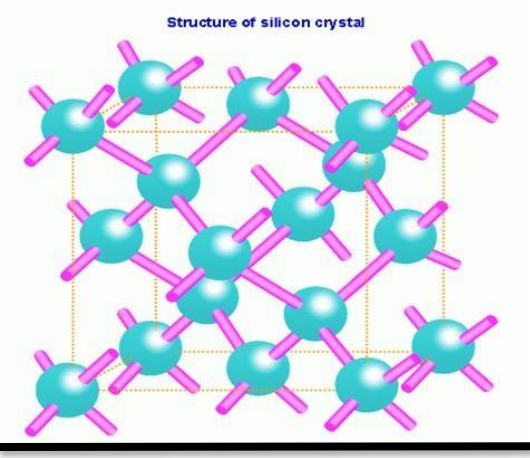
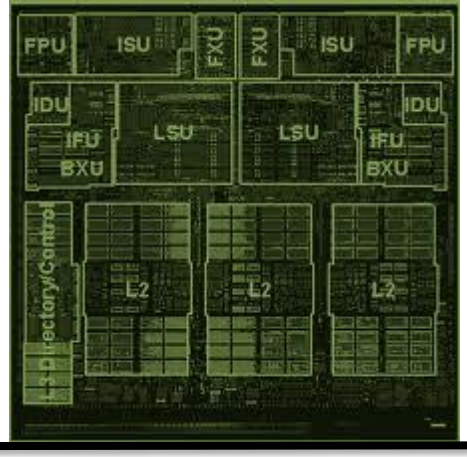
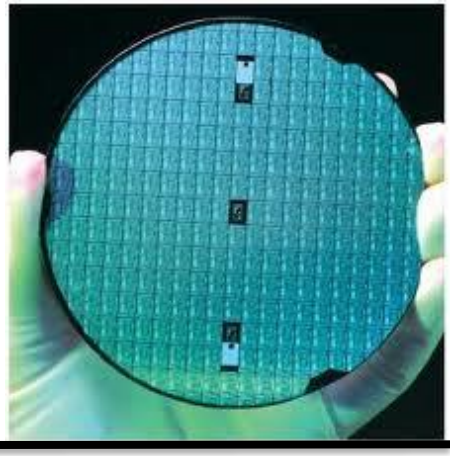






ECE 122A

VLSI Principles

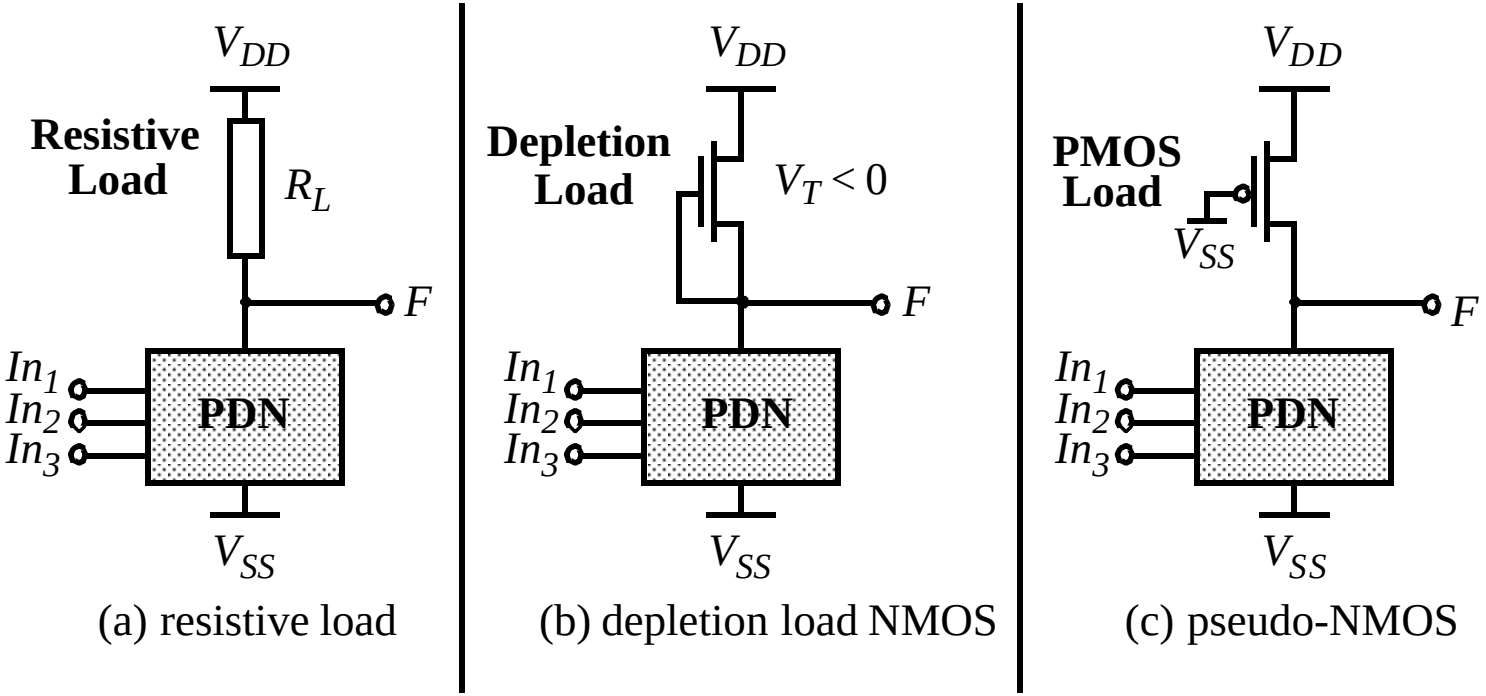
Lectures 14/15

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

Ratioed Logic

Ratioed Logic

Need N+1 transistors vs 2N for complementary CMOS

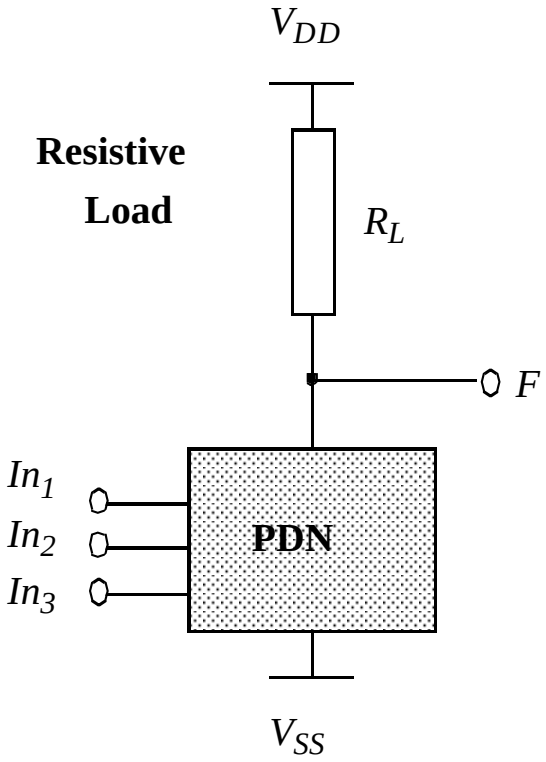


Note: a depletion mode NMOS is normally ON…an n-type channel connects the source and drain and a negative gate bias is needed to turn it off…..

Goal: to reduce the number of devices over complementary CMOS

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

Ratioed Logic: Resistive Load

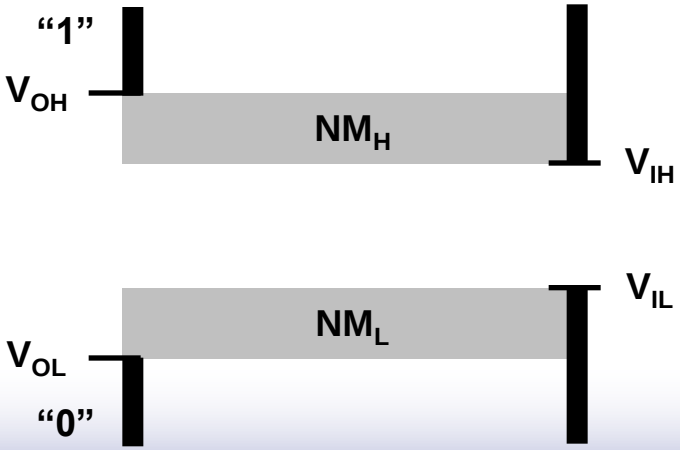


- N transistors + Load
- $V_{OH} = V_{DD}$
- $V_{OL} = \dfrac{R_{PN}}{R_{PN} + R_L}$
- Assymetrical response
- Static power consumption
- $t_{pL} = 0.69\, R_L C_L$

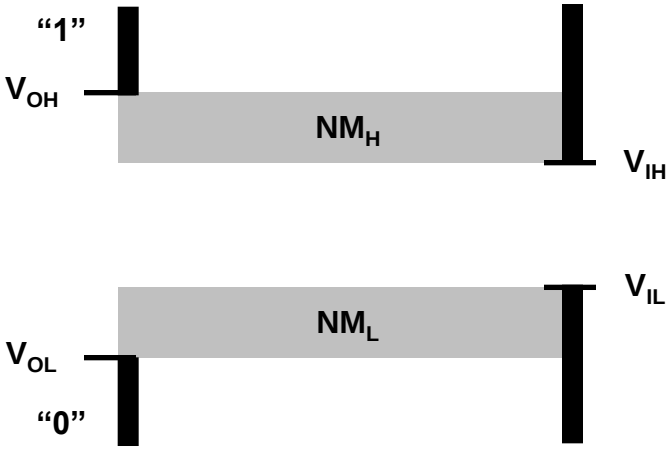
Recall a voltage divider circuit….

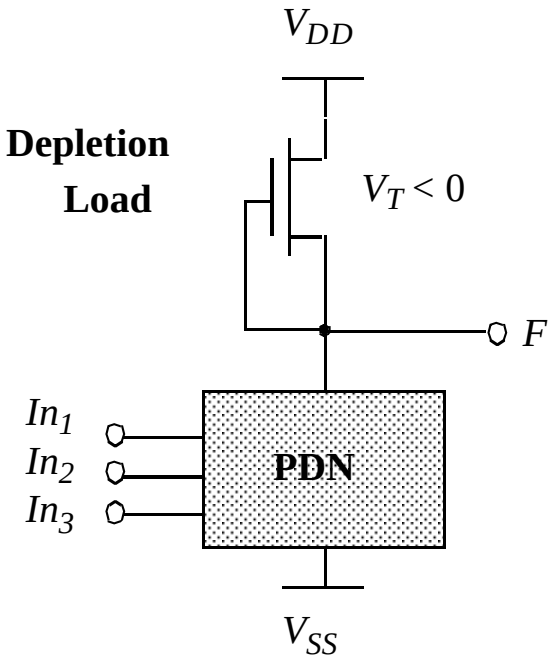
Ideally V_{OL} should be as small as possible. Hence, R_L should be large…

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

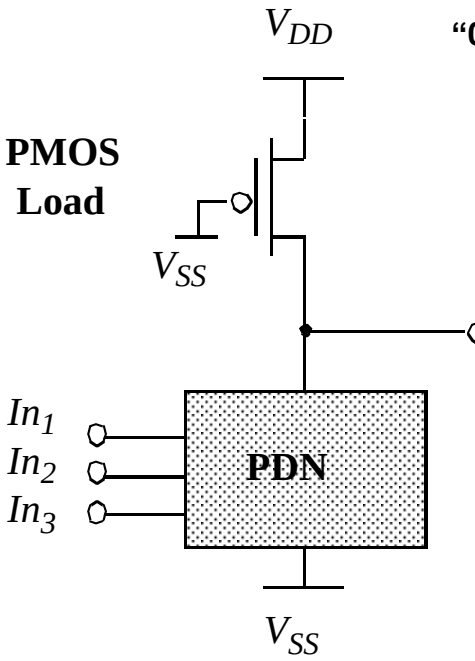


Active Loads





depletion load NMOS



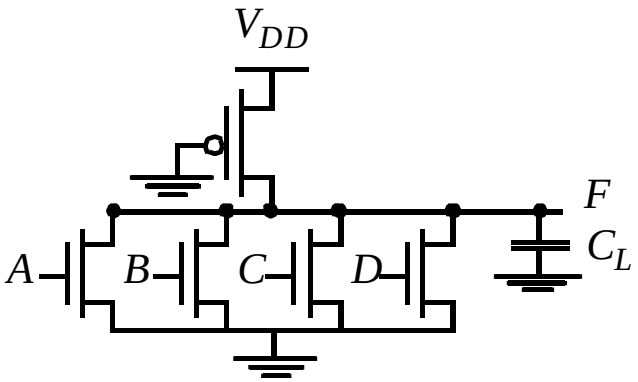
pseudo-NMOS

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

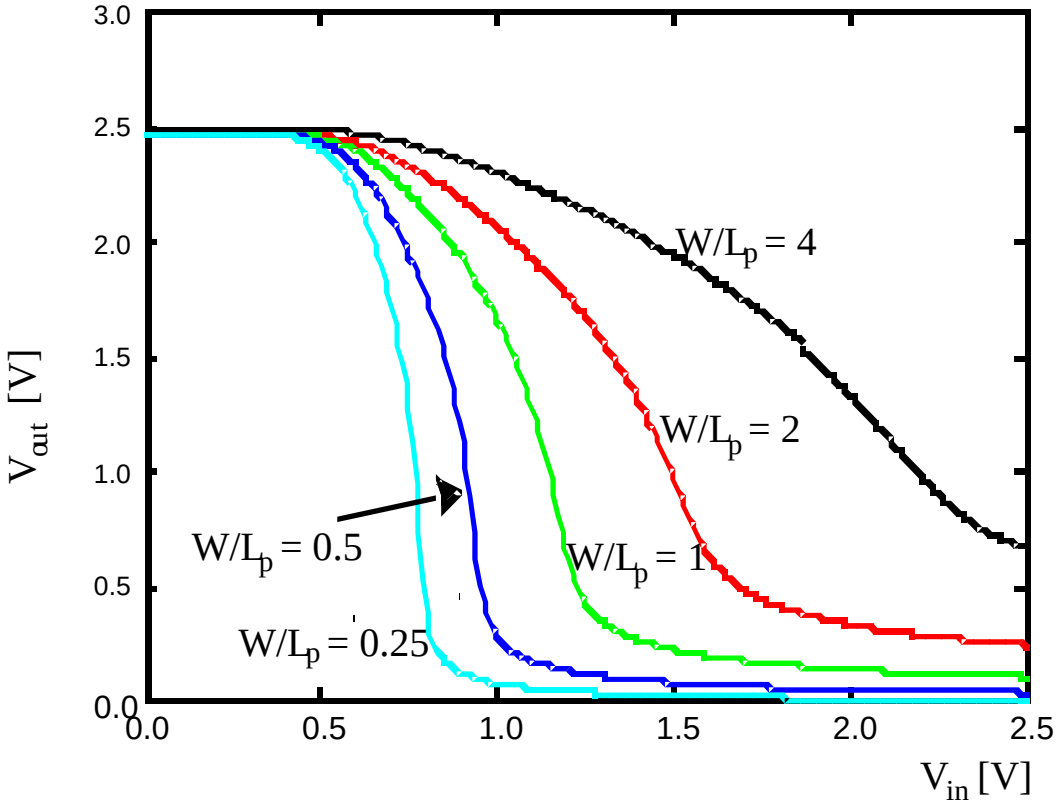
$$V_{OL} = \mu_p/\mu_n\, W_p/W_n\, V_{DSATp}$$

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

SMALLER AREA & LOAD BUT STATIC POWER DISSIPATION!!!

Pseudo-NMOS VTC

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

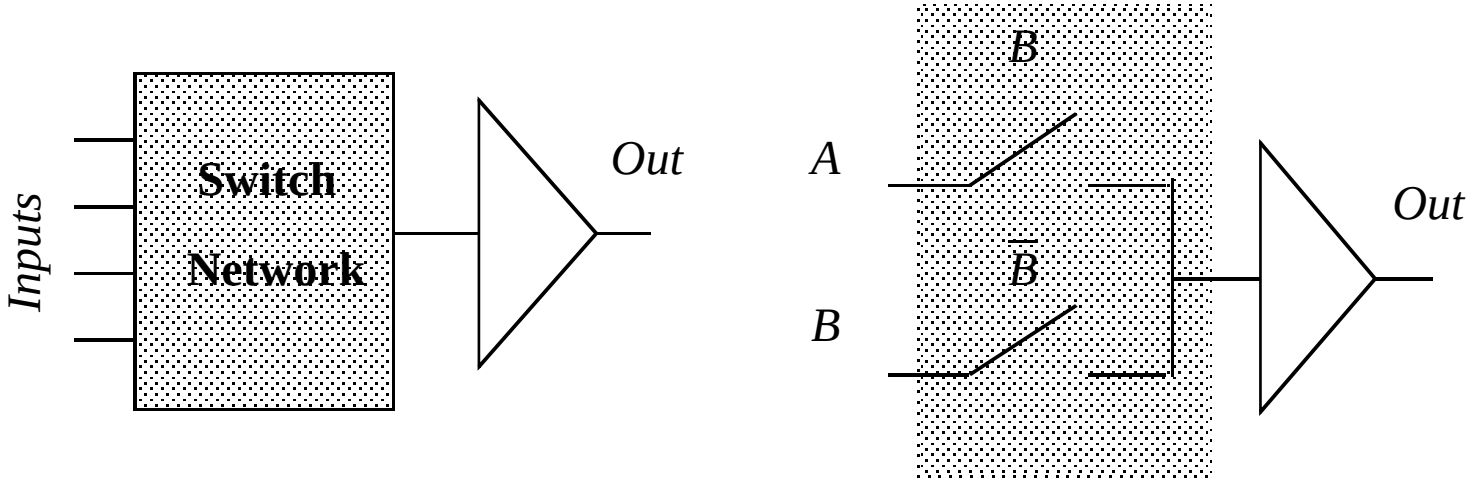


$$V_{OL} = \frac{R_{PN}}{R_{PN} + R_L}$$

Note: for V_{OL} to be low, we want large R_L or small W/L_p

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

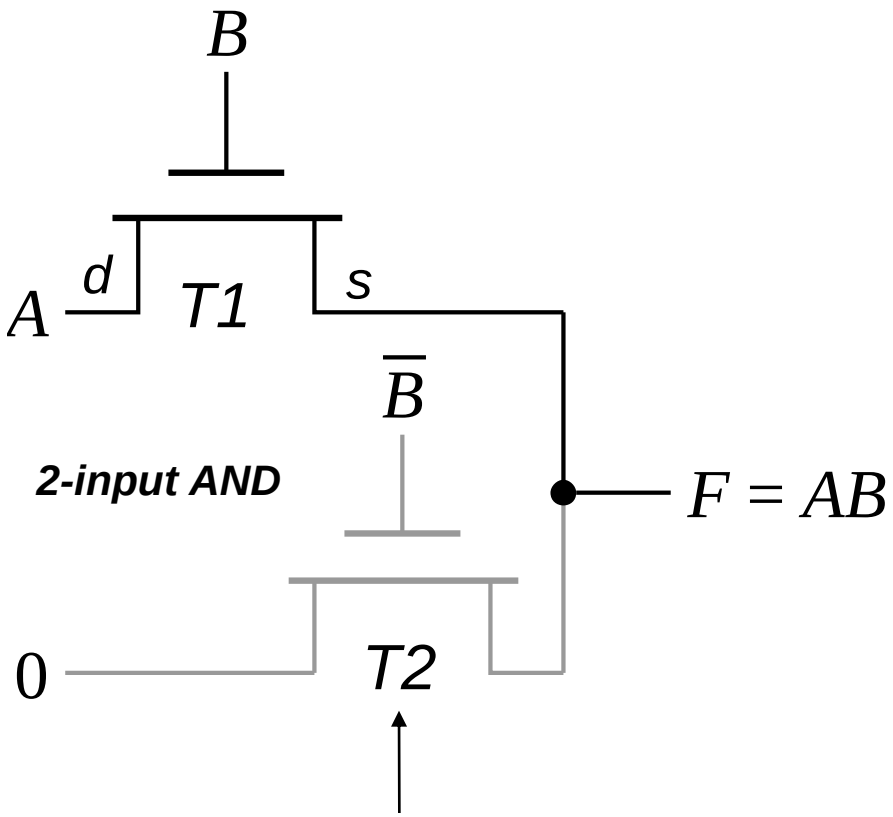
Pass-Transistor Logic



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

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

Example: AND Gate



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.*

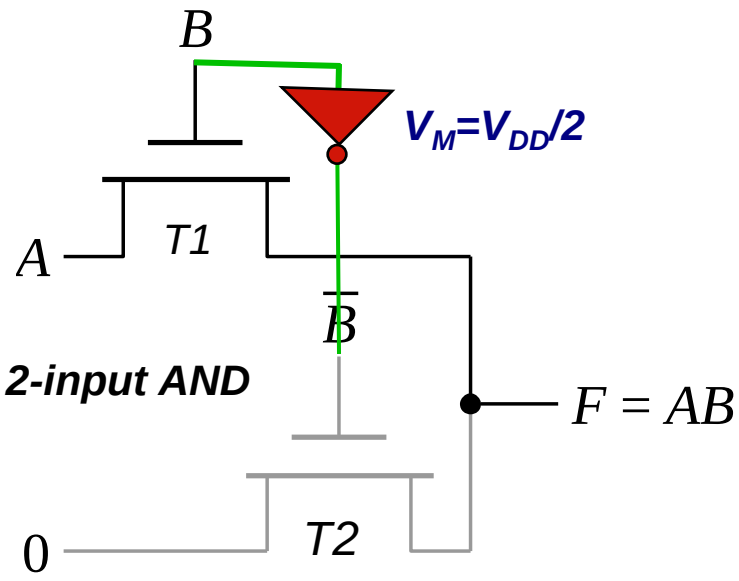
Need 6 to implement in static CMOS (4 for NAND and 2 for INV)

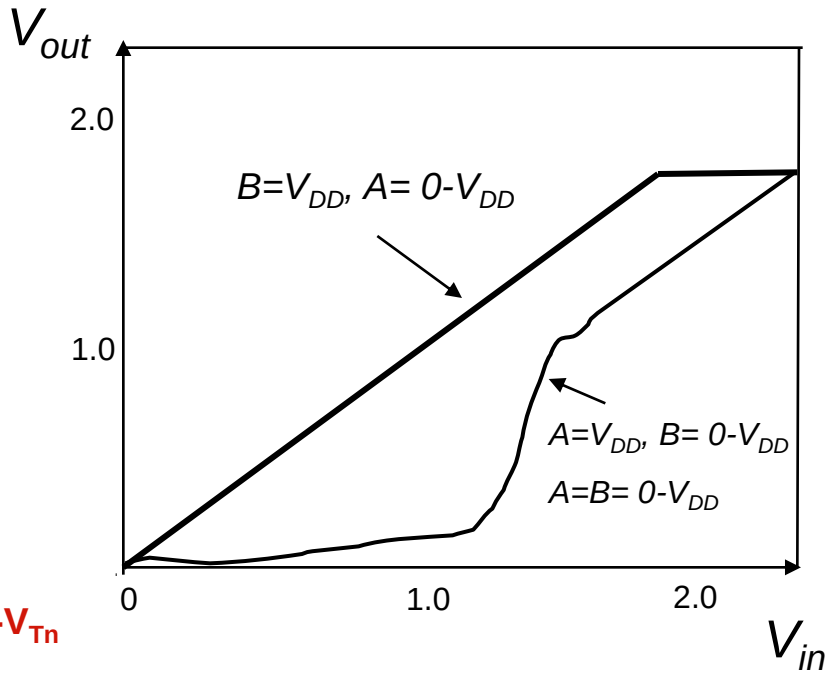
Ensures that gate is static: provides low impedance path when B=0

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





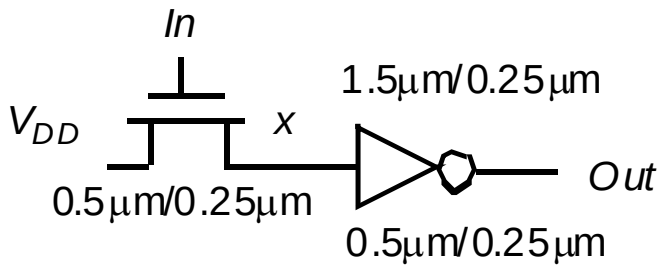
When $B=V_{DD}$, T1 is ON until the input reaches V_{DD}-V_{Tn}

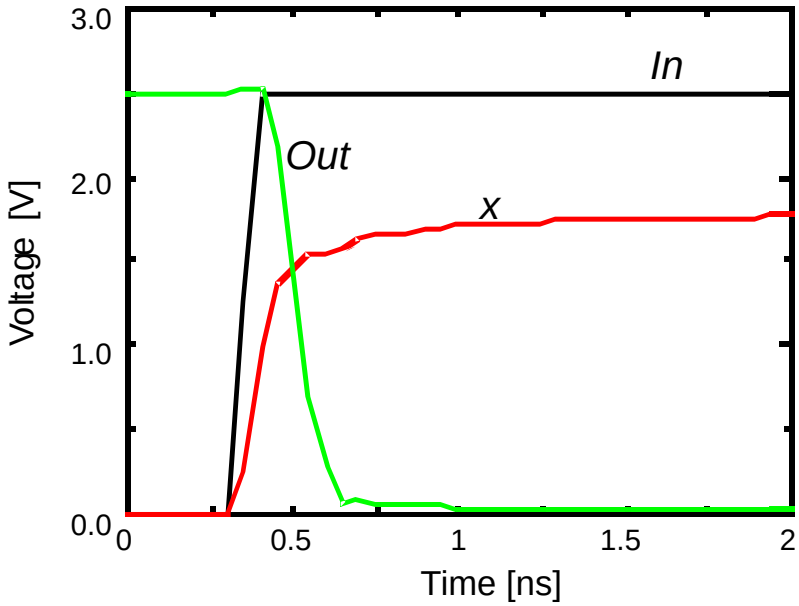
When $A=V_{DD}$, and B makes a transition from 0 to 1, T2 is turned on until $V_{DD}/2$ and Output =0. Once T2 is turned off, output follows the input B minus a threshold drop.

VTC of Pass Transistor Logic is data dependent

NMOS-Only Logic

Voltage Drop Issue:

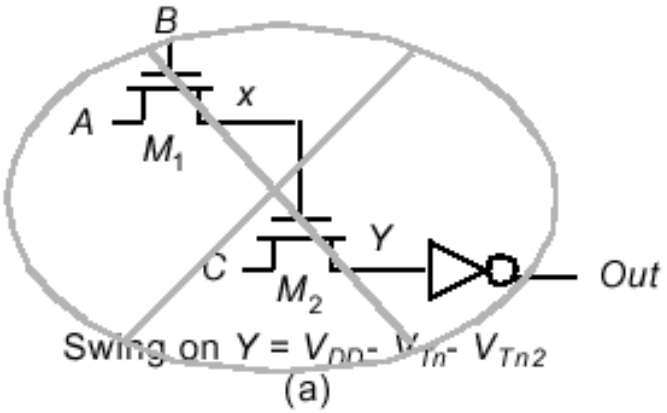


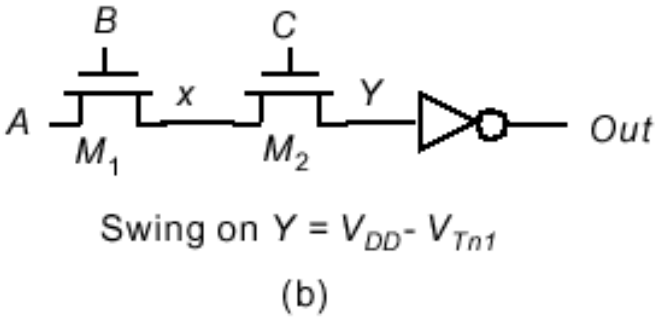


$$V_x = V_{dd} - V_{Tn}(V_x)$$

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

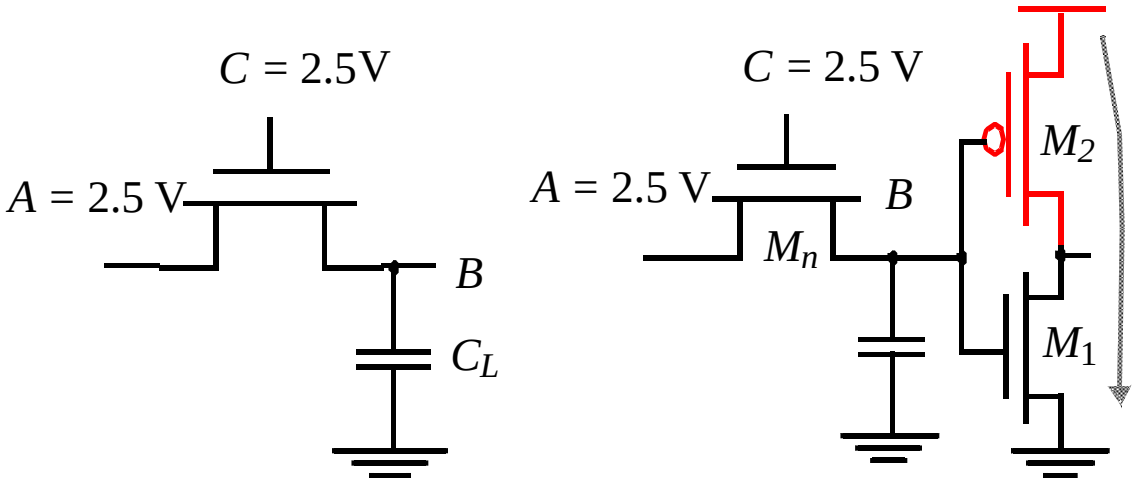




Let B=V_{DD}, A=1 (NMOS (M_1) pulling up node X):
$V_x = V_{DD}-V_{tn1}$
Let C=1 (NMOS (M_2) pulling up node Y):
$V_Y = (V_{DD}-V_{tn1})-V_{tn2}$

Let B=C=V_{DD}, A=1
$V_X = V_{DD}-V_{tn1}$ &
$V_Y = V_{DD}-V_{tn2} = V_{DD}-V_{tn1}$
(assuming $Vt_{n1}=Vt_{n2}$)

NMOS-only Switch



Lower switching energy!

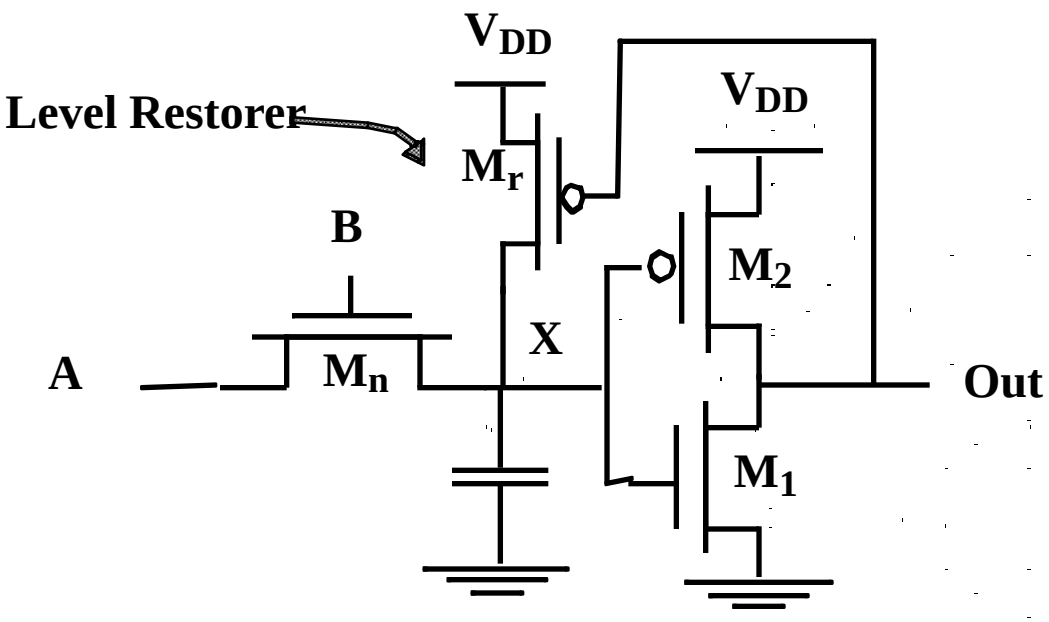
Difficult to switch off the PMOS!

V_B does not pull up to 2.5V, but 2.5V - V_{Tn}

Voltage loss causes static power consumption

NMOS has higher threshold than PMOS (body effect)

Solutions to the Voltage Drop Problem:
Solution 1: Level Restoring Transistor



Pass Transistor Logic suffers from static power dissipation and reduced NMs

At B=V_{DD}, if A: 0 to V_{DD}
$V_x = V_{DD}-V_{Tn}$, Out=0,
M_r=ON and $V_x=V_{DD}$

Eliminates static power in the Inverter

No static power between M_r and M_n

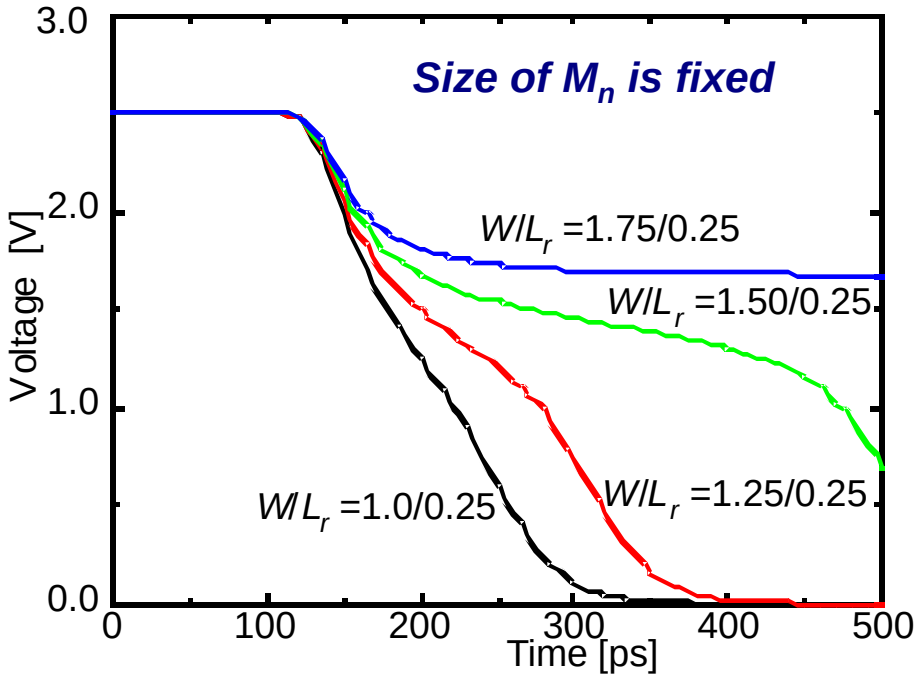
- **Advantage: Full Swing**

- **Restorer adds capacitance, takes away pull down current at X (for high to low transition at X, M_n must be stronger than M_r), can slow down gate**

- **Ratio problem**

Restorer Sizing

Need to size M_n and M_r to bring $V_x < V_M$ ($=V_{DD}/2$) (V_M is a function of R1 and R2) *R1 and R2 are the equivalent on-resistances of M1 and M2*

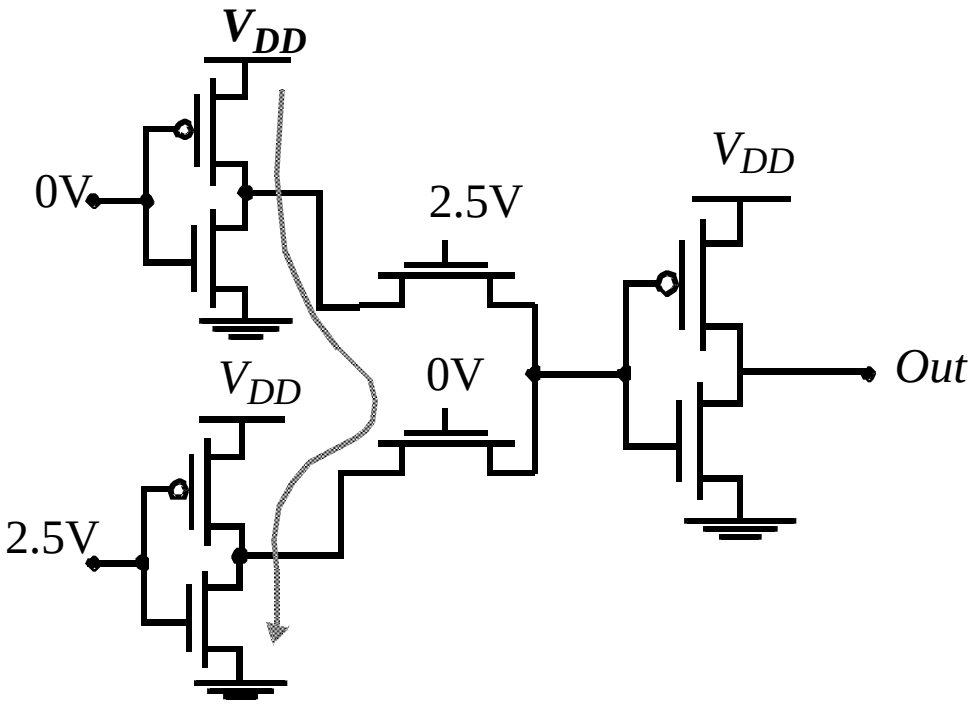


- **Upper limit on restorer size** when too large (R_r too small), V_x can't be brought below V_M
- Pass-transistor pull-down can have several transistors in stack

Transient Response: V_x vs. time

Solution 2: Single Transistor Pass Gate with $V_T=0$

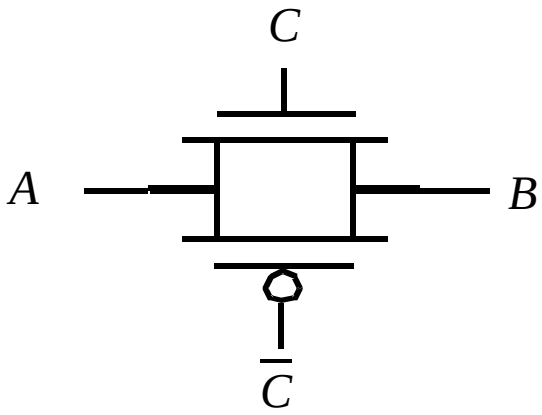
Only Pass Transistor Devices have $V_T=0$

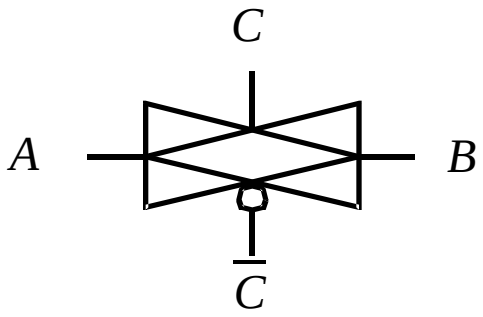


But even if $V_T=0$, there is still body effect…which prevents full swing!

WATCH OUT FOR LEAKAGE CURRENTS in the IDLE State!!!

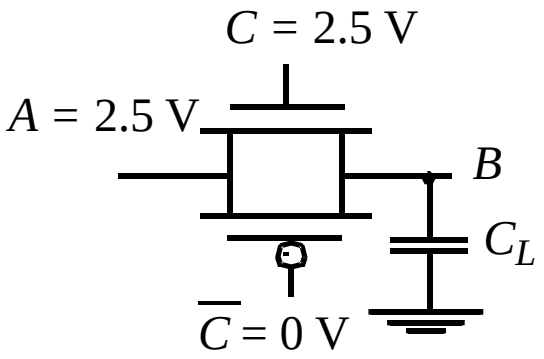
Solution 3: Transmission Gate





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)

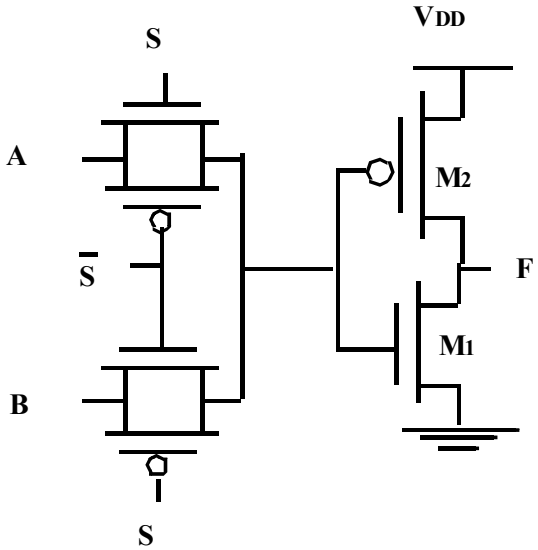


Because of the PMOS, C_L charges to Vdd

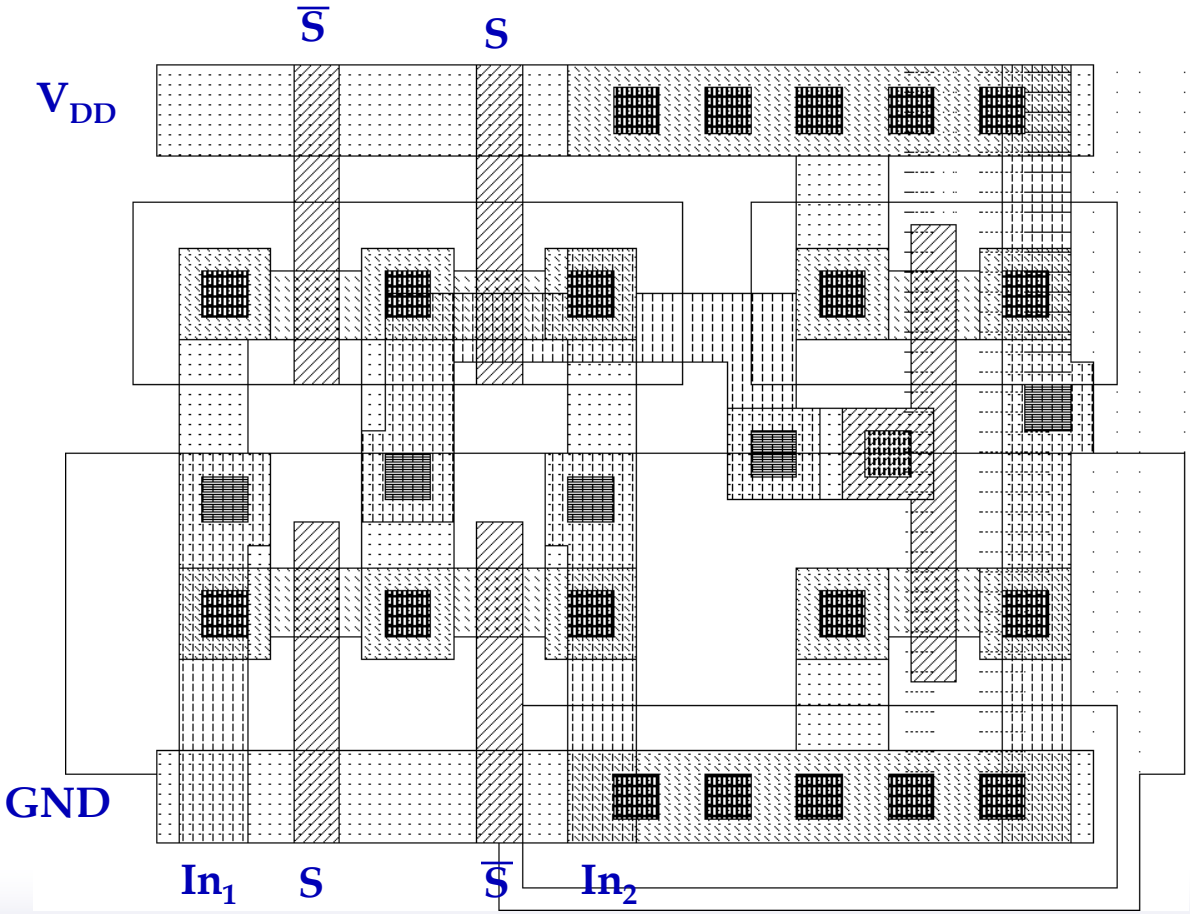
Because of NMOS C_L discharges to 0

Pass-Transistor Based Multiplexer

2-input MUX



$$\overline{F} = (A.S + B.\overline{S})$$



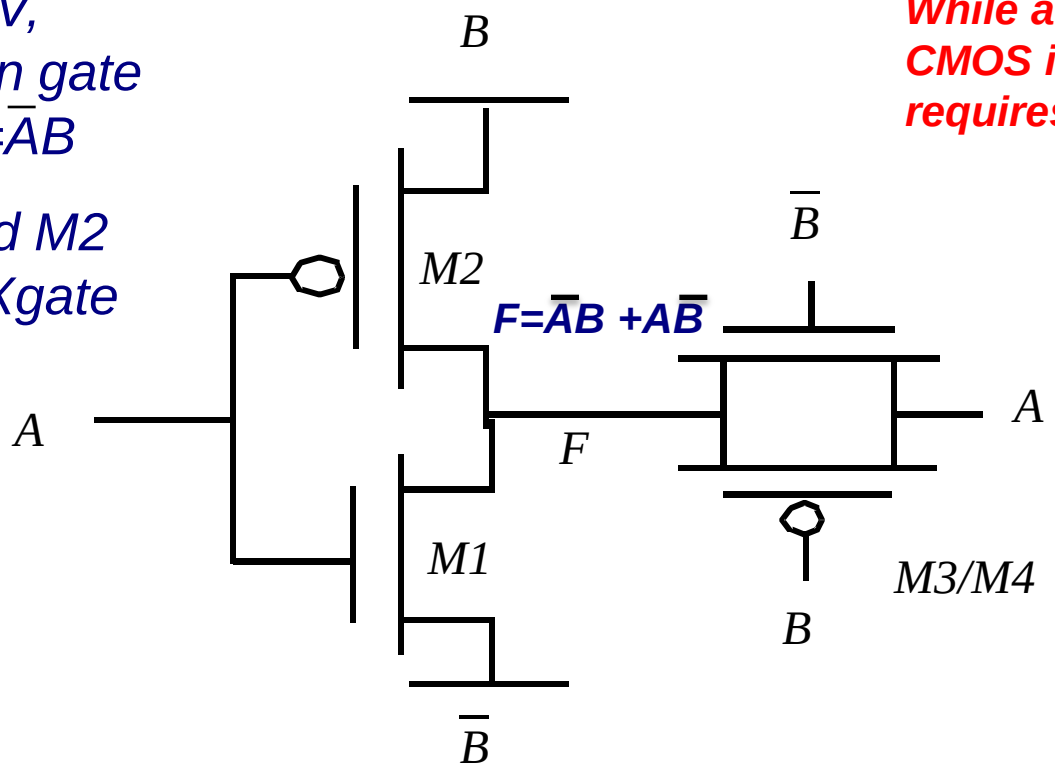
Transmission Gate XOR

B=1: *M1 and M2 act as an INV, Transmission gate is off and F=$\bar{A}B$*

B=0: *M1 and M2 are off and Xgate is on, F=$A\bar{B}$*

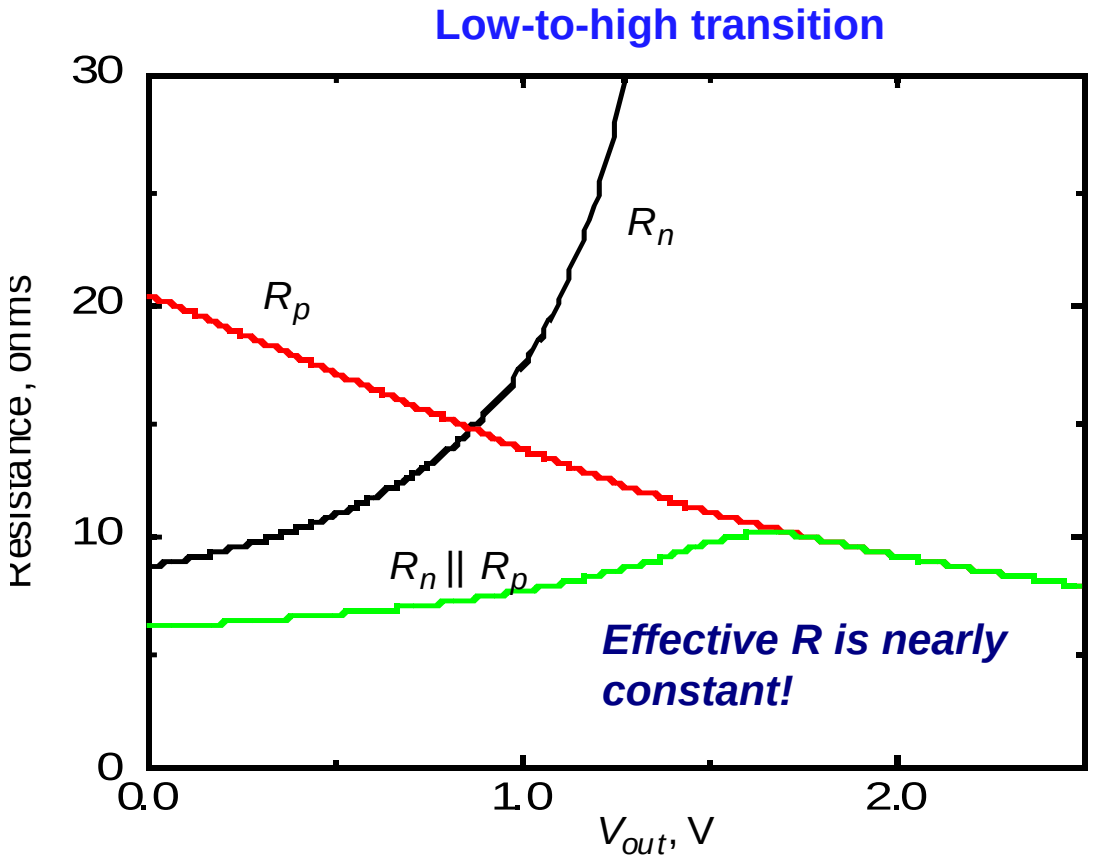
Implementation requires only 6 transistors!

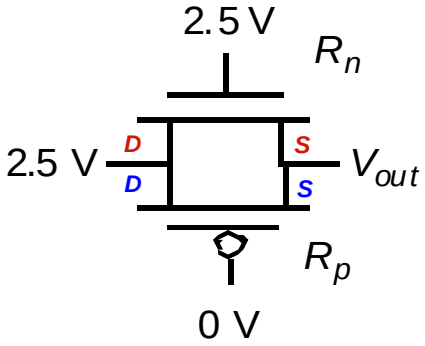
While a complementary CMOS implementation requires 10 transistors



Fast adder circuits and registers can also be implemented with Xgates

Resistance of Transmission Gate





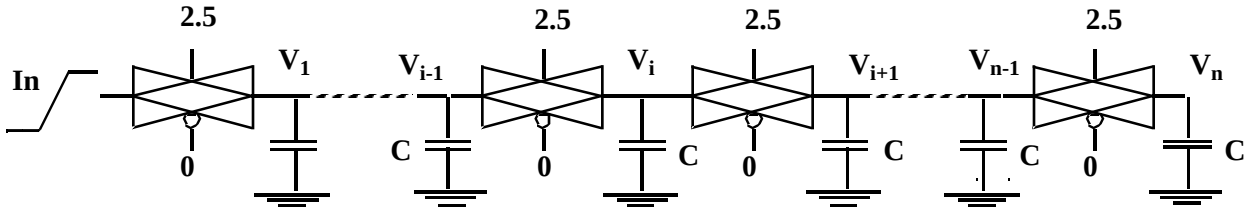
$R_n = (V_{DD}-V_{out})/I_{dn}$

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

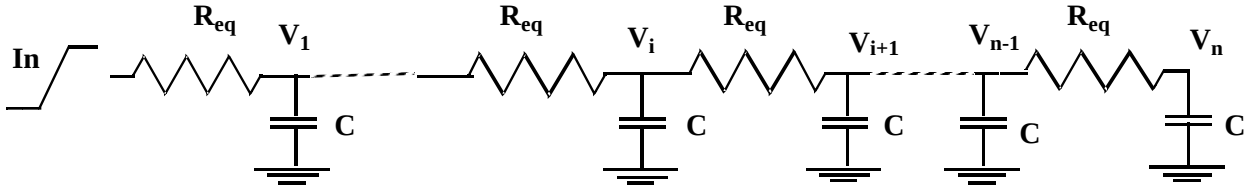
When Vout is low, NMOS is working, hence Rn dominates the equivalent resistance….similarly Rp dominates when Vout is high…..

Delay in Transmission Gate Networks

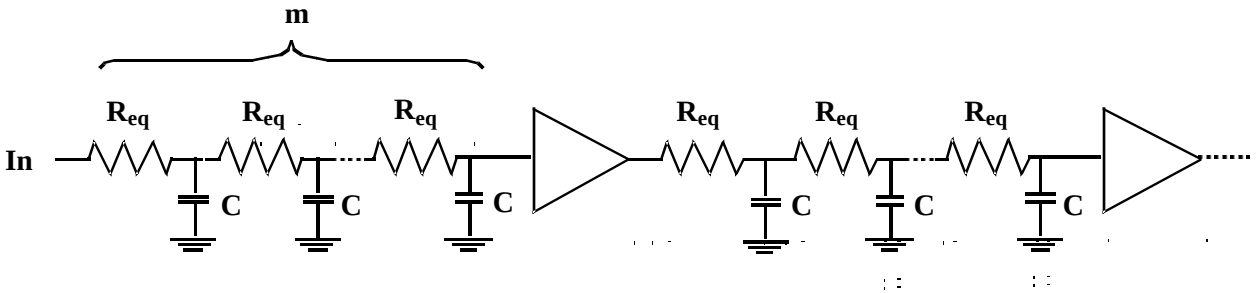
Delay of a chain of n Xgates (used in adders and deep MUXes) can be modeled using Elmore delay



(a) A chain of n Xgates



(b) Equivalent RC representation



(c) Buffer insertion in a chain of Xgates to lower delay

Delay Optimization

- **Delay of RC chain**

$$t_p = 0.69 \sum_{k=0}^{n} CR_{eq} k = 0.69 CR_{eq} \frac{n(n+1)}{2}$$

- **Delay of Buffered Chain**

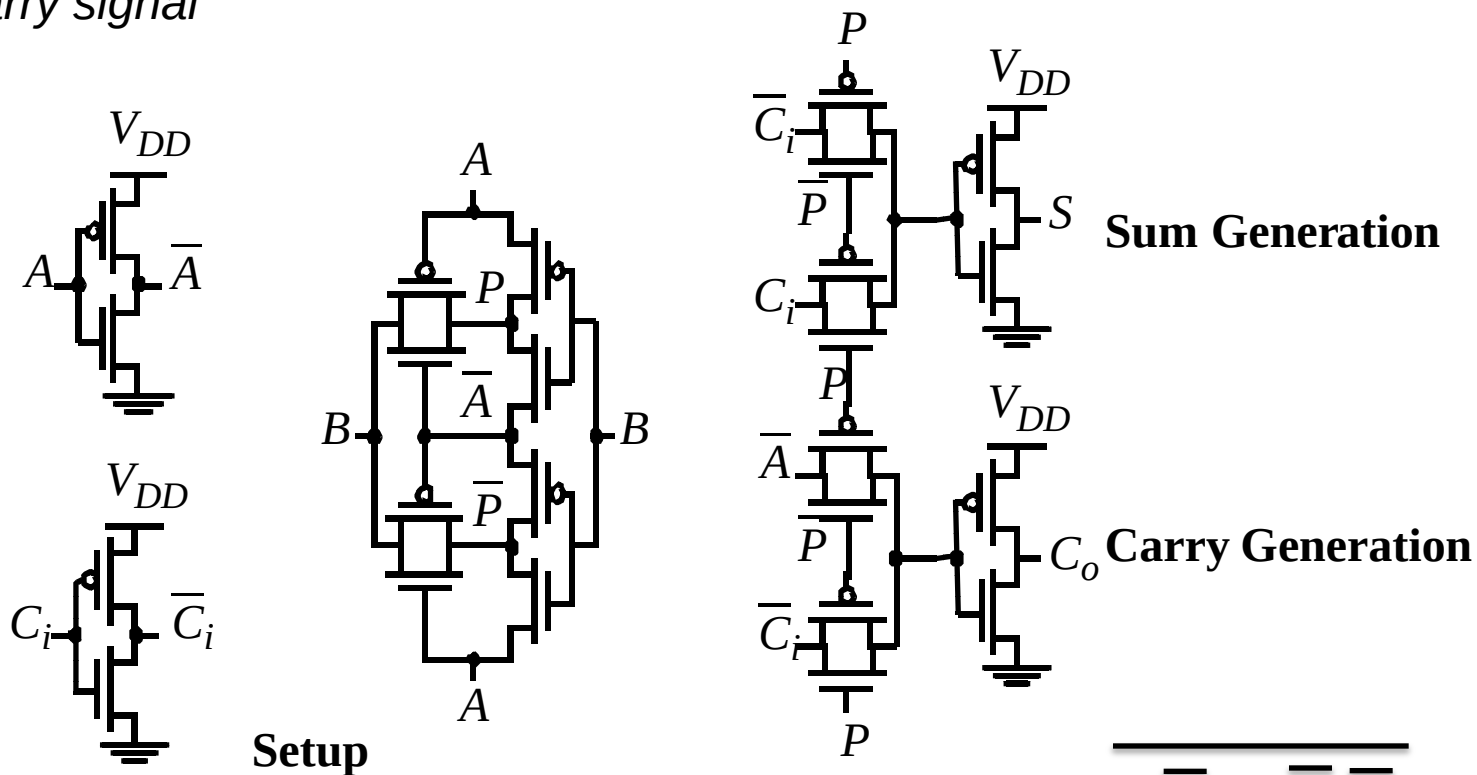
$$t_p = 0.69 \left\lfloor \frac{n}{m} CR_{eq} \frac{m(m+1)}{2} \right\rfloor + \left(\frac{n}{m} - 1 \right) t_{buf}$$

$$= 0.69 \left[CR_{eq} \frac{n(m+1)}{2} \right] + \left(\frac{n}{m} - 1 \right) t_{buf}$$

$$m_{opt} = 1.7 \sqrt{\frac{t_{pbuf}}{CR_{eq}}}$$

Transmission Gate Full Adder

P: propagation signal
C: carry signal



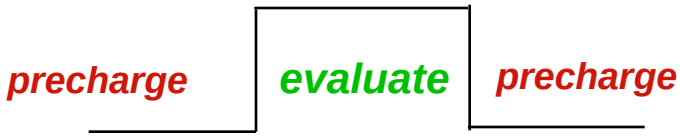
$$C_0 = \overline{P\,\overline{C}_i + \overline{P}\,\overline{A}}$$

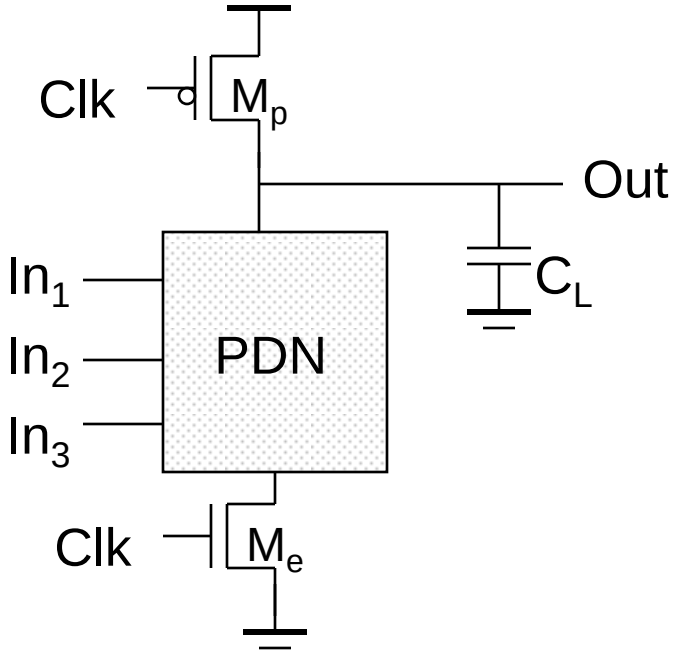
Similar delays for sum and carry

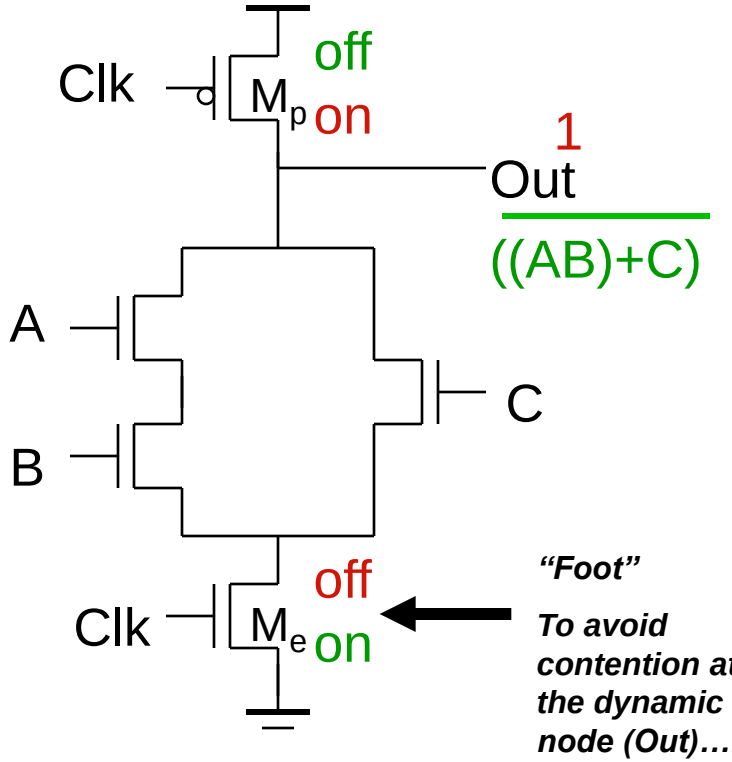
Dynamic CMOS

- 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 2*n* (*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







Two phase operation

Precharge (Clk = 0)

Evaluate (Clk = 1)

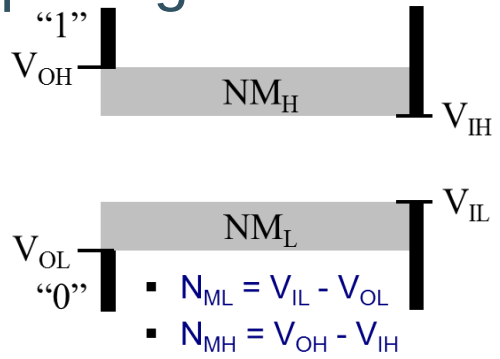
$Out = \overline{CLK} + \overline{(AB)+C} \cdot CLK$

Conditions on Output

- During evaluation phase, the only possible path between output node and supply rail is to ground. Hence, 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 (if PDN is 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)
- Full swing outputs (V_{OL} = GND and V_{OH} = V_{DD})
- Non-ratioed - sizing of the devices does not affect the logic levels
- PDN starts to work as soon as the input signals exceed V_{Tn}, so V_M, V_{IH} and V_{IL} equal to V_{Tn}
- Low noise margin (NM_L)
- Needs a precharge/evaluate clock



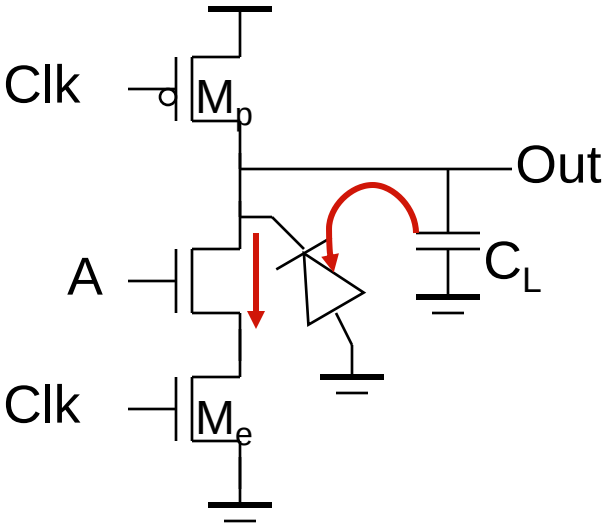
- Faster switching speeds:
  - reduced load capacitance due to lower input capacitance (C_{in}) resulting from lower number of transistors per gate and single transistor load per fan-in (reduced logical effort, 2/3 for a 2-input dynamic NOR)
  - no I_{sc}, so all the current provided by PDN goes into discharging C_L

Properties of Dynamic Gates

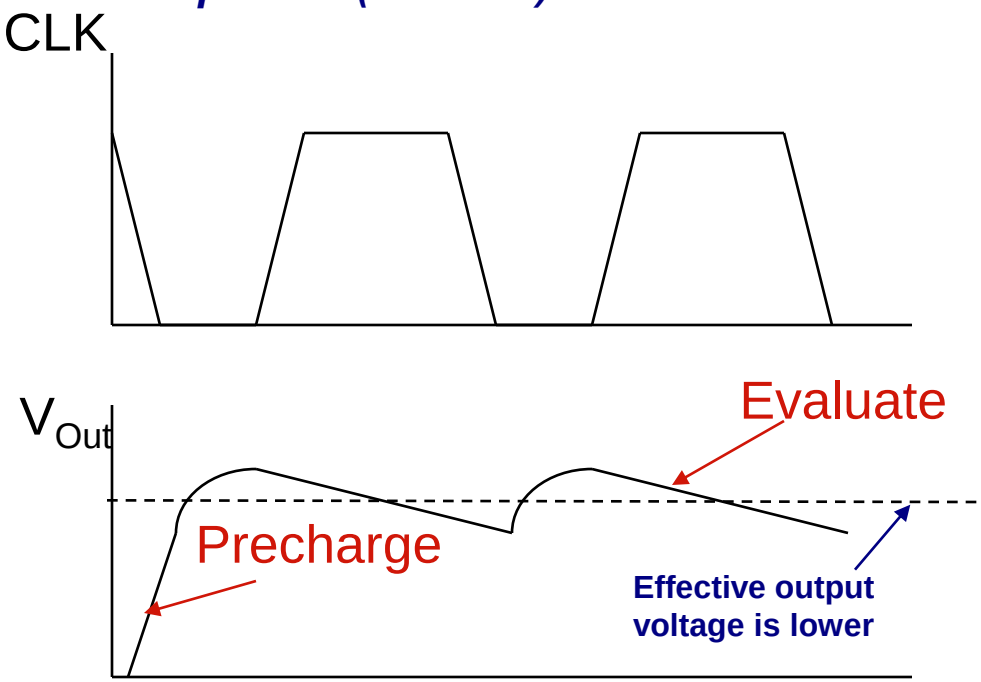
- Advantages:
  - Lower physical capacitance: uses fewer transistors
  - No glitching (dynamic gates can have at most one transition per CLK cycle)
  - Only consumes dynamic power…..no static current path ever exists between V_{DD} and GND (including P_{sc})
- In spite of the above…..overall power dissipation usually higher than static CMOS
  - CLK power can be significant: extra load on Clk + transition every CLK cycle
  - Number of transistors is more than the minimal set required for implementing logic
  - Higher switching activity due to higher transition probabilities

Issues in Dynamic Design 1: Charge Leakage

A dynamic inverter



Due to leakage, a minimal CLK rate required…(few KHz)



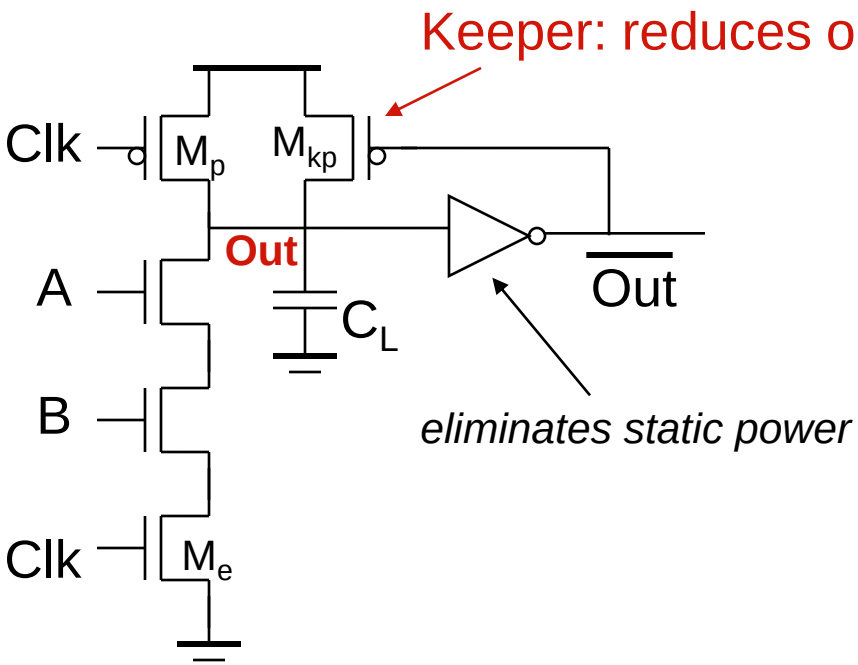
Leakage sources: reverse biased diode and subthreshold

Dominant component is subthreshold current

Note: leakage of precharge PMOS can partially compensate for the charge loss at the dynamic node

Solution to Charge Leakage

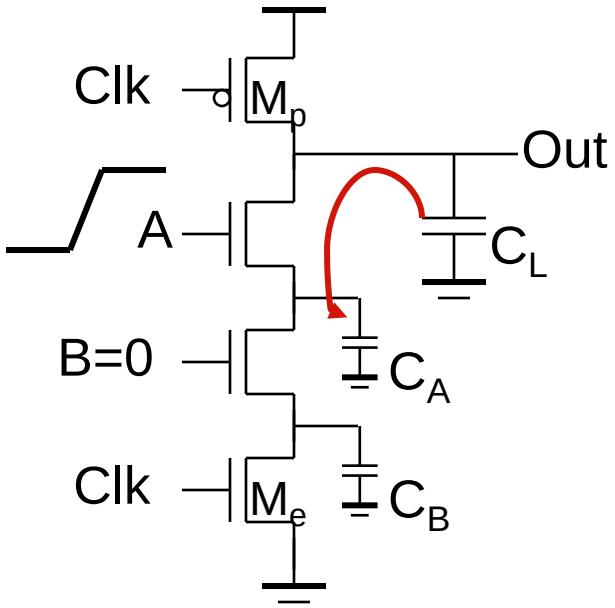
Same approach as level restorer for pass-transistor logic



Contention between keeper and PDN--- strength of keeper must be less than that of the PDN to lower the out node well below the switching threshold of the next gate. Hence keeper size should be small.

H. F. Dadgour and K. Banerjee, "A Novel Variation-Tolerant Keeper Architecture for High-Performance Low-Power Wide Fan-in Dynamic Gates," IEEE Transactions on VLSI Systems, Vol. 18, No. 11, pp. 1567-1577, 2010.

Issues in Dynamic Design 2: Charge Sharing



Charge stored originally on C_L is redistributed (shared) over C_L and C_A leading to reduced robustness

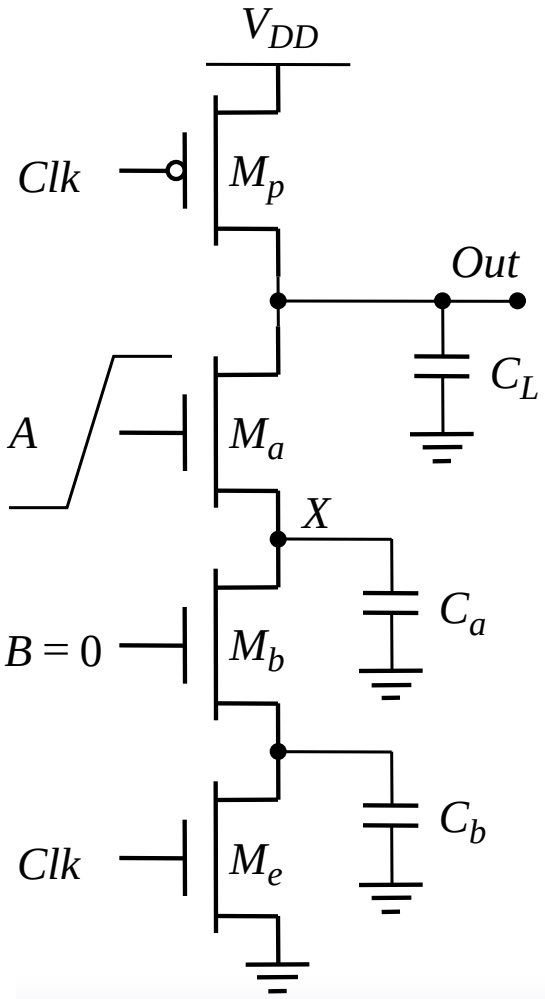
Output node voltage drops and cannot be recovered due to the dynamic nature of the circuit.

Charge Sharing

All inputs = 0 during pre-charge

Initial conditions: $V_{out}(t=0)=V_{DD}$ and $V_x(t=0)=0$

2 possible scenarios:



case 1) if $\Delta V_{out} < V_{Tn}$

Final value of V_X

From charge conservation….

$$C_L V_{DD} = C_L V_{out}(t) + C_a(V_{DD} - V_{Tn}(V_X))$$

or

$$\Delta V_{out} = V_{out}(t) - V_{DD} = -\frac{C_a}{C_L}(V_{DD} - V_{Tn}(V_X))$$

case 2) if $\Delta V_{out} > V_{Tn}$

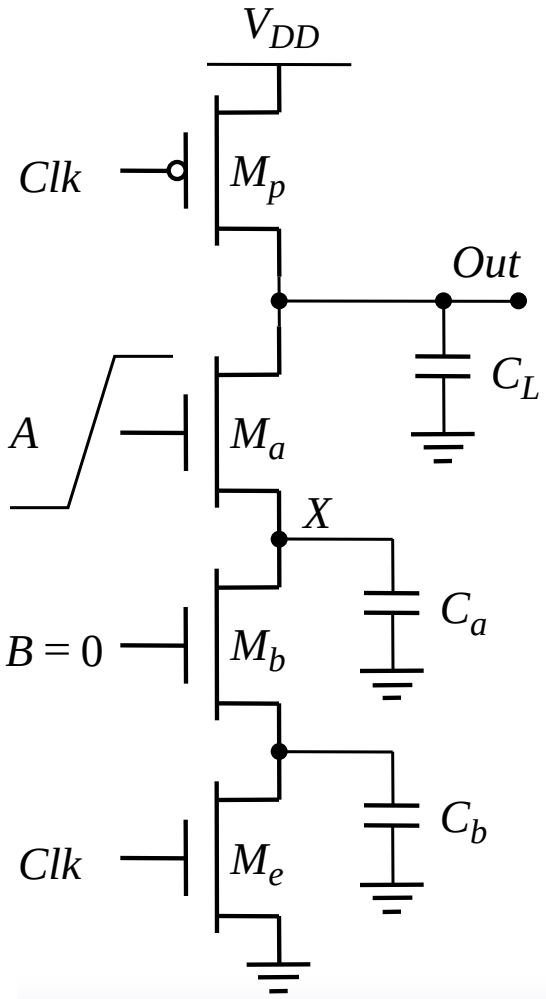
V_{OUT} and V_X then reach the same value…..

From charge conservation….

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

Which of these scenarios is valid?

Charge Sharing



Initial conditions: V_{out} (t=0)=V_{DD} and V_x(t=0)=0

2 possible scenarios:

$\Delta V_{out} < V_{Tn}$ ….case I

$\Delta V_{out} > V_{Tn}$ ….case II

Which of these scenarios is valid?

First find the capacitance ratio: C_a/C_L

The boundary condition between the two cases can be determined by setting $\Delta V_{out} = V_{Tn}$ (in the expression for case II). Hence,

$$\frac{C_a}{C_L} = \frac{V_{Tn}}{V_{DD} - V_{Tn}}$$

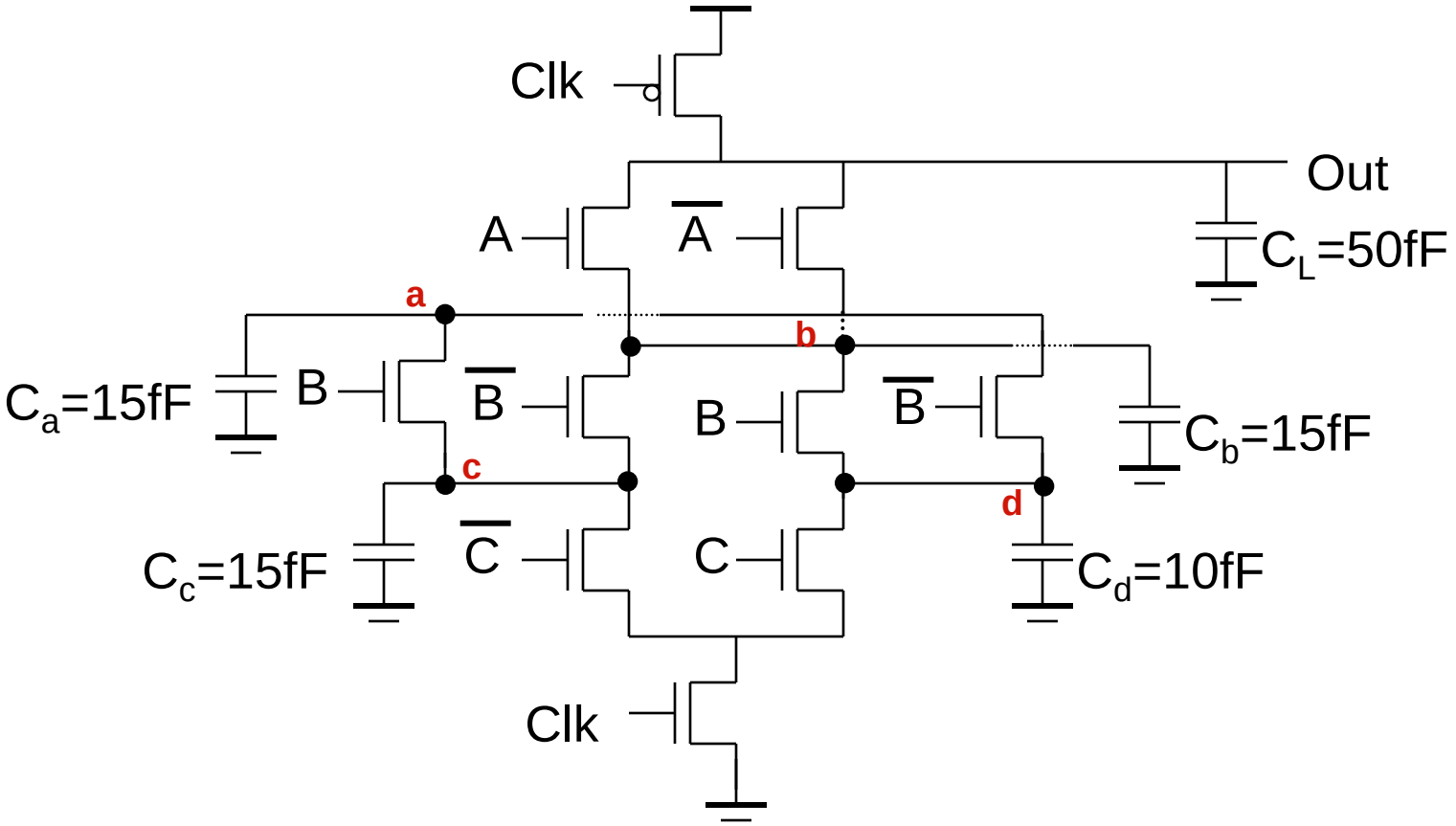
Case I holds when the C_a/C_L ratio is smaller than the value defined above, otherwise Case II holds.

Overall, it is desirable to keep $\Delta V_{out} < |V_{Tp}|$ ---since the output of dynamic gate might be connected to a static inverter---low level of V_{out} will cause static power consumption. Also, V_{out} must not go below V_M of the inverter.

Charge Sharing Example

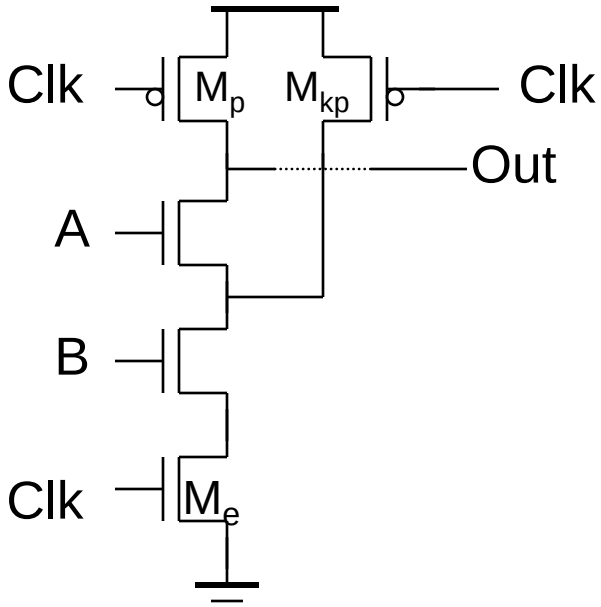
Dynamic 3-input EXOR gate

$$Out = A \oplus B \oplus C$$



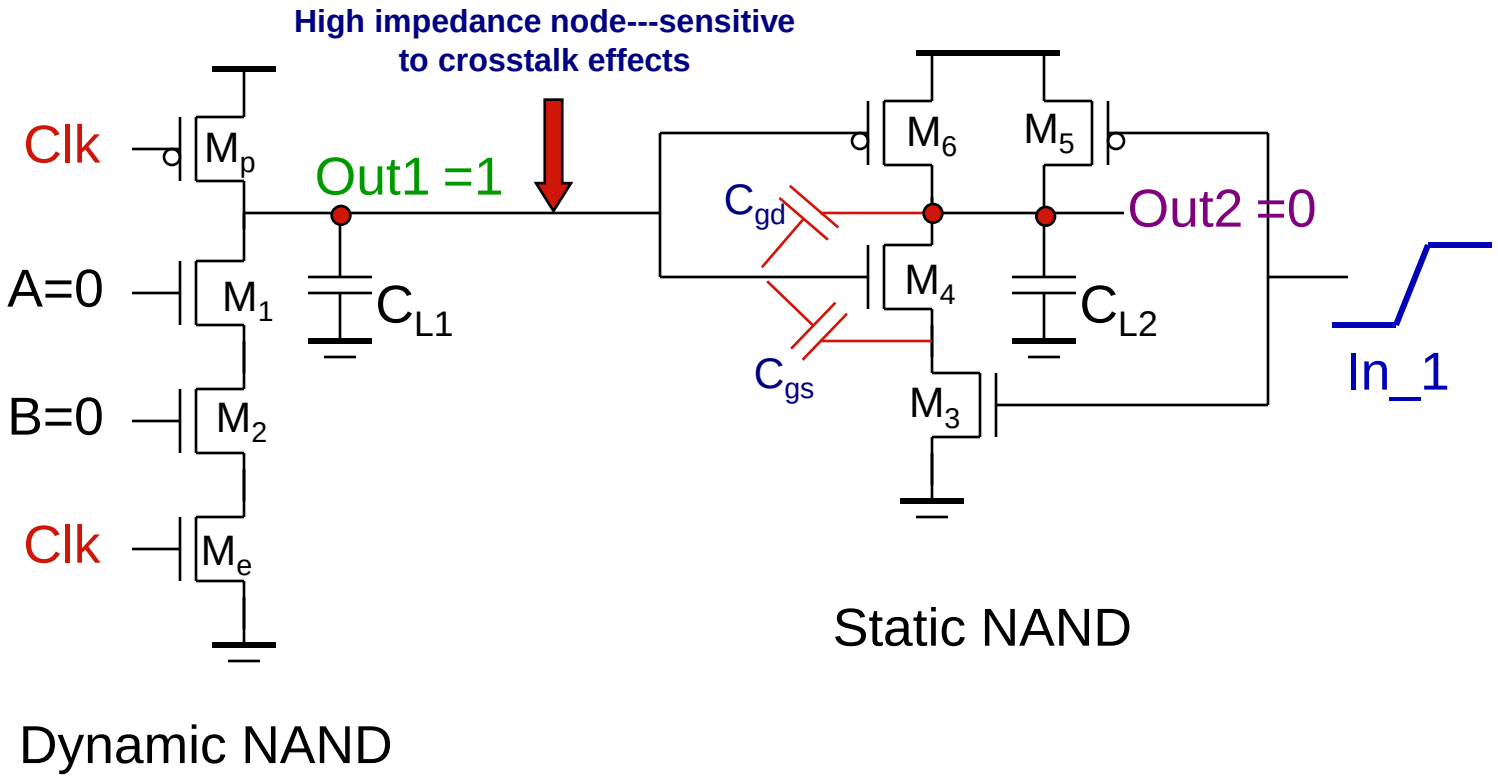
Worst case change in Output is obtained by exposing the maximum number of internal capacitances to the output: this happens for $A\overline{B}C$ or $\overline{A}BC$

Solution to Charge Redistribution



Precharge internal nodes (to V_{DD}) using a clock-driven transistor (at the cost of increased area and power)

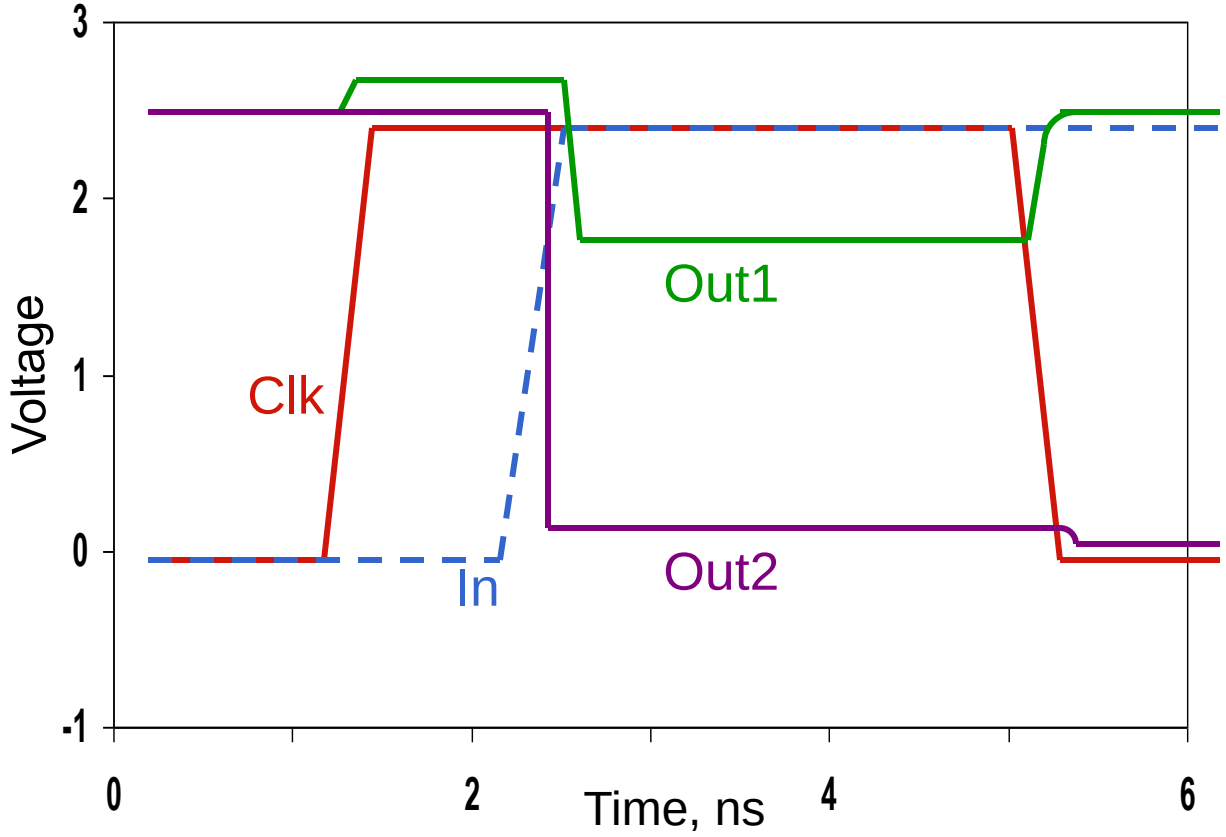
Issues in Dynamic Design 3: Backgate (Output-to-input) Coupling



Capacitive coupling between dynamic node Out1 and H-L transition at Out2 (when In_1 goes high) through the gate-drain and gate-source capacitance of M4

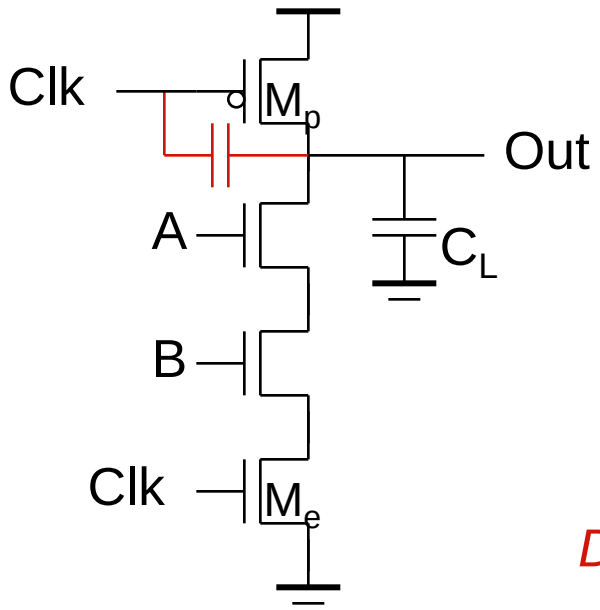
Backgate Coupling Effect

Simulation result



Coupling causes dynamic node Out1 to drop significantly, which further prevents Out2 from dropping all the way to zero---static power dissipation.

Issues in Dynamic Design 4: Clock Feedthrough



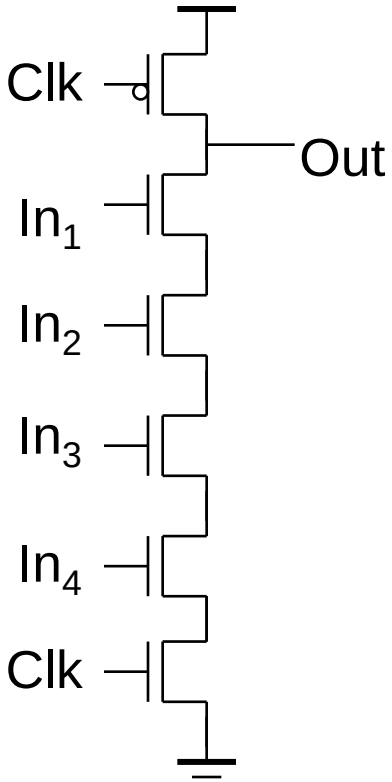
Coupling between Out and Clk input of the precharge device due to gate to drain capacitance (includes both overlap and channel).

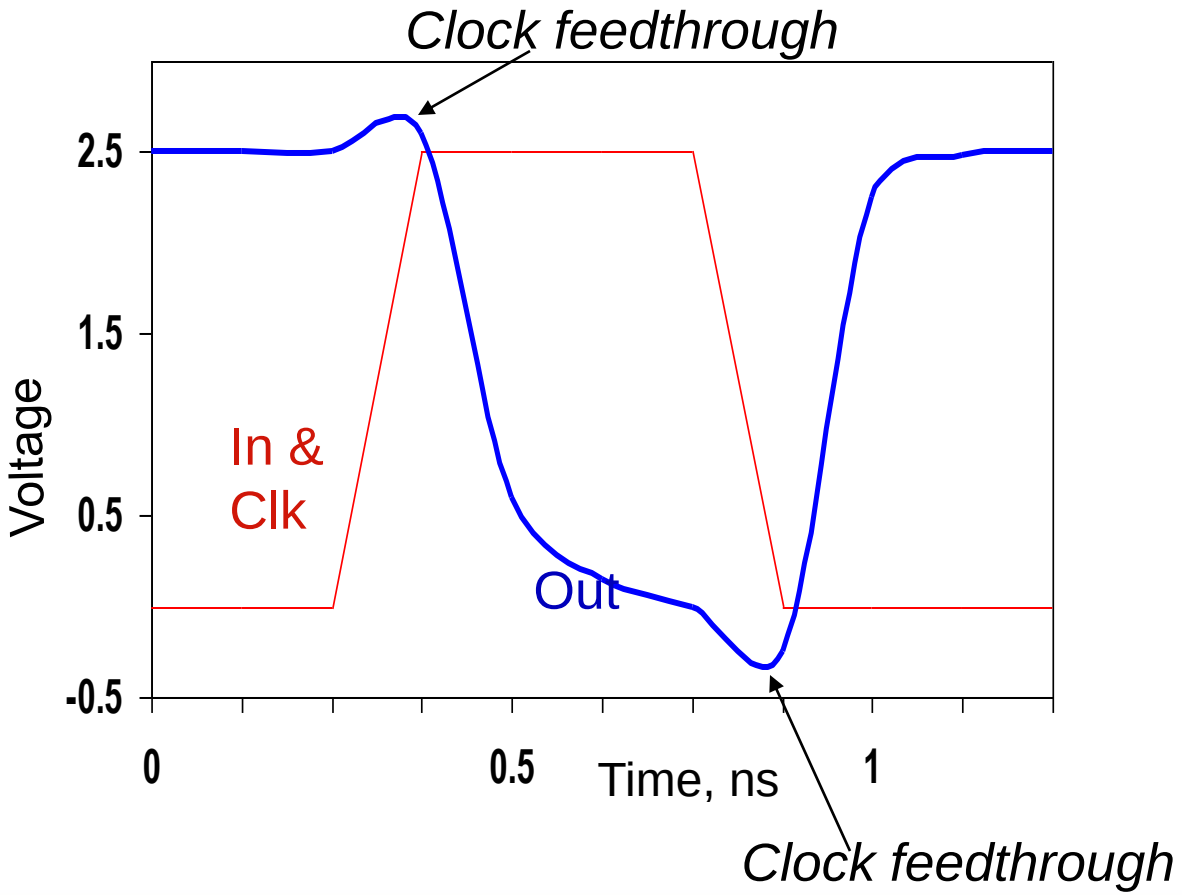
Hence, voltage of Out can rise above V_{DD} on the L-H Clk transition (assuming PDN is off). The fast rising (and falling edges) of the clock couple to Out.

Dynamic circuits need careful simulation!

Clk feedthrough can cause normally reverse biased junction diodes of the precharge transistor to become forward biased---causing electron injection into the substrate that can be collected by a nearby high-impedance node in the 1 state, eventually resulting in faulty operation.

Clock Feedthrough



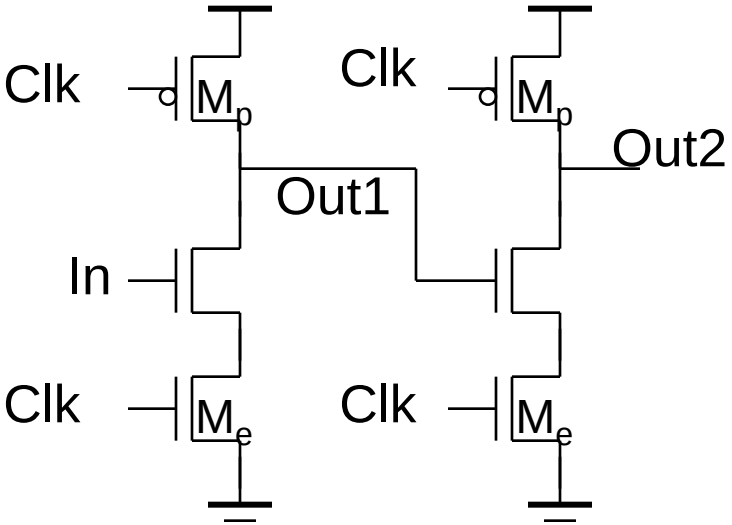


Other Effects

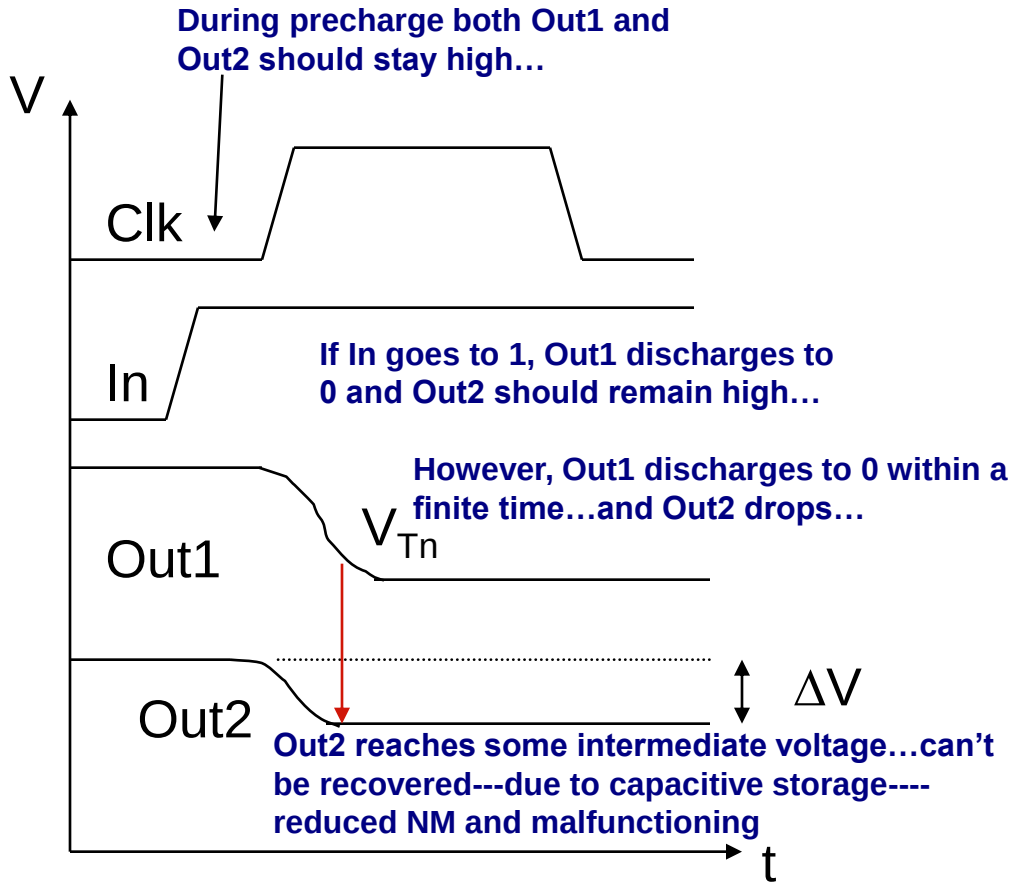
- Capacitive coupling
- Substrate coupling
- Minority charge injection
- Supply noise (ground bounce)

Cascading Dynamic Gates

Simple cascoding doesn't work…



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

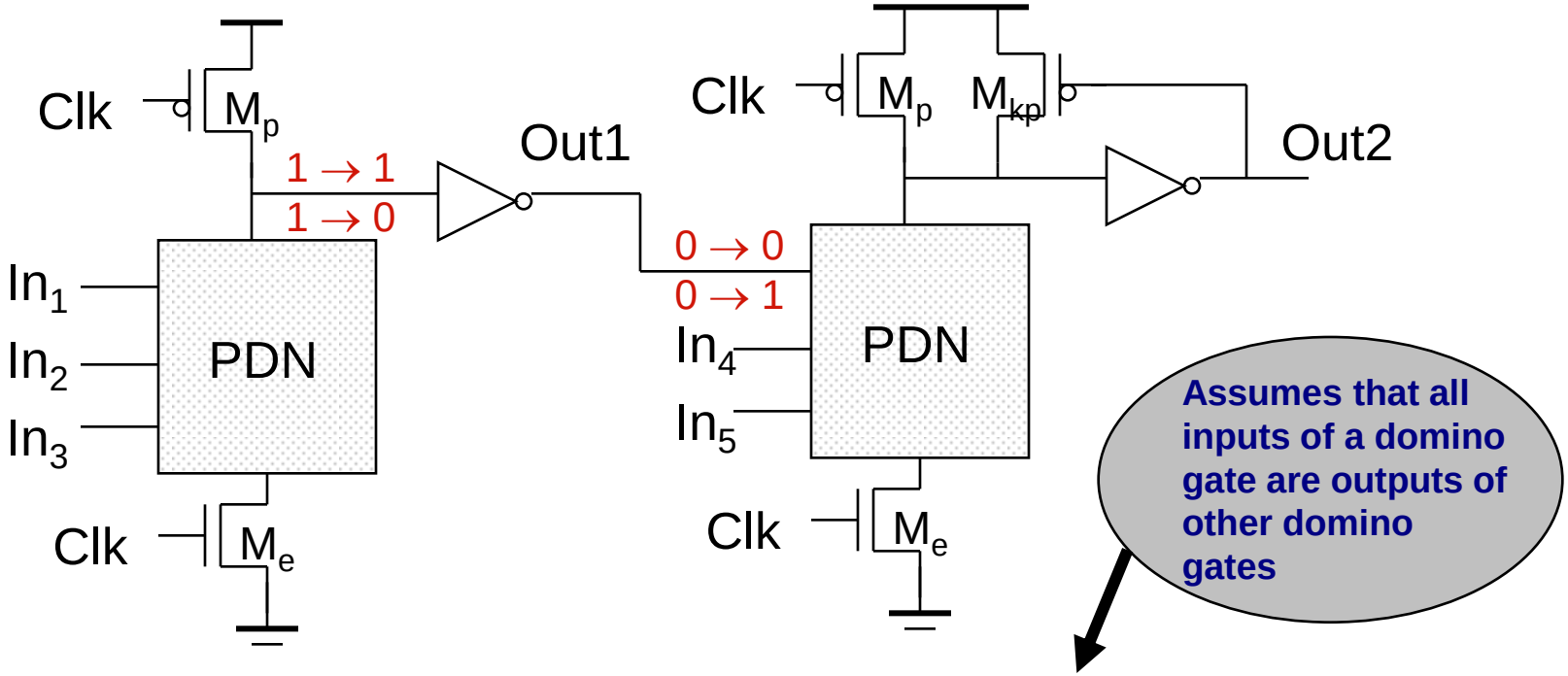


Solution: Set all inputs to 0 during precharge

For correct operation only 0 → 1 transitions should be allowed at inputs!

Domino Logic

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

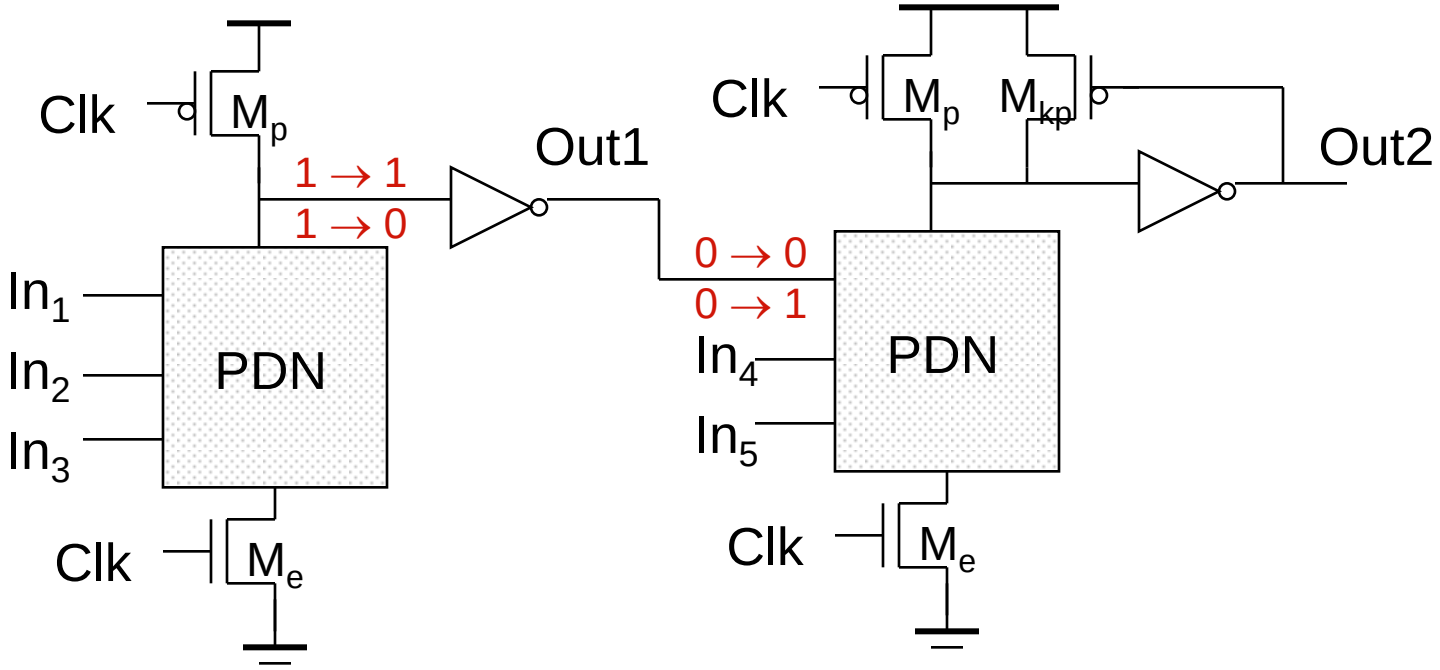


All inputs (are outputs of other Domino gates) are set to 0 at the end of precharge phase

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.

Domino Logic

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

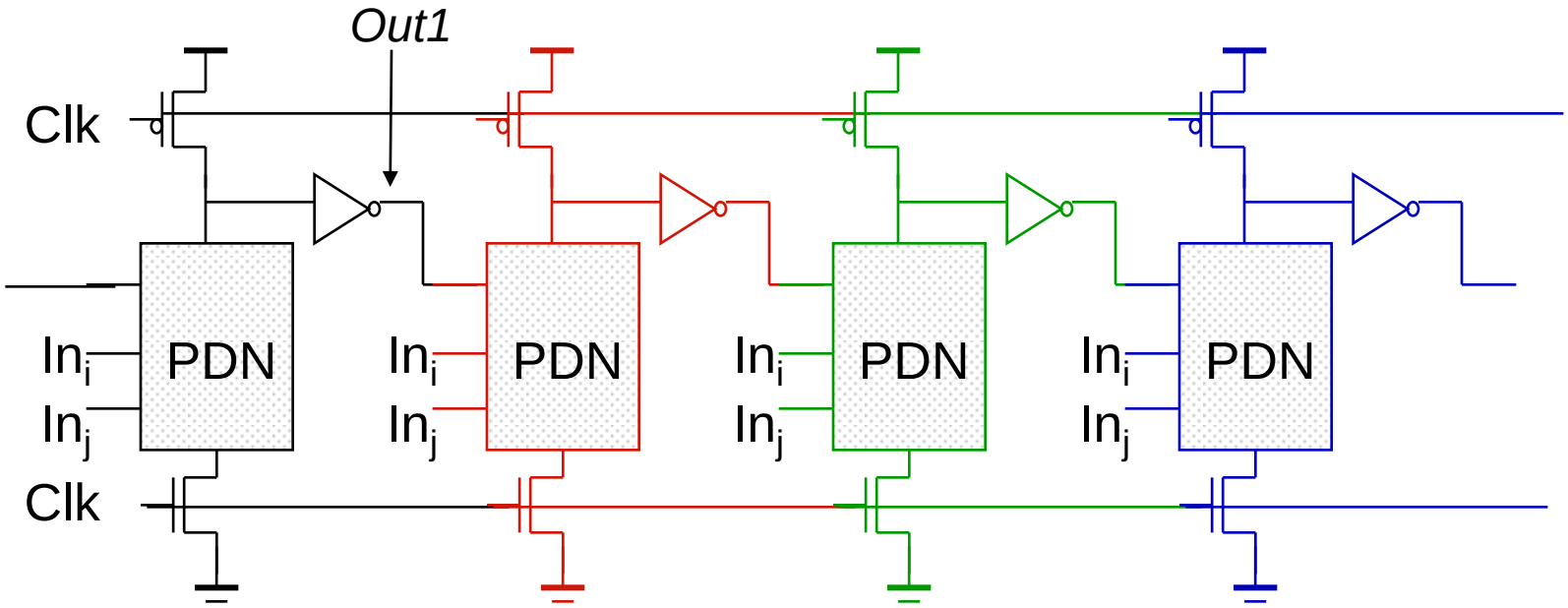


The static inverter reduces the capacitance of the dynamic output node by separating internal and load capacitances…..it also increases the NM (due to the low-impedance output)

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

Why Domino?

A Domino chain



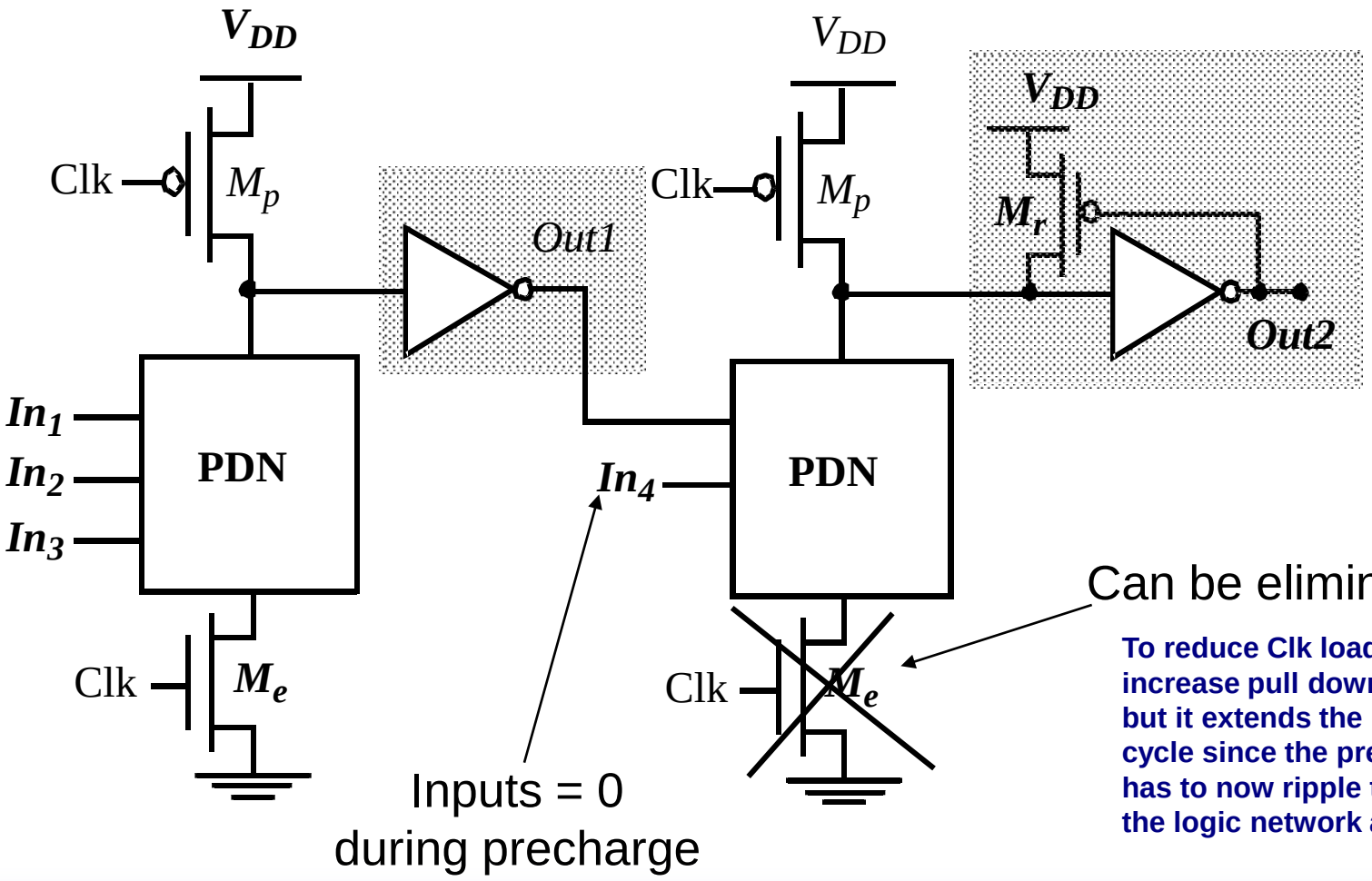
Precharge: all inputs=0

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!***

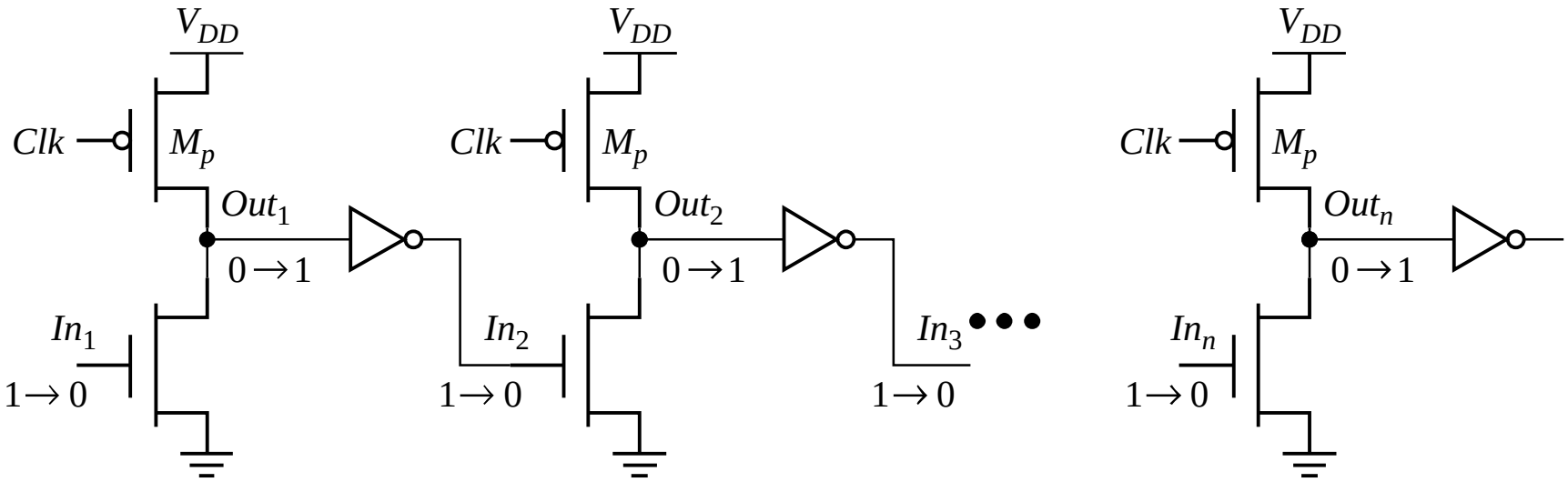
Properties of Domino CMOS Logic

- Only non-inverting logic can be implemented (due to the static inverter)
  - Major limitation
  - Can be overcome using dual-rail domino (an expensive solution)
- Very high speed
  - Only rising edge delays, and $t_{pHL}=0$
  - static inverter can be skewed to match the fanout, which is already much smaller than in the complimentary case, since only a single gate capacitance needs to be accounted for per fan-out gate.
  - Input capacitance reduced – smaller logical effort

Designing with Domino Logic



Footless Domino



If In_1=1, out_1=0 and In_2=1

On the falling edge of CLK, let In_1 =0: but it takes two gate delays for In_2 to be 0, during which second gate cannot pre-charge its output (Out_2), since PDN is fighting the precharge-PMOS

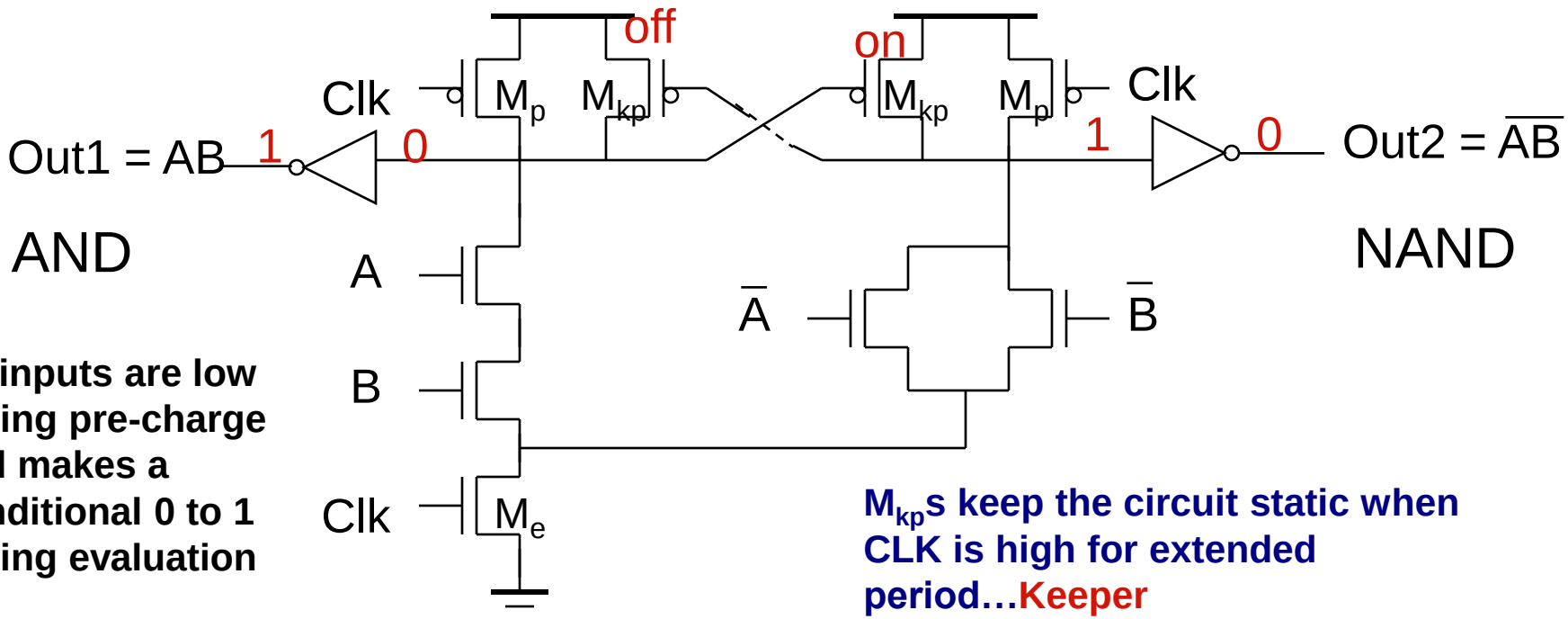
Time taken to precharge equals the critical path delay!
Better to use the evaluation device….

Pre-charge is rippling – short-circuit current
A solution is to delay the clock for each stage

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

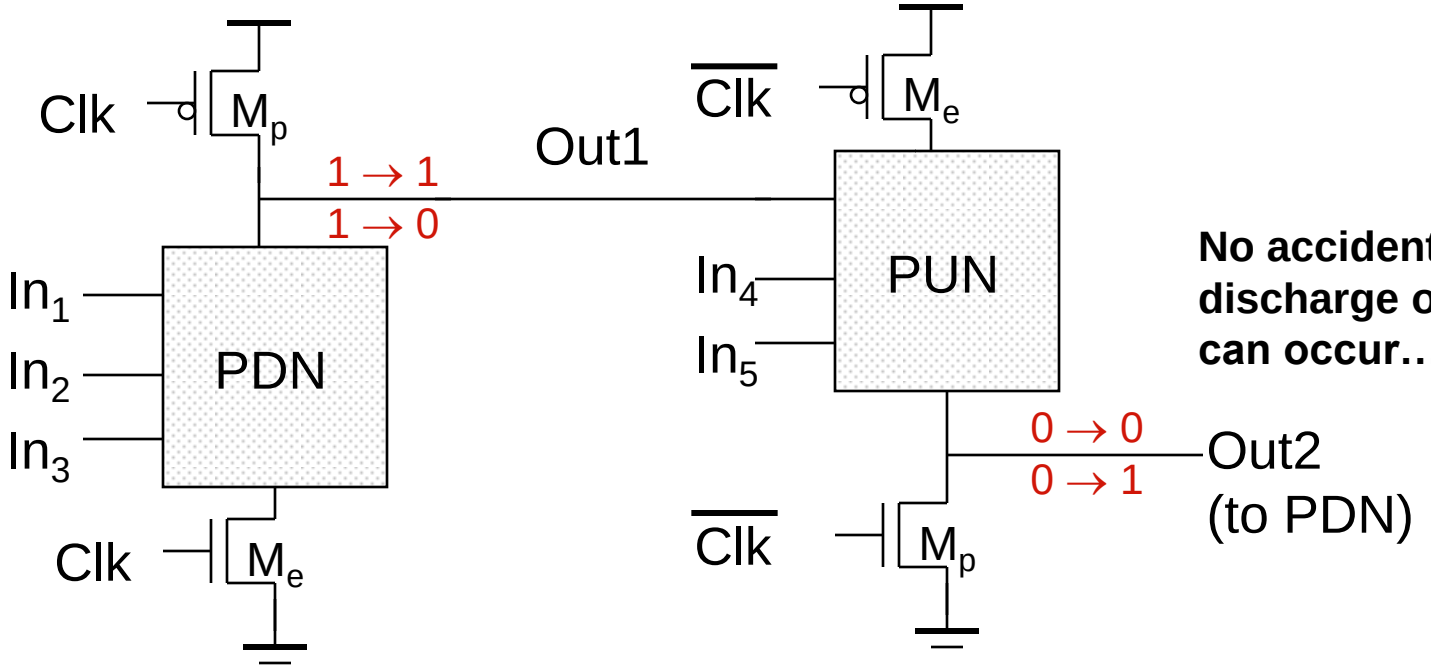
Possible to implement any arbitrary function….but comes at the expense of increased power since a transition is guaranteed every CLK cycle irrespective of the input values….either Out1 or Out2 must make a 0 to 1 transition.

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