

---

# **Different Types of Logic Design**

# CMOS disadvantages

---

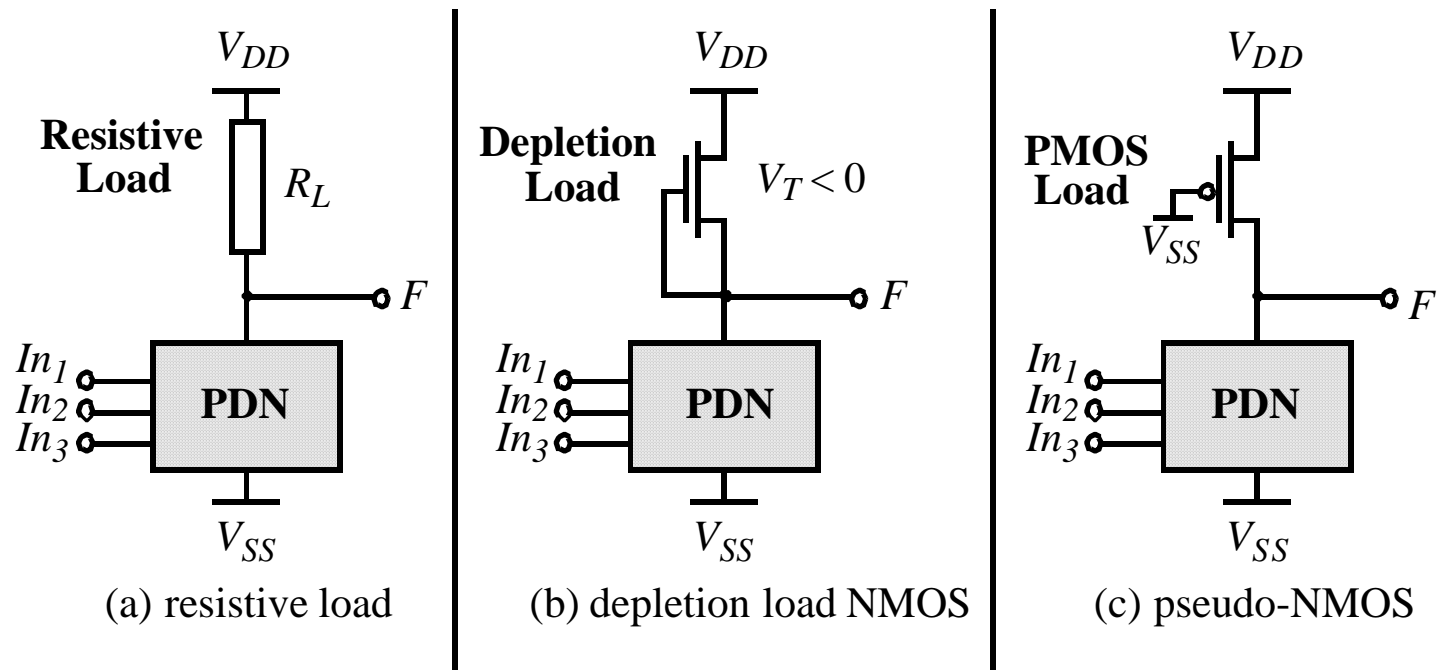
- ❑ For N-input CMOS gate,  $2N$  transistors required
  - Each input connects to an NMOS and PMOS transistor
  - Large input capacitance: limits fanout
- ❑ Large fan-in gates: always have long transistor stack in PUN or PDN
  - Limits pullup or pulldown delay
  - Requires very large transistors



## Ratioed Logic

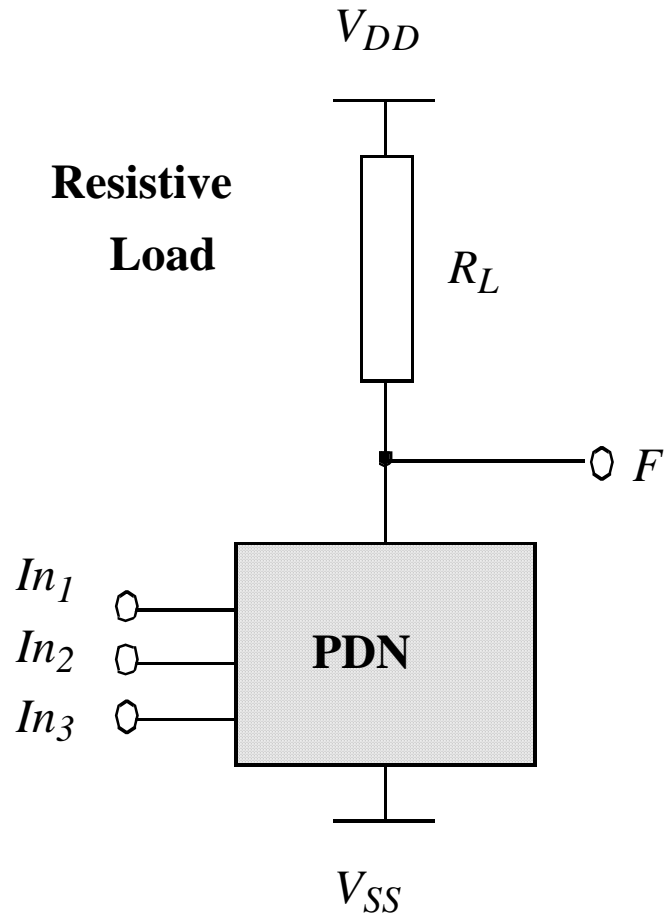
Ratioed logic is an attempt to reduce the number of transistors required to implant a given logic function, often at the cost of reduced robustness and extra power dissipation

# Ratioed Logic



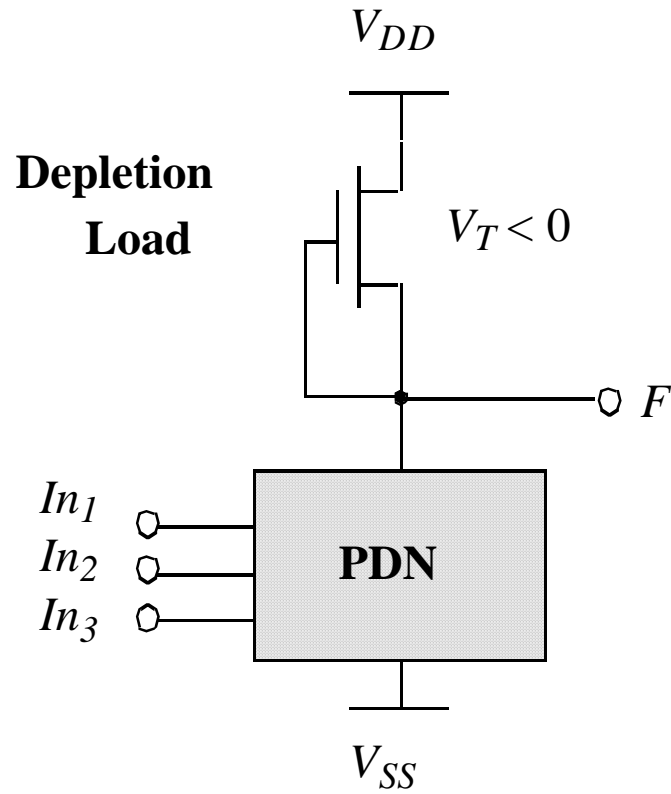
**Goal: to reduce the number of devices over complementary CMOS**

# Ratioed Logic

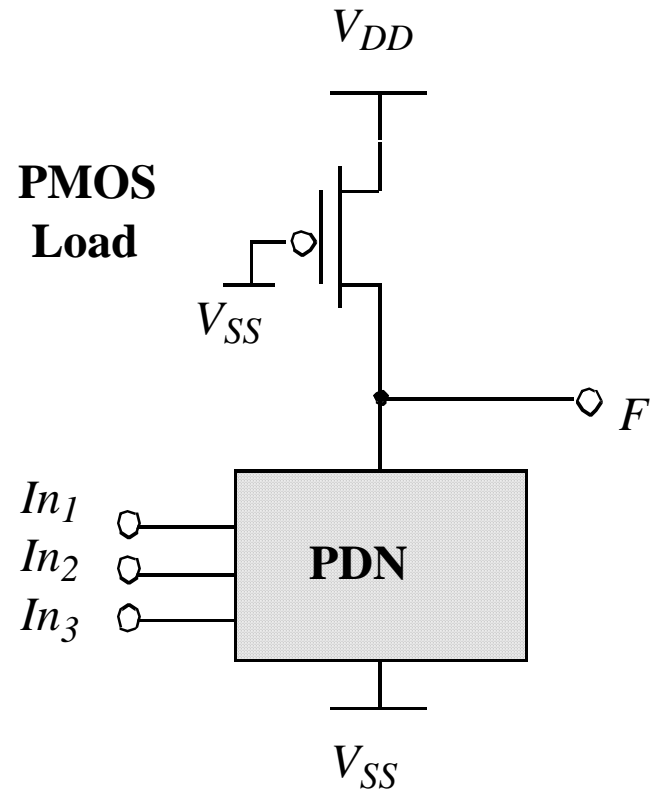


- **N transistors + Load**
- $V_{OH} = V_{DD}$
- $V_{OL} = \frac{R_{PN}}{R_{PN} + R_L}$
- **Assymetrical response**
- **Static power consumption**
- $t_{pL} = 0.69 R_L C_L$

# Active Loads



depletion load NMOS



pseudo-NMOS

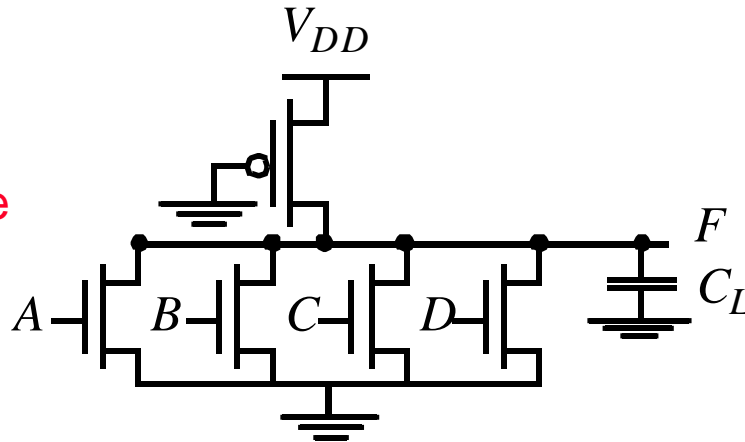
# Pseudo-NMOS logic

---

- ❑ Pseudo-NMOS: replace PMOS PUN with single “always-on” PMOS device
- ❑ Some problems as pseudo-NMOS inverter:
  - $V_{OL}$  larger than 0
  - static power when PDN is on
- ❑ Advantages
  - Replace large PMOS stacks with single device
  - Reduces overall gate size, input capacitance
  - Especially useful for wide-NOR structures

# Pseudo-NMOS

Overall functionality of the gates depend on the NMOS and PMOS size



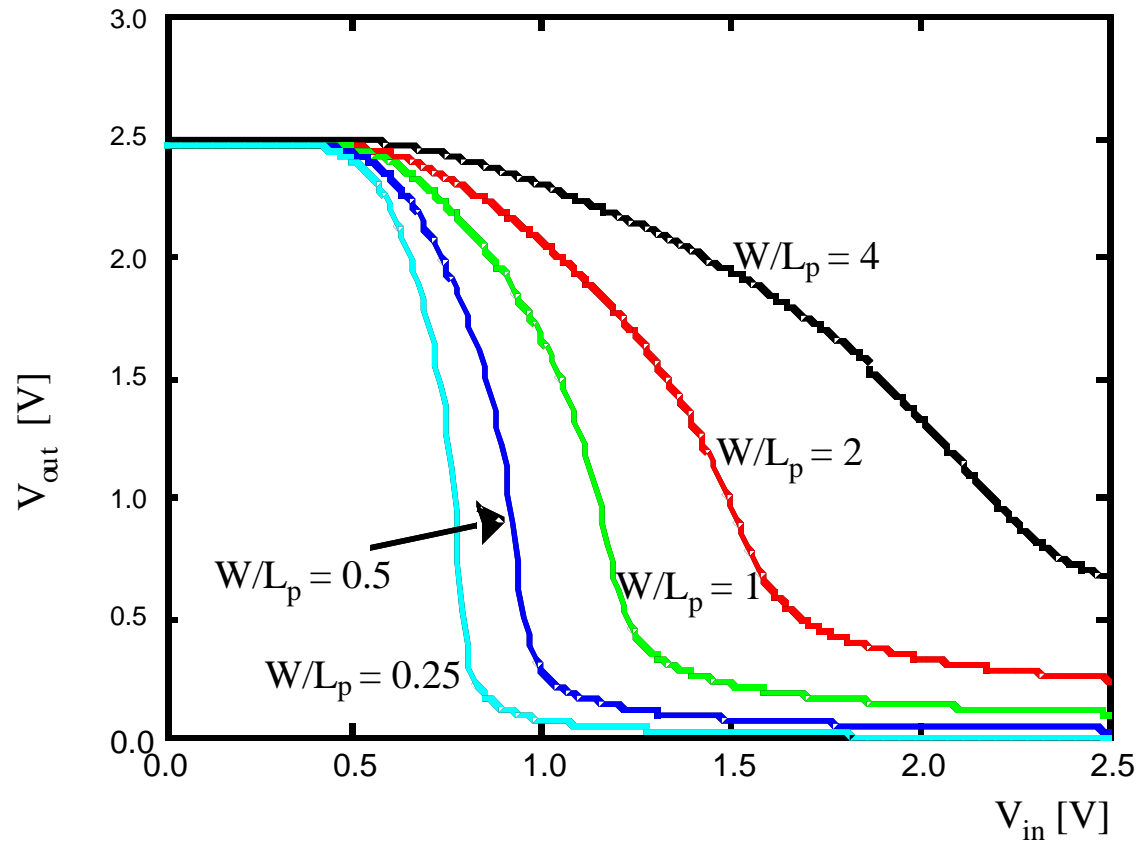
$$V_{OH} = V_{DD} \text{ (similar to complementary CMOS)}$$

$$k_n \left( (V_{DD} - V_{Tn}) V_{OL} - \frac{V_{OL}^2}{2} \right) = \frac{k_p}{2} (V_{DD} - |V_{Tp}|)^2$$

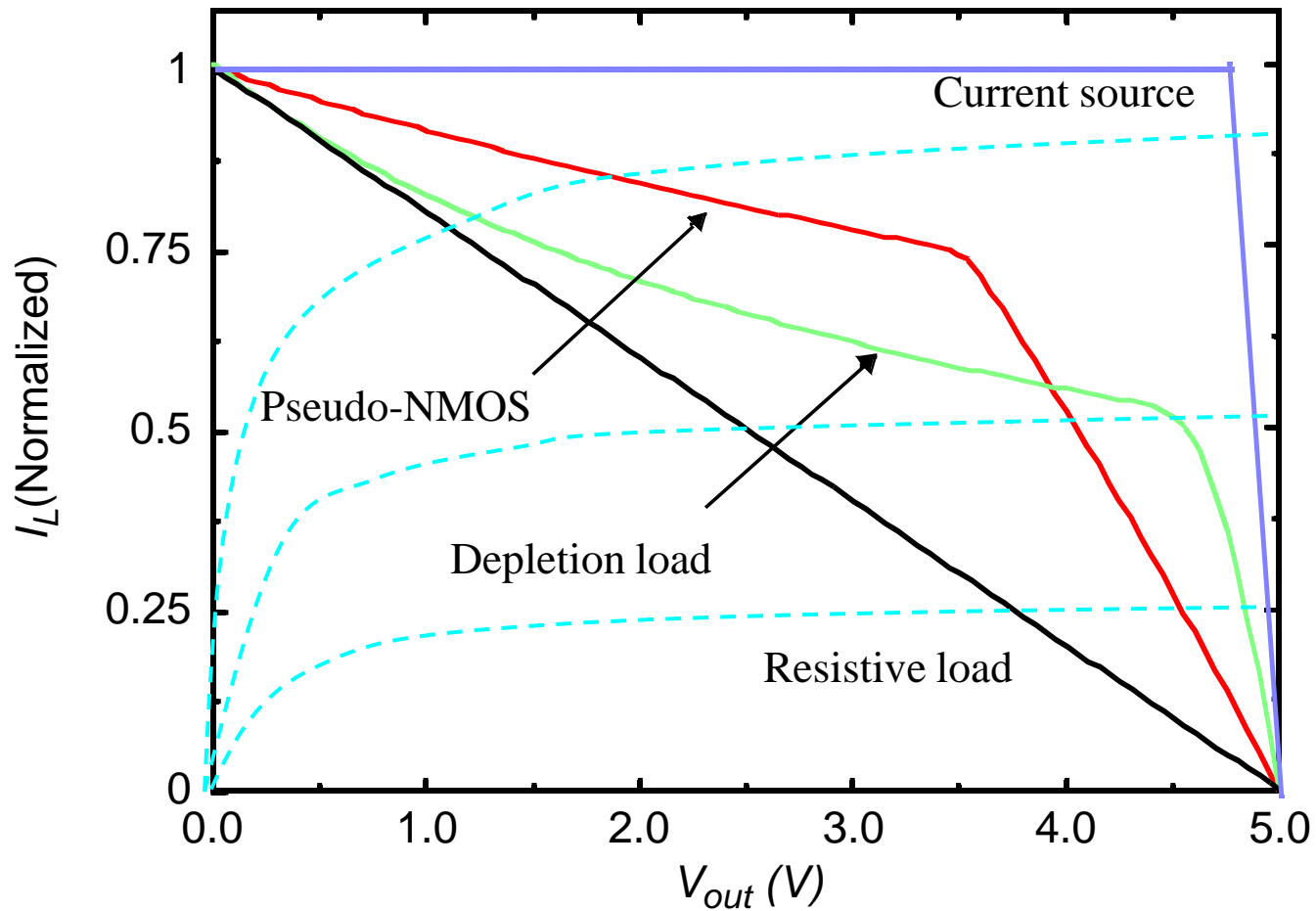
$$V_{OL} = (V_{DD} - V_T) \left[ 1 - \sqrt{1 - \frac{k_p}{k_n}} \right] \text{ (assuming that } V_T = V_{Tn} = |V_{Tp}|) \neq 0$$

***SMALLER AREA & LOAD BUT STATIC POWER DISSIPATION!!!***

# Pseudo-NMOS VTC



# Load Lines of Ratioed Gates



# 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 on  $n + 2$  ( $n+1$  N-type + 1 P-type) transistors

# Static vs Dynamic Storage

---

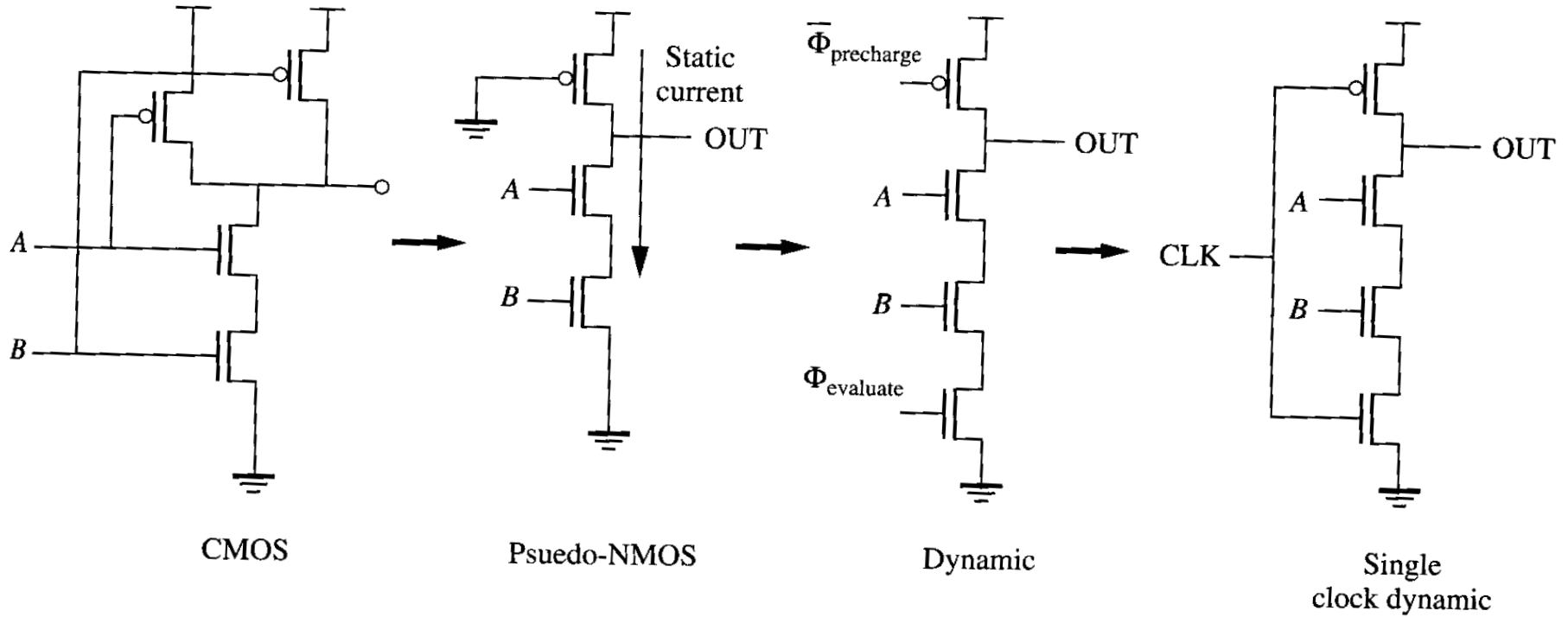
## ❑ Static storage

- preserve state as long as the power is on
- have positive feedback (**regeneration**) with an internal connection between the output and the input
- useful when updates are infrequent (clock gating)

## ❑ Dynamic storage

- store state on parasitic capacitors
- only hold state for short periods of time (milliseconds)
- require periodic refresh
- usually simpler, so higher speed and lower power

# Evolution from static to dynamic logic gate



# Dynamic CMOS

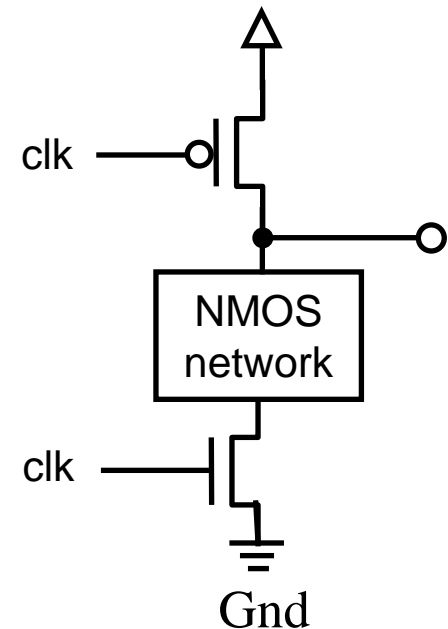
---

## ❑ Advantages:

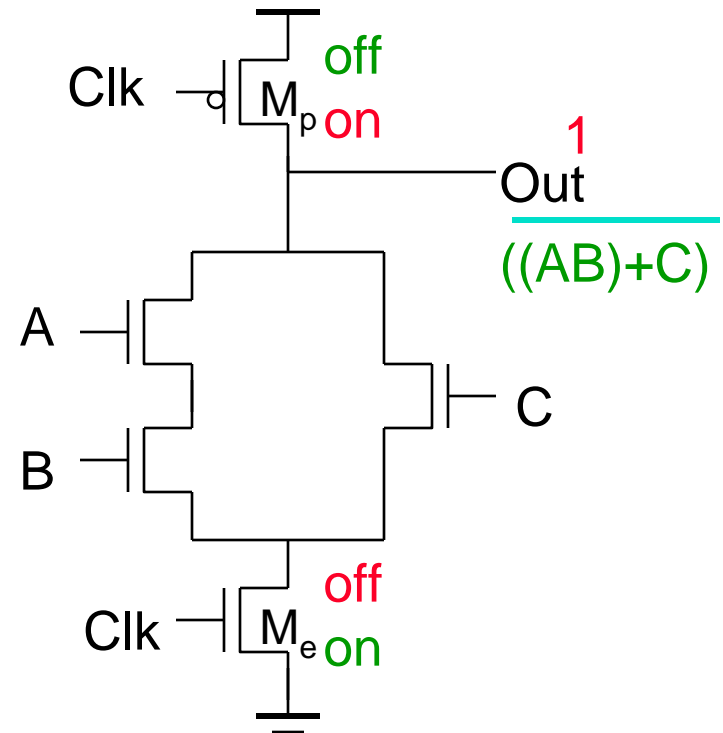
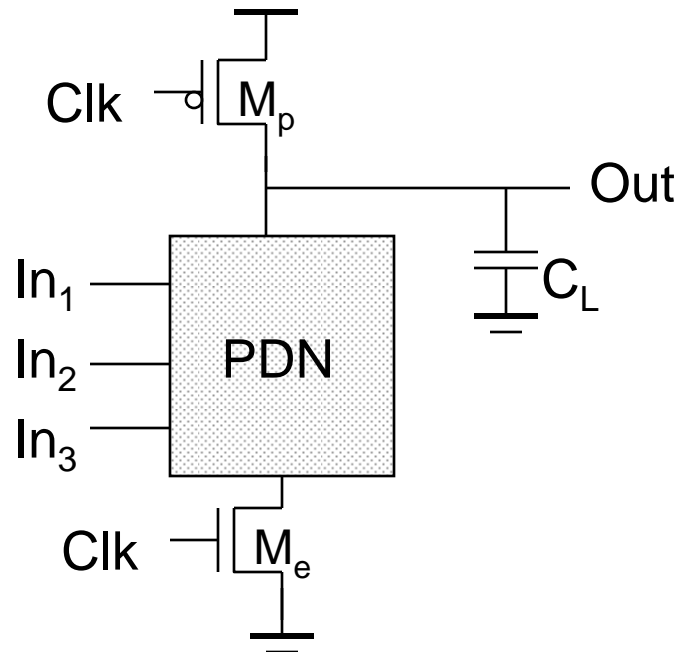
- Faster – why?
  - Reduced input load
  - No switching contention
- Less layout area

## ❑ Disadvantages:

- Charge leakage
- Charge sharing
- Capacitive coupling
- Cannot be cascaded
- Complicated timing/clocking
- Higher power
- Lower noise margins



# Dynamic Gate



Two phase operation

Precharge ( $Clk = 0$ )

Evaluate ( $Clk = 1$ )

# 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  $C_L$

This behavior is fundamentally different than the static counterpart that always has a low resistance path between the output and one of the power rails.

# Properties of Dynamic CMOS Gates

---

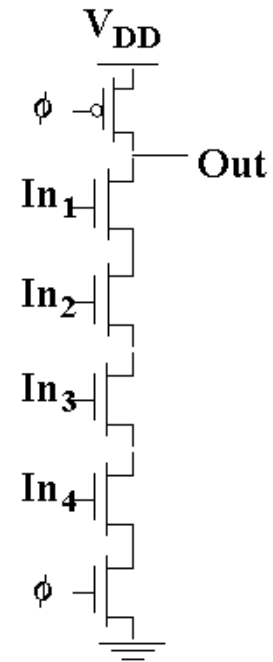
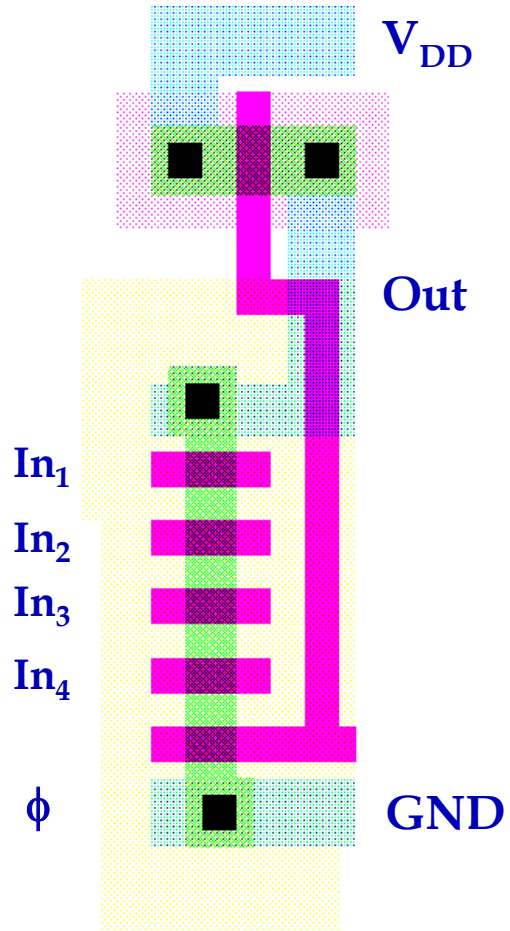
- ❑ Logic function is implemented by the PDN only
  - number of transistors is  $N + 2$  (versus  $2N$  for static complementary CMOS)
- ❑ Full swing outputs ( $V_{OL} = \text{GND}$  and  $V_{OH} = V_{DD}$ )
- ❑ Non-ratioed - sizing of the devices does not affect the logic levels
- ❑ Faster switching speeds
  - reduced load capacitance due to lower input capacitance ( $C_{in}$ )
  - reduced load capacitance due to smaller output loading ( $C_{out}$ )
  - no  $I_{sc}$ , so all the current provided by PDN goes into discharging  $C_L$

# Properties of Dynamic Gates, con't

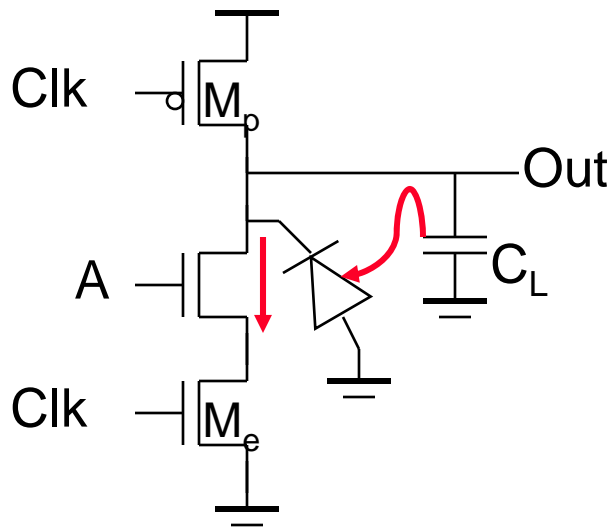
---

- ❑ 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

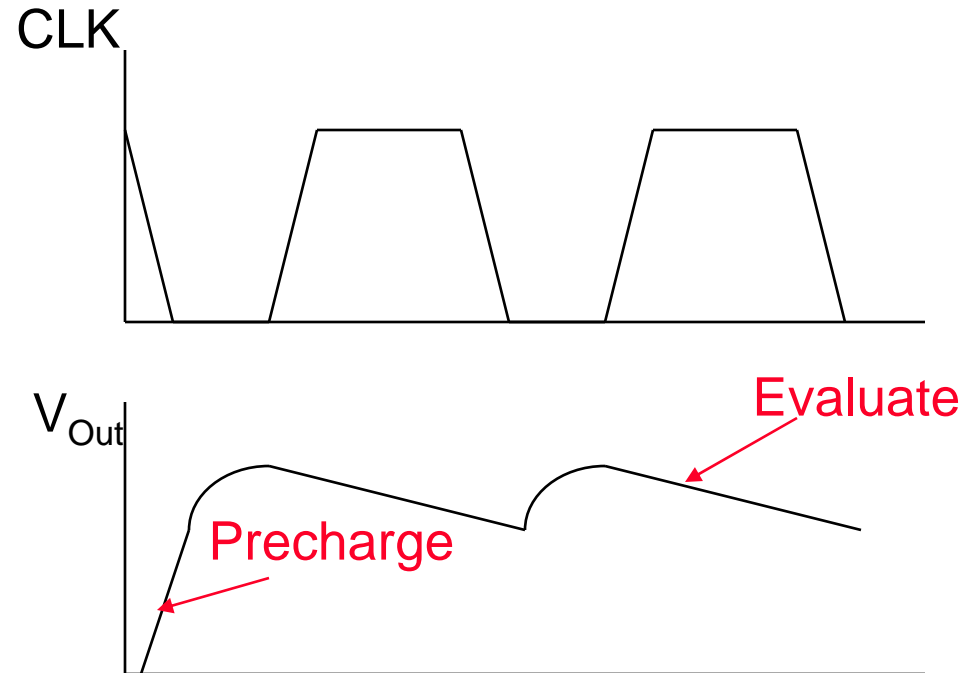
# Dynamic 4 Input NAND Gate



# Issues in Dynamic Design 1: Charge Leakage



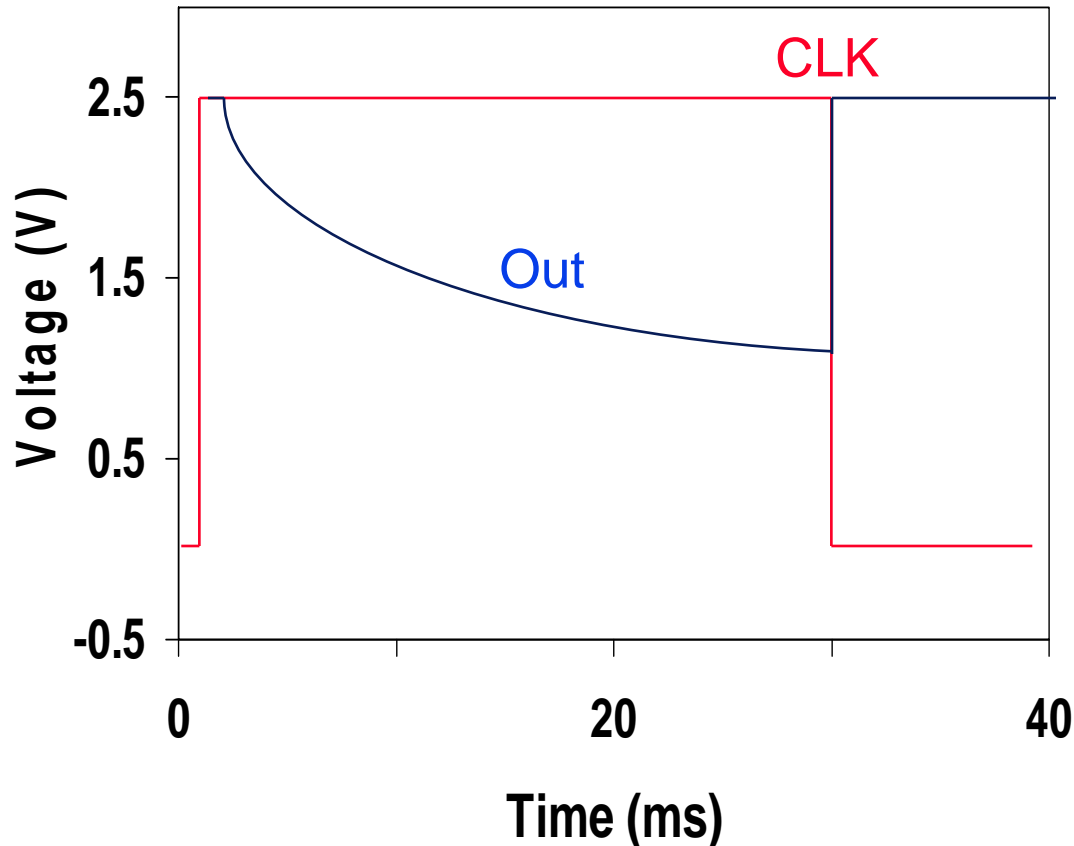
Leakage sources



*Dominant component is subthreshold current*

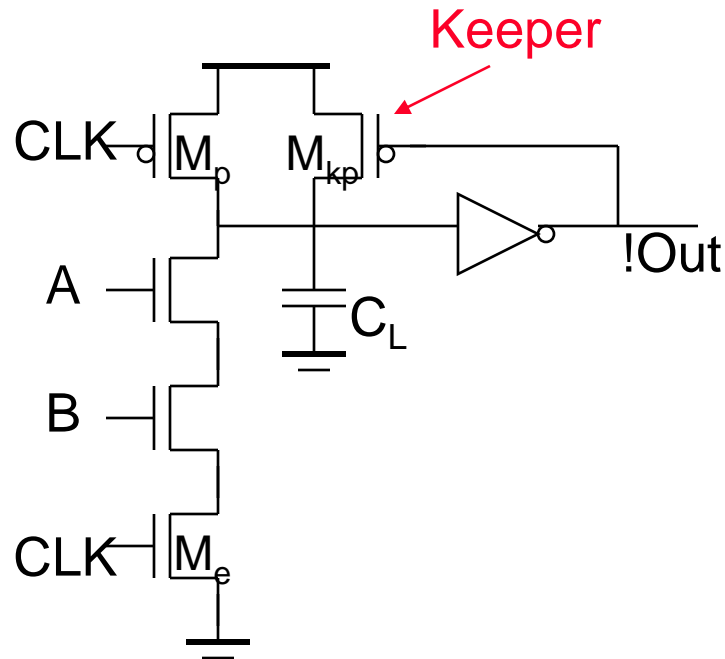
# Impact of Charge Leakage

- ❑ 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

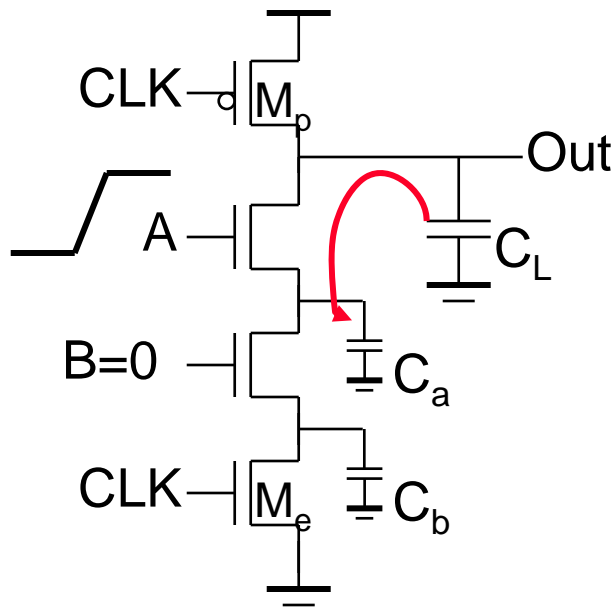
- ❑ **Keeper** compensates for the charge lost due to the pull-down leakage paths.



Same approach as level restorer for pass transistor logic

# Issues in Dynamic Design 2: Charge Sharing

---

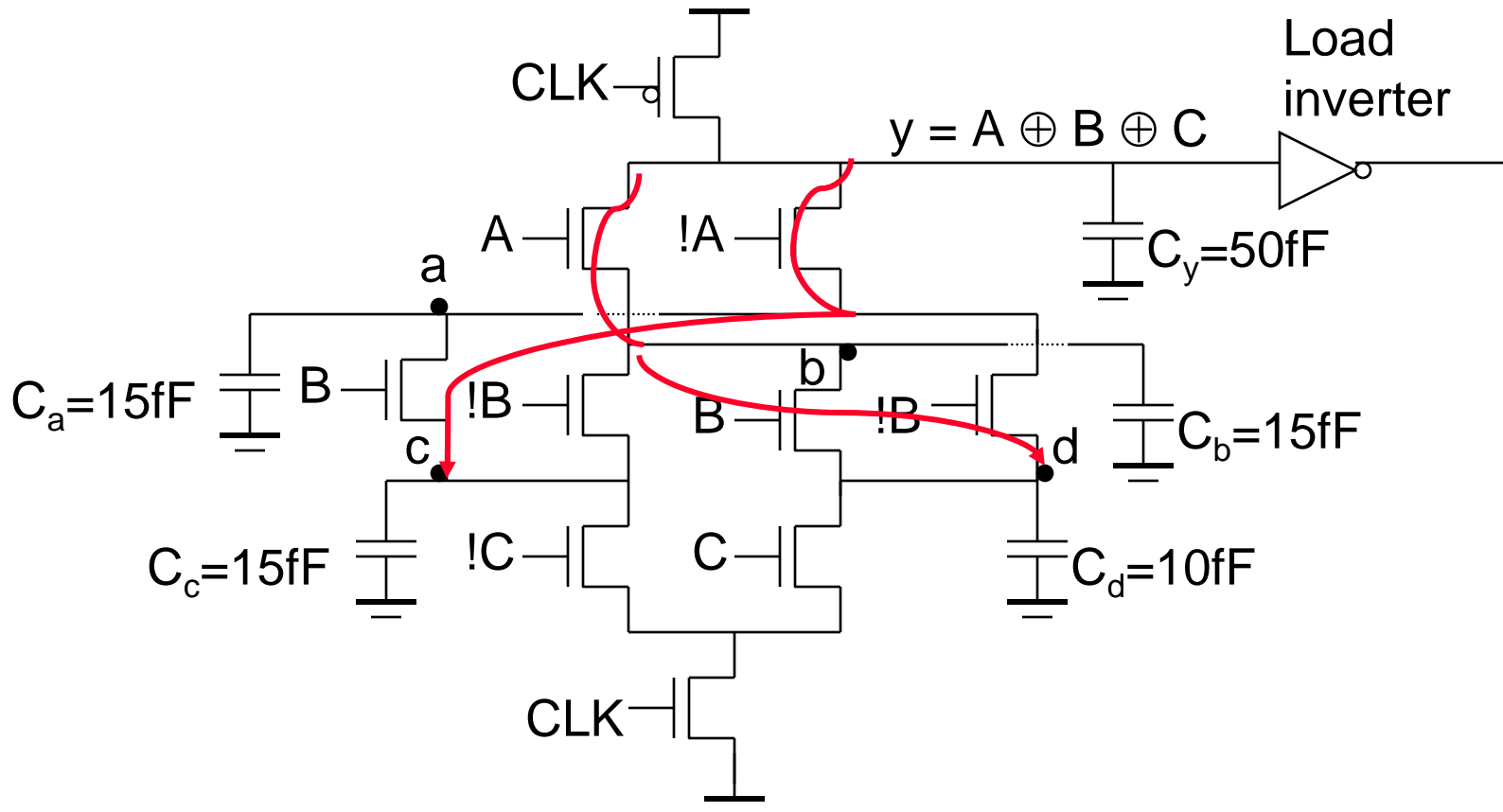


Charge stored originally on  $C_L$  is redistributed (shared) over  $C_L$  and  $C_A$  leading to static power consumption by downstream gates and possible circuit malfunction.

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 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.)

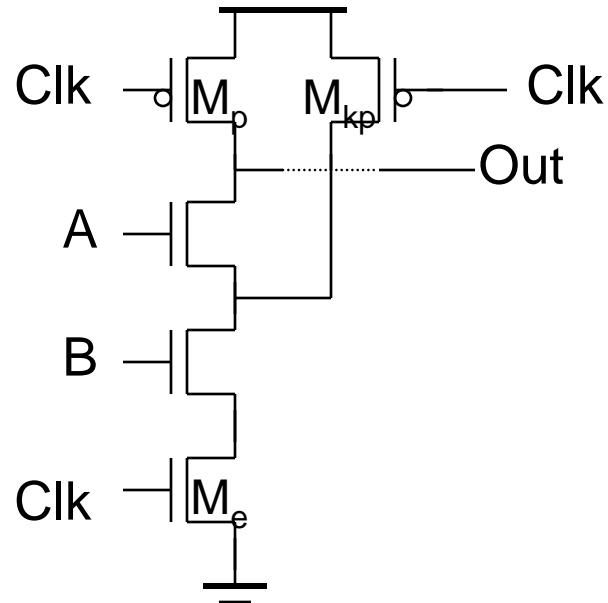


$$\Delta V_{\text{out}} = -V_{\text{DD}} \left( \frac{C_a + C_c}{(C_a + C_c) + C_y} \right)$$

$$= -2.5V \cdot \left( \frac{30}{(30+50)} \right) = -0.94V, \text{ so the output drops to } -2.5 + 0.94 = -1.56V$$

# Solution to Charge Redistribution

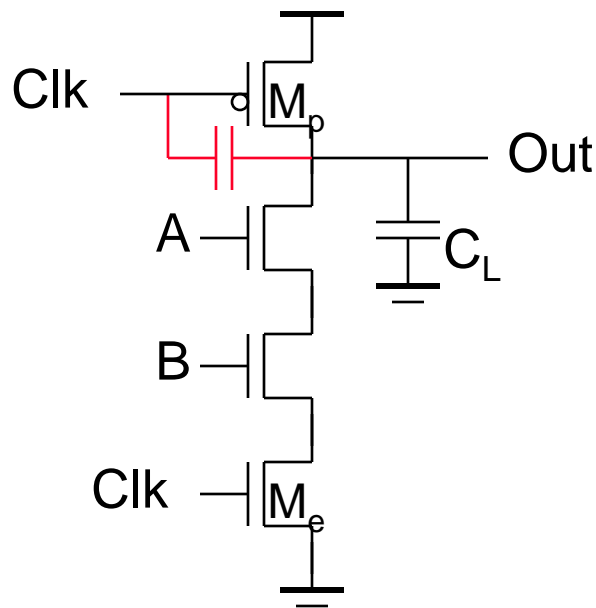
---



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

# Issues in Dynamic Design 4: Clock Feedthrough

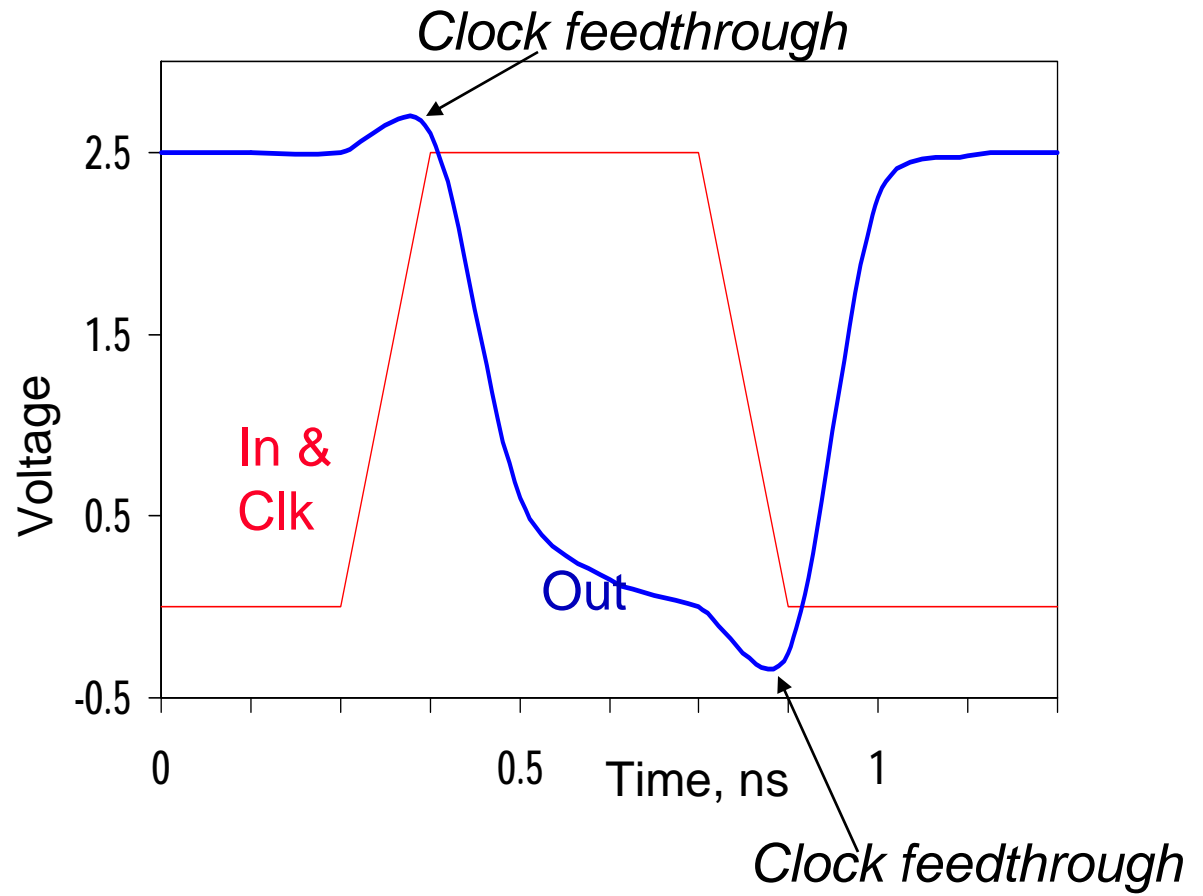
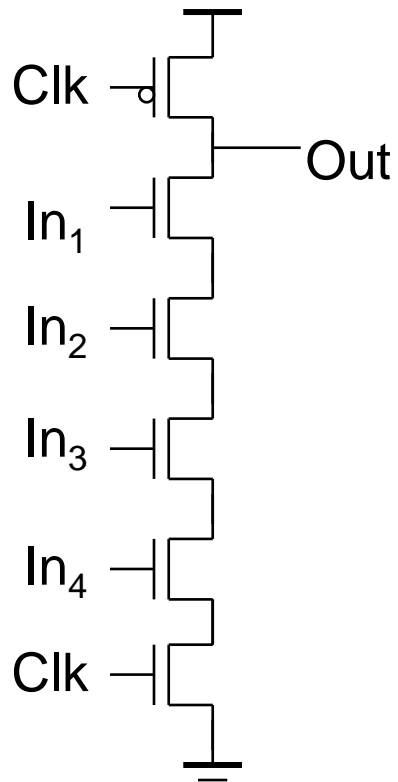
---



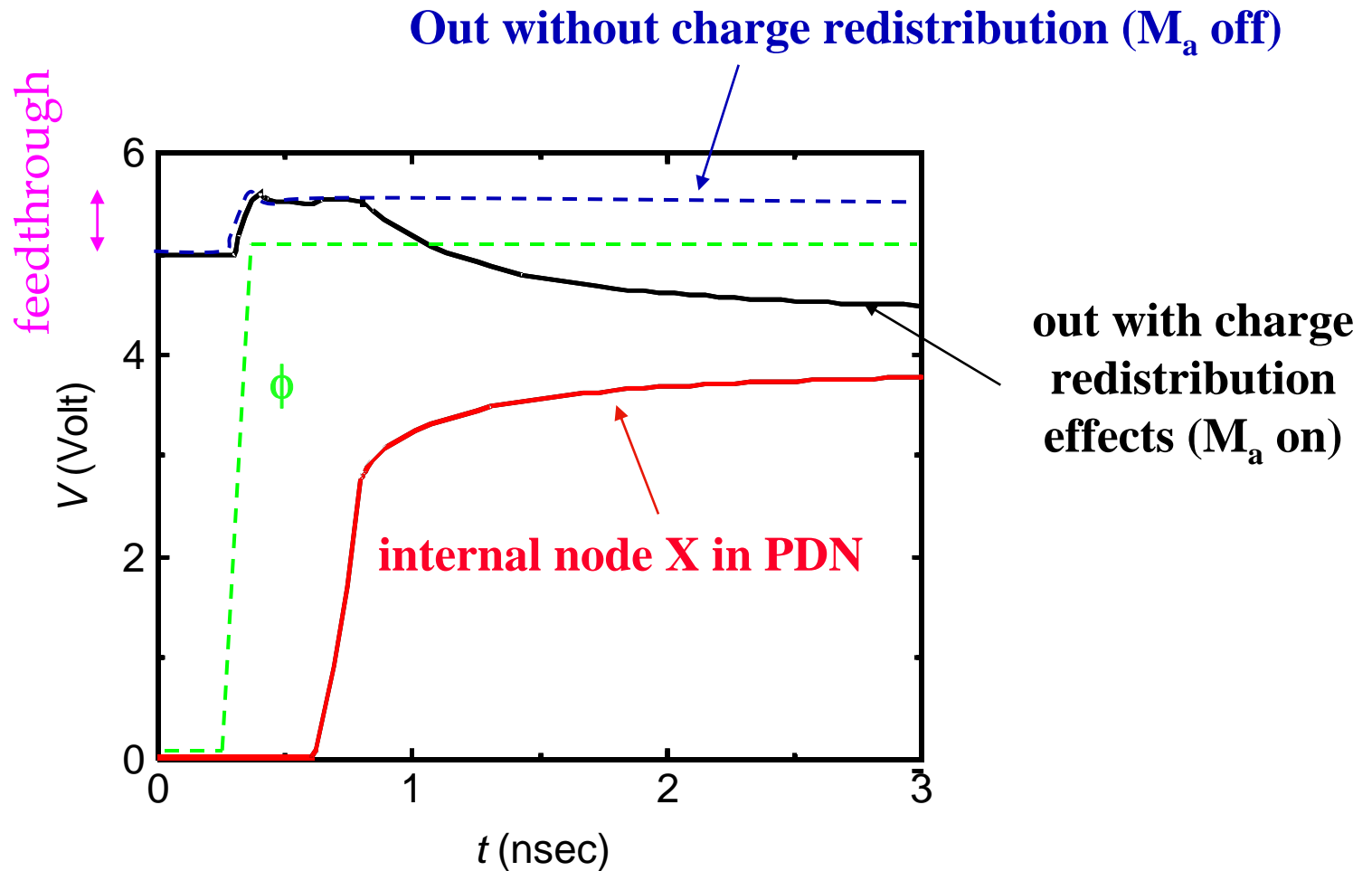
Coupling between Out and Clk input of the precharge device due to the gate to drain capacitance. So voltage of Out can rise above  $V_{DD}$ . The fast rising (and falling edges) of the clock **couple** to Out.

The danger of the clock feedthrough is that it may cause the normally reverse bias junction diodes of the precharged transistor to become forward bias.

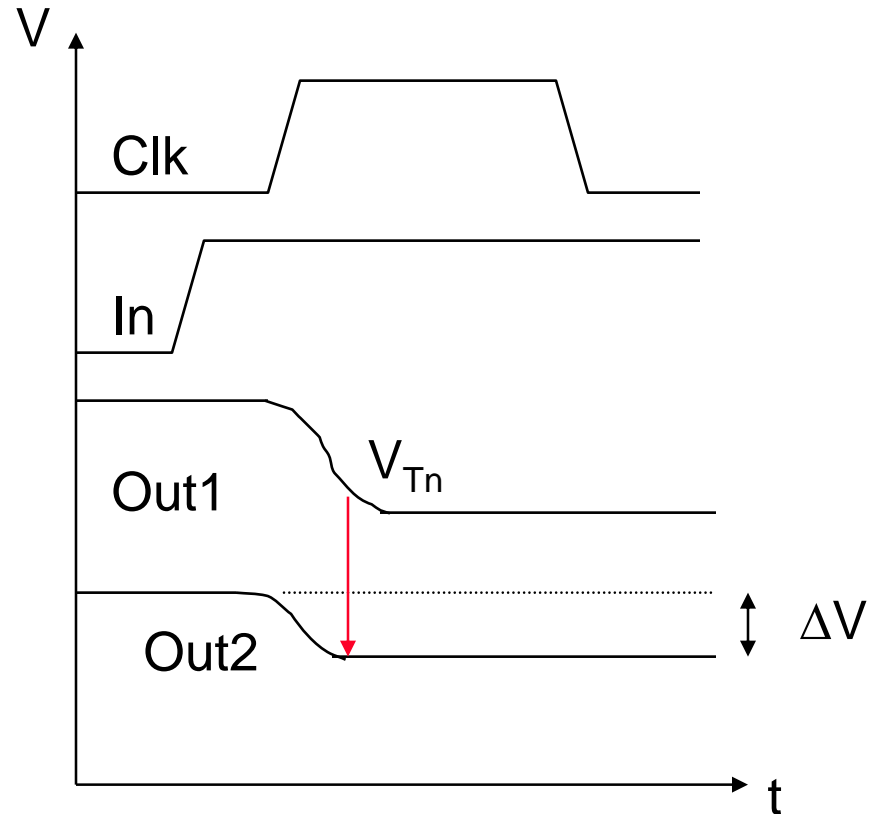
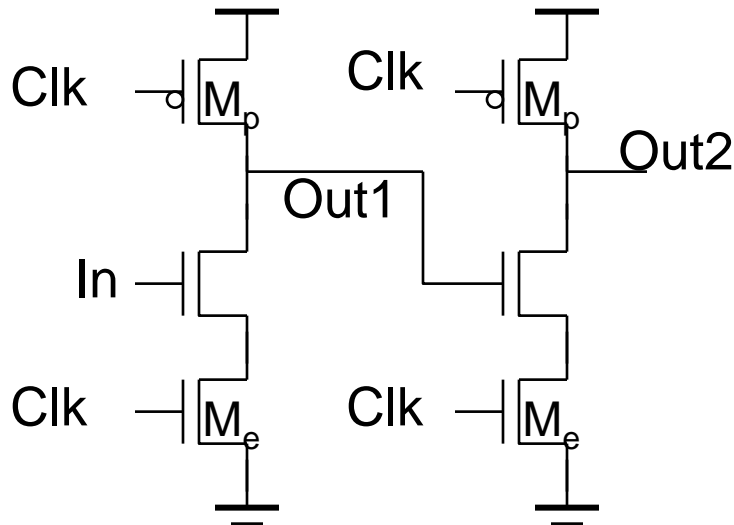
# Clock Feedthrough



# Clock Feedthrough and Charge Sharing



# Cascading Dynamic Gates issues



Out<sub>2</sub> should remain at  $V_{DD}$  since Out<sub>1</sub> transitions to 0 during evaluation. However, since there is a finite propagation delay for the input to discharge Out<sub>1</sub> to GND, the second output also starts to discharge.

Only one stage at a time should make a 1 to 0 transition!

The second dynamic inverter turns off (PDN) when Out<sub>1</sub> reaches  $V_{Tn}$ .

Only 0 → 1 transitions allowed at inputs!



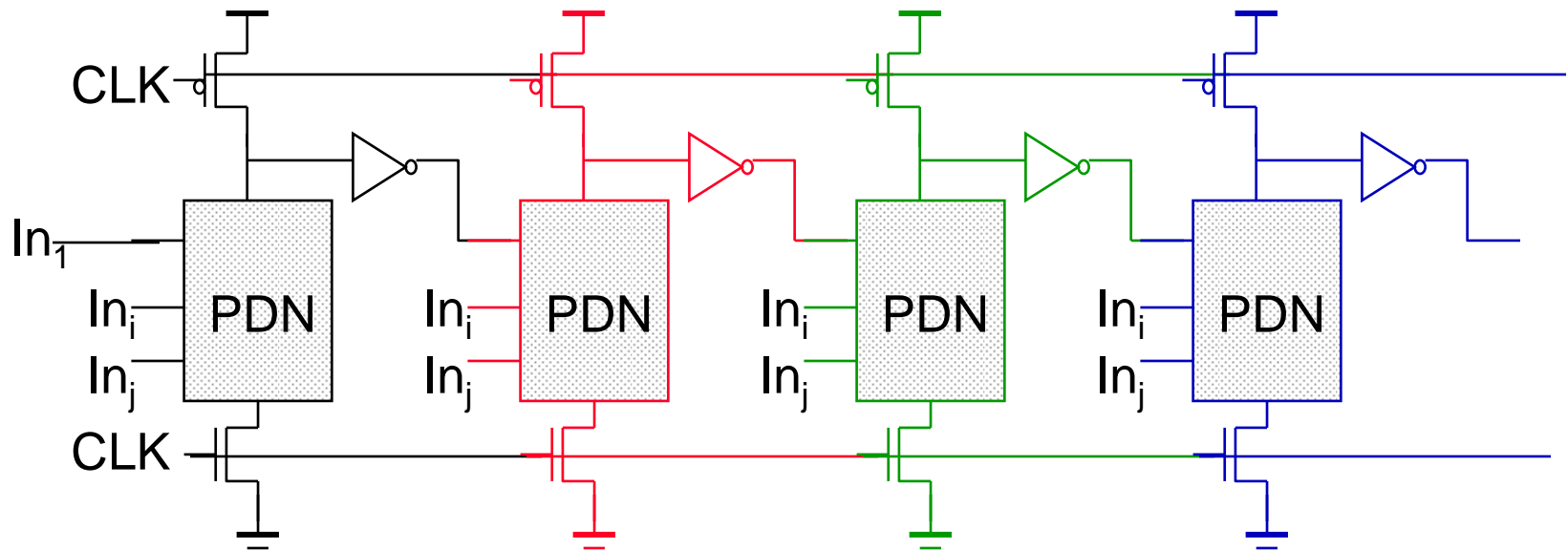
# Domino Logic

---

- ❑ Solves problem of cascading dynamic gates, but is non-inverting
  - Add an inverter between dynamic gates
    - Inverter drives the gate's fanout – increased performance
  - Sometimes the inverter is replaced with a more complex static CMOS gate
  - Static CMOS gate improves dynamic noise margins
- ❑ Solve non-inverting problem by implementing both  $F$  and  $\overline{F}$  separately
  - Area/power doubles

# Why Domino?

---



During the precharge phase all input will be turned off because all buffer output are 0.

During the evaluation phase, each buffer output at most can make one transition from 0 to 1.

Like falling dominos!