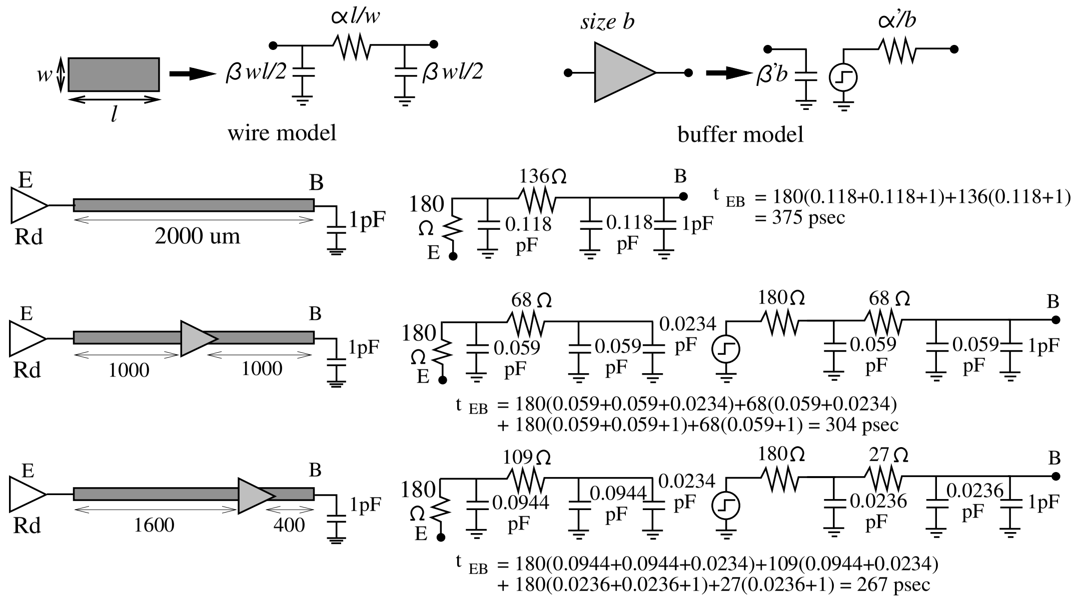
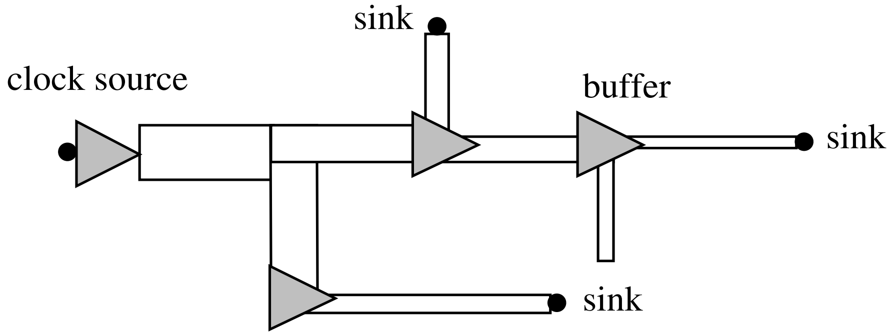
Example Elmore Delay for Buffered Wires

- Wire: $\alpha = 0.068\Omega/\mu m$, $\beta = 0.118 fF/\mu m^2$; buffer: $\alpha' = 180\Omega$/unit size, $\beta' = 23.4 fF$/unit size; driver resistance $R_d = 180\Omega$; unit-sized wire, buffer.



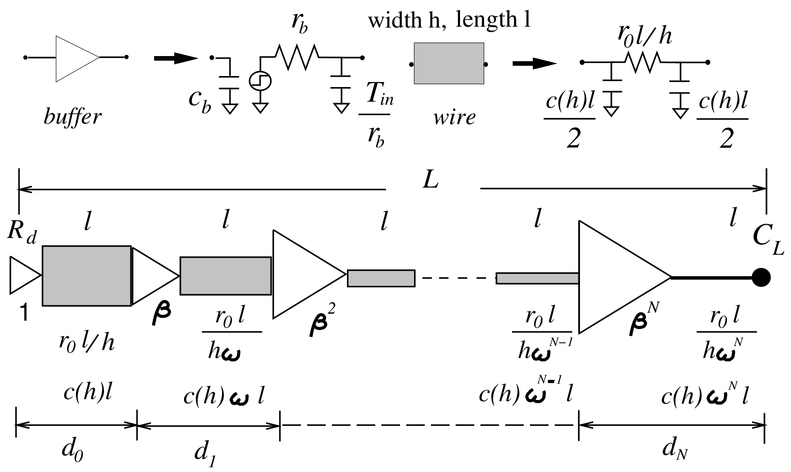
Wire Sizing & Buffering for Delay Optimization

- Size wires and buffers to optimize delay.
- Trade-off among area, delay, power, clock skew, clock-skew sensitivity, crosstalk, crosstalk sensitivity, etc.
- Popular techniques:
  - Discrete wire/buffer sizing: dynamic programming
  - Continuous wire/buffer sizing: mathematical programming (e.g., Lagrangian relaxation)



Finding Optimum Wire & Buffer Sizes

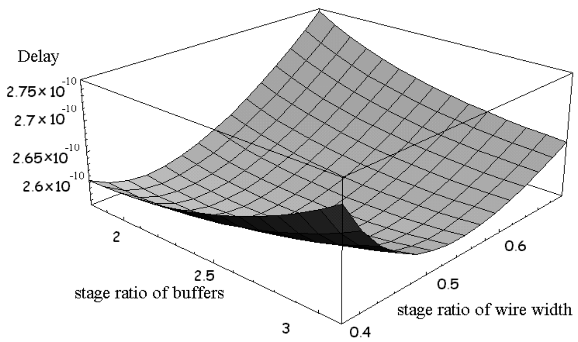
- **Input:** Wire length L, driver resistance R_d, load capacitance C_L, unit wire area capacitance c_0, unit wire fringing capacitance c_f, unit-sized wire resistance r_0, unit-size capacitance of a buffer c_b, unit-size buffer resistance r_b, intrinsic buffer delay T_{in}, and the number of buffers N.
- **Objective:** Determine the stage ratio β for buffer sizes and the stage ratio ω for wire widths such that the wire delay is minimized.



Wire/Buffer Size Ratios for Delay Optimization

$$
\begin{aligned}
D_N(\beta,\omega) &= R_d(c_0h + c_f)l + \beta R_d c_b + NT_{in} + \beta(N-1)r_b c_b + \frac{r_0 l C_L}{h\omega^N} + \frac{r_b}{\beta^N}C_L \\
&+ \frac{1}{2}(N+1)(r_0c_0l^2 + \frac{r_0c_fl^2}{h})\sum_{i=1}^{N}\frac{r_b}{\beta^i}(c_0\omega^i hl + c_fl) + \sum_{i=1}^{N}\frac{\beta^i}{\omega^i}\frac{r_0lc_b}{h}.
\end{aligned}
$$

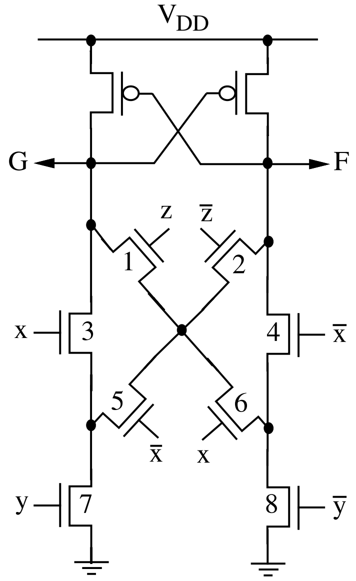
- The delay of a wire $D_N(\beta,\omega)$ is a convex function of the stage ratio β for buffer sizes and the stage ratio ω for wire widths.
- Can apply efficient search techniques (e.g., binary search) to find the optimum ratios.



Example PR Delay Computation

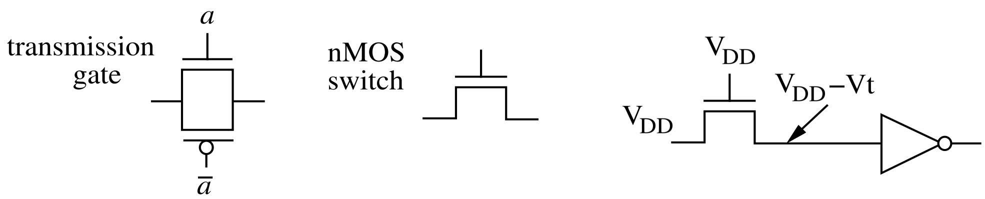
Using the Penfield-Rubenstein dealy model, estimate the delay for F to go from 1 to 0, when the inputs x, y, and z take on values of 001.

- Output loads at F, G: 0.25 pF; transistor on-resistance: 5$K\Omega$; gate capacitance: 0.01 pF; drain, source capacitance: 0.005 pF.



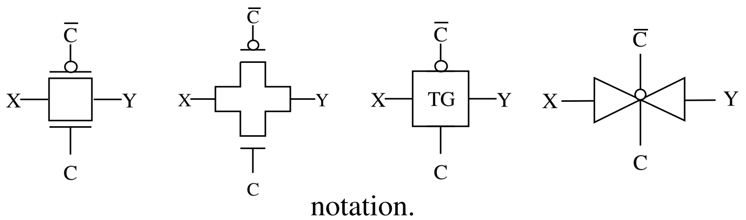
Switch Logic

- Can implement Boolean formulas as networks of switches.
- Can build switches from MOS transistors or transmission gates.
- Transmission gates transmit logic 0 and 1 from drain to source equally well.
  - n-type transistor conducts logic 0.
  - p-type transistor conducts logic 1.
- n-type switch has source-drain voltage drop when conducting.
  - Conducts logic 0 perfectly.
  - Introduces threshold drop into logic 1.

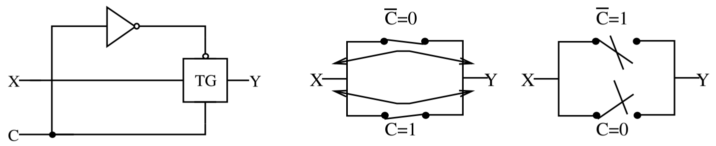


CMOS Transmission Gate (TG)

- A parallel n-channel and p-channel device driven by **opposite** control signals has two states:
  - **OFF (open)** when the drive signal is "0"
  - **ON (closed)** when the drive signal is "1"
- Current can flow either way through the TG (**could be dangerous**).



notation.

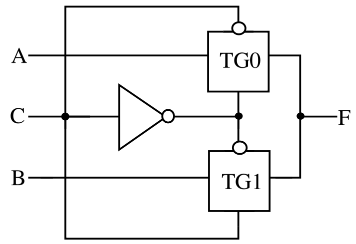


Control signals are mutually complemented.

C = 1: X and Y are connected;
C = 0: X and Y are disconnected.

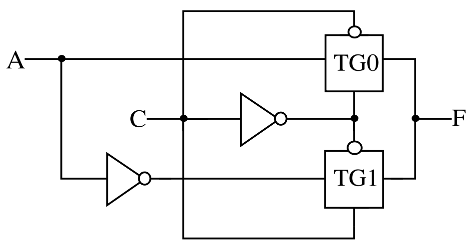
CMOS Logic with TGs

- Transmission gate-based multiplexer: ($C = 1 \Longrightarrow$ select B; $C = 0 \Longrightarrow$ select A)



$F(A, B, C) = A\overline{C} + BC$

- Transmission gate-based XOR logic:

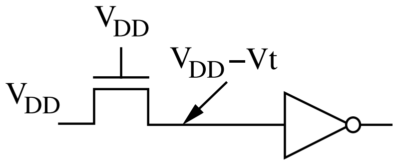


A	C	TG1	TG0	F
0	0	No path	Path	0
0	1	Path	No path	1
1	0	No path	Path	1
1	1	Path	No path	0

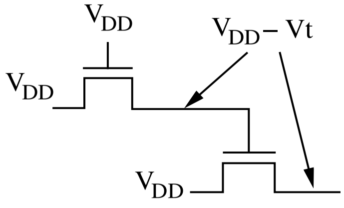
$F(A, C) = A \oplus C$

n-type Switch

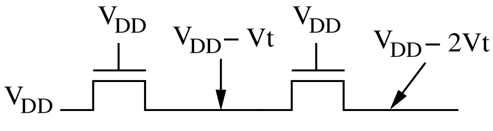
- Switch underdrives static gate, but gate restores logic levels.



- Voltage drop causes next stage to be turned on weakly or signal degradation.



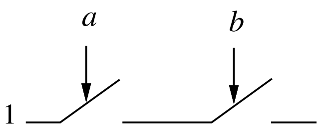
weaker control signal



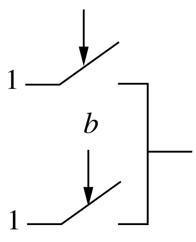
signal degradation

Switch Networks

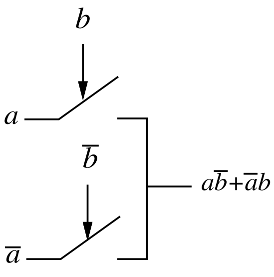
- Control inputs control the switches—a switch is closed when its control input is 1.
- If the switch network output is not connected to power supply through switch path, output is undefined (so called **pseudo**).
- Switch network inputs may be connected to power supply or logic signals.



pseudo–AND



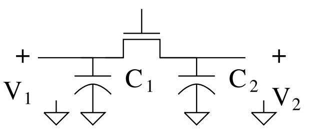
pseudo–OR

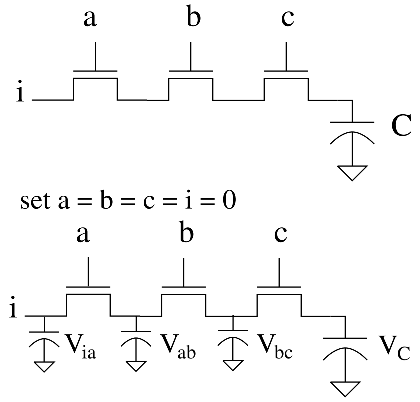


a switch network

Charge Sharing

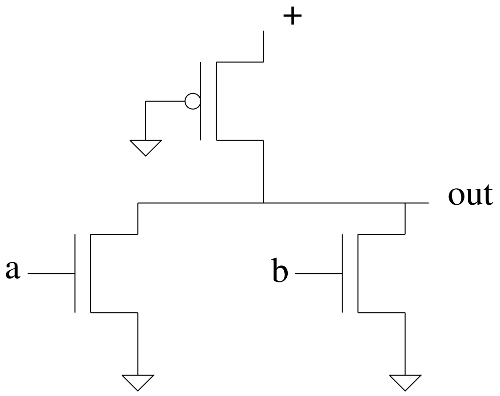
- Interior nodes in a switch network may not be driven.
- Charge can accumulate on parasitic capacitances.
- At undriven nodes, charge is divided according to capacitance ratio.
- Shared charge can produce erroneous output values.
- To avoid charge sharing, make sure that for every input combination there is a path from the power supply to the output.





Pseudo-nMOS Logic

- Use a p-type transistor as a resistor, n-type network for pulldowns.
- Characteristics
  - Consumes static power (both pullup and pulldown are ON, forming a conducting path from V_{DD} to V_{SS}).
  - Has much smaller pullup network than static gate.
  - Pulldown time is longer since both pullup and pulldown are fighting to determine the gate's output.

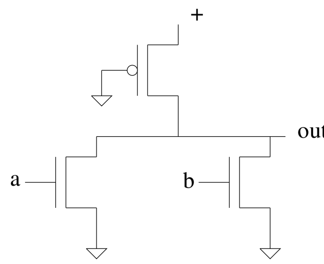


Transistor Ratio for Pseudo-nMOS

- Assume $V_{OL} = 0.25(V_{DD} - V_{SS})$.
- Must choose p, n transistor sizes to create effective resistances of the required ratio.
- For logic 0 output, pullup and pulldown form a voltage divider.
  - Pullup is in linear region; pulldown is in saturation.
  - Pullup and pulldown have same current flowing through them.
  - Equate two currents to get the p, n transistor ratio.
- For 0.5 μm, 3.3 V parameters:

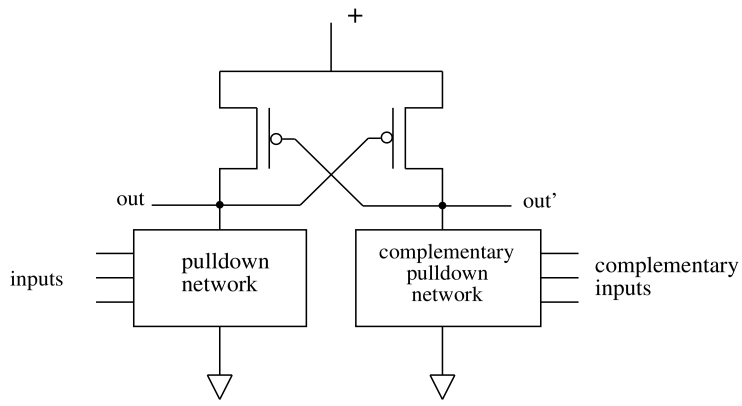
$$\frac{W_p/L_p}{W_n/L_n} = 3.9.$$

- For series n-type pulldown, we need larger transistors.



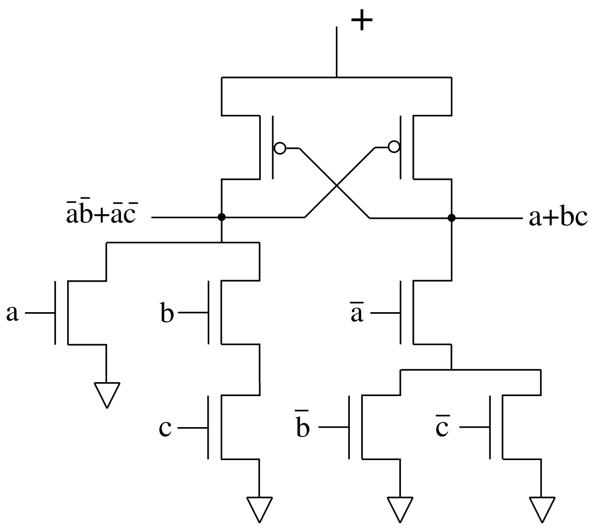
Differential Cascade Voltage Switch (DCVS) Logic

- Uses latch to compute output quickly.
- Requires true/complement inputs, produces true/complement outputs.
- Characteristics
  - Static logic, thus consumes no dynamic power.
  - Has much smaller pullup network than static gate, but larger pulldown.



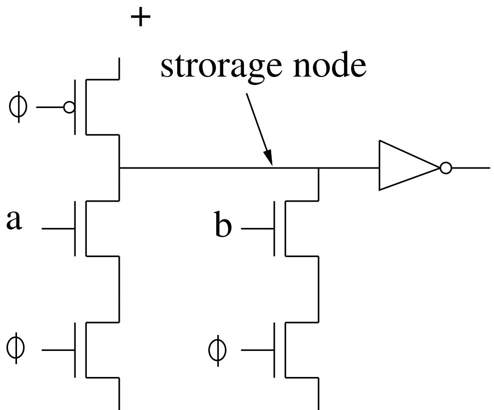
DCVS Logic Example

- Exactly one of true/complement pulldown networks will complete a path to the power supply.
- Pulldown network will lower output voltage, turning on other p-type transistor, which also turns off p-type transistor for node which is going down.



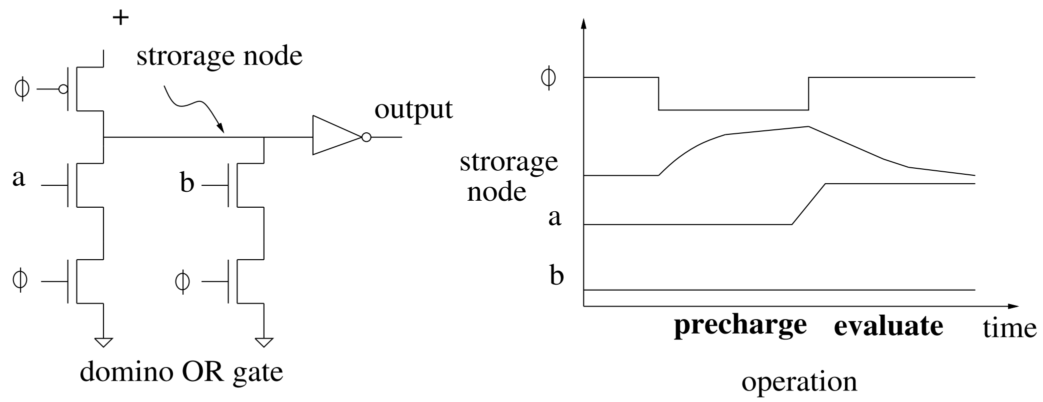
Precharged Circuit

- Precharged logic uses stored charge to help evaluation.
- Precharged node, selectively discharge it.
- Take advantage of higher speed of n-type transistors.
- Requires multiple phases for evaluation.
- Characteristics: low area and high speed, but sensitive to noise.



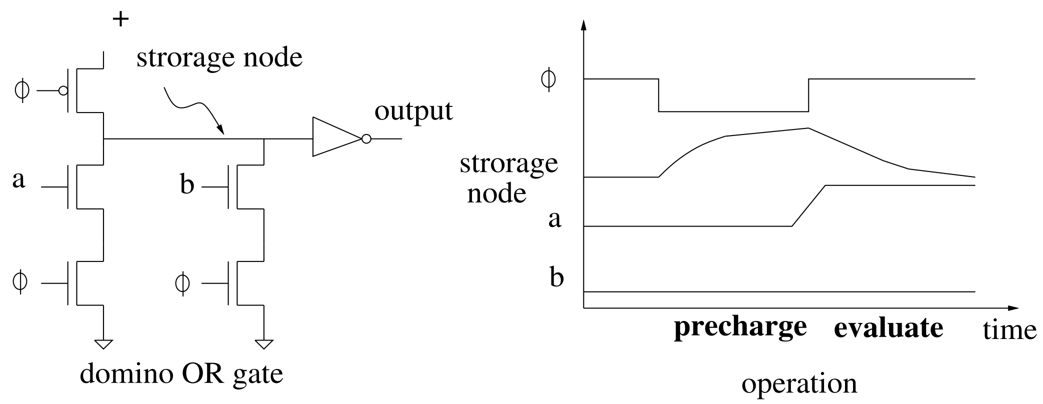
Domino Logic

- Use precharged clock ϕ to compute output in two phases
  - **Precharge:** when ϕ goes low, the p-type transistor starts charging the storage node (precharge capacitance); inverter ensures that output goes low.
  - **Evaluate:** when ϕ goes high, precharging stops and the evaluation phase begins; storage node may be pulled down, so output goes up.



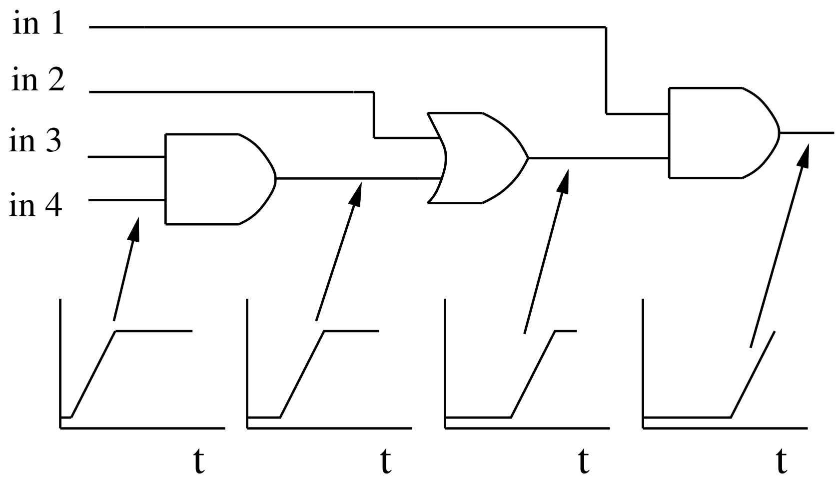
Properties of Domino Logic

- Domino gates' outputs must be monotonically increasing; glitch causes storage node to discharge.
- Not a complete logic family, since it cannot be inverted.
- About 30% faster than static logic.
- Lower noise margin.
- About 2X power dissipation.



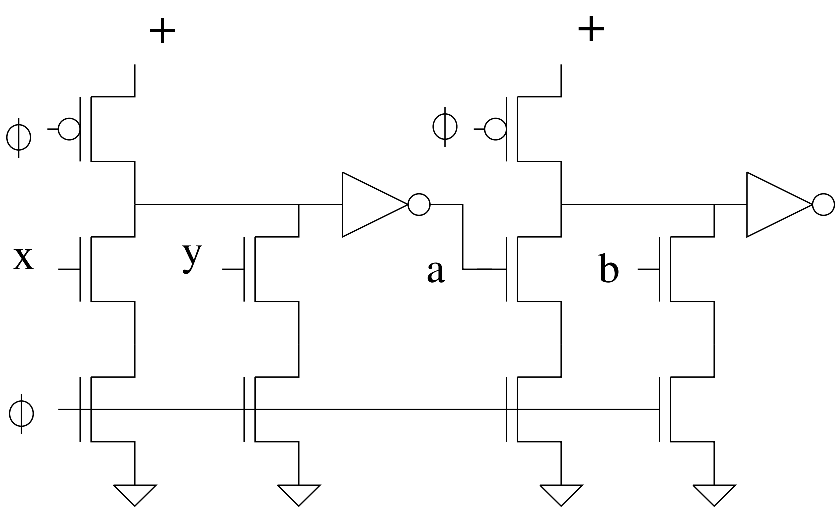
Domino Effect

- Gate outputs fall in sequence.



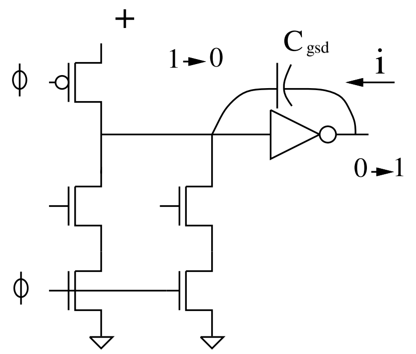
Domino Buffer

- Output inverter is needed for two reasons
  - **Make sure that outputs start low so that domino output can be connected to another domino gate.**
  - Protects storage node from outside influence.

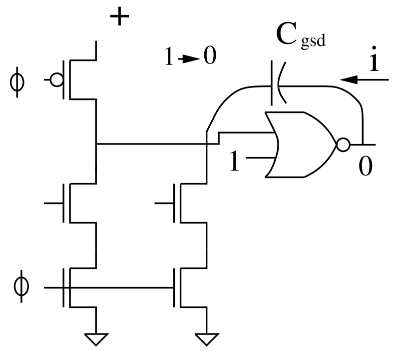


Domino Buffer

- Output inverter is needed for two reasons
  - Make sure that outputs start low so that domino output can be connected to another domino gate.
  - **Protects storage node from outside influence.**
- Inverting buffer isolates storage node. Storage node and inverter have correlated values.



output and storage node are correlated, can design circuit to withstand the effects of coupling capacitance



output and storage node are NOT correlated

Stored Charge in Domino Logic

- Charge can be stored in source/drain connections between pull-downs.
- Stored charge can be sufficient to affect the storage node.
- Can be avoided by precharging the internal pulldown network nodes along with the storage node.
- Must also avoid charge leakage through the substrate.