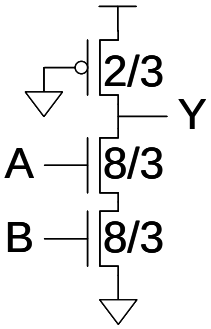


Inverter



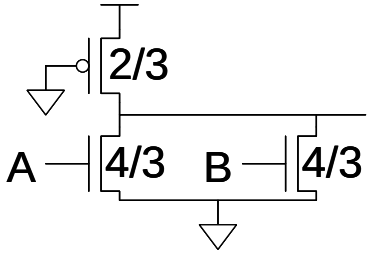
$g_u = 4/3$
$g_d = 4/9$
$g_{avg} = 8/9$
$p_u = 6/3$
$p_d = 6/9$
$p_{avg} = 12/9$

NAND2



$g_u = 8/3$
$g_d = 8/9$
$g_{avg} = 16/9$
$p_u = 10/3$
$p_d = 10/9$
$p_{avg} = 20/9$

NOR2



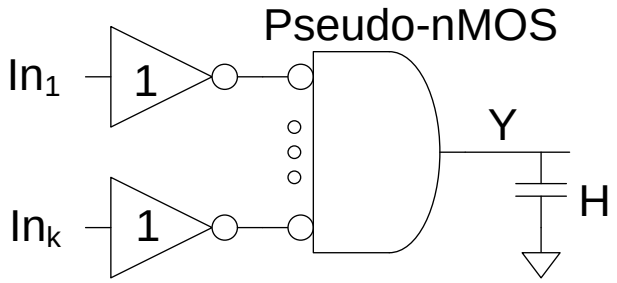
$g_u = 4/3$
$g_d = 4/9$
$g_{avg} = 8/9$
$p_u = 10/3$
$p_d = 10/9$
$p_{avg} = 20/9$

Inverter : $I_u = (1/3)I$, $C_{inv\text{-}u} = 1$, $g_u = (4/3)/1$, $p_u = (2/3+4/3)/1$

$I_d = I$, $C_{inv\text{-}d} = 3$, $g_d = (4/3)/3$, $p_d = (2/3+4/3)/3$

Pseudo-nMOS Design

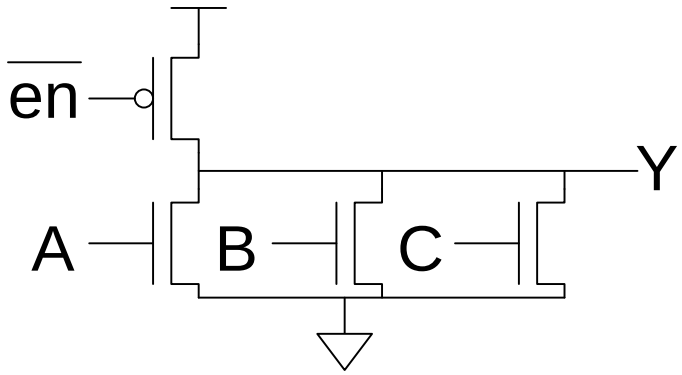
- Ex: Design a k-input AND gate using INV+ seudo-nMOS NOR. Find the delay driving a fanout of H



- G = 1 * 8/9 = 8/9
- F = GBH = 8H/9
- P = 1 + (4+8k)/9 = (8k+13)/9
- N = 2
- $D = NF^{1/N} + P = \frac{4\sqrt{2H}}{3} + \frac{8k+13}{9}$

Pseudo-nMOS Power

- Pseudo-nMOS draws power whenever Y = 0
  - Called static power $P = I \bullet V_{DD}$
  - A few mA / gate * 1M gates would be a problem
  - This is why nMOS went extinct!
- Use pseudo-nMOS sparingly for wide NORs
- Turn off pMOS when not in use

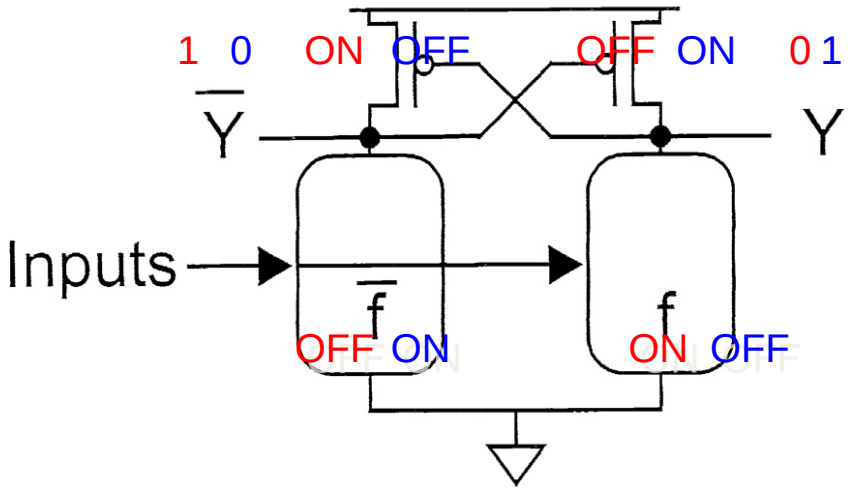


Outline

1. Static CMOS
2. Ratioed Circuits
3. **Cascode Voltage Switch Logic**
4. Dynamic Circuits
5. Pass-Transistor Circuits
6. Circuit Pitfalls

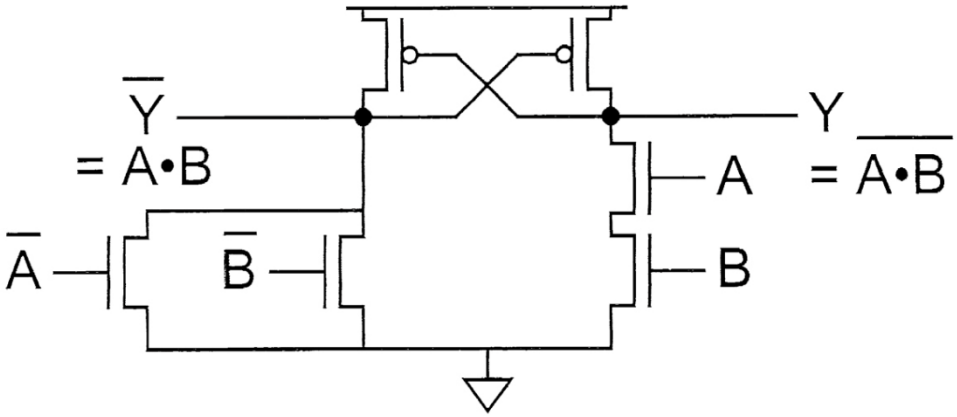
Cascode Voltage Switch Logic

- Differential Cascode Voltage Switching Logic (DCVS, DCVSL)
  - Seeks the performance of ratioed circuits without the static power consumption
  - Use both true and complementary input signals and compute both true and complementary outputs

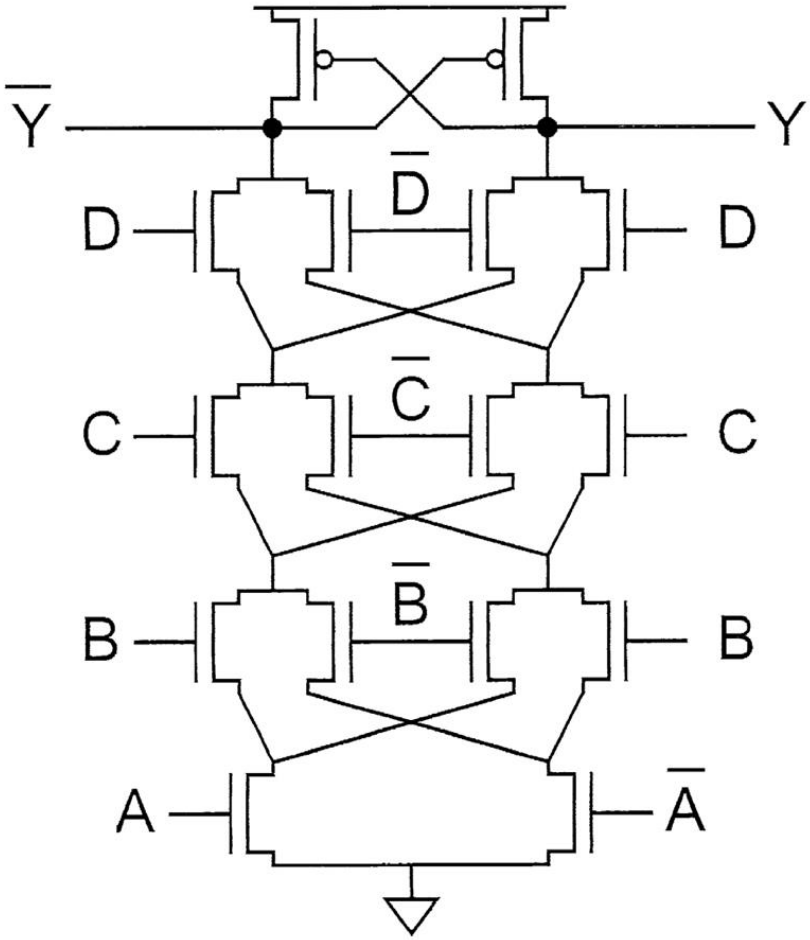


Cascode Voltage Switch Logic

2-input AND/NAND gate



4-input XOR/XNOR gate



Outline

1. Static CMOS
2. Ratioed Circuits
3. Cascode Voltage Switch Logic
4. **Dynamic Circuits**
5. Pass-Transistor Circuits
6. Circuit Pitfalls

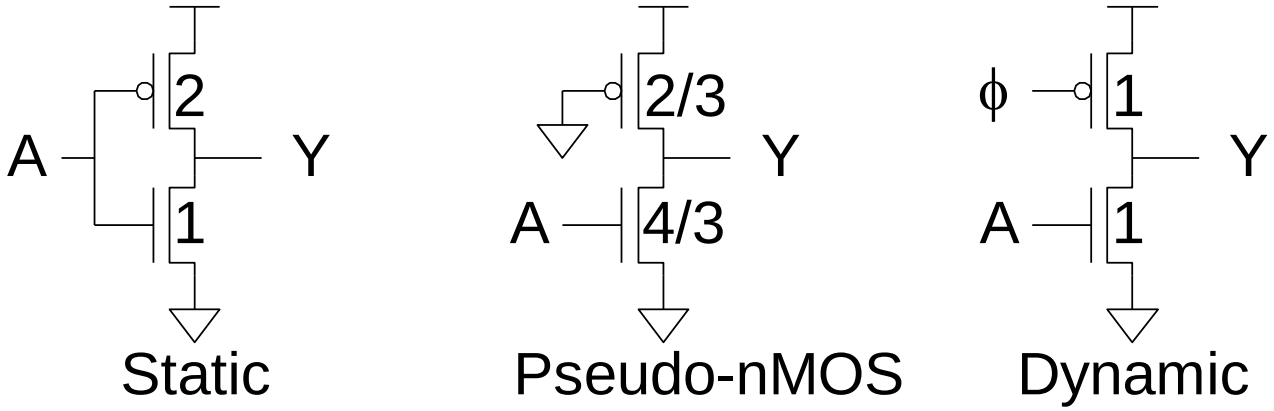
Dynamic CMOS

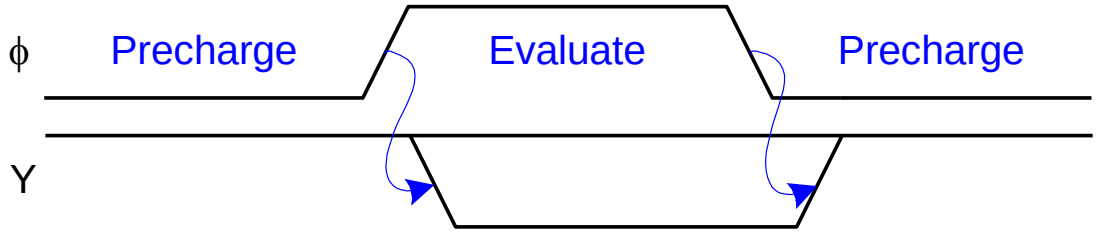
- In static circuits at every point in time (except when switching) the output is connected to either GND or V_{DD} via a low resistance path.
  - fan-in of N requires 2N devices

- Dynamic circuits rely on the temporary storage of signal values on the capacitance of high impedance nodes.
  - requires only N + 2 transistors
  - takes a sequence of precharge and conditional evaluation phases to realize logic functions

Dynamic Logic

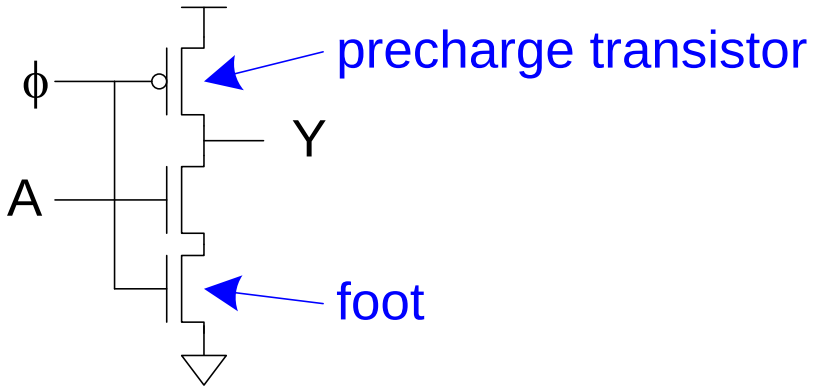
- *Dynamic* gates uses a clocked pMOS pullup
- Two modes: *precharge* and *evaluate*

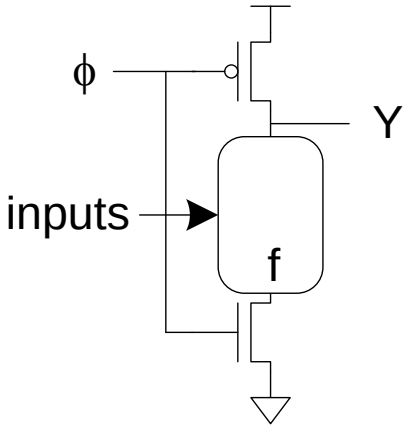




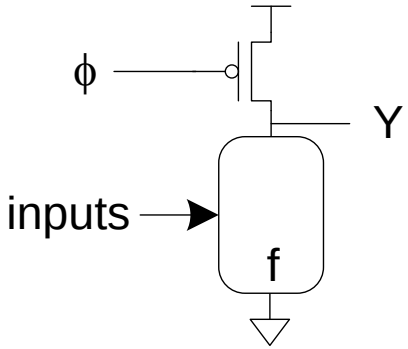
The Foot

- What if pulldown network is ON during precharge?
- Use series evaluation transistor to prevent fight.



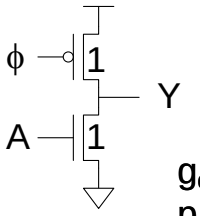
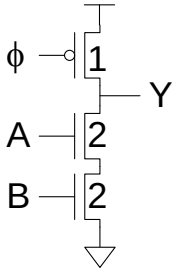
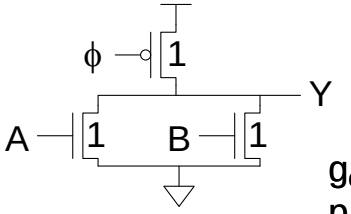
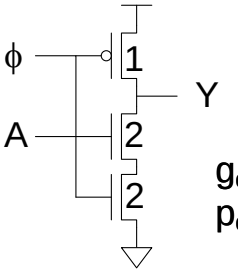
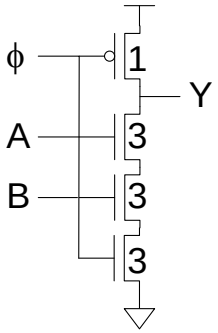
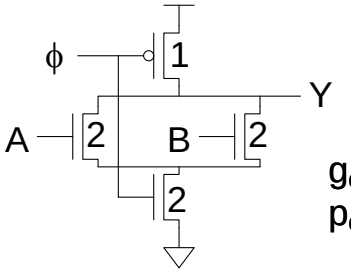


footed



unfooted

Logical Effort

	Inverter	NAND2	NOR2
unfooted	 $g_d = 1/3$ $p_d = 2/3$	 $g_d = 2/3$ $p_d = 3/3$	 $g_d = 1/3$ $p_d = 3/3$
footed	 $g_d = 2/3$ $p_d = 3/3$	 $g_d = 3/3$ $p_d = 4/3$	 $g_d = 2/3$ $p_d = 5/3$

Conditions on Output

- Once the output of a dynamic gate is discharged, it cannot be charged again until the next precharge operation.
- Inputs to the gate can make at most one transition during evaluation.
- Output can be in the high impedance state during and after evaluation (PDN off), state is stored on C_L

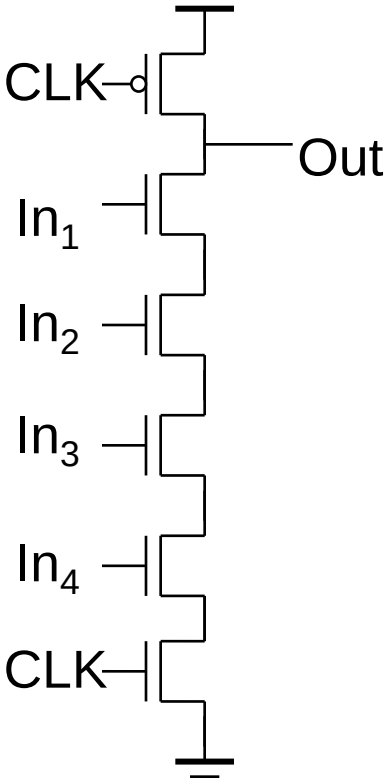
Properties of Dynamic Gates

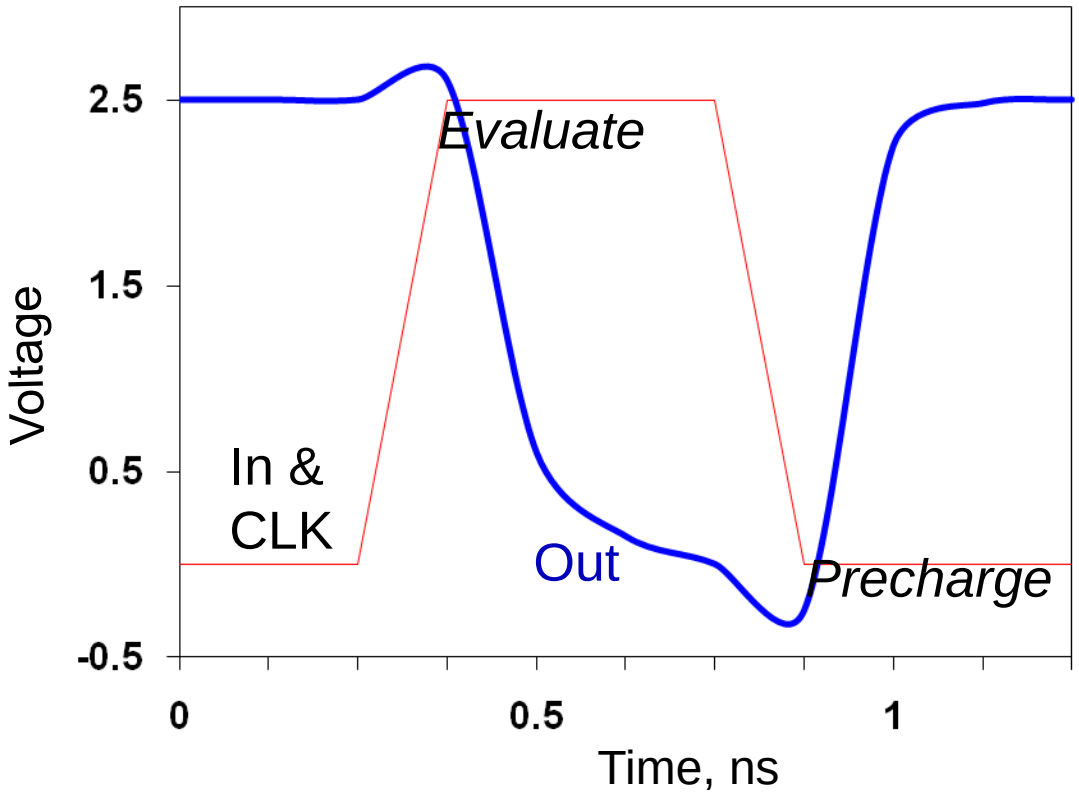
- Logic function is implemented by the PDN only
  - number of transistors is N + 2 (versus 2N for static complementary CMOS)
  - should be smaller in area than static complementary CMOS
- Full swing outputs (V_{OL} = GND and V_{OH} = V_{DD})
- Nonratioed - sizing of the devices is not important for proper functioning (only for performance)
- Faster switching speeds
  - reduced load capacitance due to lower number of transistors per gate (C_{int}) so a reduced logical effort
  - reduced load capacitance due to smaller fan-out (C_{ext})
  - no I_{sc}, so all the current provided by PDN goes into discharging C_L
  - Ignoring the influence of precharge time on the switching speed of the gate, t_{pLH} = 0 but the presence of the evaluation transistor slows down the t_{pHL}

Properties of Dynamic Gates, con't

- Power dissipation should be better
  - consumes only dynamic power – no short circuit power consumption since the pull-up path is not on when evaluating
  - lower C_L- both C_{int} (since there are fewer transistors connected to the drain output) and C_{ext} (since there the output load is one per connected gate, not two)
  - by construction can have at most one transition per cycle – no glitching
- But power dissipation can be significantly higher due to
  - higher transition probabilities
  - extra load on CLK
- PDN starts to work as soon as the input signals exceed V_{Tn}, so set V_M, V_{IH} and V_{IL} all equal to V_{Tn}
  - low noise margin (NM_L)
- Needs a precharge clock

Dynamic Behavior

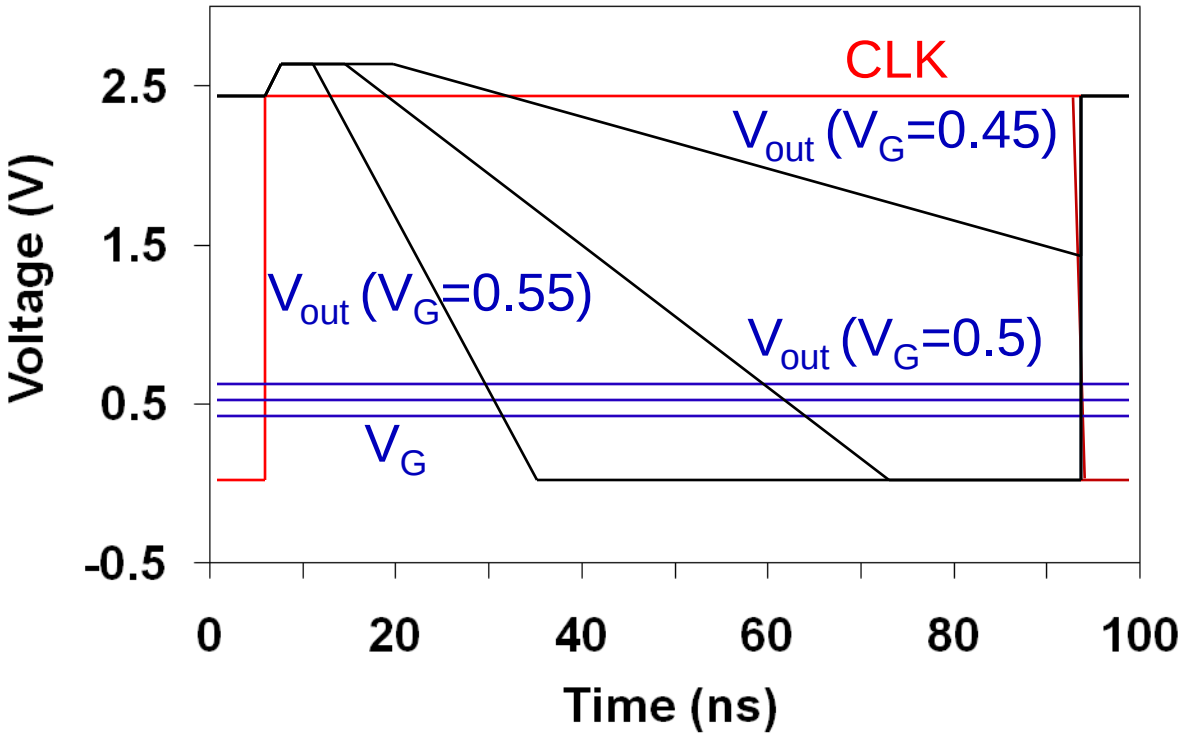




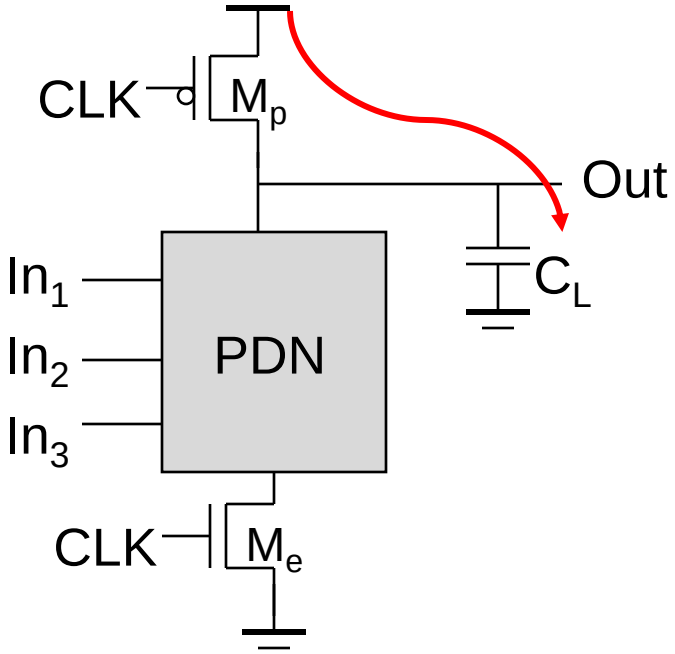
#Trns	V_{OH}	V_{OL}	V_M	NM_H	NM_L	t_{pHL}	t_{pLH}	t_p
6	2.5V	0V	V_{Tn}	2.5-V_{Tn}	V_{Tn}	110ps	0ns	83ps

Gate Parameters are Time Independent

- The amount by which the output voltage drops is a strong function of the input voltage and the available evaluation time.
  - Noise needed to corrupt the signal has to be larger if the evaluation time is short – i.e., the switching threshold is truly time independent.



Power Consumption of Dynamic Gate



Power only dissipated when previous Out = 0

Dynamic Power is Data Dependent

Dynamic 2-input NOR Gate

A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0

Assume signal probabilities

$P_{A=1} = 1/2$

$P_{B=1} = 1/2$

Then transition probability

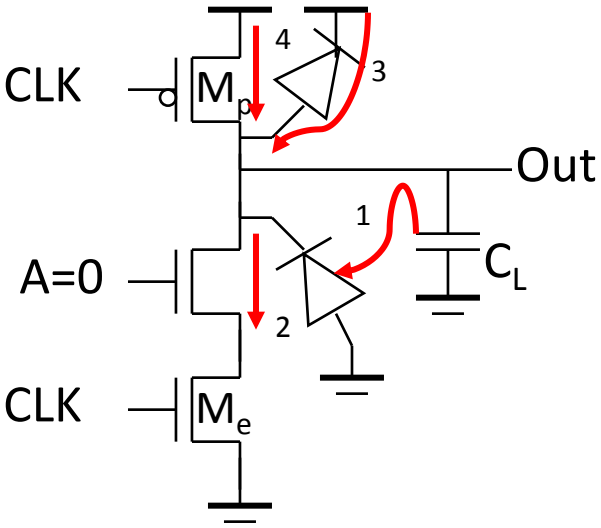
$P_{0\rightarrow1} = P_{out=0} \times P_{out=1}$

$= 3/4 \times 1 = 3/4$

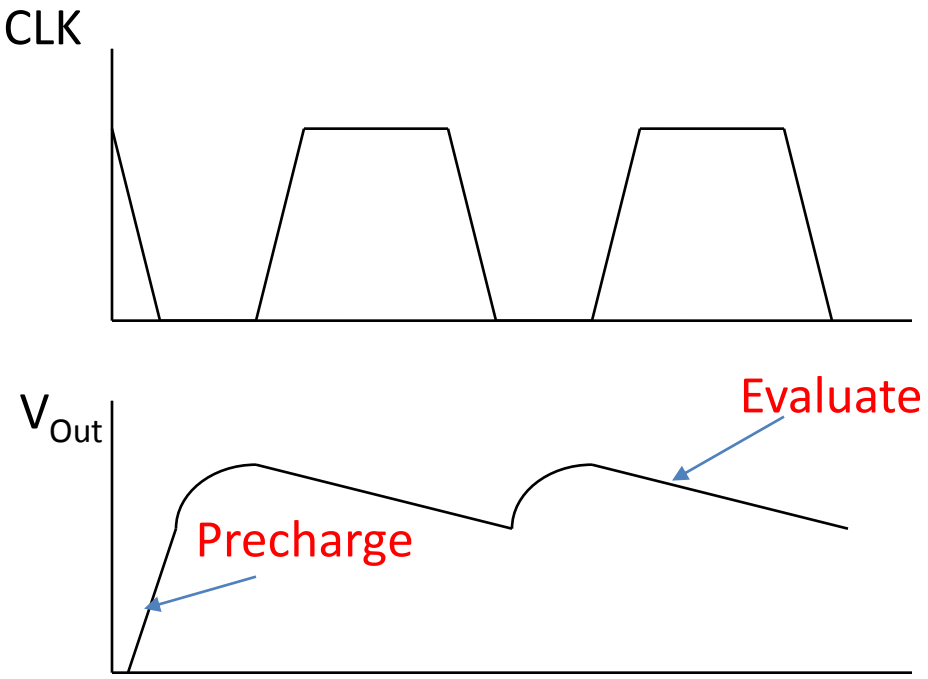
Switching activity can be higher in dynamic gates!

$P_{0\rightarrow1} = P_{out=0}$

Issues 1: Charge Leakage



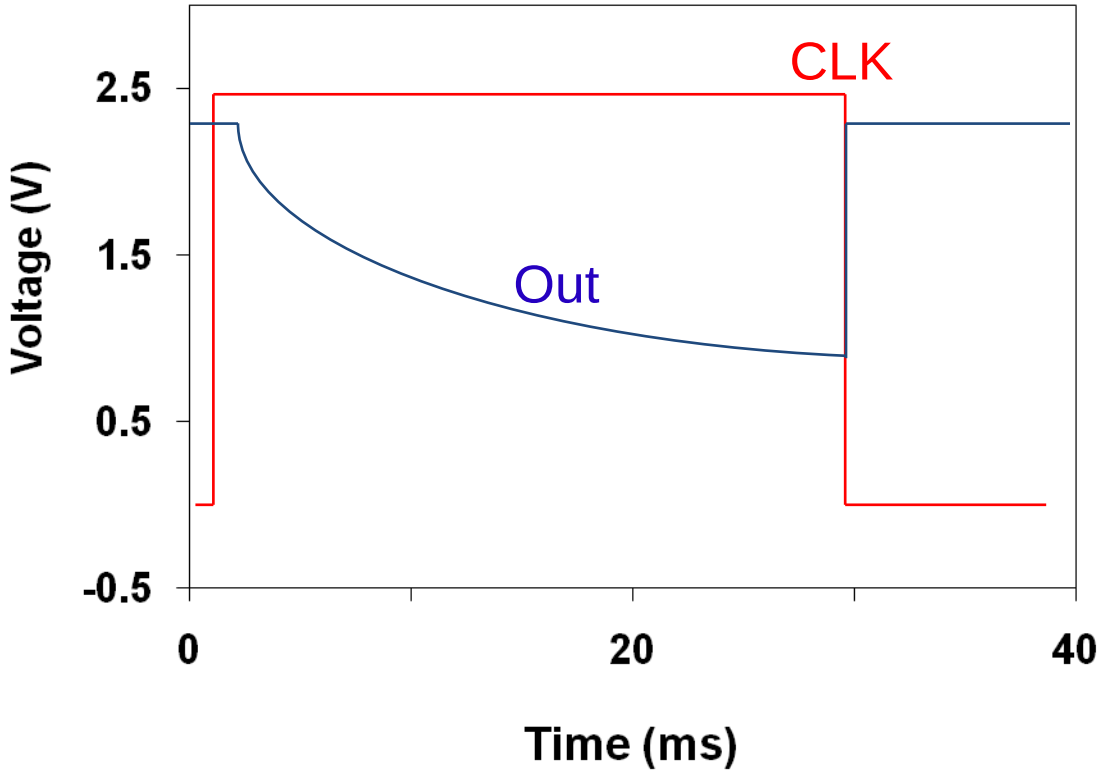
Leakage sources



Minimum clock rate of a few kHz

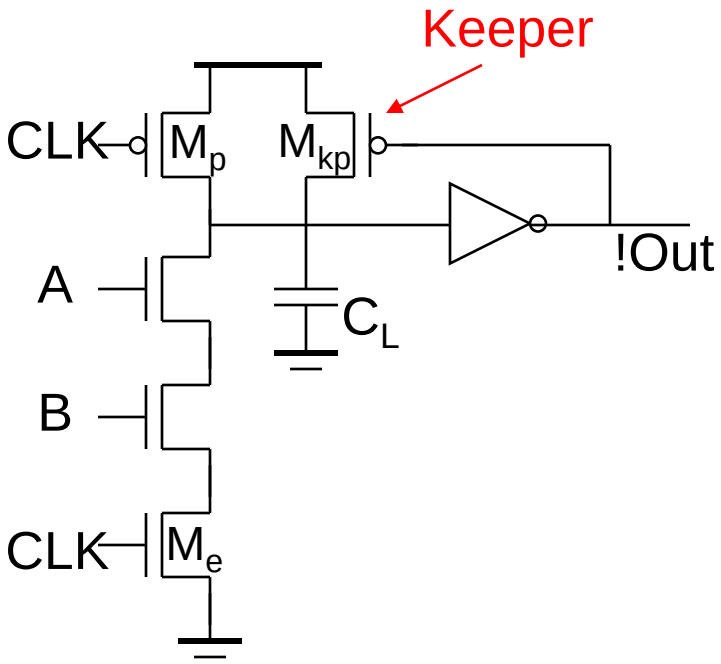
Impact of Charge Leakage

- Output settles to an intermediate voltage determined by a resistive divider of the pull-up and pull-down networks
  - Once the output drops below the switching threshold of the fan-out logic gate, the output is interpreted as a low voltage.



A Solution to Charge Leakage

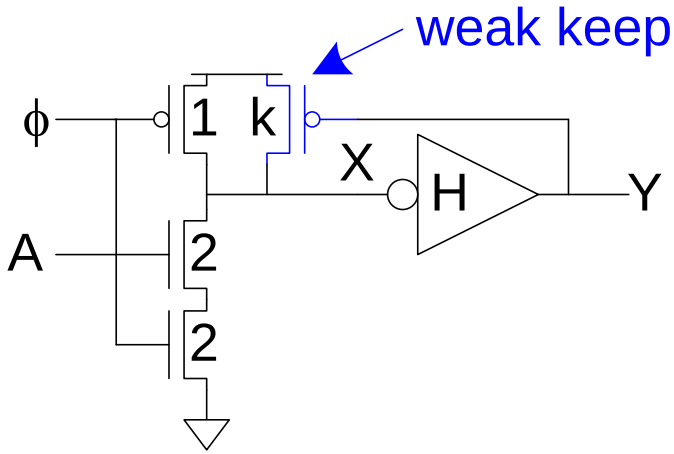
- Keeper compensates for the charge lost due to the pull-down leakage paths.



Same approach as level restorer for pass transistor logic

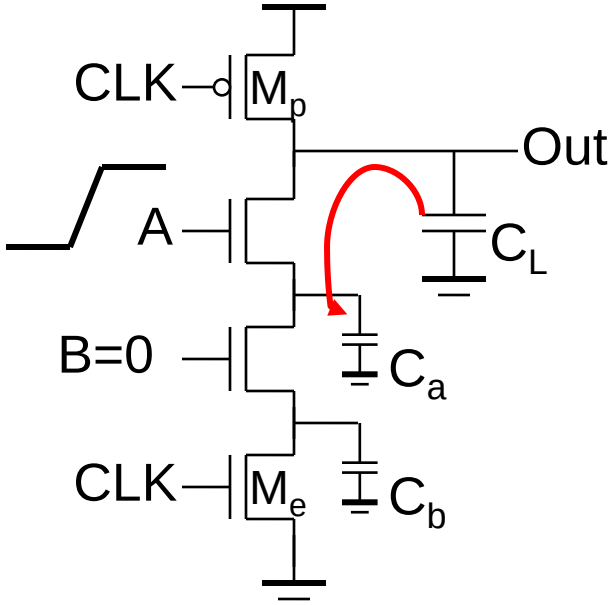
Summary: Leakage

- Dynamic node floats high during evaluation
  - Transistors are leaky ($I_{OFF} \neq 0$)
  - Dynamic value will leak away over time
  - Formerly miliseconds, now nanoseconds!
- Use keeper to hold dynamic node
  - Must be weak enough not to fight evaluation



Issues 2: Charge Sharing

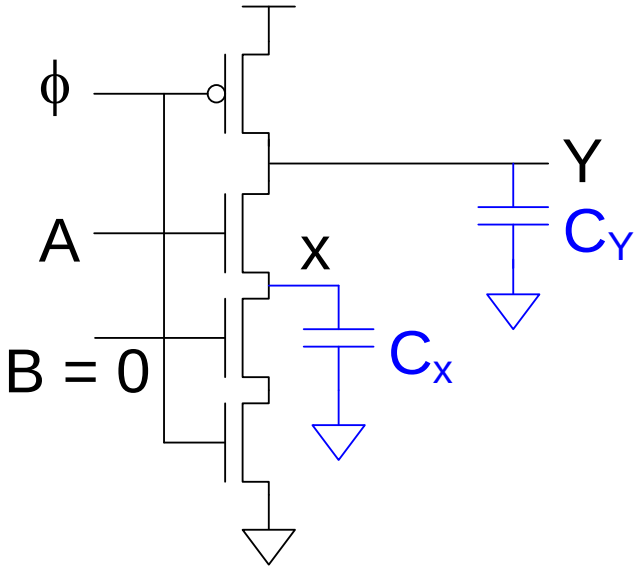
- Charge stored originally on C_L is redistributed (shared) over C_L and C_A leading to static power consumption by downstream gates and possible circuit malfunction.

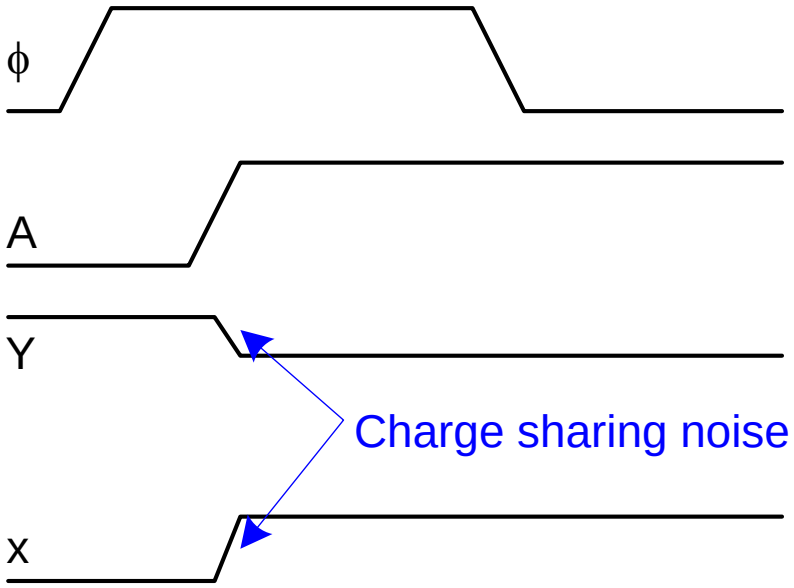


- When $\Delta V_{out} = - V_{DD} (C_a / (C_a + C_L))$ the drop in V_{out} is large enough to be below the switching threshold of the gate it drives causing a malfunction.

Charge Sharing

- Dynamic gates suffer from charge sharing

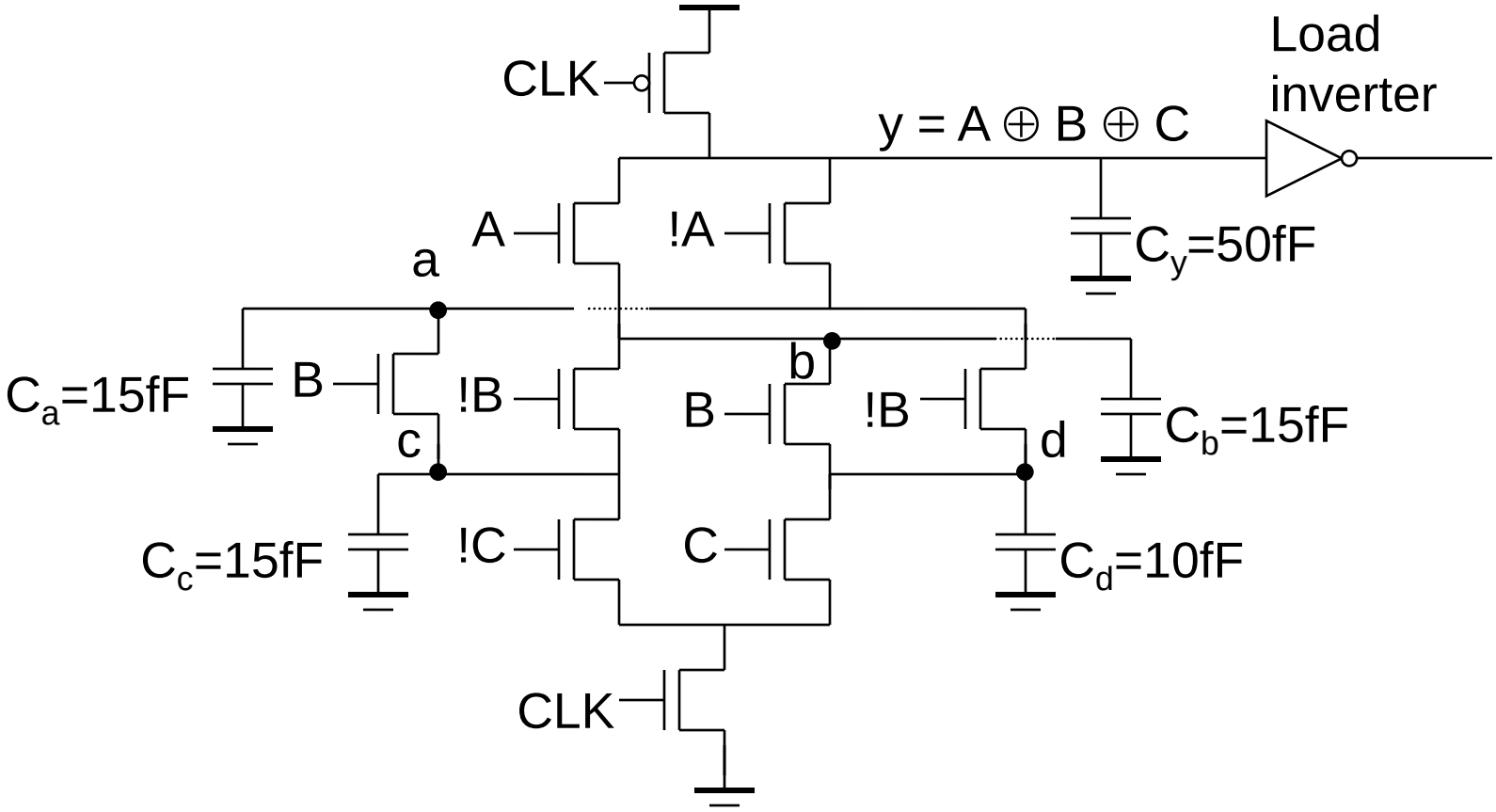




$$V_x = V_Y = \frac{C_Y}{C_x + C_Y} V_{DD}$$

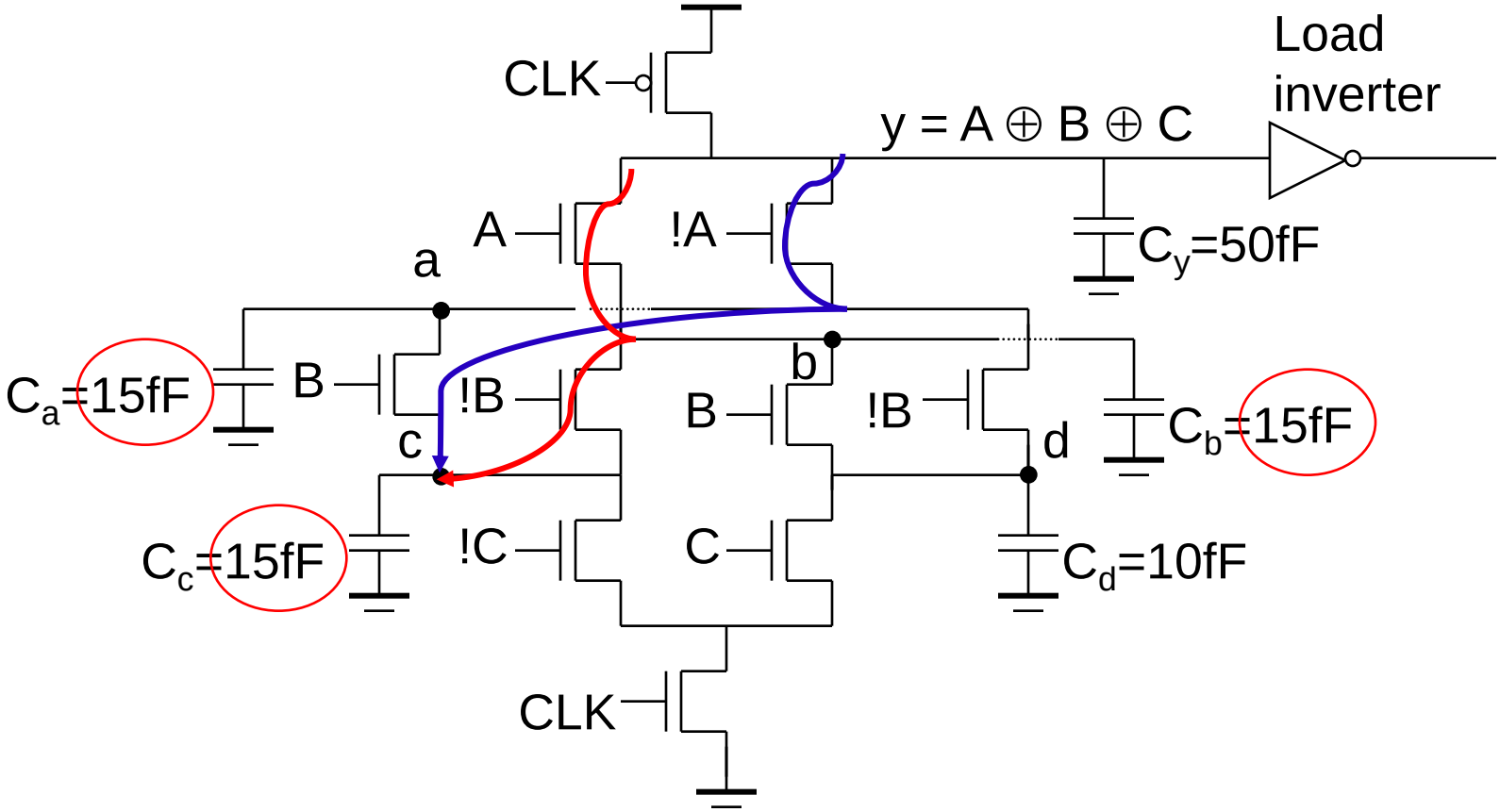
Charge Sharing Example

What is the worst case voltage drop on y? (Assume all inputs are low during precharge and that all internal nodes are initially at 0V.)



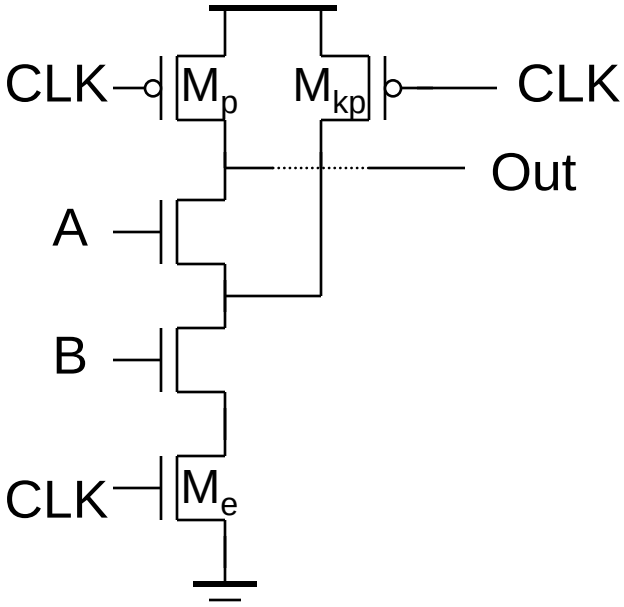
Charge Sharing Example

What is the worst case voltage drop on y? (Assume all inputs are low during precharge and that all internal nodes are initially at 0V.)



$\Delta V_{out} = - V_{DD} ((C_a + C_c)/((C_a + C_c) + C_y)) = - 2.5V*(30/(30+50)) = -0.94V$

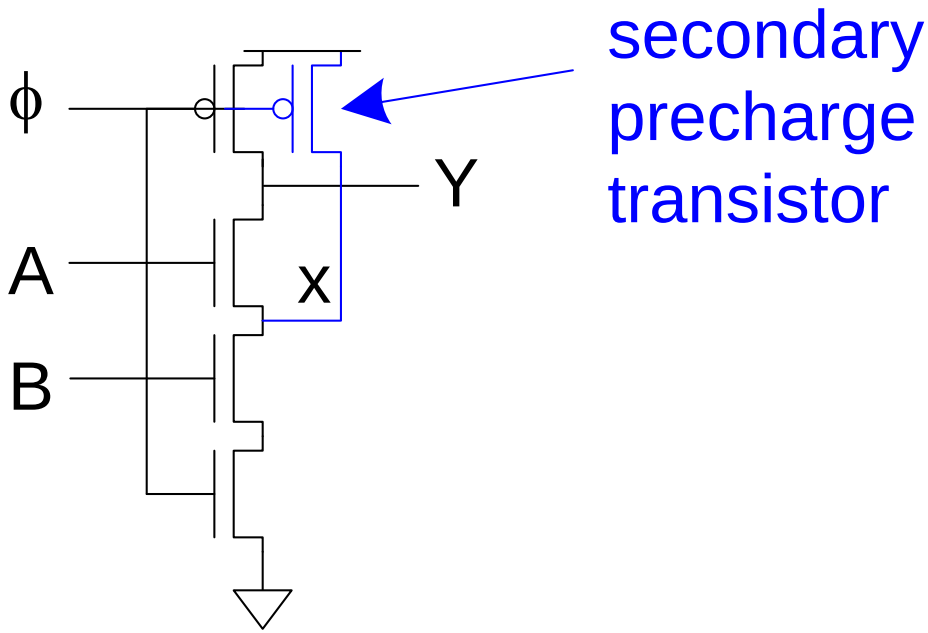
Solution to Charge Redistribution



Precharge internal nodes using a clock-driven transistor (at the cost of increased area and power)

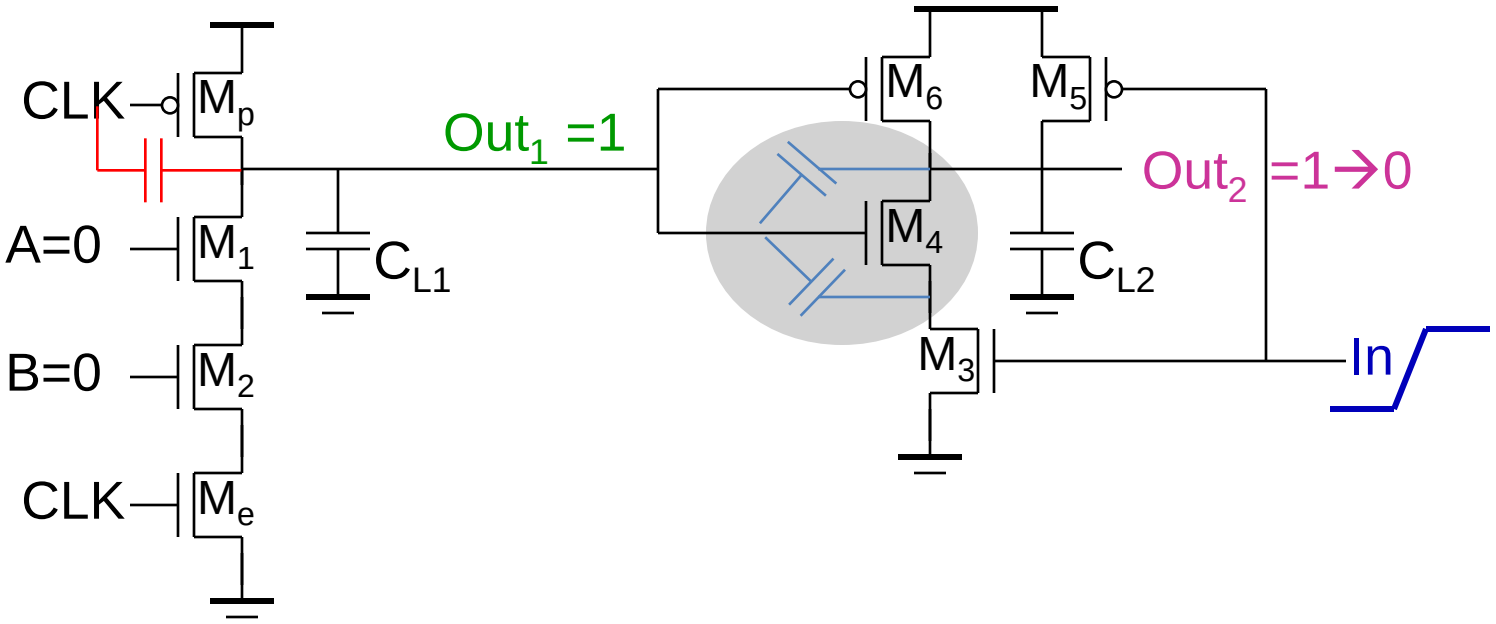
Secondary Precharge

- Solution: add secondary precharge transistors
  - Typically need to precharge every other node
- Big load capacitance C_Y helps as well



Issues 3: Backgate Coupling

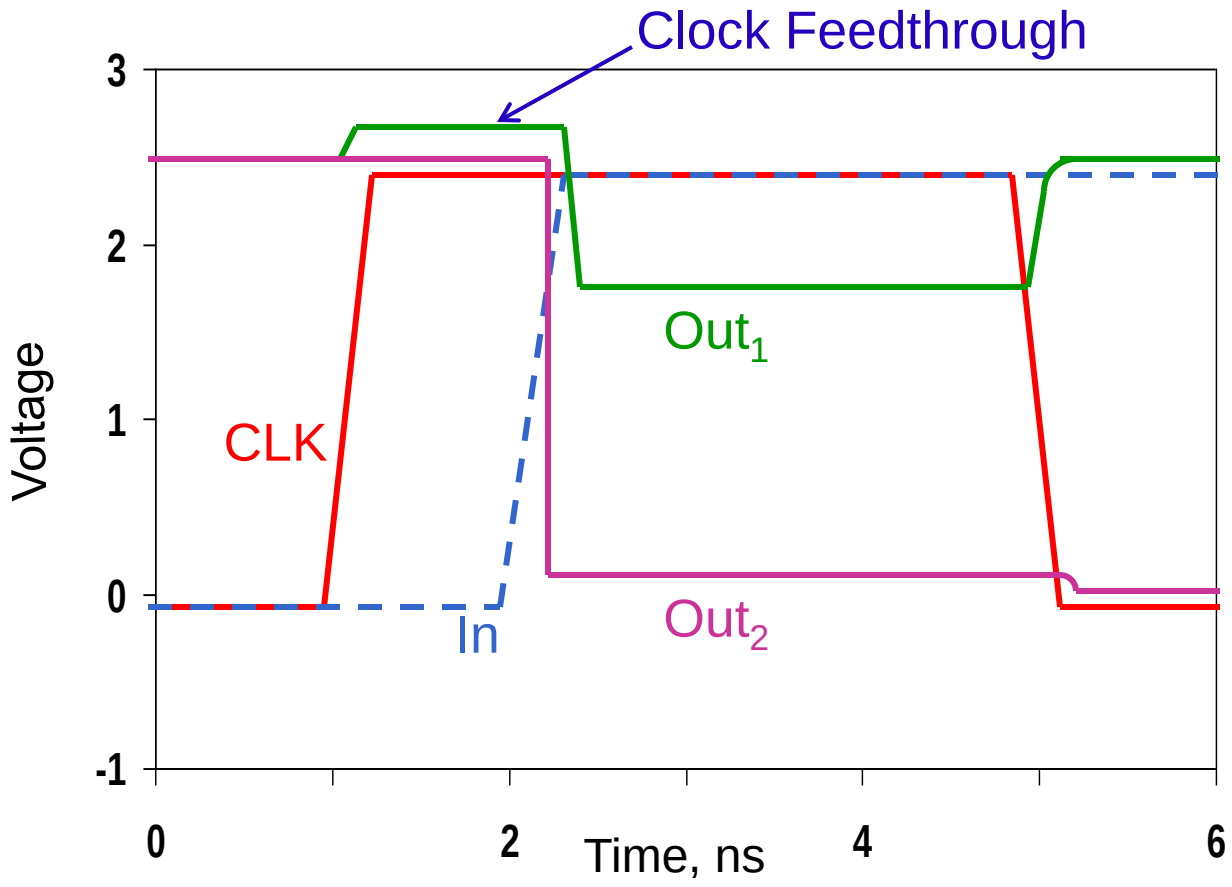
- Susceptible to crosstalk due to 1) high impedance of the output node and 2) backgate capacitive coupling
  - Out_2 capacitively couples with Out_1 through the gate-source and gate-drain capacitances of M4



Dynamic NAND Static NAND

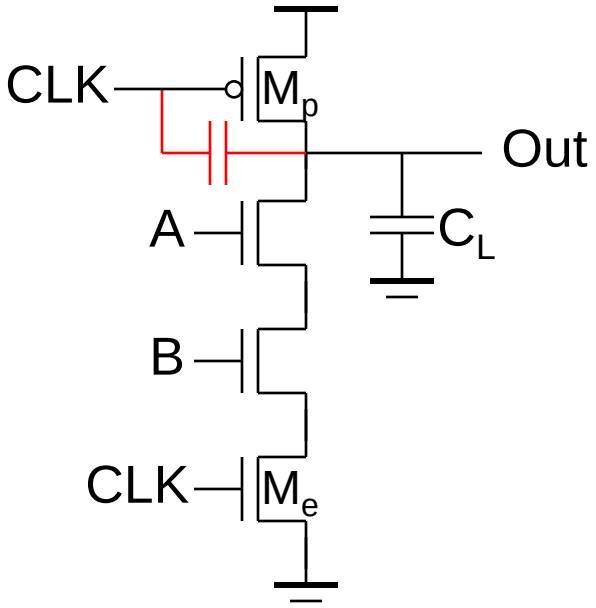
Backgate Coupling Effect

- Capacitive coupling means Out_1 drops significantly so Out_2 doesn't go all the way to ground



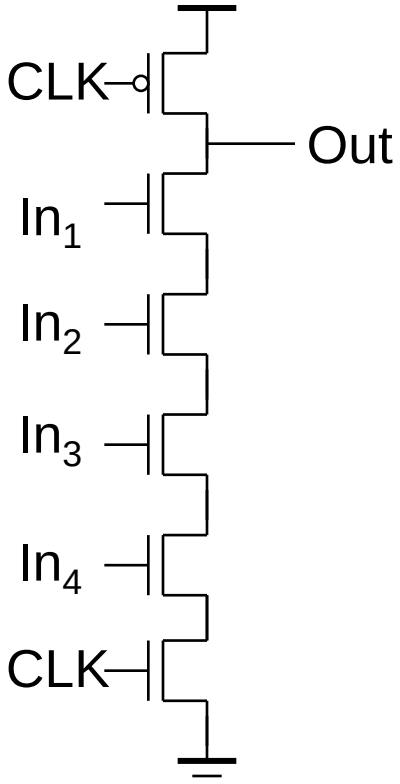
Issues 4: Clock Feedthrough

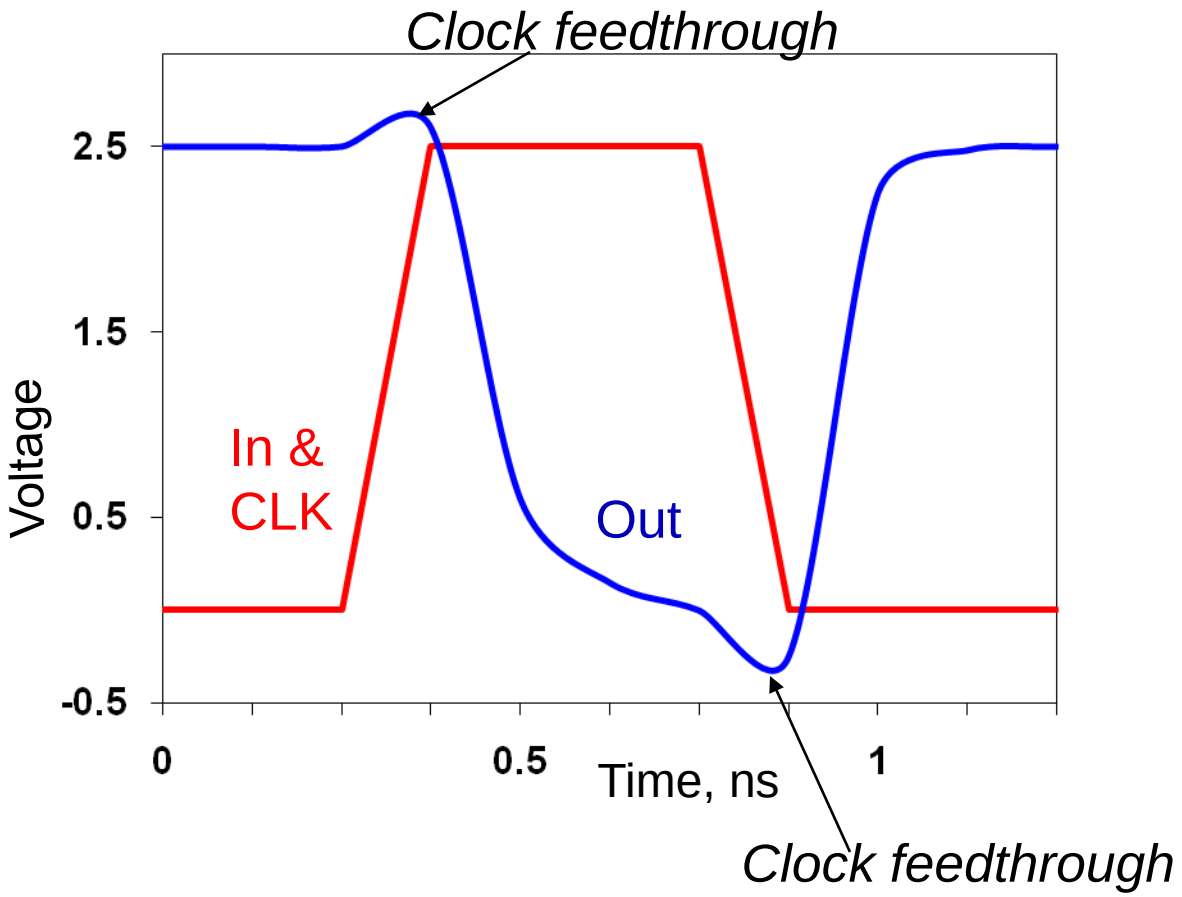
- A special case of backgate capacitive coupling between the clock input of the precharge transistor and the dynamic output node



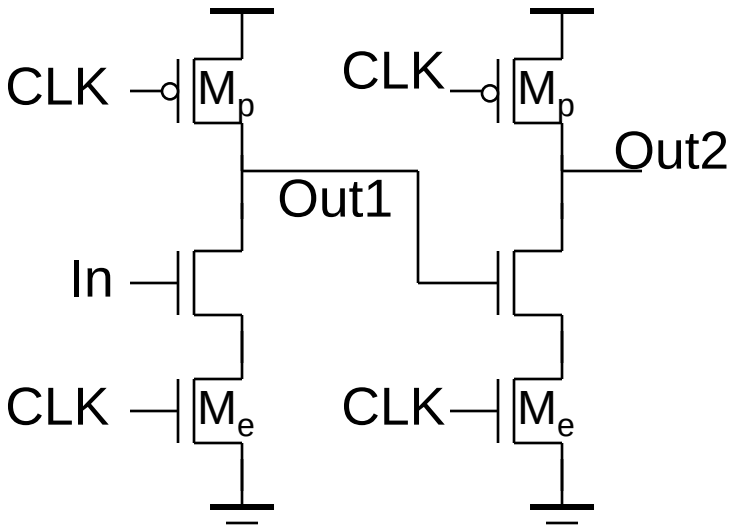
Coupling between Out and CLK input of the precharge device due to the gate-drain capacitance. So voltage of Out can rise above V_{DD}. The fast rising (and falling edges) of the clock couple to Out.

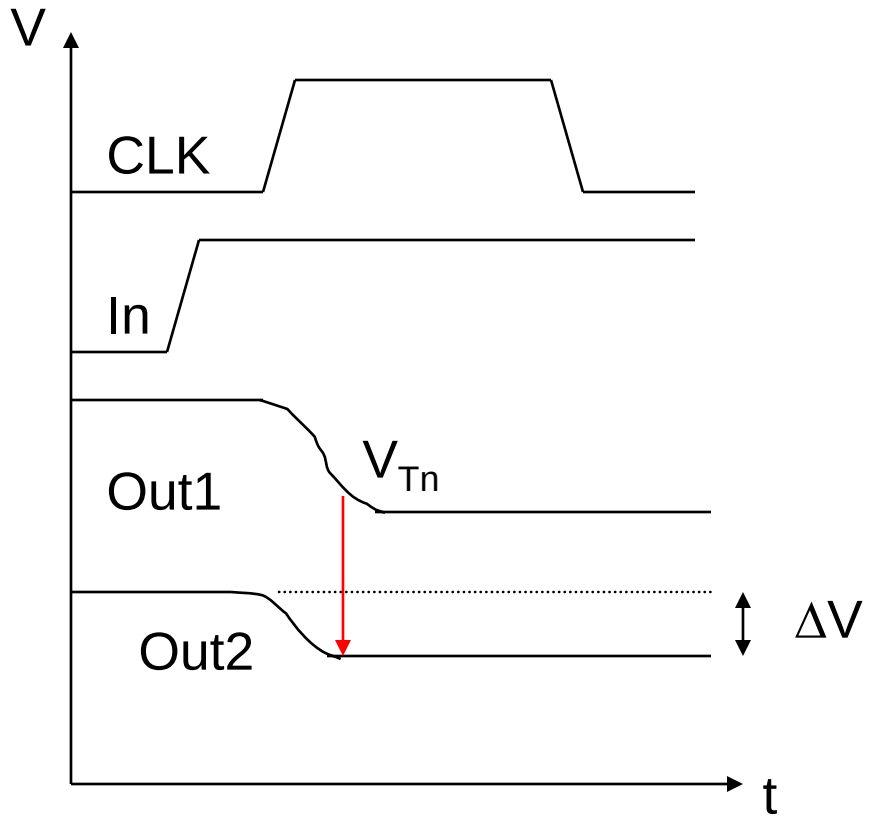
Clock Feedthrough





Issues 5: Cascading Gates

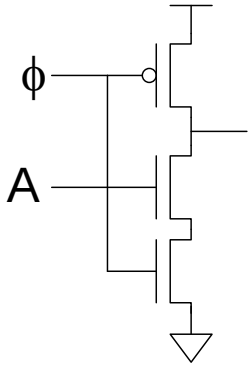


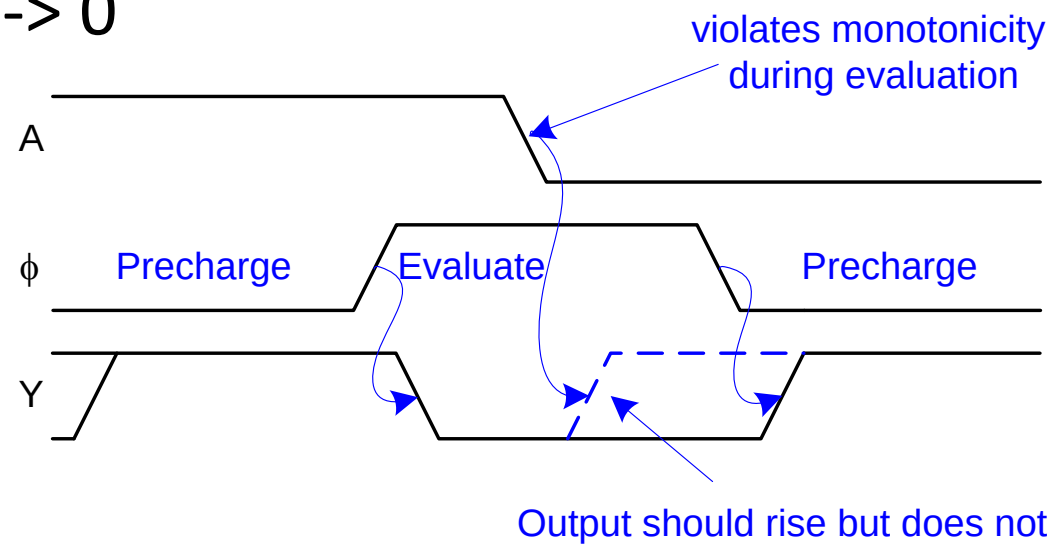


Only a single 0 → 1 transition allowed at the inputs during the evaluation period!

Monotonicity

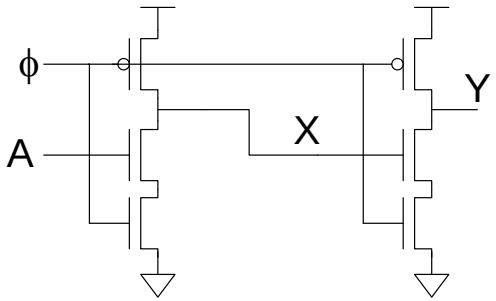
- Dynamic gates require *monotonically rising* inputs during evaluation
  - 0 -> 0
  - 0 -> 1
  - 1 -> 1
  - But not 1 -> 0

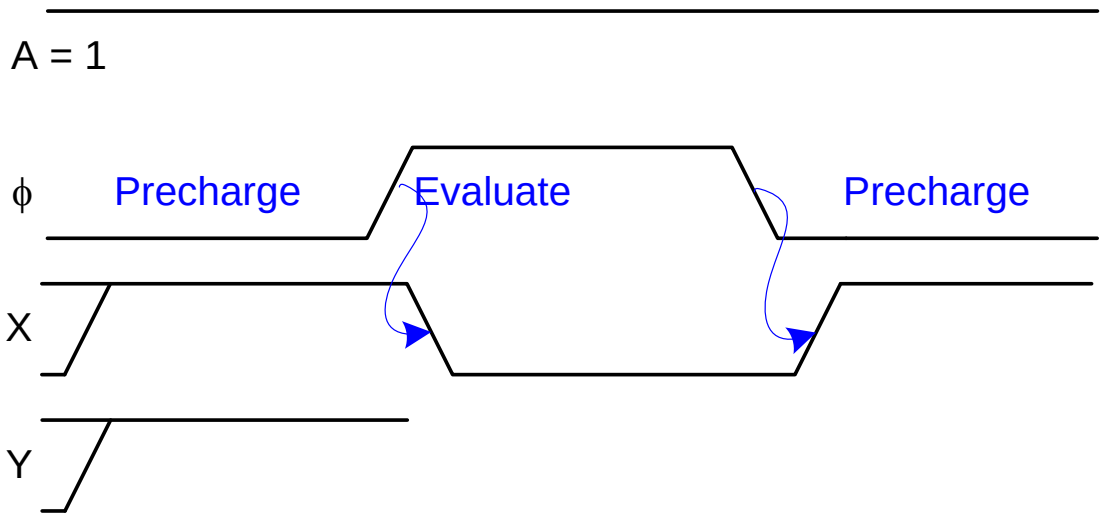




Monotonicity Woes

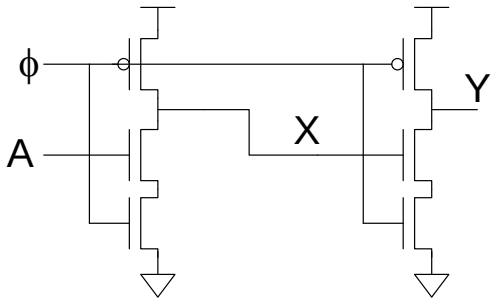
- But dynamic gates produce monotonically falling outputs during evaluation
- Illegal for one dynamic gate to drive another!

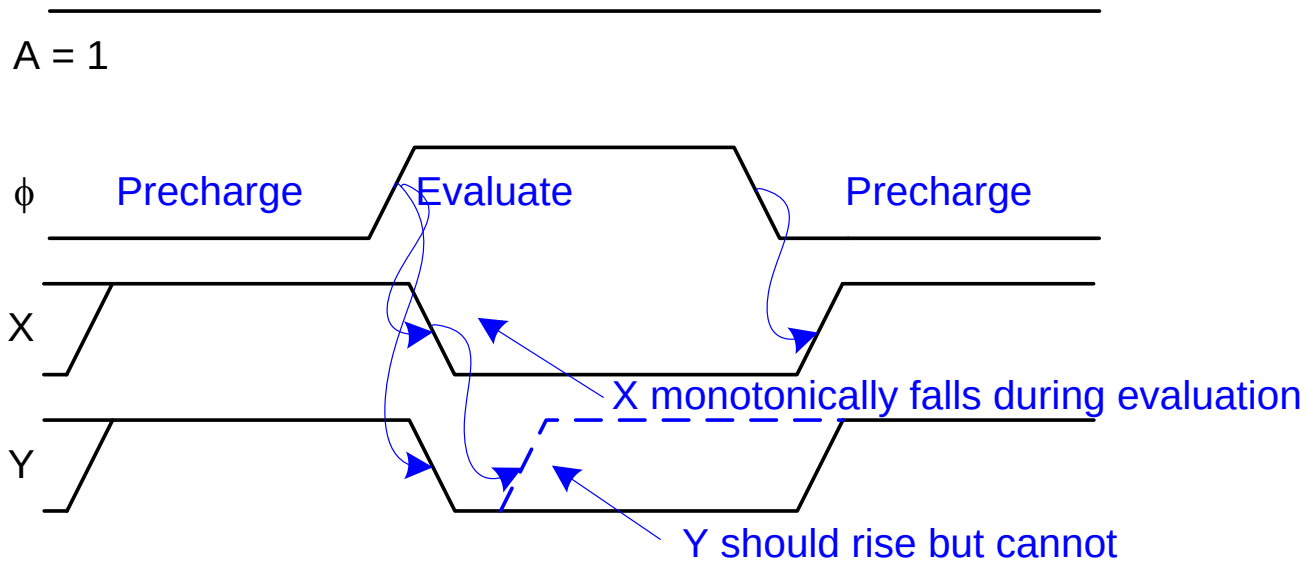




Monotonicity Woes

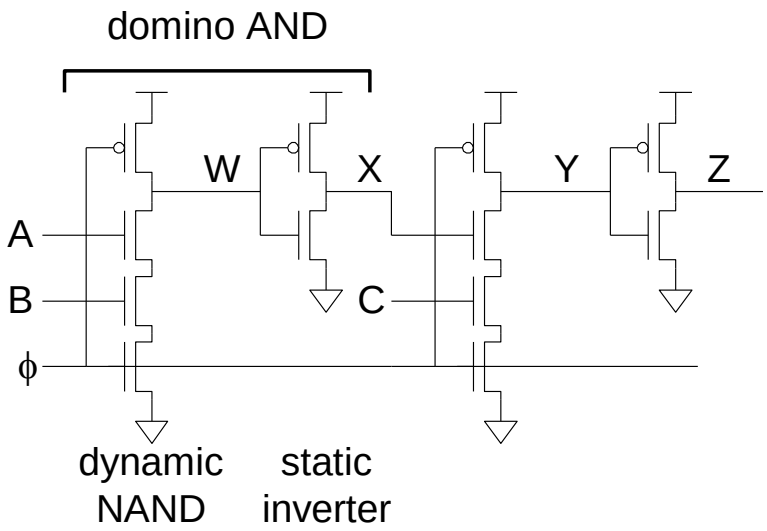
- But dynamic gates produce monotonically falling outputs during evaluation
- Illegal for one dynamic gate to drive another!

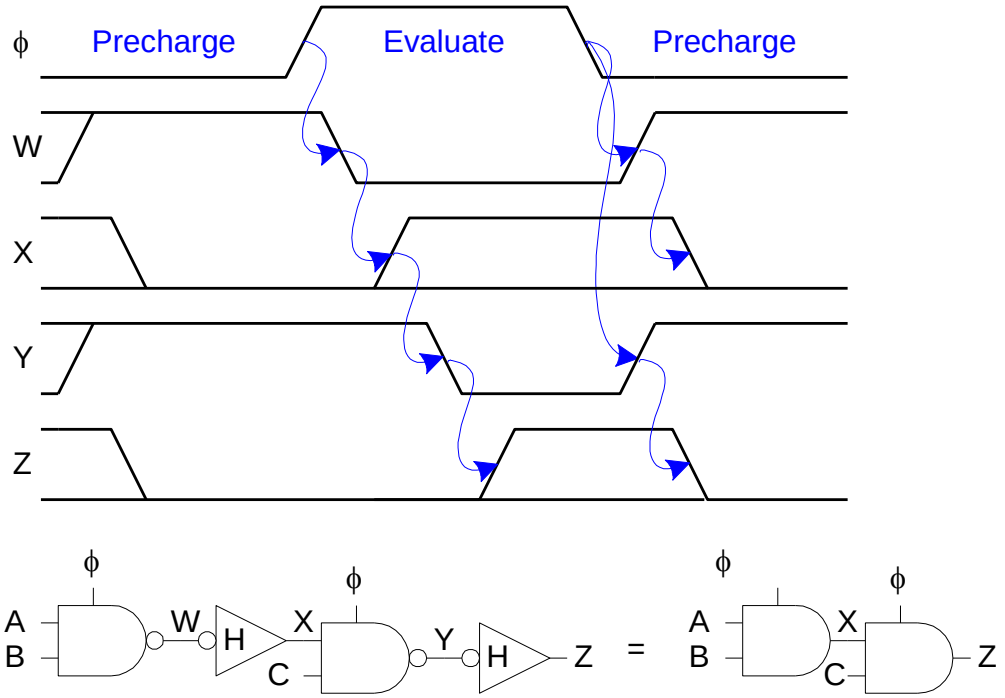




Domino Gates

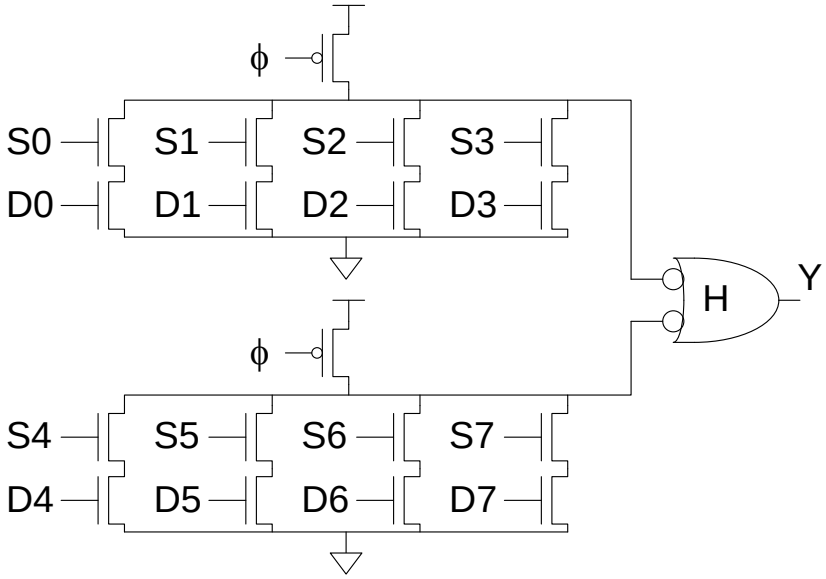
- Follow dynamic stage with inverting static gate
  - Dynamic / static pair is called domino gate
  - Produces monotonic outputs





Domino Optimizations

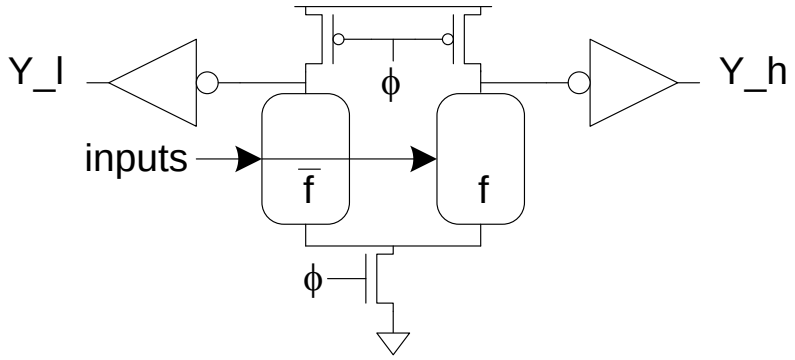
- Each domino gate triggers next one, like a string of dominos toppling over
- Gates evaluate sequentially, precharge in parallel
- Thus evaluation is more critical than precharge
- HI-skewed static stages can perform logic



Dual-Rail Domino

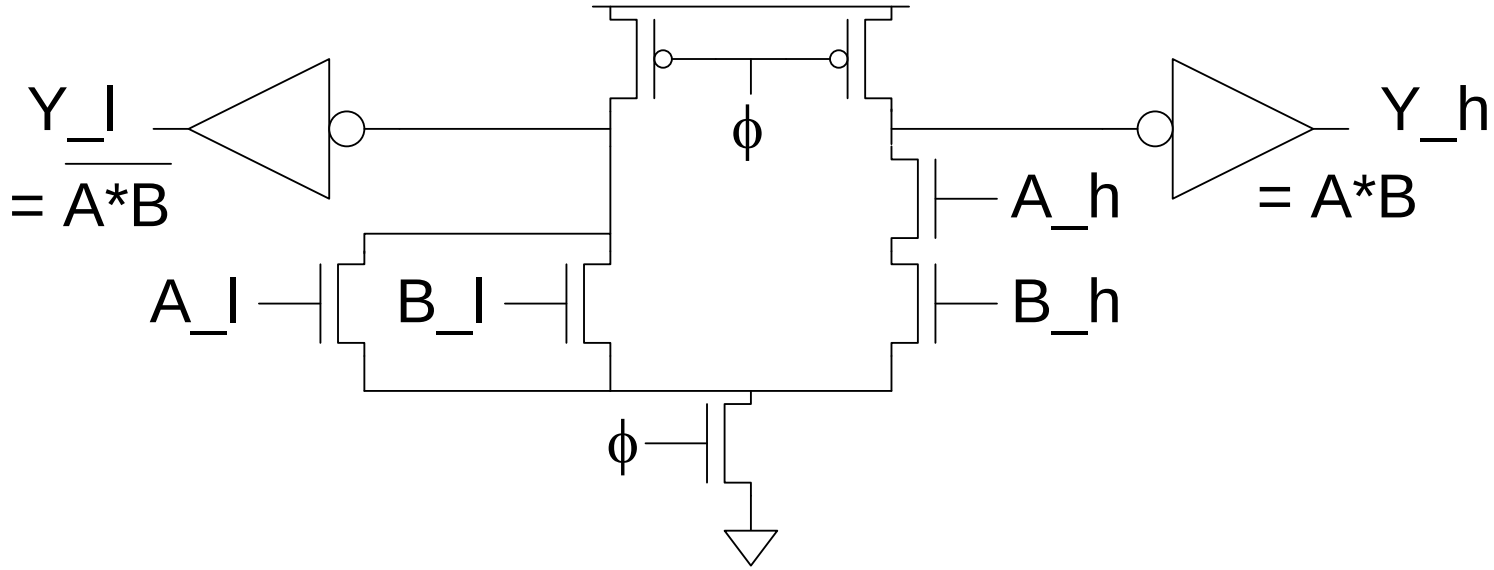
- Domino only performs noninverting functions:
  - AND, OR but not NAND, NOR, or XOR
- Dual-rail domino solves this problem
  - Takes true and complementary inputs
  - Produces true and complementary outputs

sig_h	sig_l	Meaning
0	0	Precharged
0	1	‘0’
1	0	‘1’
1	1	invalid



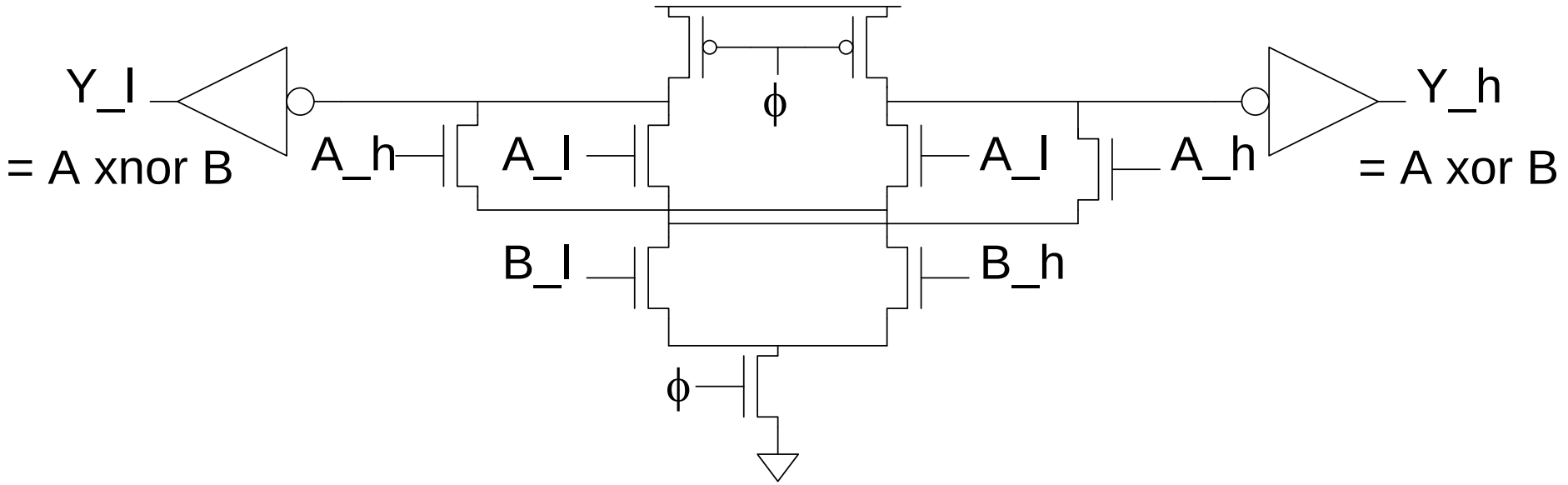
Example: AND/NAND

- Given A_h, A_l, B_h, B_l
- Compute Y_h = A * B, Y_l = ~(A * B)
- Pulldown networks are conduction complements



Example: XOR/XNOR

- Sometimes possible to share transistors



Noise Sensitivity

- Dynamic gates are very sensitive to noise
  - Inputs: $V_{IH} \approx V_{tn}$
  - Outputs: floating output susceptible noise
- Noise sources
  - Capacitive crosstalk
  - Charge sharing
  - Power supply noise
  - Feedthrough noise
  - And more!

Domino Summary

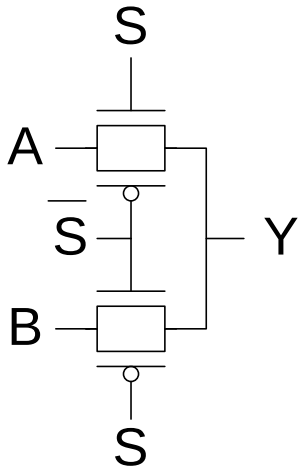
- Domino logic is attractive for high-speed circuits
  - 1.5 – 2x faster than static CMOS
  - But many challenges:
    - Monotonicity
    - Leakage
    - Charge sharing
    - Noise
- Widely used in high-performance microprocessors

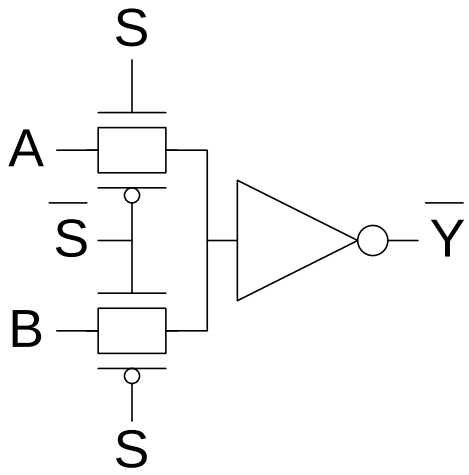
Outline

1. Static CMOS
2. Ratioed Circuits
3. Cascode Voltage Switch Logic
4. Dynamic Circuits
5. **Pass-Transistor Circuits**
6. Circuit Pitfalls

Pass Transistor Circuits

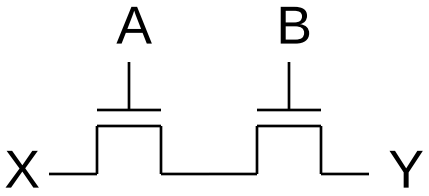
- Use pass transistors like switches to do logic
- Inputs drive diffusion terminals as well as gates
- CMOS + Transmission Gates:
  - 2-input multiplexer
  - Gates should be restoring



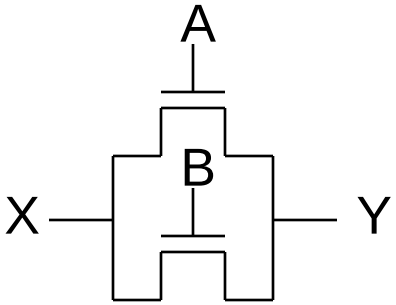


NMOS Transistors in Series/Parallel

- Primary inputs drive both gate and source/drain terminals
- NMOS switch closes when the gate input is high



X = Y if A and B

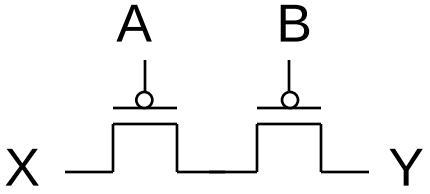


X = Y if A or B

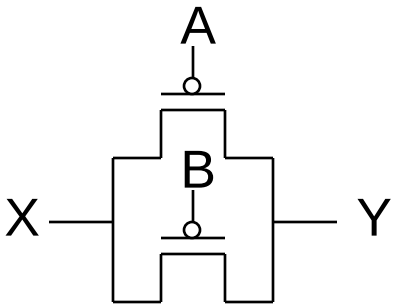
- Remember - NMOS transistors pass a strong 0 but a weak 1

PMOS Transistors in Series/Parallel

- Primary inputs drive both gate and source/drain terminals
- PMOS switch closes when the gate input is low



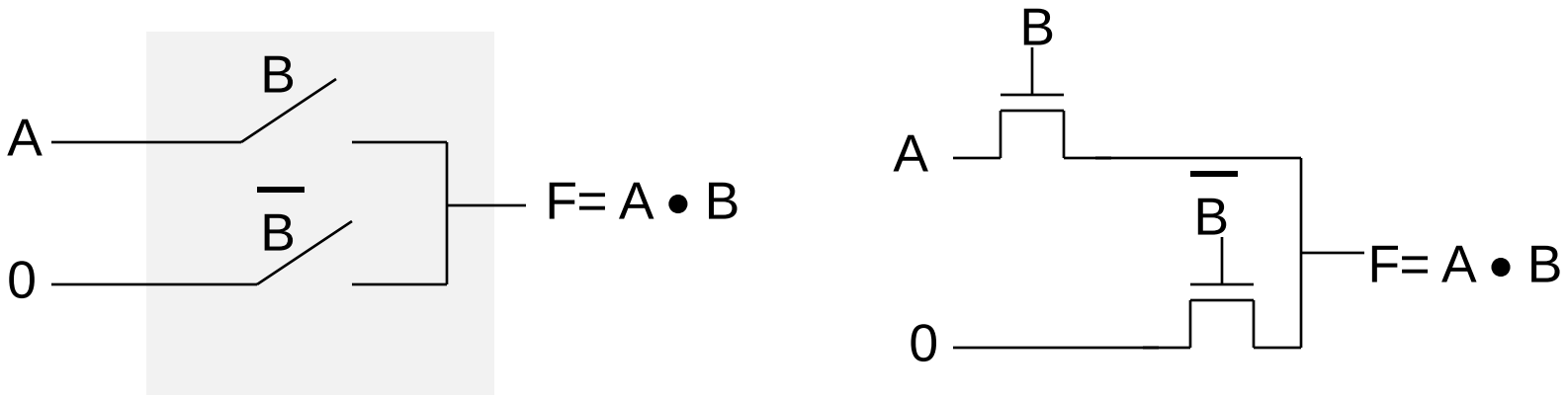
$X = Y$ if $\overline{A}$ and $\overline{B} = \overline{A + B}$



$X = Y$ if $\overline{A}$ or $\overline{B} = \overline{A \bullet B}$

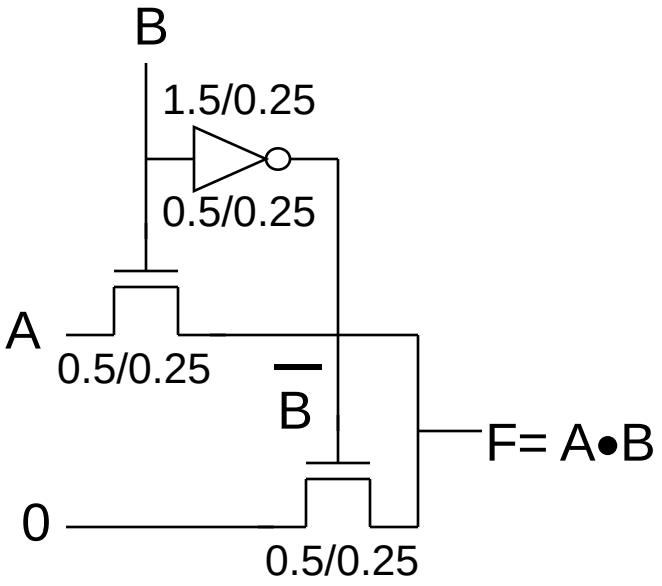
- Remember - PMOS transistors pass a strong 1 but a weak 0

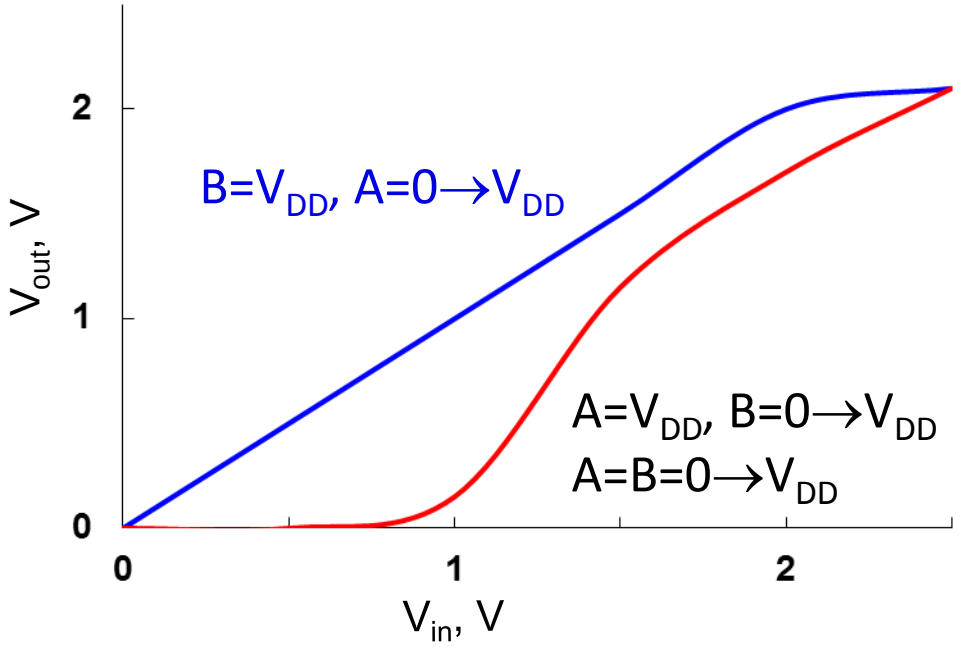
Pass Transistor (PT) Logic



- Gate is static – a low-impedance path exists to both supply rails under all circumstances
- N transistors instead of 2N
- No static power consumption
- Ratioless
- Bidirectional (versus undirectional)

VTC of PT AND Gate

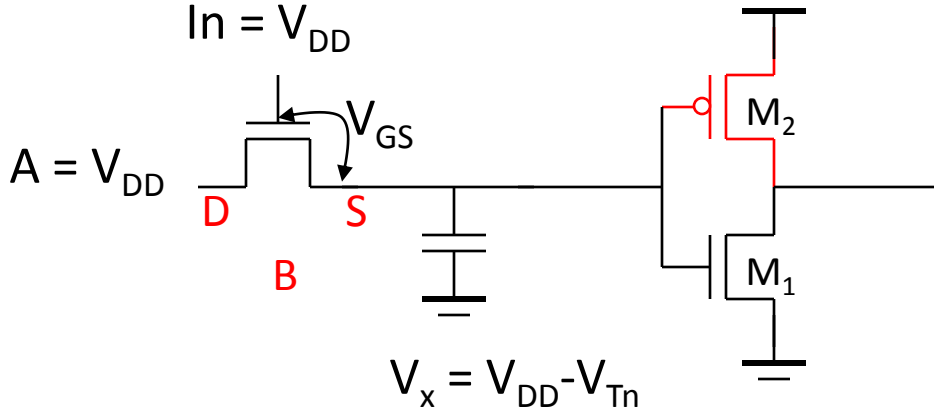




- Pure PT logic is not regenerative - the signal gradually degrades after passing through a number of PTs (can fix with static CMOS inverter insertion)

NMOS Only PT Driving an Inverter

- V_x does not pull up to V_{DD}, but $V_{DD} - V_{Tn}$

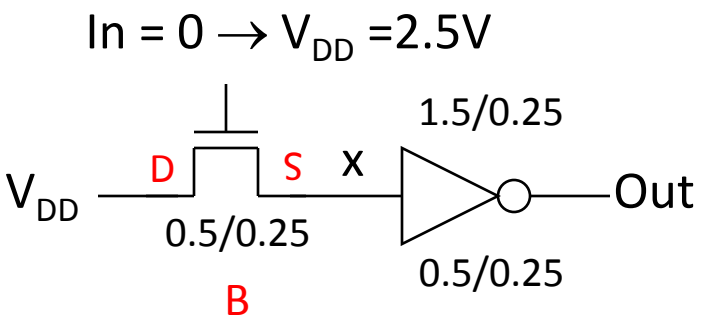


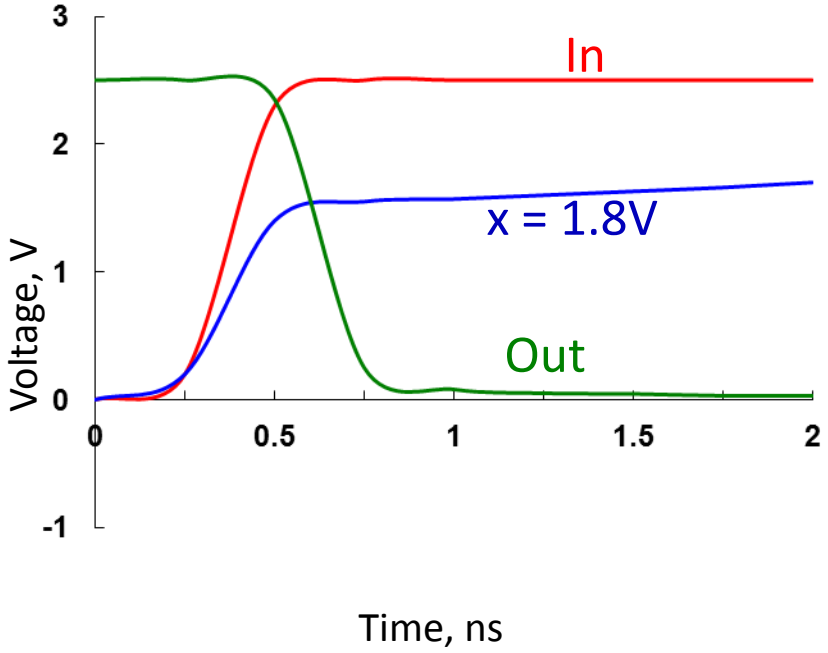
- Threshold voltage drop causes static power consumption (M_2 may be weakly conducting forming a path from V_{DD} to GND)
- Notice V_{Tn} increases for pass transistor due to body effect (V_{SB})

Voltage Swing of PT Driving an Inverter

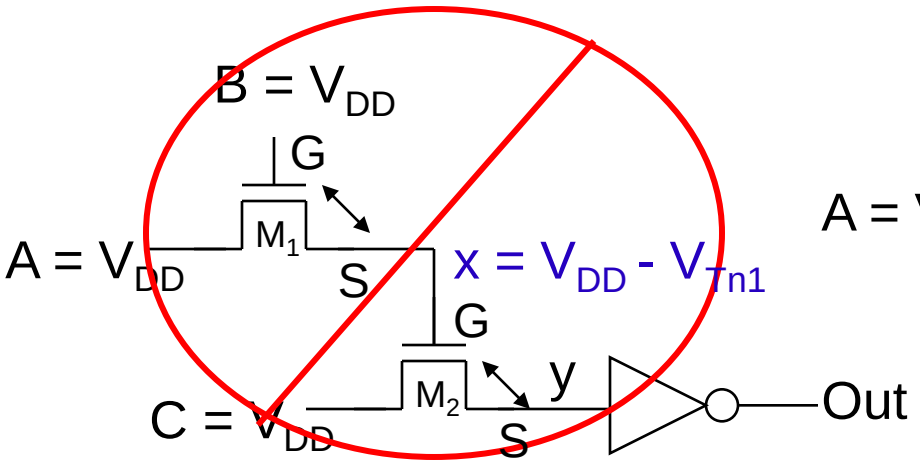
- Body effect – large V_{SB} at x - when pulling high (B is tied to GND and S charged up close to V_{DD})
- So the voltage drop is even worse

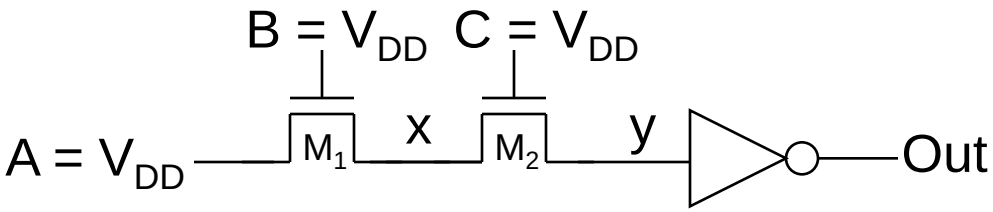
$$V_x = V_{DD} - (V_{Tn0} + \gamma(\sqrt{(|2\phi_f| + V_x)} - \sqrt{|2\phi_f|}))$$





Cascaded NMOS Only PTs





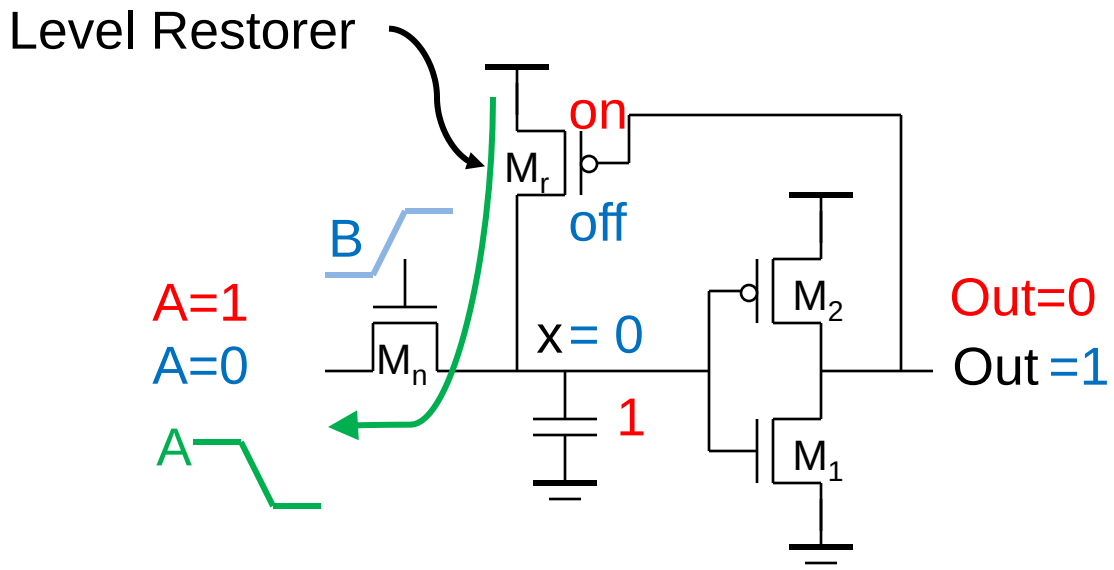
Swing on y = $V_{DD} - V_{Tn1} - V_{Tn2}$

Swing on y = $V_{DD} - V_{Tn1}$

- Pass transistor gates should never be cascaded as on the left
- Logic on the right suffers from static power dissipation and reduced noise margins

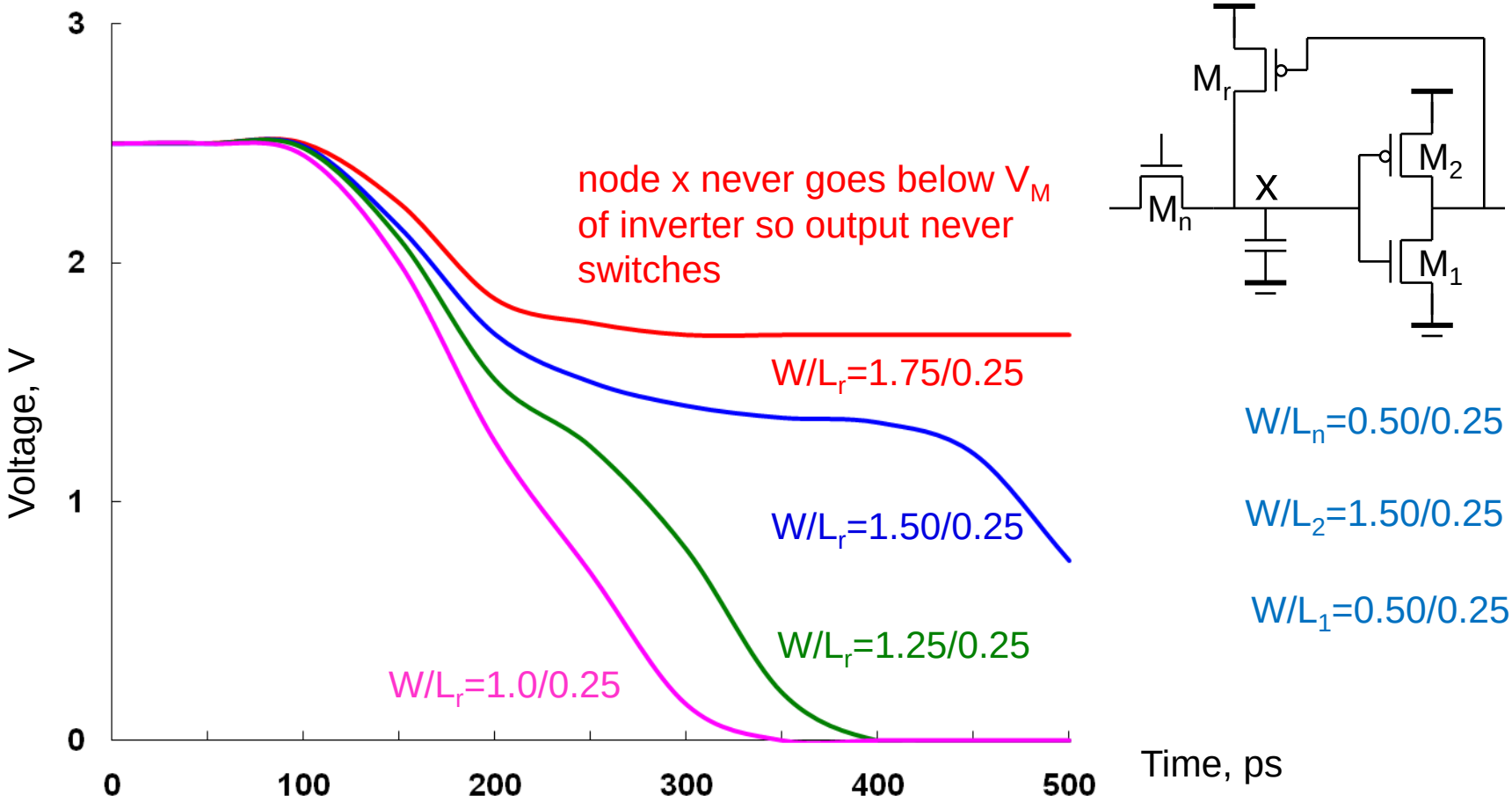
Solution 1: Level Restorer

- Full swing on x (due to Level Restorer) so no static power consumption by inverter
- No static backward current path through Level Restorer and PT since Restorer is only active when A is high
- For correct operation M_r must be sized correctly (ratioed)



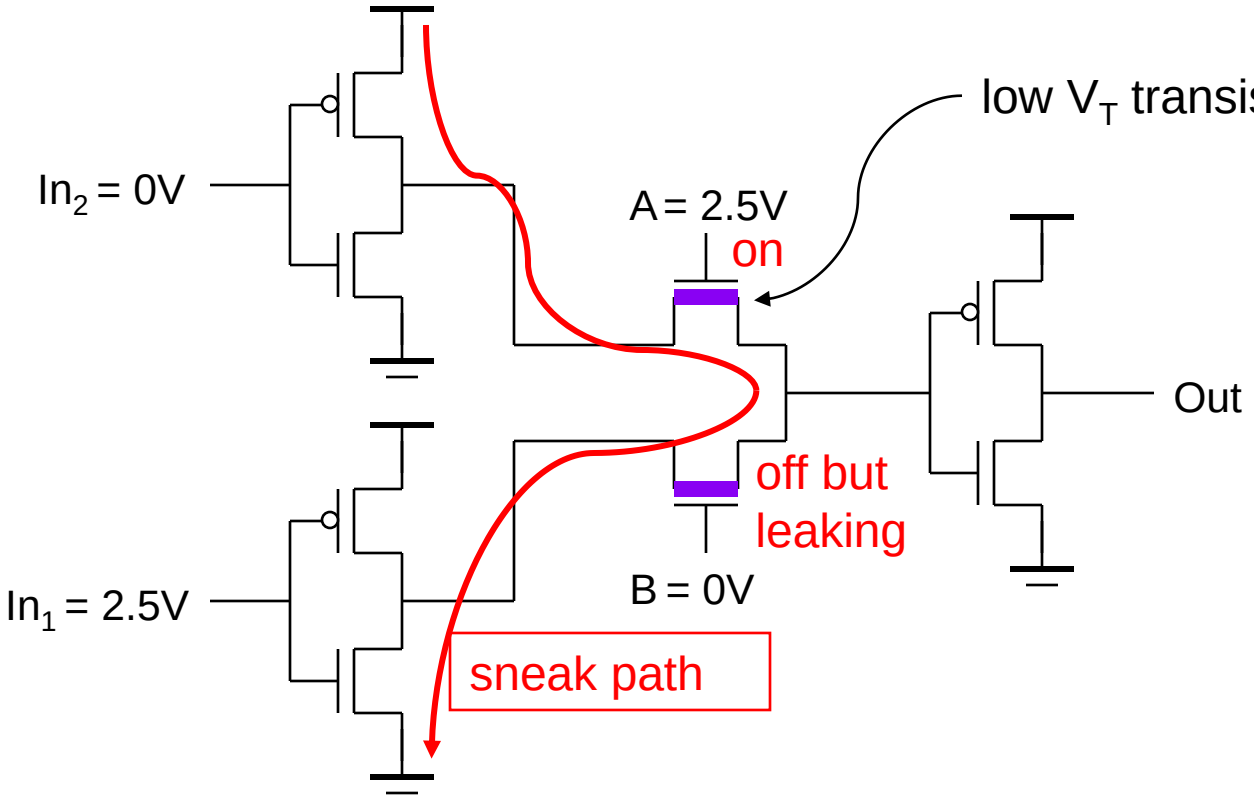
Restorer Circuit Transient Response

- Restorer has speed and power impacts: increases the capacitance at x, slowing down the gate; increases t_r (but decreases t_f)



Solution 2: Multiple V_T Transistors

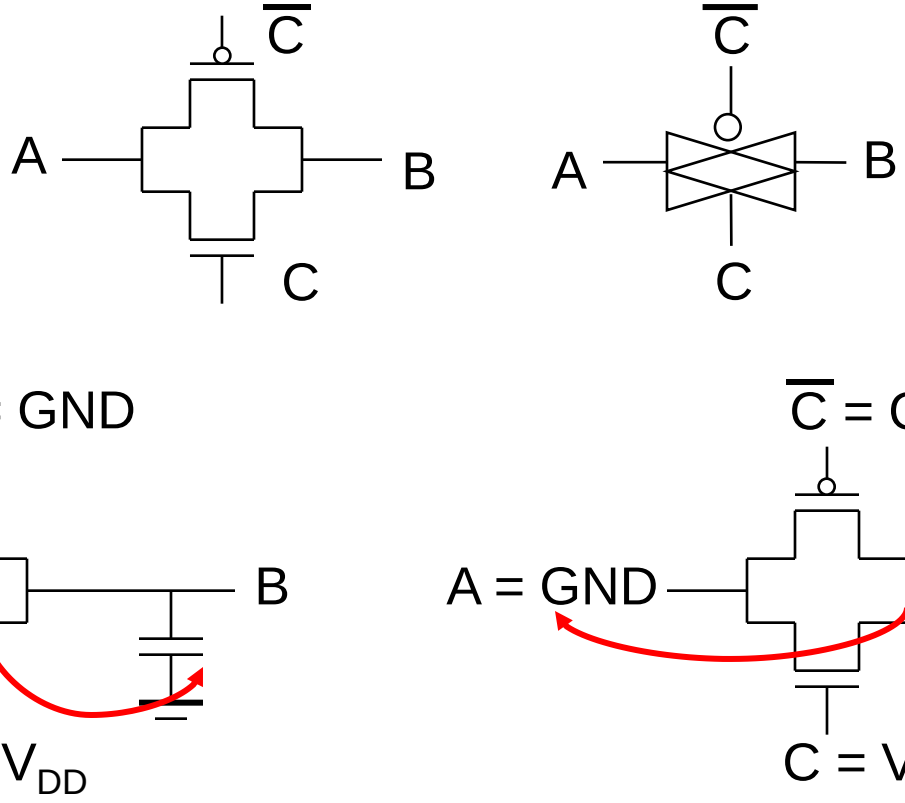
- Technology solution: Use (near) zero V_T devices for the NMOS PTs to eliminate *most* of the threshold drop (body effect still in force preventing full swing to V_{DD})



- Impacts static power consumption due to subthreshold currents flowing through the PTs (even if V_{GS} is below V_T)

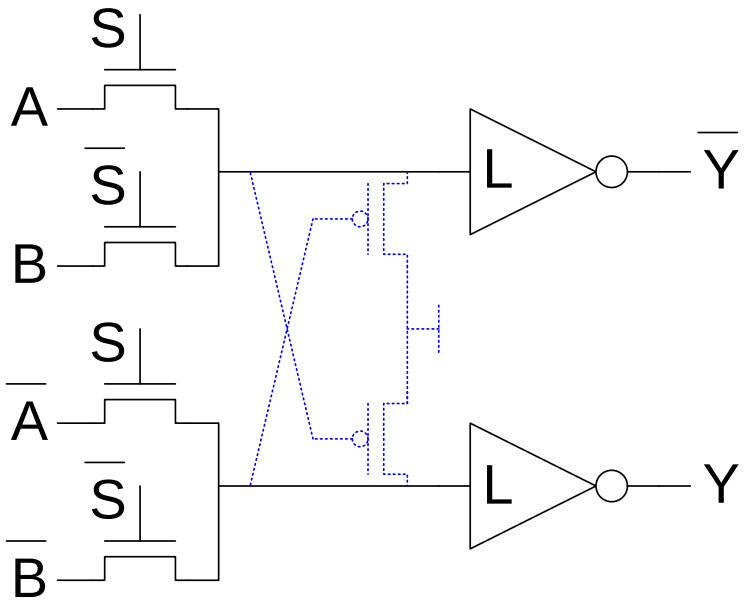
Solution 3: Transmission Gates (TGs)

- Most widely used solution
- Full swing *bidirectional* switch controlled by the gate signal C, A = B if C = 1



Solution 4: CPL

- **C**omplementary **P**ass-transistor **L**ogic
  - Dual-rail form of pass transistor logic
  - Avoids need for ratioed feedback
  - Optional cross-coupling for rail-to-rail swing



Outline

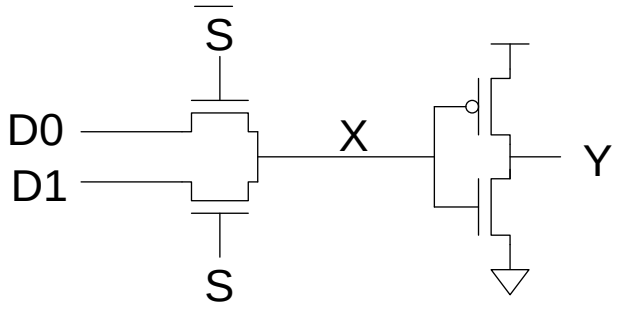
1. Static CMOS
2. Ratioed Circuits
3. Cascode Voltage Switch Logic
4. Dynamic Circuits
5. Pass-Transistor Circuits
6. **Circuit Pitfalls**

Introduction

- Circuit Pitfalls
  - Detective puzzle
  - Given circuit and symptom, diagnose cause and recommend solution
  - All these pitfalls have caused failures in real chips
- Noise Budgets
- Reliability

Threshold Drop

- Circuit
  - 2:1 multiplexer

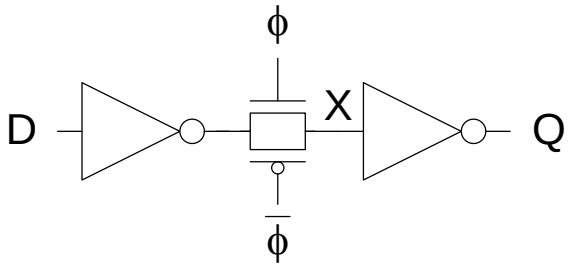


- Symptom
  - Mux works when selected D is 0 but not 1.
  - Or fails at low V_{DD}.
  - Or fails in SF corner.

- Principle: Threshold drop
  - X never rises above V_{DD}-V_t
  - V_t is raised by the body effect
  - The threshold drop is most serious as V_t becomes a greater fraction of V_{DD}.
- Solution: Use transmission gates, not pass transistors

Leakage

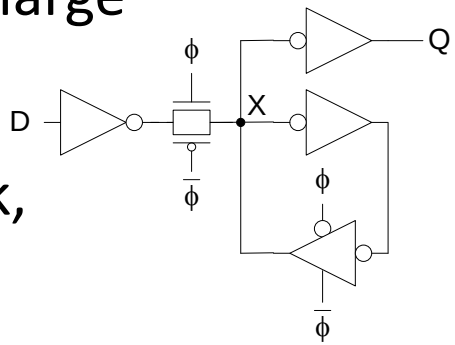
- Circuit
  - Latch



- Symptom
  - Load a 0 into Q
  - Set ϕ = 0
  - Eventually Q spontaneously flips to 1

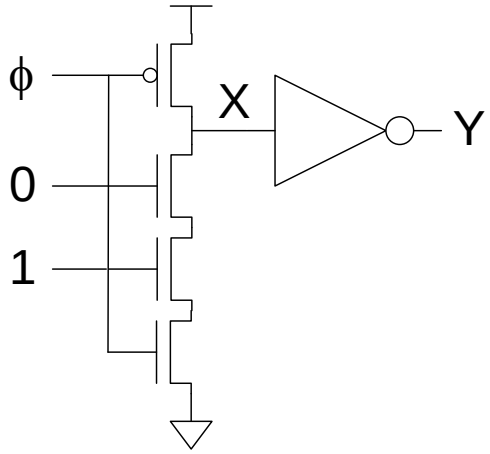
❑ Principle: Leakage
  - X is a dynamic node holding value as charge on the node
  - Eventually subthreshold leakage may disturb charge

❑ Solution: Stabilize node with feedback
  - Or periodically refresh node (requires fast clock, not practical processes with big leakage)



Leakage

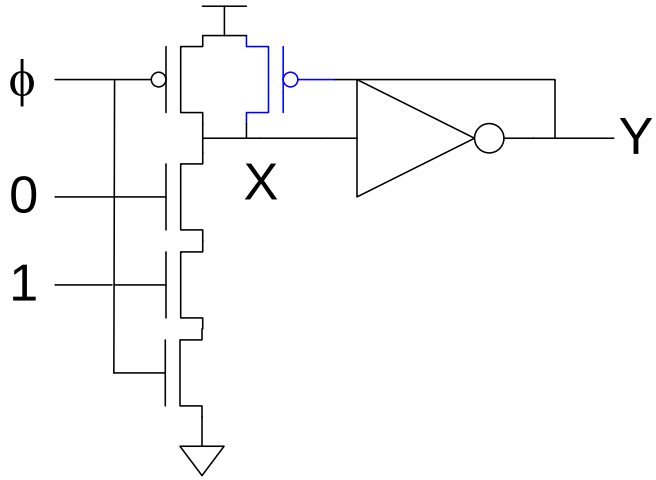
- Circuit
  - Domino AND gate



- Symptom
  - Precharge gate (Y=0)
  - Then evaluate
  - Eventually Y spontaneously flips to 1

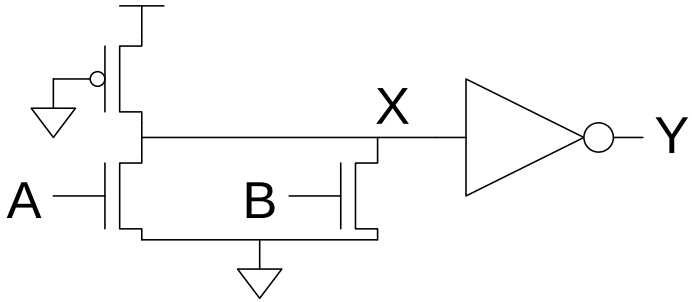
❑ Principle: Leakage
  - X is a dynamic node holding value as charge on the node
  - Eventually subthreshold leakage may disturb charge

❑ Solution: Keeper



Ratio Failure

- Circuit
  - Pseudo-nMOS OR



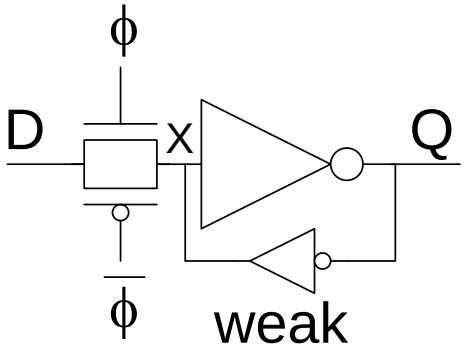
- Symptom
  - When only one input is true, Y = 0.
  - Perhaps only happens in SF corner.

❑ Principle: Ratio Failure
  - nMOS and pMOS fight each other.
  - If the pMOS is too strong, nMOS cannot pull X low enough.

❑ Solution: Check that ratio is satisfied **in all corners**

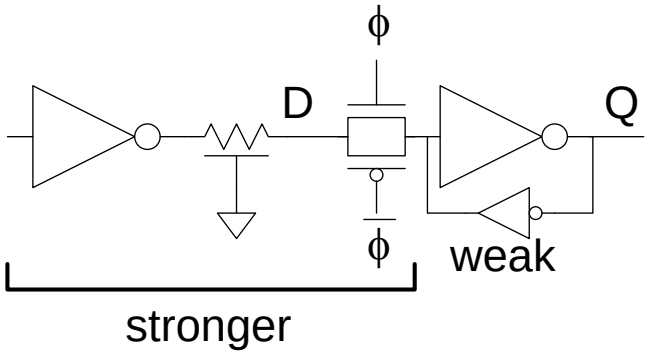
Ratio Failure

- Circuit
  - Latch



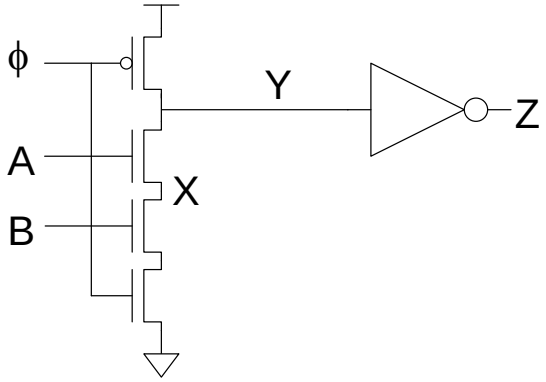
- Symptom
  - Q stuck at 1.
  - May only happen for certain latches where input is driven by a small gate located far away.

- ❑ Principle: Ratio Failure (again)
  - Series resistance of D driver, wire resistance, and tgate must be much less than weak feedback inverter.
- ❑ Solutions: Check relative strengths
  - Avoid unbuffered diffusion inputs where driver is unknown

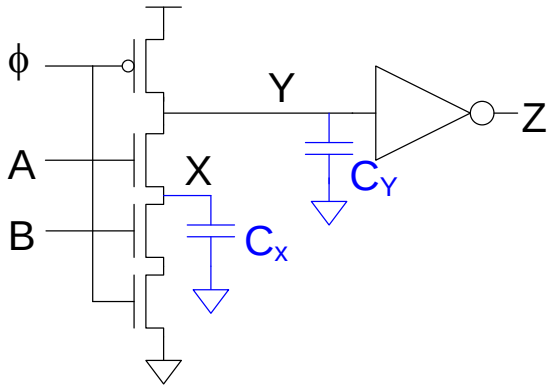


Charge Sharing

- Circuit
  - Domino AND gate



- Symptom
  - Precharge gate while A = B = 0, so Z = 0
  - Set ϕ = 1
  - A rises
  - Z is observed to sometimes rise



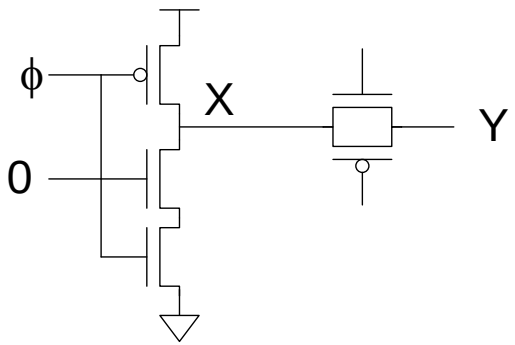
- Principle: Charge Sharing
  - If X was low, it shares charge with Y
- Solutions: Limit charge sharing

$$V_x = V_Y = \frac{C_Y}{C_x + C_Y} V_{DD}$$

  - Safe if $C_Y >> C_X$
  - Or precharge node X too

Charge Sharing

- Circuit
  - Dynamic gate + latch

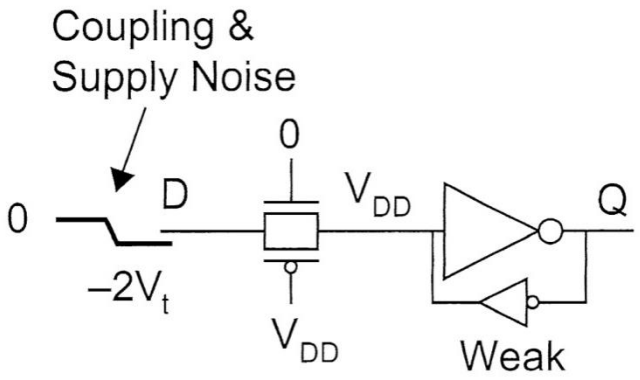


- Symptom
  - Precharge gate while transmission gate latch is opaque
  - Evaluate
  - When latch becomes transparent, X falls

- ❑ Principle: Charge Sharing
  - If Y was low, it shares charge with X
- ❑ Solution: Buffer dynamic nodes before driving transmission gate

Diffusion Input Noise

- Circuit
  - Latch

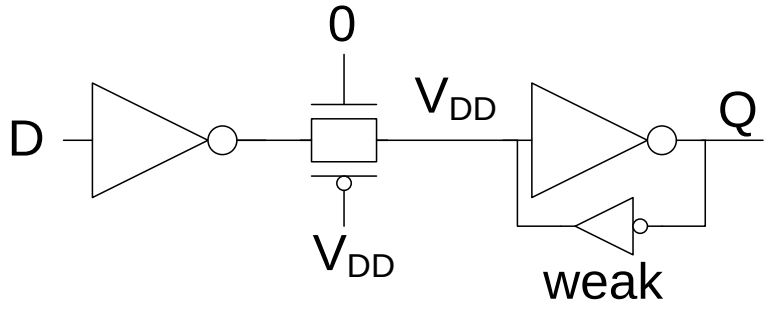


- Symptom
  - Q changes while latch is opaque
  - Especially if D comes from a far-away driver

❑ Principle: Diffusion Input Noise Sensitivity

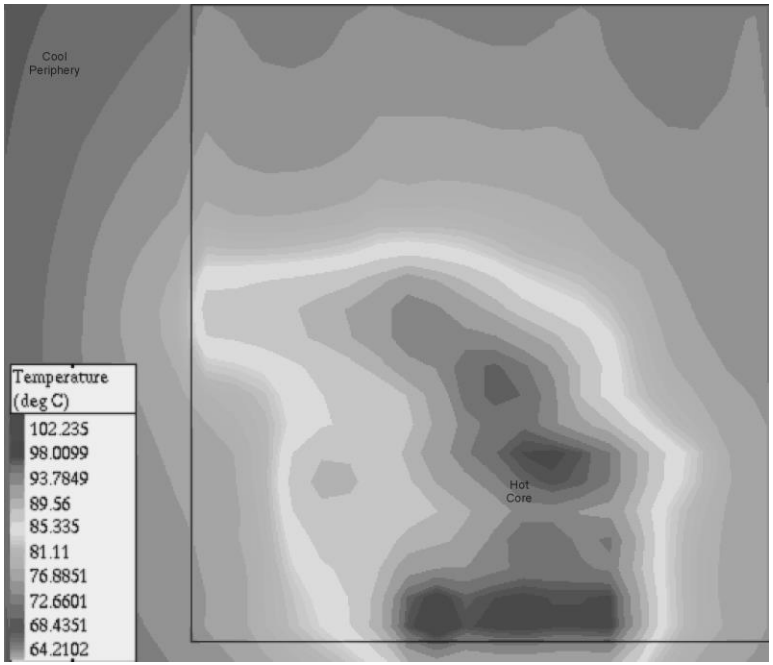
- If $D < -V_t$, transmission gate turns on
- Most likely because of power supply noise or coupling on D

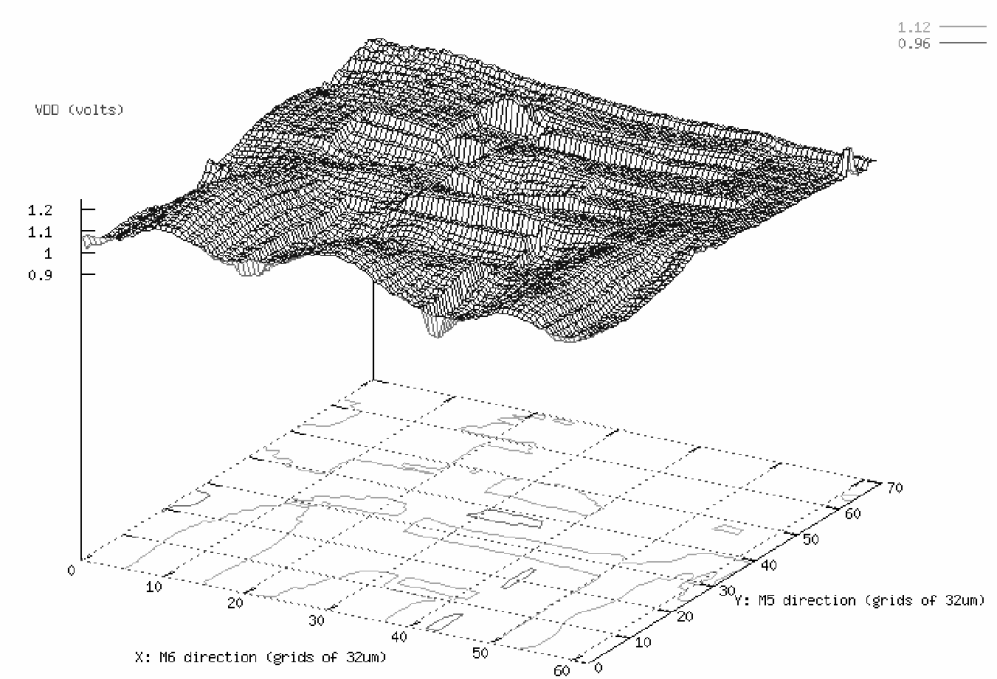
❑ Solution: Buffer D locally



Hot Spot

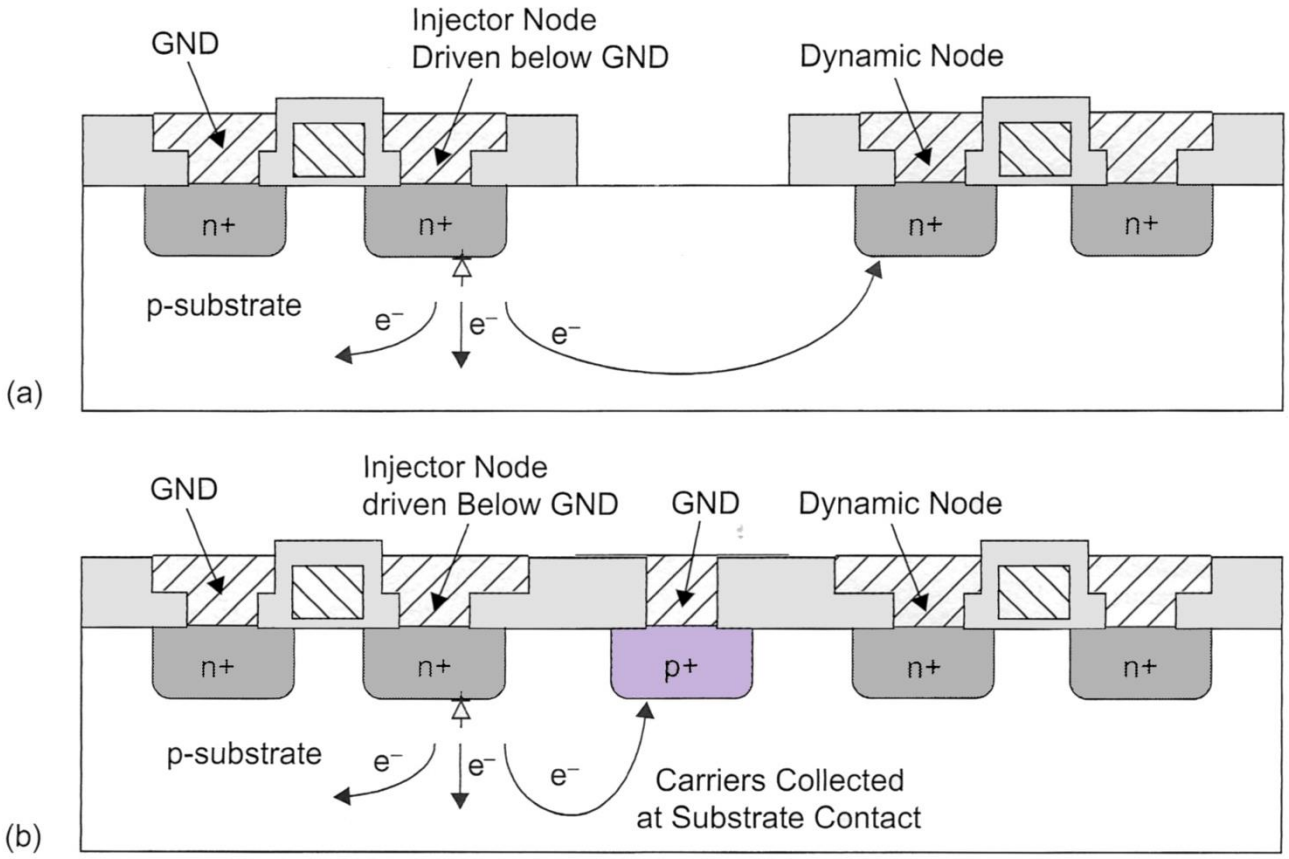
- Nonuniform power dissipation (even within overall power budget)





Minority Carrier Injection

- Minority injection caused by forward biased p-n junction
- Solution: Use guard ring to collect the excess minority



Back-Gate Coupling

- Dynamic gates drive multiple-input static CMOS gates
- Solution : Drive input closer to the rail