

VLSI Design

Lecture 9: MOS Logic Families

Shaahin Hessabi

Department of Computer Engineering

Sharif University of Technology

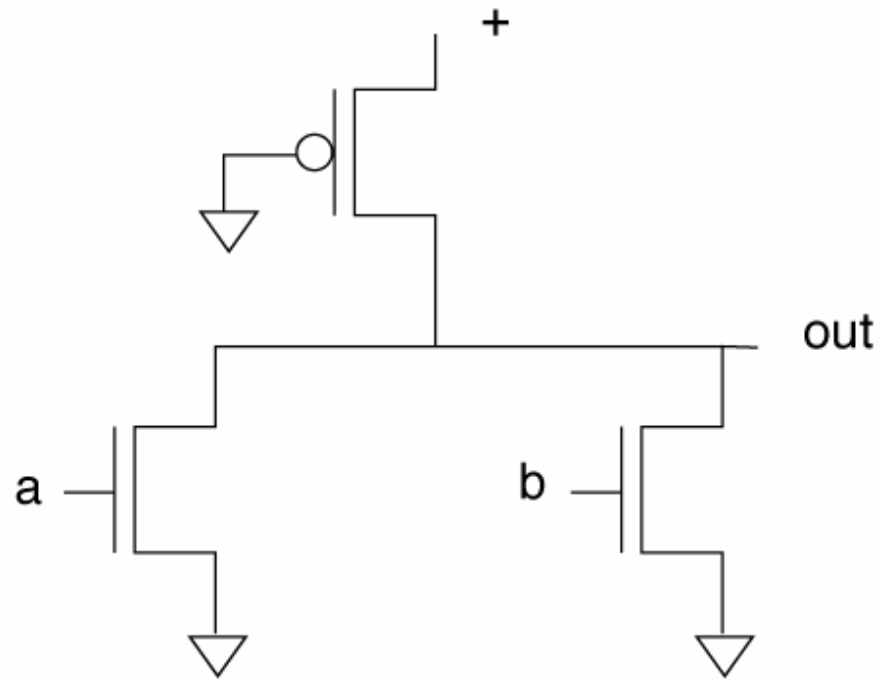
Adapted, with modifications, from lecture notes prepared
by the author (from Prentice Hall PTR)

Topics

- Pseudo-nMOS gates.
- DCVS logic.
- Domino gates.

Pseudo-NMOS

- Uses a p-type transistor as a resistive pullup, n-type network for pulldowns.

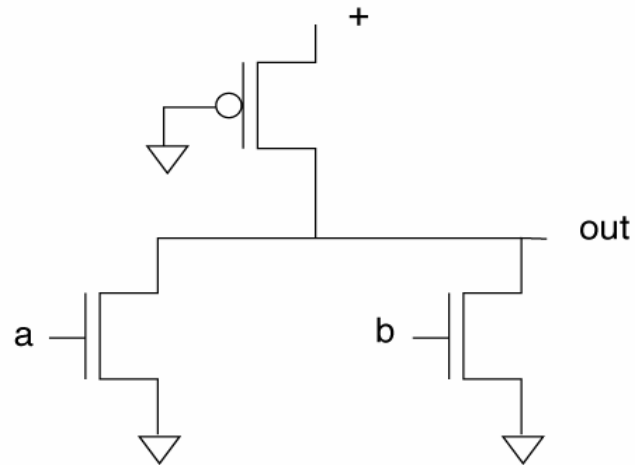


Characteristics

- Consumes static power.
- Has much smaller pullup network than static gate.
- Pulldown time is longer because pullup is fighting.

Output voltages

- Logic 1 output is always at V_{DD} .
- Logic 0 output is above V_{SS} .
- $V_{OL} = 0.25 (V_{DD} - V_{SS})$ is one plausible choice.

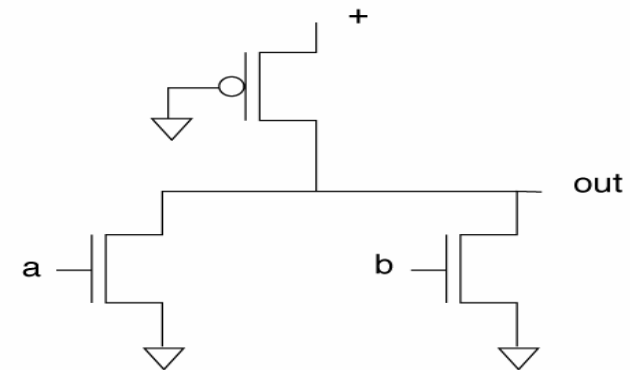


Producing output voltages

- For logic 0 output, pullup and pulldown form a voltage divider.
- Must choose n, p transistor sizes to create effective resistances of the required ratio.
- Effective resistance of pulldown network must be computed in worst case; series n-types means larger transistors.

Transistor ratio calculation

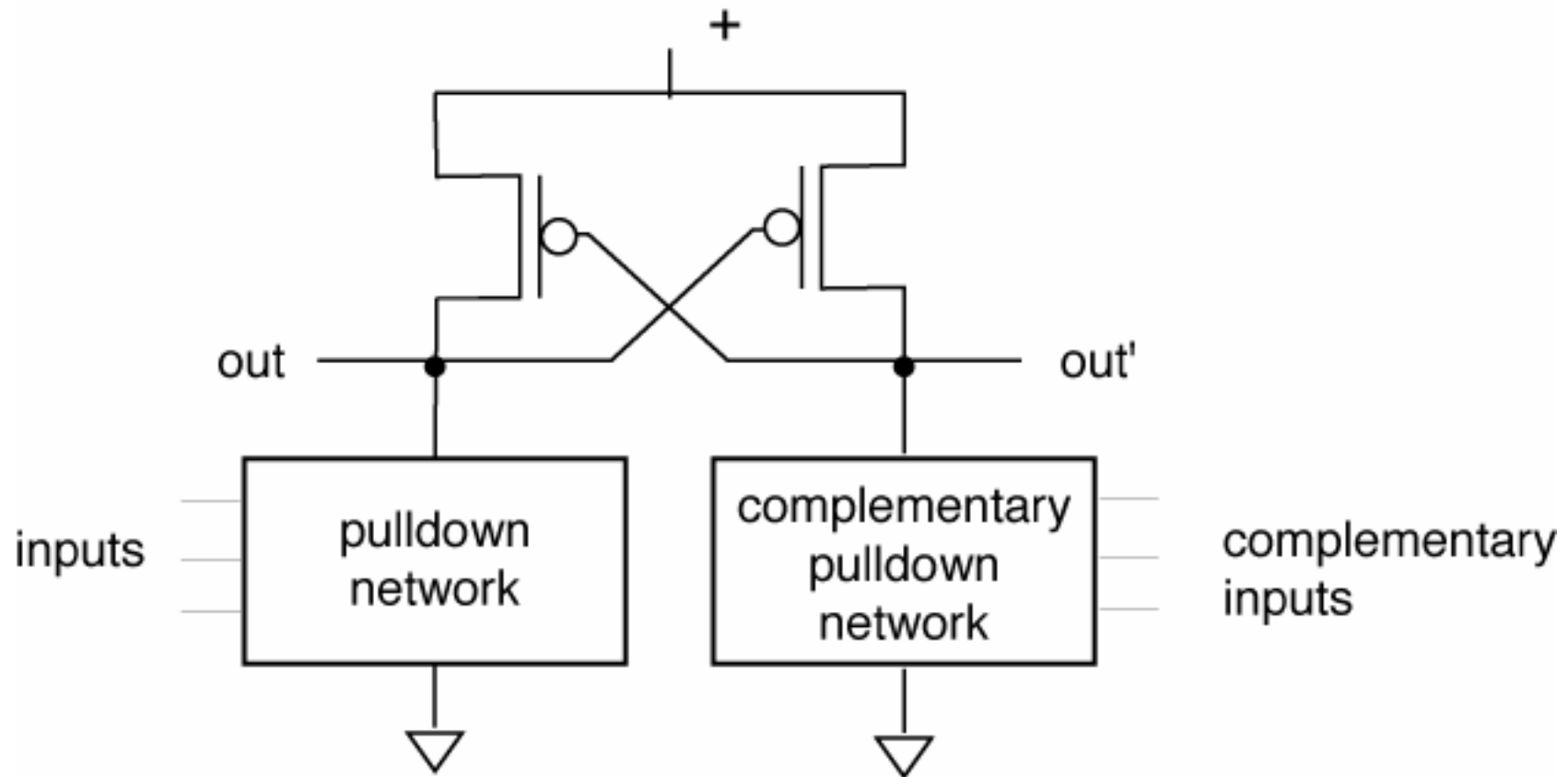
- For creating logic 0 output, (initially):
 - pullup is in linear region, $V_{ds} = V_{out} - (V_{DD} - V_{SS})$;
 - pulldown is in saturation $V_{ds} = (V_{DD} - V_{SS})$
- Pullup and pulldown have same current flowing through them; i.e., $-I_{dp} = I_{dn}$
- For equal noise margins, using $0.5\ \mu\text{m}$ parameters, 3.3V power supply:
 - $(W_p/L_p) / (W_n/L_n) = 3.9$



DCVS logic

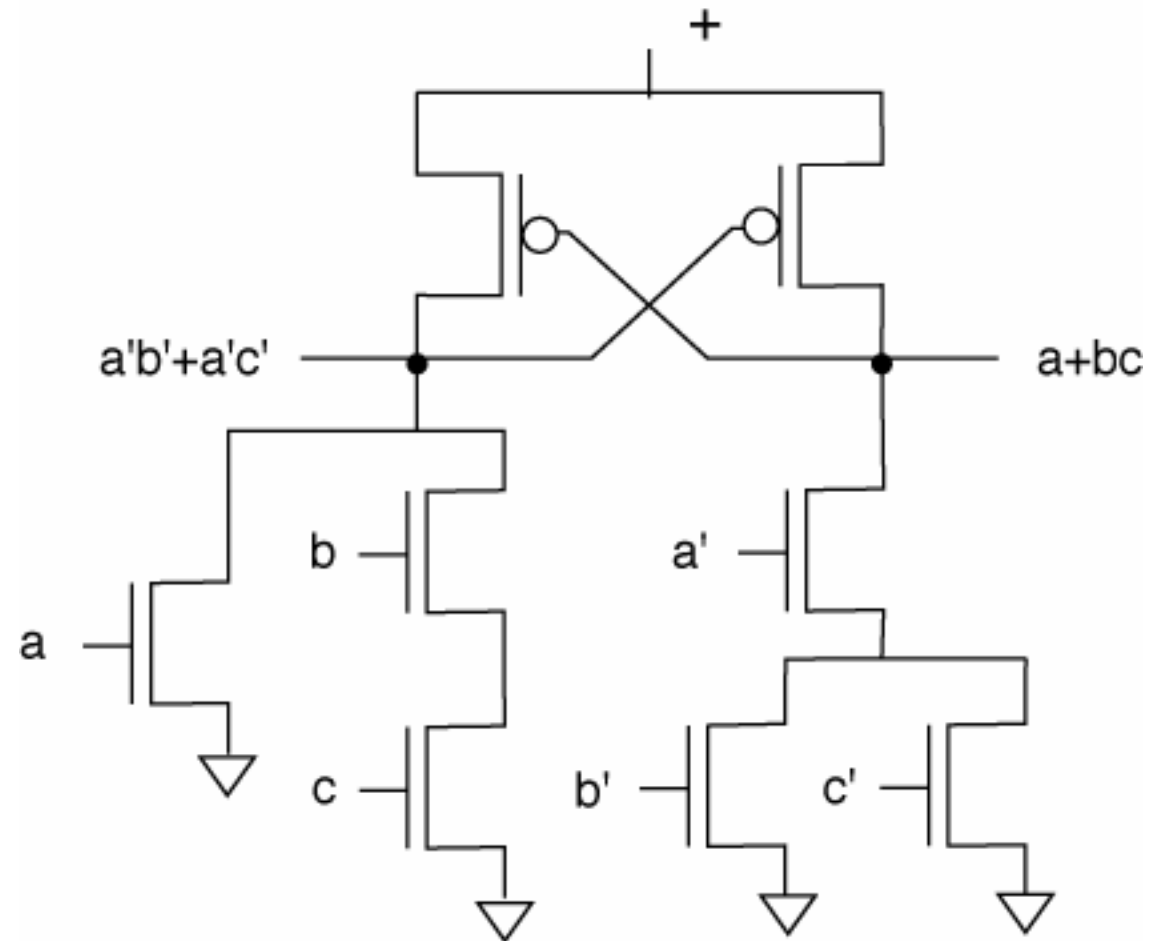
- DCVSL = differential cascode voltage switch logic.
- Static logic.
 - consumes no static power (like standard CMOS)
- Uses latch to compute output quickly.
- Requires true/complement inputs, produces true/complement outputs.

DCVS structure



DCVS operation

- 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, which also turns off p-type for node which is going down.

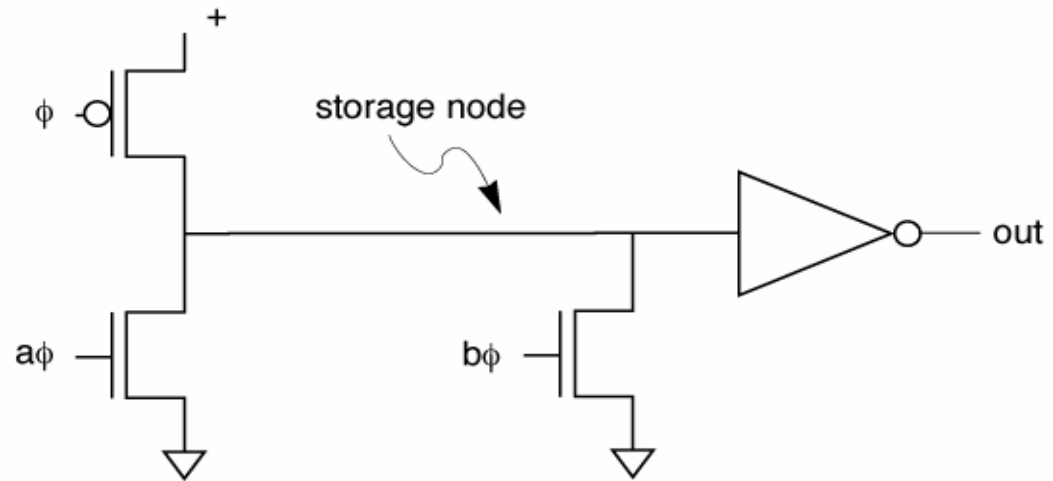


Precharged logic

- Precharged logic uses stored charge to help evaluation.
- Precharge node, selectively discharge it.
- Take advantage of higher speed of n-types.
- Requires multiple phases for evaluation.

Domino logic

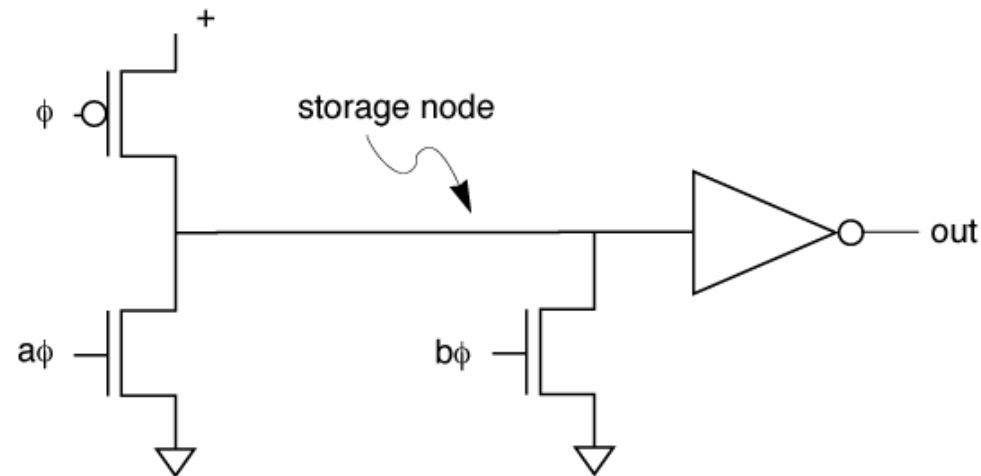
- Uses precharge clock to compute output in two phases:
 - precharge;
 - evaluate.



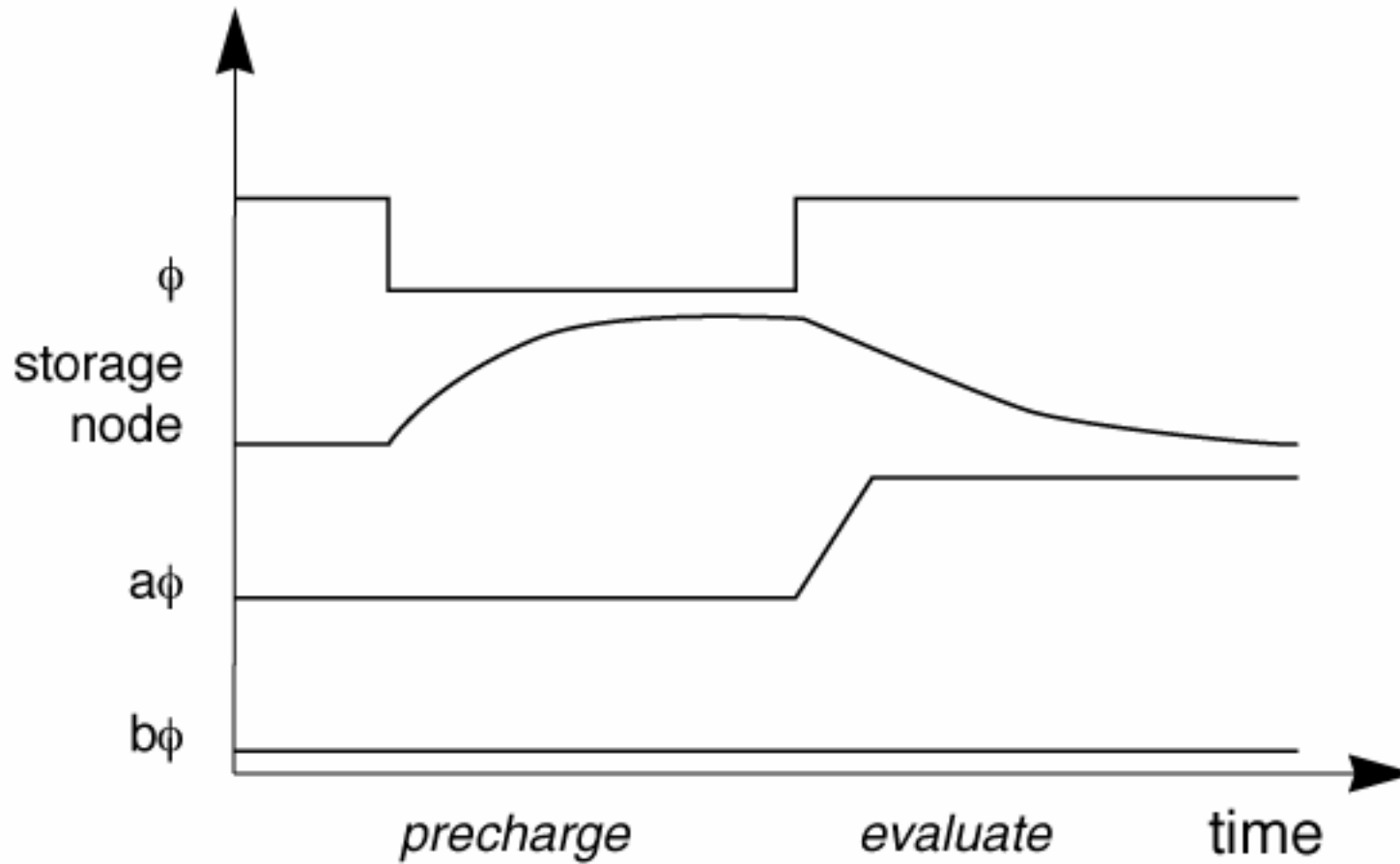
- Not a complete logic family; cannot invert.

Domino phases

- Controlled by clock ϕ .
- Precharge: p-type pullup precharges the storage node; inverter ensures that output goes low.
- Evaluate: storage node may be pulled down, so output goes up.

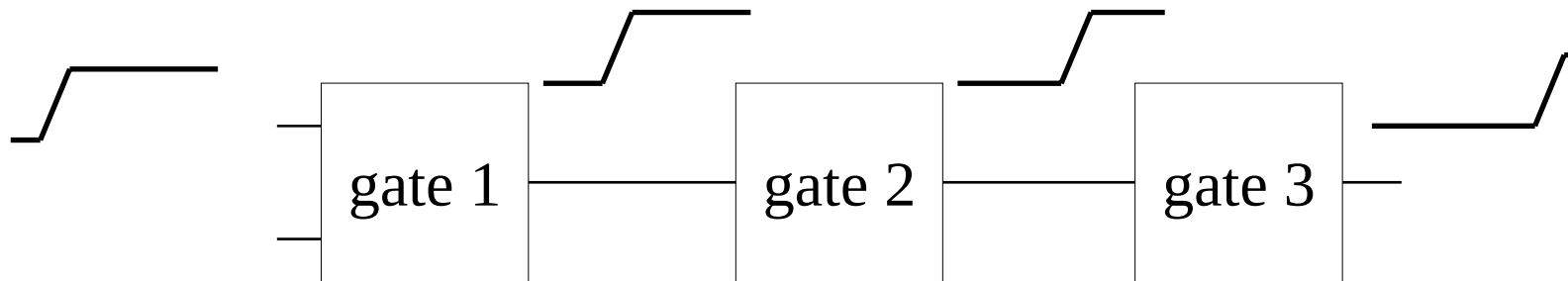


Domino operation



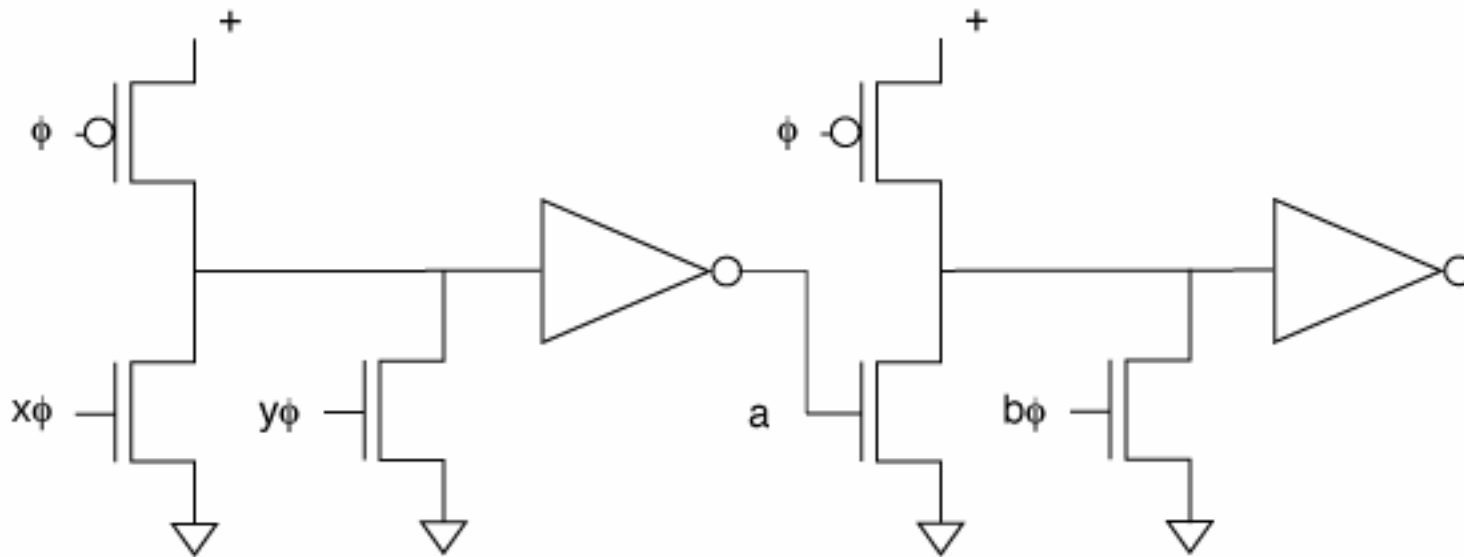
Domino effect

Gate outputs fall (rise) in sequence:



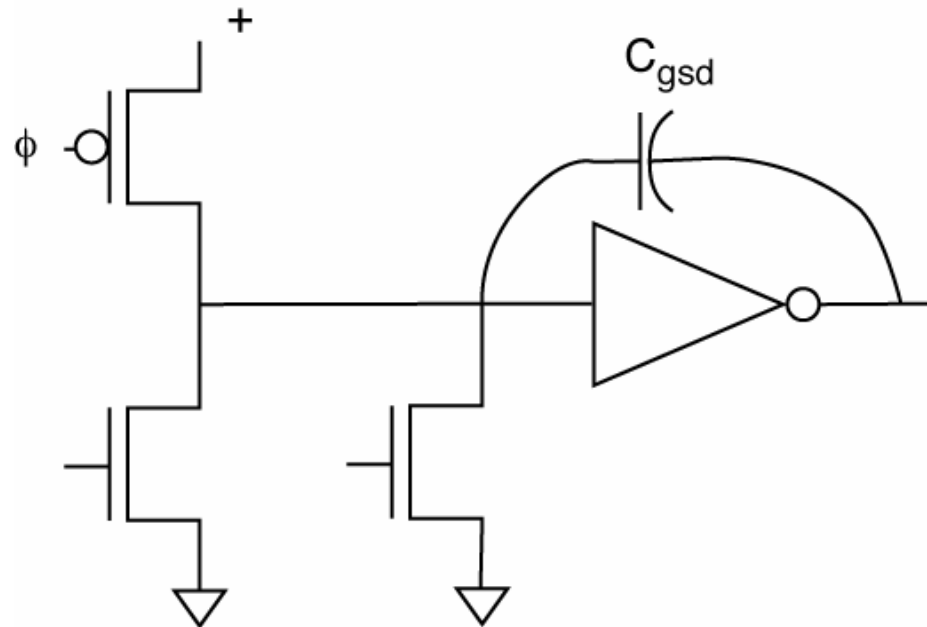
Monotonicity

- Domino gates inputs must be monotonically increasing: glitch causes storage node to discharge.



Output buffer

- Inverting buffer isolates storage node.
- Storage node and inverter have correlated values.



Domino buffer

- Output inverter is needed for two reasons:
 1. make sure that outputs start low, go high so that domino output can be connected to another domino gate;
 - Can it be avoided by using an NMOS (controlled by ϕ) in series with the pull-down network?
 - (consider two cascaded gates)
 2. protects storage node from outside influence.

Using domino logic

- Can rewrite logic expression using DeMorgan's Laws:
 - $(a + b)' = a'b'$
 - $(ab)' = a' + b'$
- Add inverters to network inputs/outputs as required.

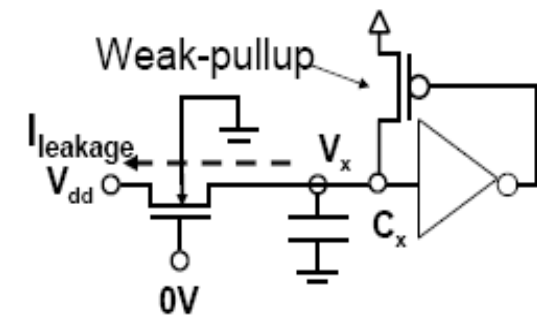
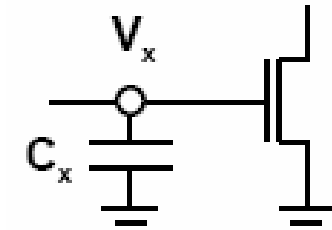
Charge-Storage Principle

- Node X holds charge for “long” periods
- t_H : time to bring node X from V_{dd} to $0.5 V_{dd}$
- C_x : dynamic node capacitance = $C_{ox} WL$

$$t_H = \frac{C_x (0.5 V_{dd})}{I_{leakage}}$$

$$= \frac{(6 \times 10^{-7})(0.36 \times 0.25 \times 10^{-8})(0.5 \times 2.5)}{1 \times 10^{-12}} = 0.675 \text{ ms}$$

- $t_{clk} \ll t_H$ for dynamic circuits to work.
- Future trends:
 - C_x decreases; V_{dd} decreases; $I_{leakage}$ increases.
 - This implies t_H decreases but so does t_{clk} .
- Use a keeper transistor

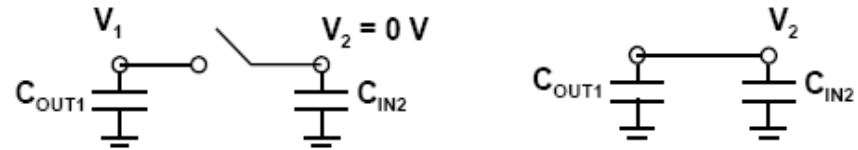


Domino and stored charge

- Charge can be stored in source/drain connections between pulldowns.
- Stored charge can be sufficient to affect precharge node.
- Can be averted by precharging the internal pulldown network nodes along with the precharge node.

Charge-sharing

■ What is the value of V_2 ?



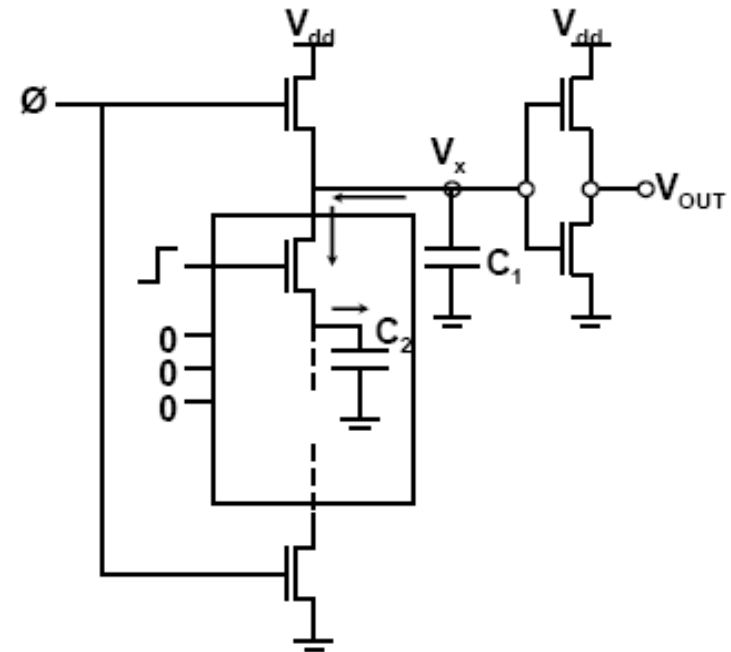
$$C_{out1}V_1 = (C_{out1} + C_{in2})V_2$$

$$\frac{C_{out1}}{C_{out1} + C_{in2}}V_1 = V_2$$

$$V_1 \approx V_2 \quad \text{iff} \quad C_{out1} \gg C_{in2}$$

■ Solution:

- Make $C_1 \gg C_2$
- Reduce V_t of the output inverter
- Use a keeper/bleeder transistor
- Use Multiple precharge transistors



Dynamic logic vs. Static Logic

- Faster. Used in data-paths of high performance microprocessors.
- Smaller area
- Extra clock signal. Consumes extra power.
- Dynamic power depends upon probability of logic values rather than probability of a transition
- Susceptible to noise. Careful physical design required.