University of Southern California

Viterbi School of Engineering

EE477L

MOS VLSI Circuit Design

Dynamic Logic Circuits

Shahin Nazarian

Spring 2013

Dynamic Logic

- Dynamic logic circuits
  - depend on temporary (transient) storage of charge in parasitic node capacitances
  - need periodic updating of internal node voltage levels
  - require periodic clock signals in order to control charge refreshing
  - use a common clock signal in order to enable us to synchronize the operations of various circuit blocks
  - have smaller area than the static logic implementation

Dynamic Logic Circuit Types

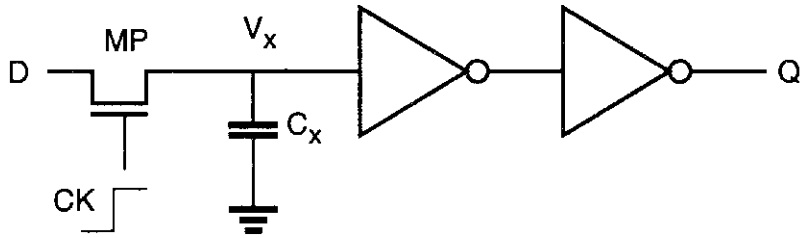
- Ratioed vs Ratioless

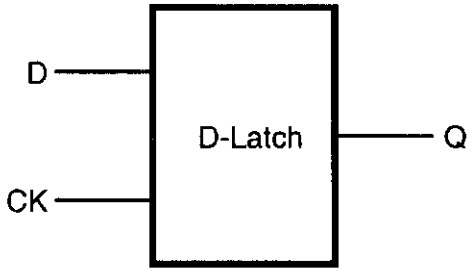
- Dynamic Pass Transistor Logic
- Dynamic Transmission Gate Logic

- Circuit Reviewed in EE477L
  - Domino CMOS
  - Multiple Output Domino
  - Dual-Rail Domino
  - NP-Domino (NORA)
  - Zipper
  - TSPC

Example 9.1 – Dynamic D-latch

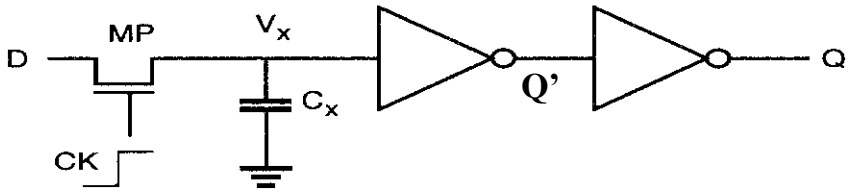
- Consider the dynamic D-latch circuit shown below
  - The parasitic input capacitance C_x plays an important role in the dynamic operation of this circuit
  - When CK=1, MP turns on, C_x is either charged up, or charged down through the pass transistor MP, depending on the input (D) voltage level. The output (Q) assumes the same logic level as the input
  - When CK=0, MP is off and C_x is isolated from D. The amount of charge stored in C_x during the previous cycle determines V_Q
  - Node X is also called a soft node
    - Nature of soft node makes the dynamic circuit more vulnerable to single-event upsets (SEUs) caused by α-particle or cosmic ray hits in integrated circuits

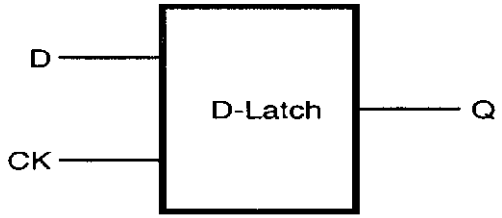




Example 9.1 (Cont.)

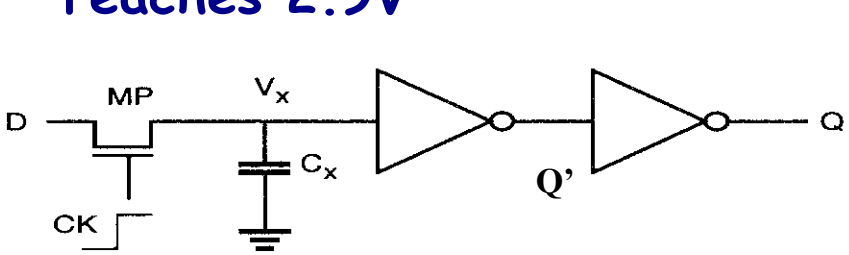
- Consider V_{DD}=5 V and assume that the VTC's of both inverters are identical, V_{OL}=0V, V_{IL}=2.1V, V_{IH}=2.9V, V_{OH}=5.0V, and $V_{T,n}$=0.8V
  - When CK=1 and $V_{in}=V_{OH}$=5 V:
    - MP is conducting and C_x is charged up to a logic high level
    - nMOS is a poor conductor for logic "1" and its output voltage V_x will be lower than V_{OH}
    - V_x=5.0-0.8=4.2V (which is more than V_{IH} of the first inverter, therefore $V_{Q'}$ is very close to V_{OL}=0V
    - Consequently Q becomes a logic "1", i.e., $V_Q=V_{DD}$

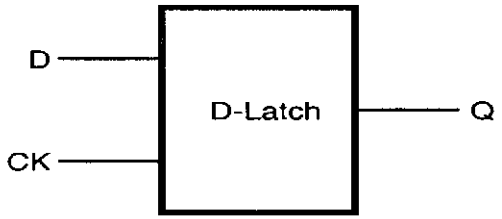




Example 9.1 (Cont.)

- When CK=0 (MP is off)
  - Initially X is high and Q is high. V_X starts to drop because of charge leakage from the soft node
  - To keep Q at logic "1", the voltage level at node X cannot be allowed to drop lower than V_{IH}=2.9 V
  - Clock signal can be kept low for as long as it takes for V_X to drop from 4.2V to 2.9V due to charge leakage. To avoid an erroneous output, the charge stored in C_X must be restored or refreshed to its original level before V_X reaches 2.9V

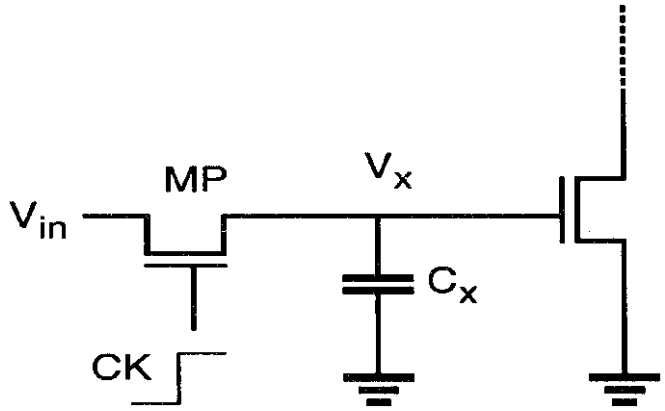




- This example shows the dynamic-charge storage principle in D-latch is quite feasible for preserving output state during inactive clock phase, assuming leakage currents are small

Basic Principle of Pass Transistor Circuits

- The two possible operations when clock is high (CK=1)
  - Logic "1" transfer (charging up C_x to a logic-high level)
  - Logic "0" transfer (charging down C_x to a logic-low level)
- When CK=0, MP ceases to conduct and the charge stored in C_x continues to determine the output level of the inverter



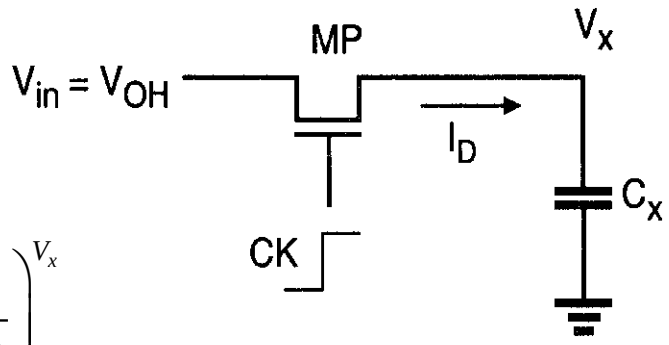
The basic building block for NMOS dynamic logic, which consists of an NMOS pass transistor driving the gate of another NMOS transistor

Logic "1" Transfer

- Assume initially $V_x(t=0) = 0V$
- Apply $V_{in}=V_{OH}=V_{DD}$
- Clock signal goes from 0 to V_{DD} at t=0 and MP starts to conduct
- Since $V_{DS}=V_{GS}$, **MP is in saturation**
- MP starts to charge up C_x:
  - To simplify analysis we neglect the substrate bias effect ($V_{SB}=V_X$) on V_T

$$C_x \frac{dV_x}{dt} = \frac{k_n}{2}(V_{DD} - V_x - V_{T,n})^2$$

$$\int_0^t dt = \frac{2C_x}{k_n}\int_0^{V_x} \frac{dV_x}{(V_{DD} - V_x - V_{T,n})^2} = \frac{2C_x}{k_n}\left(\frac{1}{(V_{DD} - V_x - V_{T,n})}\right)_0^{V_x}$$



Equivalent circuit for the logic "1" transfer

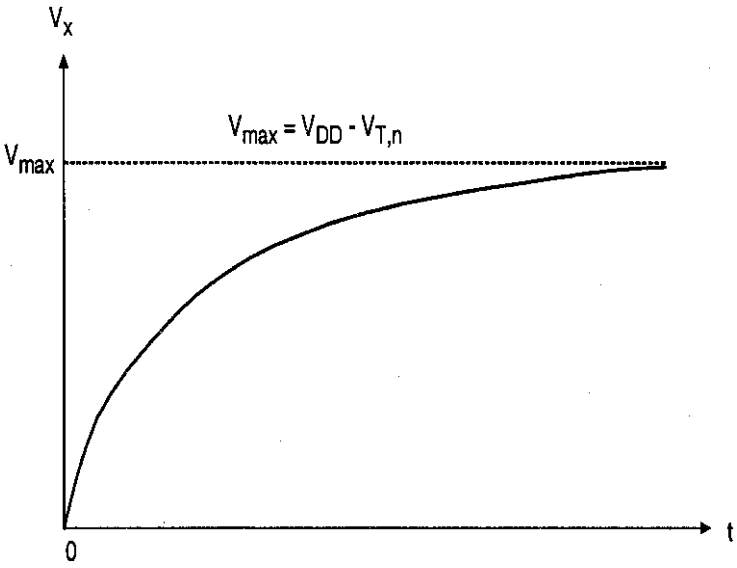
Logic "1" Transfer (Cont.)

$$t = \frac{2C_x}{k_n} \cdot \left[\left(\frac{1}{V_{DD} - V_x - V_{T,n}} \right) - \left(\frac{1}{V_{DD} - V_{T,n}} \right) \right]$$

$$V_x(t) = (V_{DD} - V_{T,n}) \cdot \frac{\left(\frac{k_n (V_{DD} - V_{T,n})}{2C_x} \right) t}{1 + \left(\frac{k_n (V_{DD} - V_{T,n})}{2C_x} \right) t}$$

- **MP turns off when $V_x = V_{max}$ since at this point its gate-to-source voltage will be equal to the nMOS threshold voltage**

$$V_{\max} = V_x \big|_{t \to \infty} = V_{DD} - V_{T,n}$$
$$= V_{DD} - V_{T0,n} - \gamma \left(\sqrt{|2\phi_F| + V_{\max}} - \sqrt{|2\phi_F|} \right)$$

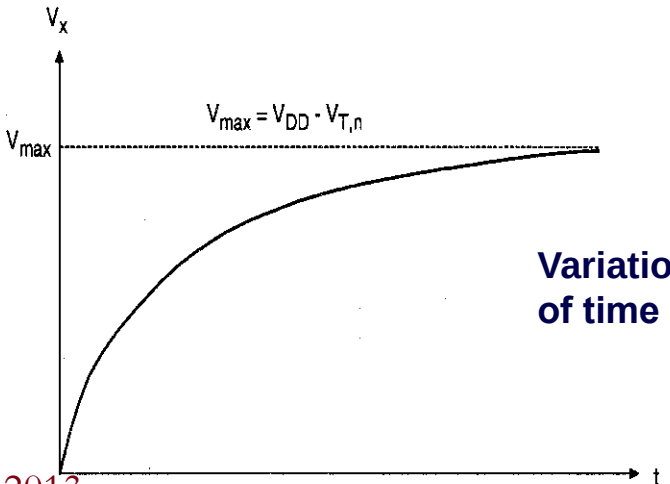


Variation of V_x as a function of time during logic "1" transfer

Logic "1" Transfer (Cont.)

$$V_{\max} = V_x|_{t\to\infty} = V_{DD} - V_{T,n}$$
$$= V_{DD} - V_{T0,n} - \gamma\left(\sqrt{|2\phi_F| + V_{\max}} - \sqrt{|2\phi_F|}\right)$$

- Note that the rise time will be underestimated if $V_{T0,n}$ is used (i.e., if body bias is ignored) because drain current of nMOS is decreased due to body bias effect
- The fact that V_x has an upper limit of $V_{max}=(V_{DD}-V_{T,n})$ has a significant implication for circuit design (shown later using the chain of pass transistors)



Variation of V_x as a function of time during logic "1" transfer

Rise Time Calculation

- **Rise time calculation for V_x: $t_{10\%}=0.1V_{max}$, $t_{90\%}=0.9V_{max}$**
  - **Note that $V_{T,n}$ in equations below refers to $V_{T,n}(V_{max})$, and not $V_{T,n}(0.1V_{max})$ or $V_{T,n}(0.9V_{max})$**

$$t_{10\%} = \frac{2C_x}{k_n}\cdot\left[\left(\frac{1}{0.9(V_{DD}-V_{T,n})}\right)-\left(\frac{1}{V_{DD}-V_{T,n}}\right)\right] = \frac{2C_x}{k_n(V_{DD}-V_{T,n})}\cdot\left(\frac{1}{9}\right)$$

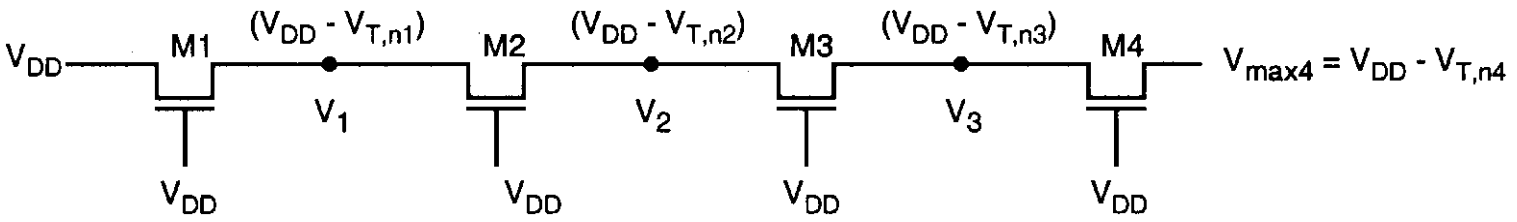
$$t_{90\%} = \frac{2C_x}{k_n(V_{DD}-V_{T,n})}\cdot(9)$$

$$\tau_{rise} = t_{90\%}-t_{10\%} = \frac{2C_x}{k_n(V_{DD}-V_{T,n})}\cdot\left(9-\frac{1}{9}\right)$$

$$\tau_{rise} = 17.78\frac{C_x}{k_n(V_{DD}-V_{T,n})}$$

A Chain of Series-Connected Pass Transistors

- Consider the following case in which a logic "1" at the input node is being transferred through a chain of cascaded pass transistors
  - Assume that initially V1 through V4 are zero
  - Pass transistors are identical: $V_{T,n1}=V_{T,n2}=V_{T,n3}=\ldots$
  - M1 is in saturation –> $V_{max1}=V_{DD} - V_{T,n1}$; M2 operates at saturation boundary (linear)–> $V_{max2}=V_{DD} - V_{T,n2}$
  - The voltage at the end of the pass transistor chain will become one threshold voltage less than V_{DD} regardless of the number of pass transistors in the chain

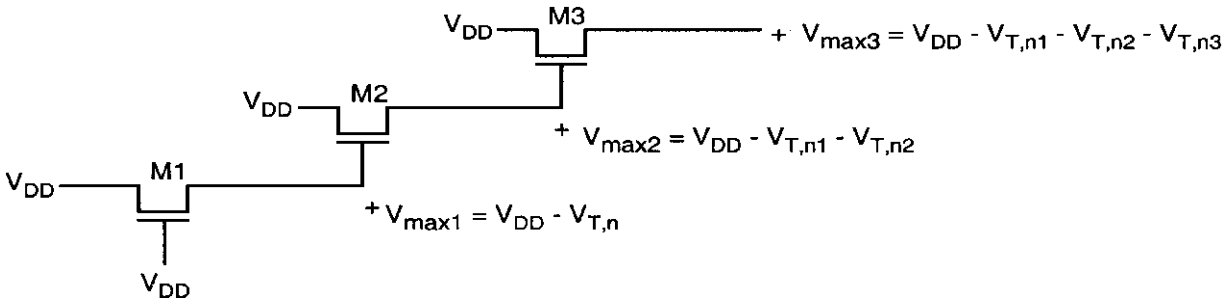


Node voltages in a pass-transistor chain during logic "1" transfer

A Chain of Pass Transistors Driving Each Other

- Different case: output of each pass transistor drives the gate of another pass transistor
  - $V_{max1}=V_{DD}-V_{T,n1}$
  - M2 operates in saturation region, $V_{max2}=V_{DD}-V_{T,n1}-V_{T,n2}$
  - Each stage causes a significant loss of voltage level
  - Realistically the loss is more, considering substrate bias effect, which is different in all stages (higher V_T values)

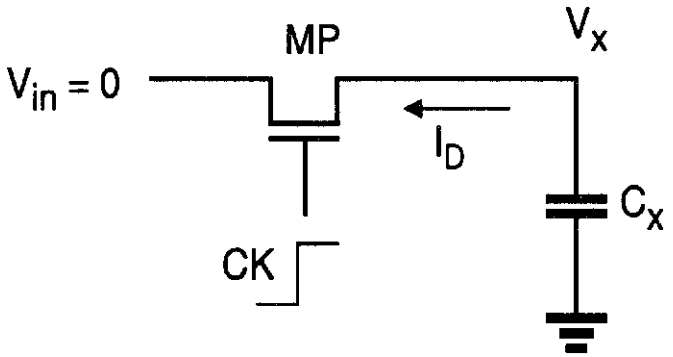
$$V_{T,n1}=V_{T0,n}+\gamma\left(\sqrt{|2\phi_F|+V_{\max 1}}-\sqrt{|2\phi_F|}\right) \qquad V_{T,n2}=V_{T0,n}+\gamma\left(\sqrt{|2\phi_F|+V_{\max 2}}-\sqrt{|2\phi_F|}\right)$$



Node voltages during logic "1" transfer ; each pass transistor is driving another pass transistor

Logic "0" Transfer

- Assume $V_x(t=0) = V_{max} = V_{DD}-V_{T,n}$
- Apply a logic "0" level to V_{in}
- Clock signal from 0 to V_{DD} at t=0, MP starts to conduct
- Direction of drain current flow through MP will be opposite to that during the charge-up event
- X corresponds to drain terminal of MP and input node will correspond to its source terminal
- $V_{GS}=V_{DD}$ and $V_{DS}=V_{max}$ ($V_{DS} < V_{GS} - V_{T,n}$) $\Rightarrow$ **MP operates in the linear region**



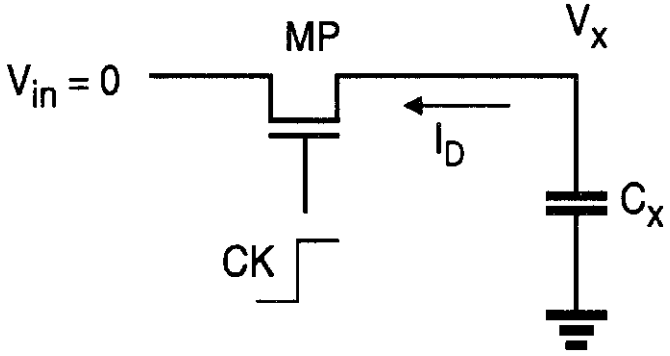
Equivalent circuit for the logic "0" transfer event

Logic "0" Transfer (Cont.)

- MP, which operates in the linear region, discharges C_x as follows:

$$-C_x \frac{dV_x}{dt} = \frac{k_n}{2}\left(2(V_{DD} - V_{T,n})V_x - V_x^2\right)$$

$$dt = -\frac{2C_x}{k_n} \cdot \frac{dV_x}{2\left(V_{DD} - V_{T,n}\right)V_x - V_x^2}$$



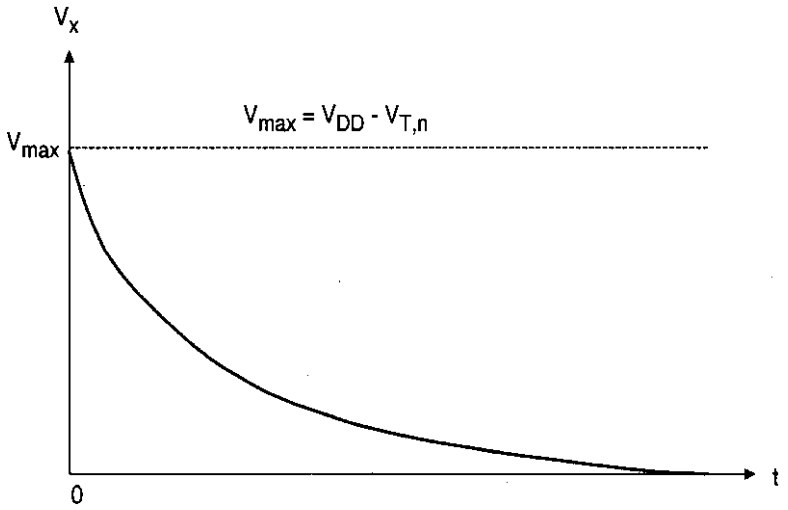
- Source voltage of MP is equal to 0 V during this event; hence, $V_{T,n}=V_{T0,n}$, but the initial condition ($V_{X0}=V_{DD}-V_{T,n}$) contains the threshold voltage with substrate bias effect
- To simplify, we use $V_{T,n}$

$$\int_0^t dt = -\frac{2C_x}{k_n}\int_{V_{DD}-V_{T,n}}^{V_x}\left(\frac{\frac{1}{2(V_{DD}-V_{T,n})}}{2(V_{DD}-V_{T,n})-V_x} + \frac{\frac{1}{2(V_{DD}-V_{T,n})}}{V_x}\right)dV_x$$

Logic "0" Transfer (Cont.)

$$t = \frac{C_x}{k_n(V_{DD} - V_{T,n})}\left[\ln\left(\frac{2(V_{DD} - V_{T,n}) - V_x}{V_x}\right)\right]_{V_{DD}-V_{T,n}}^{V_x}$$

$$= \frac{C_x}{k_n(V_{DD} - V_{T,n})}\ln\left(\frac{2(V_{DD} - V_{T,n}) - V_x}{V_x}\right)$$



Variation of V_x as a function of time during logic "0" transfer

- Unlike the charge-up case, the applied input voltage level (logic 0) can be transferred to the soft node without any modification during this event

Fall Time Calculation

- **Fall time calculation for V_x: $t_{90\%}=0.9V_{max}$, $t_{10\%}=0.1V_{max}$**

$$t_{90\%} = \frac{C_x}{k_n(V_{DD}-V_{T,n})}\ln\left(\frac{(2-0.9)(V_{DD}-V_{T,n})}{0.9(V_{DD}-V_{T,n})}\right) = \frac{C_x}{k_n(V_{DD}-V_{T,n})}\ln\left(\frac{1.1}{0.9}\right)$$

$$t_{10\%} = \frac{C_x}{k_n(V_{DD}-V_{T,n})}\ln\left(\frac{1.9}{0.1}\right)$$

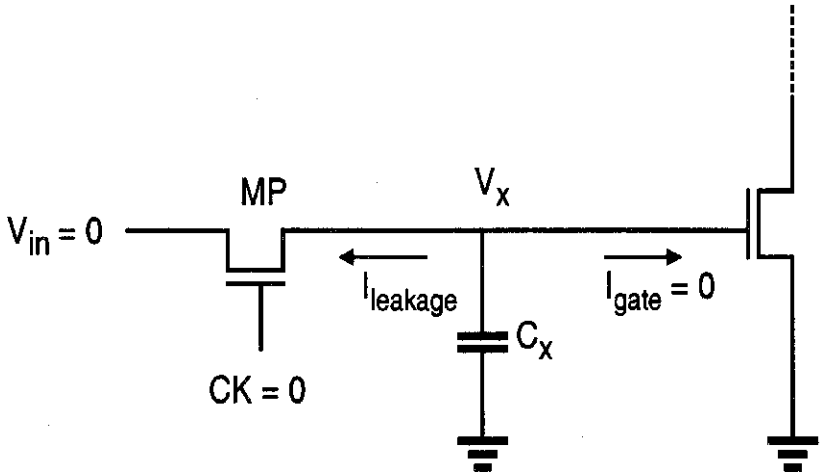
$$\tau_{fall} = t_{10\%} - t_{90\%} = \frac{C_x}{k_n(V_{DD}-V_{T,n})}\cdot\left(\ln(19)-\ln(1.22)\right)$$

$$\boxed{\tau_{fall} = 2.74\frac{C_x}{k_n(V_{DD}-V_{T,n})}}$$

- **So far we focused on logic transfer (CLK=1), now we turn our attention to the storage of logic levels at soft node X**

Charge Storage and Charge Leakage

- Consider the scenario shown below
- Assume that a logic-high voltage level has been transferred to the soft node during CK=1 and now V_{in}=CK=0V
- The charge stored in C_x will gradually leak away
  - $I_{leakage} = I_{subthreshold}(MP) + I_{reverse}(MP)$

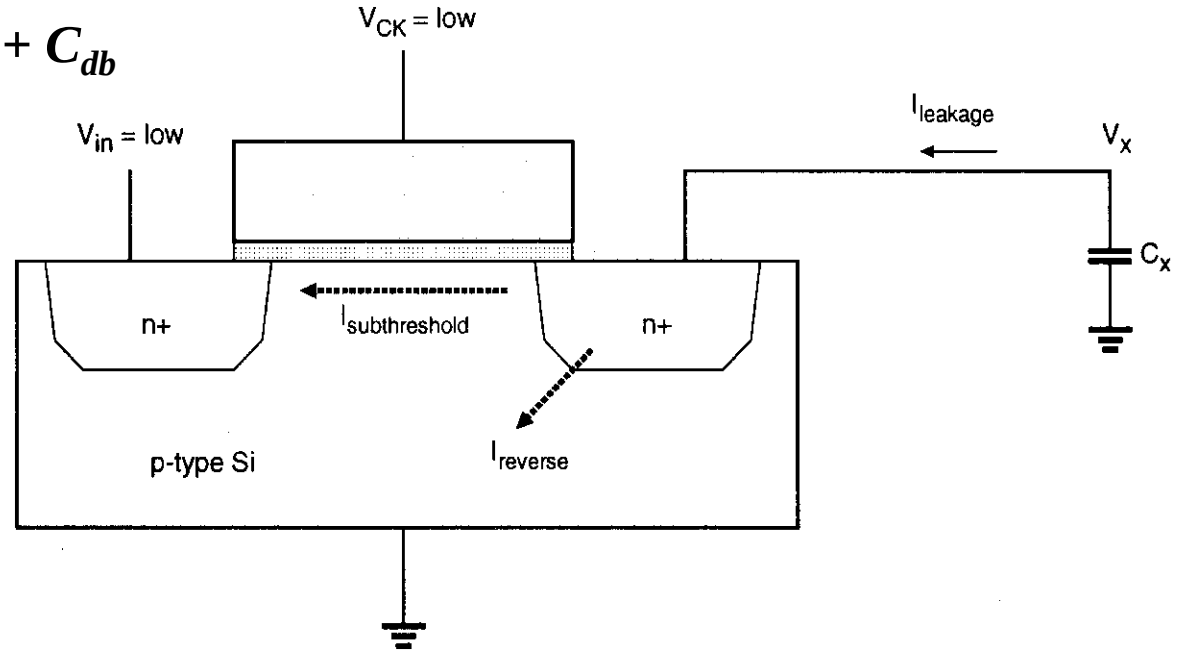


Charge leakage from the soft node

Charge Storage and Charge Leakage (Cont.)

- A portion of C_x is due to the reverse biased drain-substrate junction, which is also a function of V_x
- Other components of C_x, which are primarily due to oxide-related parasitics, can be considered constant

$$C_x = C_{gb} + C_{poly} + C_{metal} + C_{db}$$
$$= C_{in} + C_{db}$$



Simplified cross-section of the NMOS pass transistor, showing the leakage current components responsible for draining the soft-node capacitance C_x

Charge Storage and Charge Leakage (Cont.)

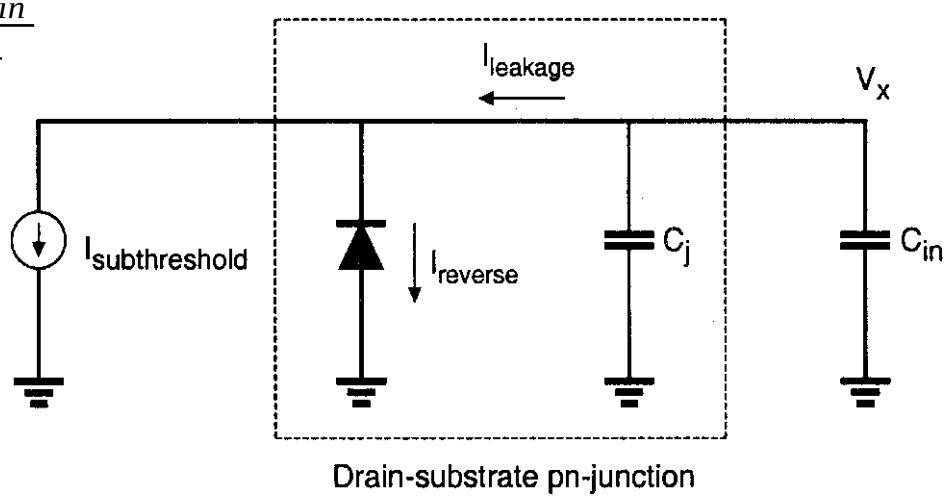
- C_{in} denotes the contact components of C_x, which are primarily due to oxide-related parasitics:

$$C_x = C_{in} + C_{db}$$

$$Q = Q_j(V_x) + Q_{in} \quad \text{where} \quad Q_{in} = C_{in} \cdot V_x$$

$$C_{in} = C_{gb} + C_{poly} + C_{metal}$$

$$I_{leakage} = \frac{dQ}{dt} = \frac{dQ_j(V_x)}{dt} + \frac{dQ_{in}}{dt}$$



Equivalent circuit used for analyzing the charge leakage process

Charge Storage and Charge Leakage (Cont.)

$$C_x = C_{in} + C_{db}$$

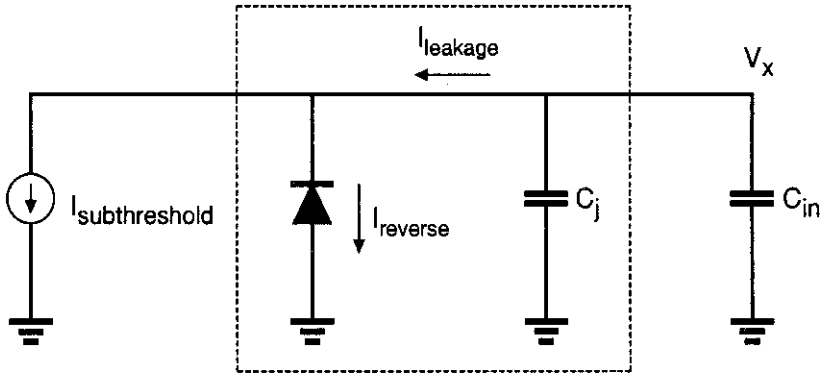
where

$$I_{leakage} = \frac{dQ_j(V_x)}{dV_x}\frac{dV_x}{dt} + C_{in}\frac{dV_x}{dt}$$

$$\frac{dQ_j(V_x)}{dV_x} = C_j(V_x) = \frac{A \cdot C_{j0}}{\sqrt{1+\frac{V_x}{\phi_0}}} = A \cdot \sqrt{\frac{q\varepsilon_{Si}N_A}{2(\phi_0 + V_x)}}$$

$$\phi_0 = \frac{kT}{q}\ln\left(\frac{N_D \cdot N_A}{n_i^2}\right)$$

$$C_{j0} = \sqrt{\frac{q\varepsilon_{Si}N_A N_D}{2(N_A + N_D)\phi_0}} \approx \sqrt{\frac{q\varepsilon_{Si}N_A}{2\phi_0}}$$



Charge Storage and Charge Leakage (Cont.)

- Assume the minimum combined soft-node capacitance is given as $C_{x,min} = C_{gb} + C_{poly} + C_{metal} + C_{db,min}$

  where $C_{db,min}$ represents the minimum junction capacitance, obtained under the bias condition $V_x = V_{max}$

- t_{hold}: We define it as the shortest time required for the soft-node voltage to drop from its initial logic-high value to the logic (switching) threshold voltage, [$V_M = V_{th} = V_{DD}/2$] due to leakage

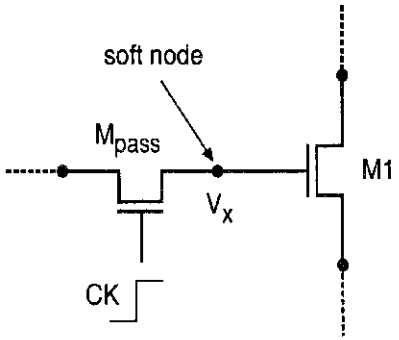
where

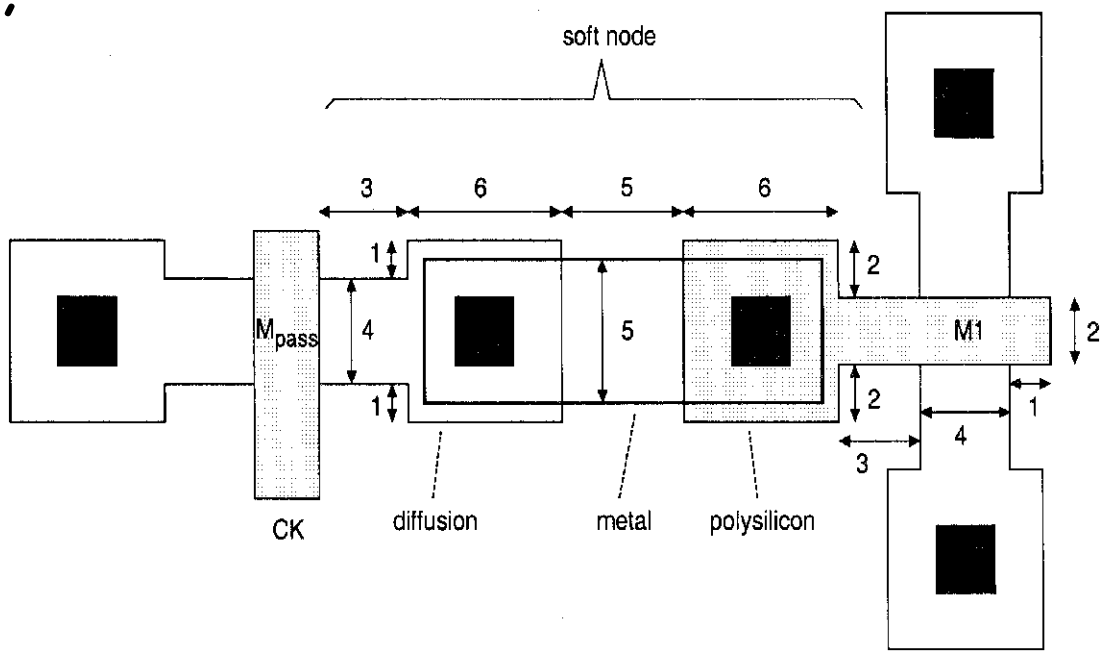
$$t_{hold} = \frac{\Delta Q_{critical,\min}}{I_{leakage,\max}}$$

$$\Delta Q_{critical,\min} = C_{x,\min}\left(V_{\max} - \frac{V_{DD}}{2}\right)$$

Example 9.2

- Consider the soft-node structure:
- V_{DD}=5 V, and the soft node initially charged up to V_{max}
  - V_{T0}=0.8 V, γ=0.4 $V^{1/2}$, $|2\Phi_F|$=0.6 V, C_{ox}=0.065fF/μm²
  - C'_{metal}=0.036 fF/μm² , C'_{poly}=0.055 fF/μm², C_{j0}=0.095 fF/μm² , C_{j0sw}=0.2 fF/μm
- Estimate the worst-case holding time





Example 9.2 (Cont.)

- $C_{gb}=C_{ox}.W.L_{mask}= = 0.065\ fF/\mu m^2 \times (8\ \mu m^2) = 0.52\ fF$ [for M1]
- $C_{metal} = 0.036\ fF/\mu m^2 \times (25\ \mu m^2) = 0.90\ fF$
- $C_{poly} = 0.055\ fF/\mu m^2 \times (6 \times 6 \mu m^2 + (3+1) \times 2\ \mu m^2) = 2.42\ fF$
- $C_{db,max} = A_{bottom} \cdot C_{j0} + P_{sidewall} \cdot C_{j0sw} = 10.56\ fF$

$$V_{\max} = 5.0 - 0.8 - 0.4\left(\sqrt{0.6 + V_{\max}} - \sqrt{0.6}\right) = 3.86V$$

$$C_{db,\min} = \frac{C_{bottom}}{\sqrt{1 + \dfrac{V_{x,\max}}{\phi_0}}} + \frac{C_{sidewall}}{\sqrt{1 + \dfrac{V_{x,\max}}{\phi_{0sw}}}} = 4.71\,fF$$

$$C_{x,min} = C_{gb} + C_{poly} + C_{metal} + C_{db,min}$$

- **$C_{x,min} = 8.55\ fF$**

Example 9.2 (Cont.)

- The amount of the critical charge drop in the soft node, assuming the logic threshold voltage of the next stage is ($V_{DD}/2$)
  - ($I_{subthreshold}$ can be calculated using equation 3.92, and $I_{reverse}$ using junction diode characteristics) Here we assume it's calculated for us as 0.85pA

$$\Delta Q_{critical} = C_{x,\min} \cdot \left(V_{x,\max} - \frac{V_{DD}}{2} \right) = 10.09\, fC$$

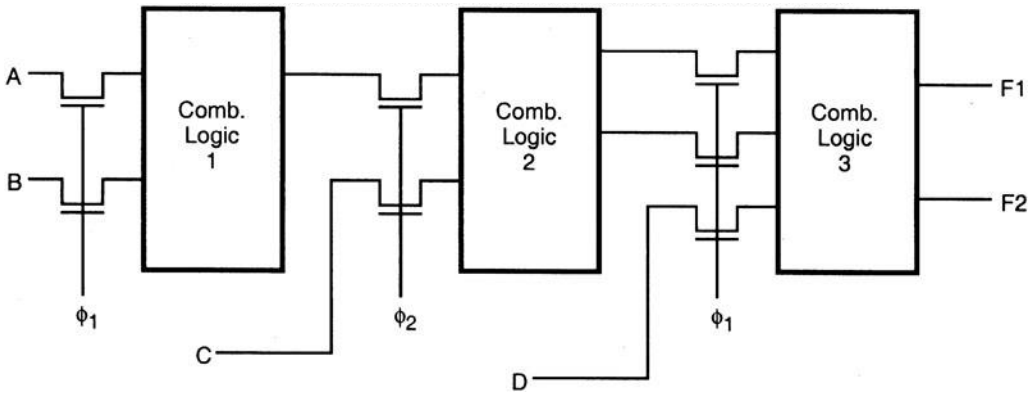
$$I_{leakage} = I_{subthreshold} + I_{reverse} = 0.85\, pA$$

$$t_{hold,\min} = \frac{\Delta Q_{critical}}{I_{leakage,\max}} = \frac{10.09\, fC}{0.85\, pA} = 11.87 ms$$

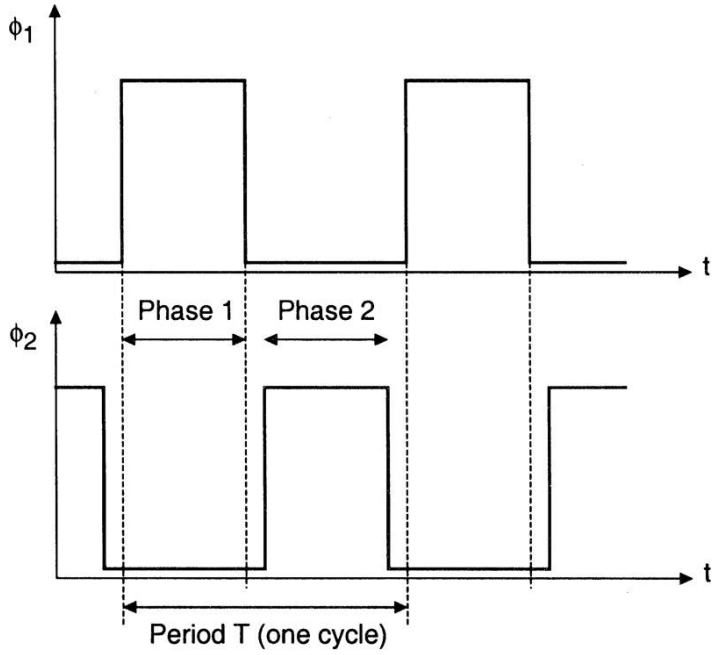
- Note than even with a small soft-node cap (8.55fF) the worst-case holding time is relatively long, so this proves the feasibility of the dynamic charge storage concept

Dynamic Pass Transistor Circuits

- Consider the generalized view of a multi-stage synchronous circuit shown below
- To drive the pass transistors, two non-overlapping clock signals are used (known as *two-phase clocking*)



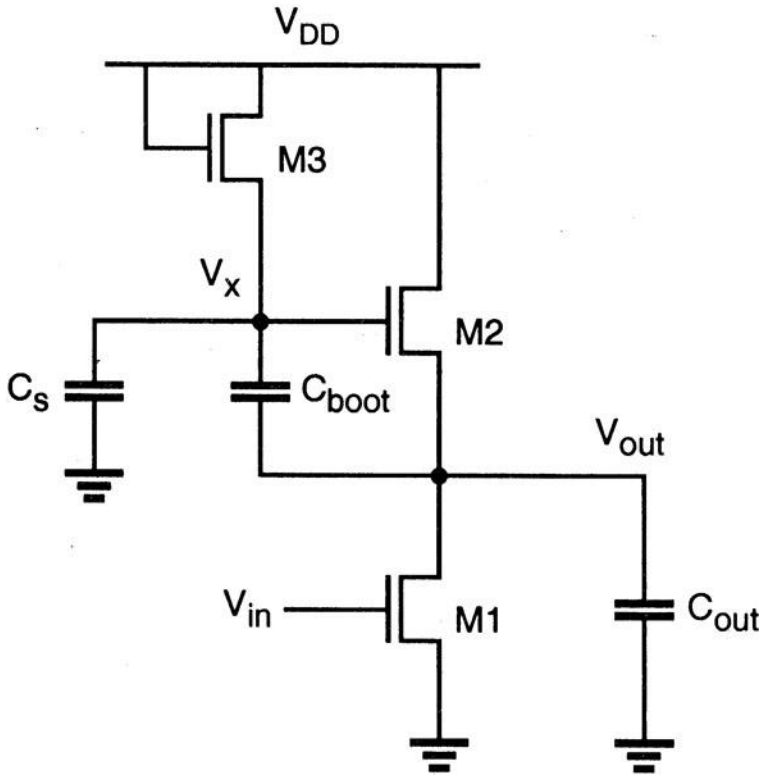
Multi-stage pass transistor logic driven by two nonoverlapping clocks



Nonoverlapping clock signals used for two-phase synchronous operation

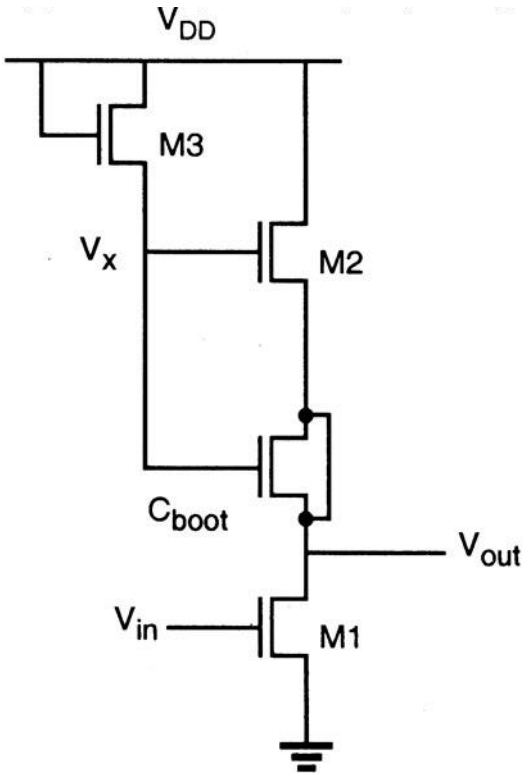
[Optional] Voltage Bootstrapping

- $V_x = V_{DD} - V_{T3}$ instead of V_{DD}



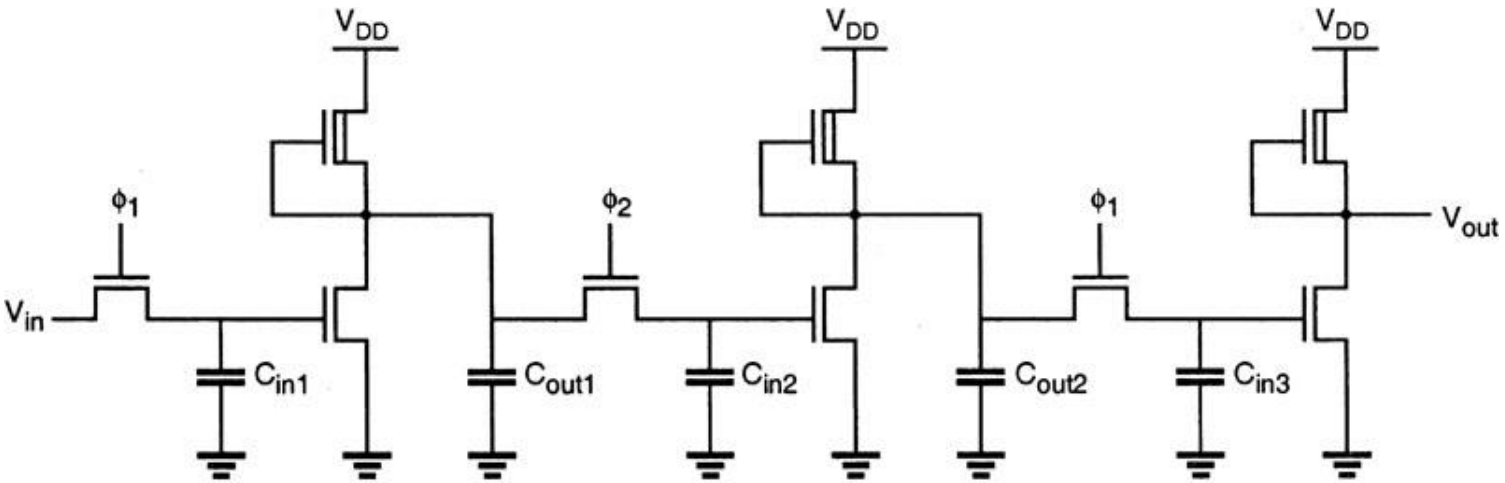
[Optional] Voltage Bootstrapping (Cont.)

- $V_{out} = V_{DD} - V_{T2}$ instead of V_{DD}



Depletion Load Dynamic Shift Register

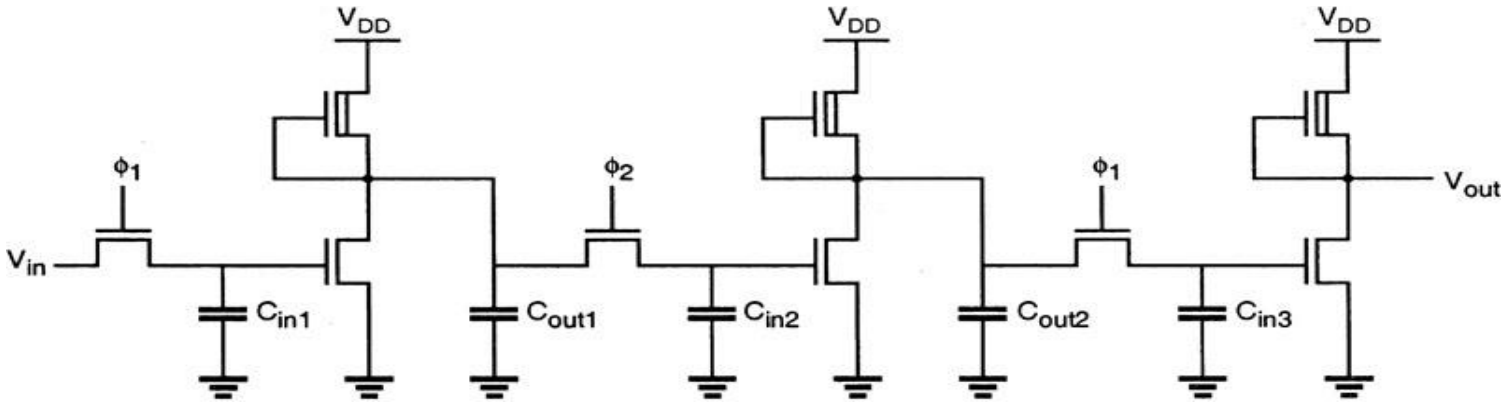
- During the active phase of φ_1, V_{in} is transferred into C_{in1}. Thus, the valid output voltage level of the first stage is determined as the inverse of the present input during this cycle
- When φ_2 becomes active during the next phase, the output voltage level of the first stage is transferred into the second stage input capacitance C_{in2} , and the valid output level of the second stage is determined



Three stages of a depletion-load nMOS dynamic shift register circuit driven with two-phase clocking

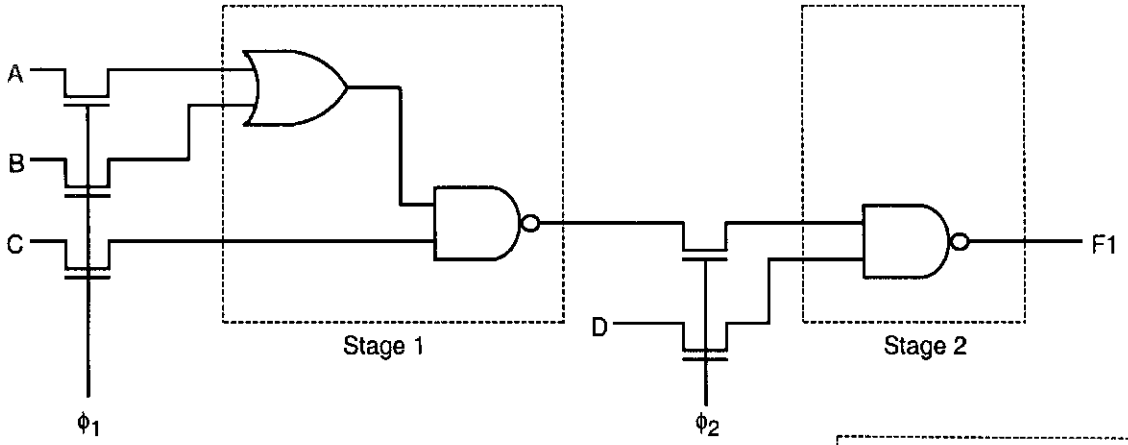
Depletion Load Dynamic Shift Register (Cont.)

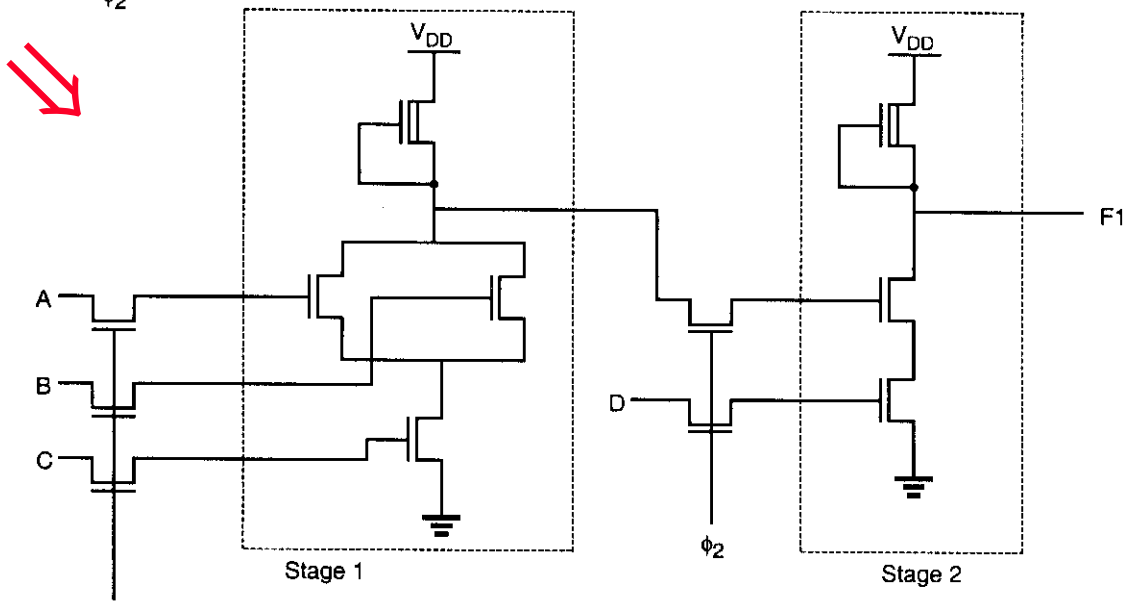
- During the active φ_2 phase, the first-stage input cap , C_{in1}, continues to retain its previous level via charge storage
- When φ_1 becomes active again, the original data bit written into register during previous cycle is transferred into the 3rd stage and the first stage can also accept the next data bit
- One half period of the clock must be long enough to allow input cap C_{in} to charge up or down and the logic level to propagate to the output by charging C_{out}
- Note that logic-high input of each inverter is one NMOS threshold voltage lower than V_{DD}



A Dynamic Two-Stage Circuit

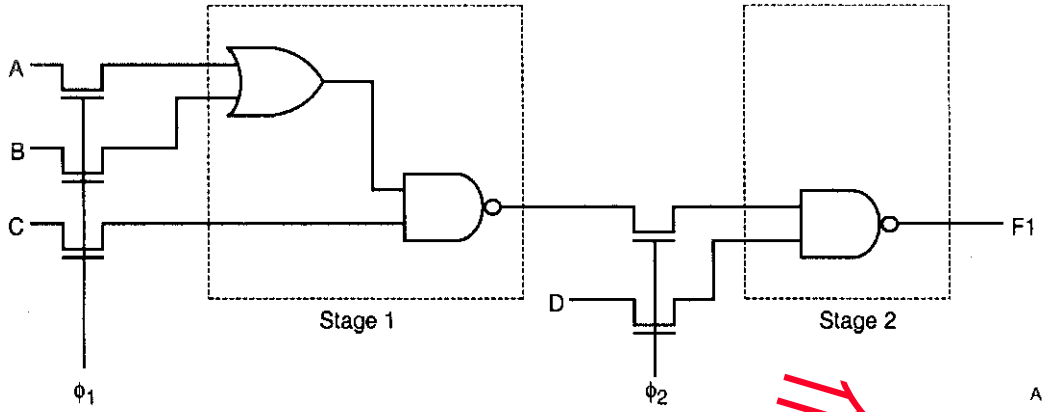
- Same operation principle used in the shift register is used for synchronous depletion NMOS complex logic1

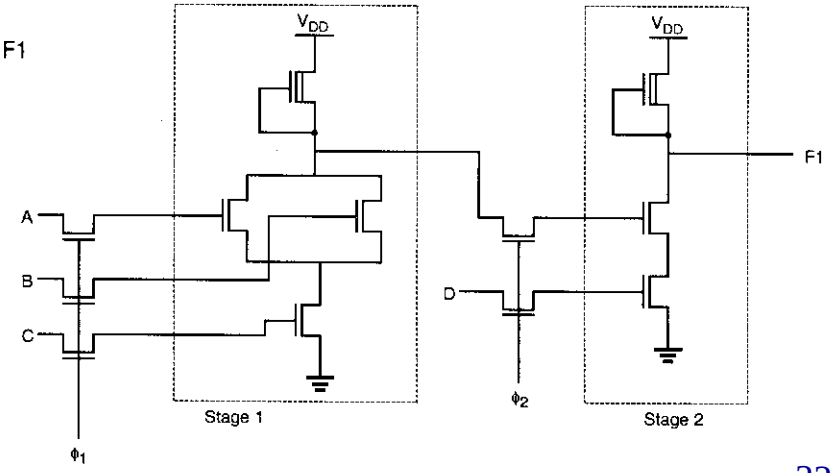




A Dynamic Two-Stage Circuit (Cont.)

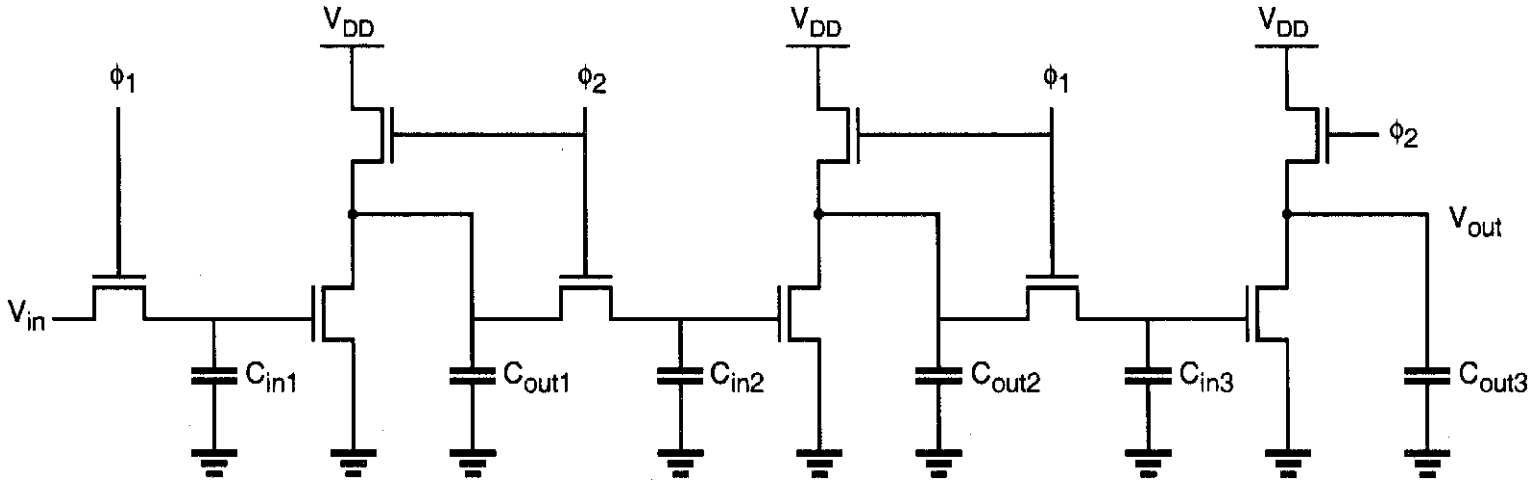
- Signal propagation delay of each logic stage may be different, therefore to guarantee the correct logic levels are propagated during each active clock cycle, the half-period length of the clock signal must be longer than the largest single stage propagation delay





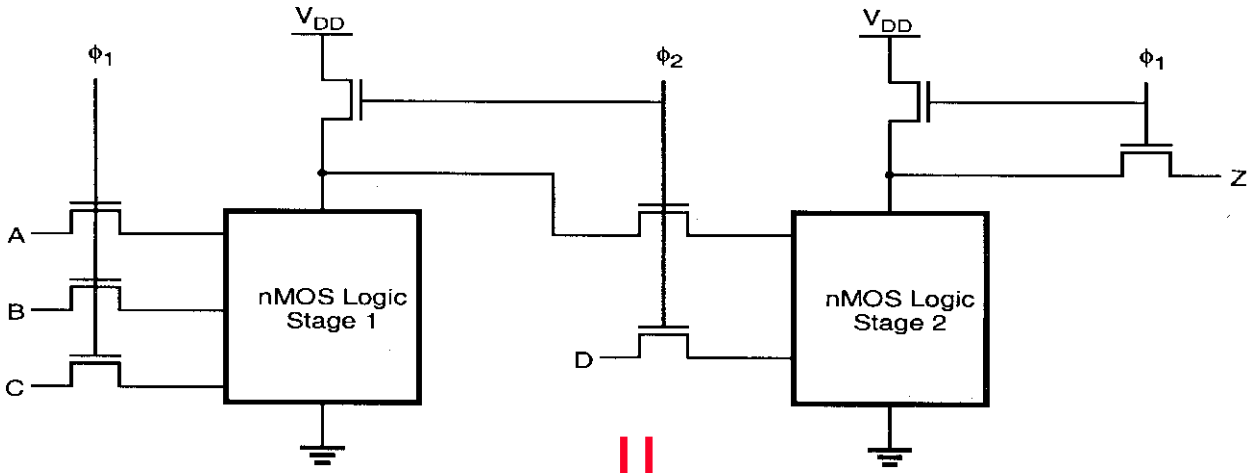
Dynamic Shift Register (Ratioed Logic)

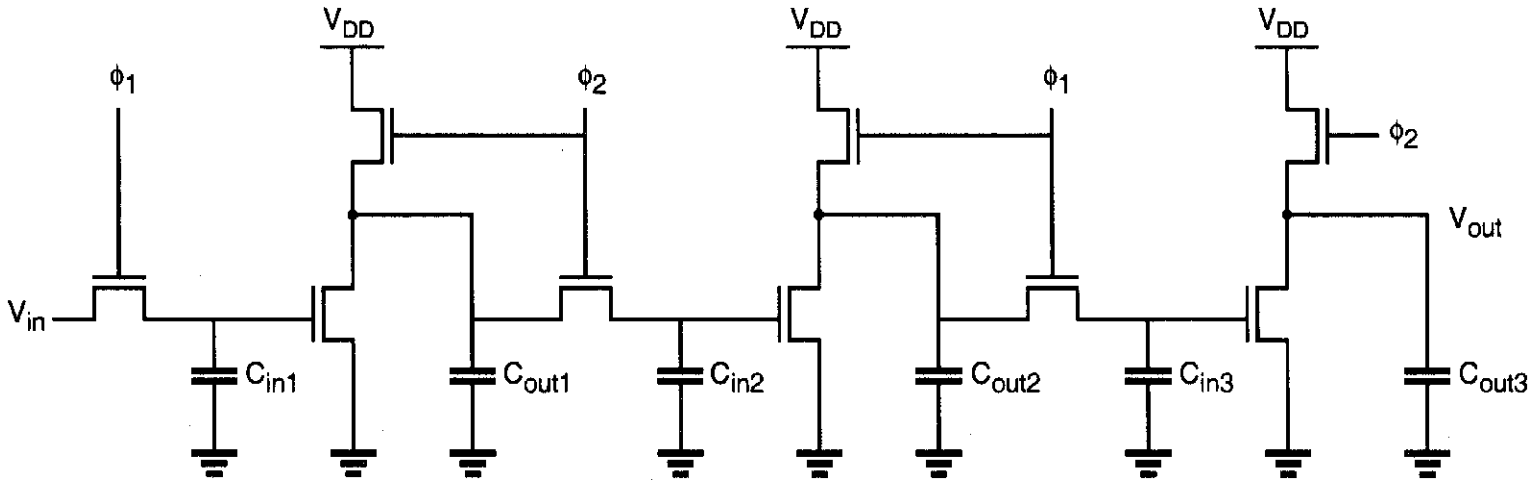
- Different implementation of shift register circuit, using enhancement-load NMOS inverters
  - Instead of biasing the load transistors with a constant gate voltage, apply the clock signal to the gate of the load transistor as well
  - Reduction in power dissipation and the silicon area compared to the depletion load NMOS logic



Ratioed Dynamic Logic (Cont.)

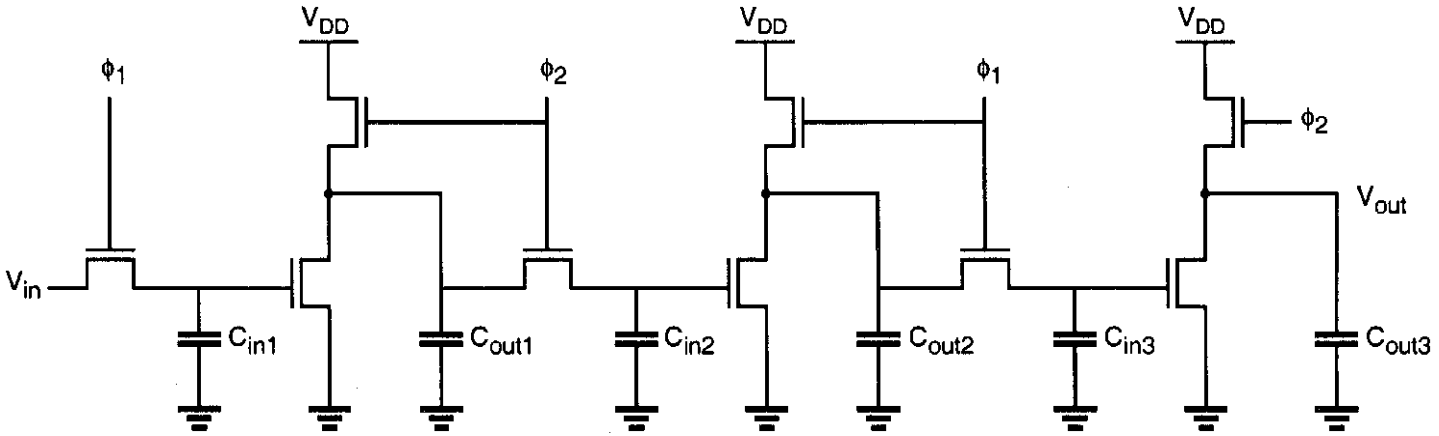
General circuit structure of ratioed synchronous dynamic logic





Ratioed Dynamic Logic (Cont.)

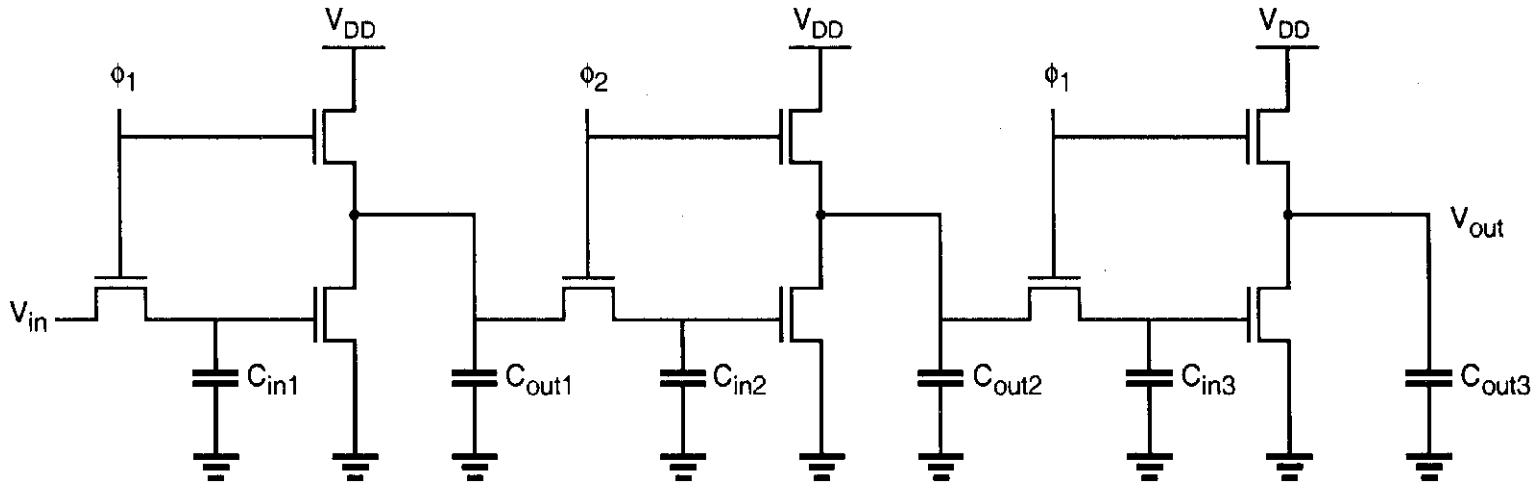
- For both the shift-register and complex logic:
  - When φ_1 is active, V_{in} is transferred to C_{in1}, enhancement-type NMOS load transistor of first stage is not active yet
  - When φ_2 is active, load transistor is turned on, pass transistor of the second stage is turned on –> output level transferred into C_{in2}
  - When φ_1 is on again, the valid output level across C_{out2} is determined and transferred into C_{in3}
- In this circuit V_{OL} of each stage is determined by the driver-to-load ratio. Therefore, this circuit is called ratioed dynamic logic



General circuit structure of ratioed synchronous dynamic logic

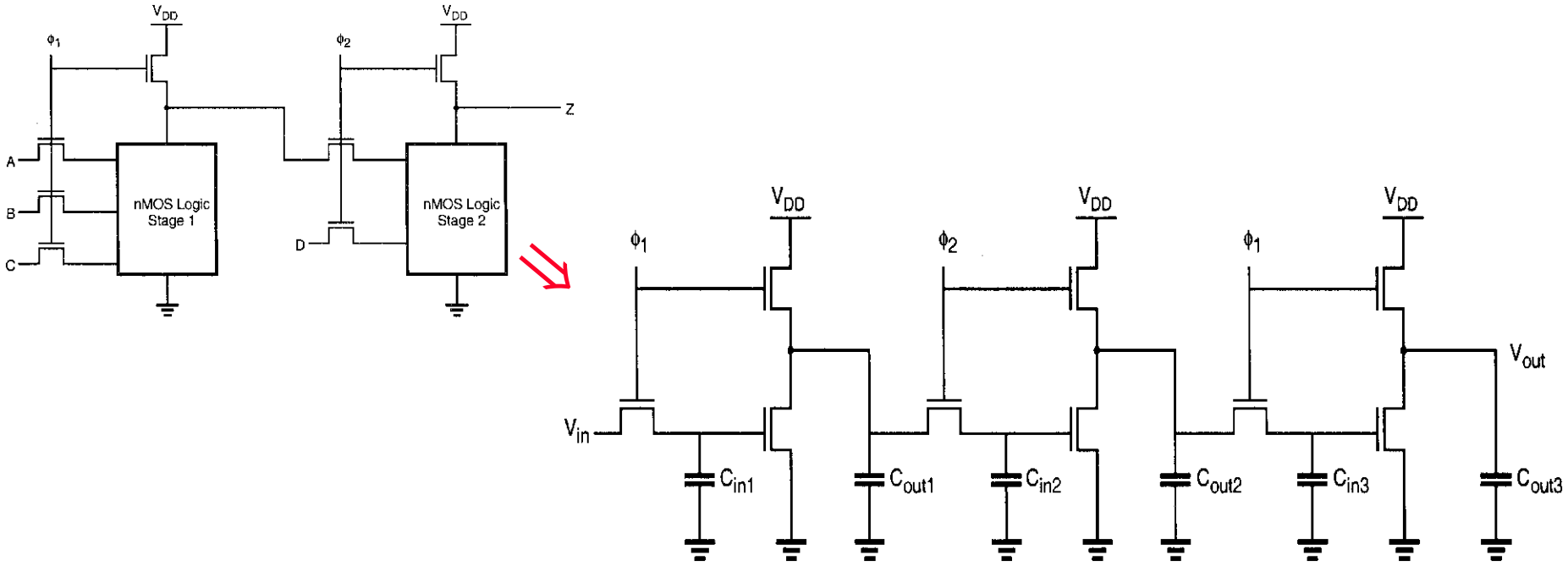
Ratioless Dynamic Logic

- Consider following enhancement-load shift register implementation where, in each stage, the input pass transistor and the load transistor are driven by the same clock phase
- When φ_1 is active, V_{in} is transferred into C_{in1}, also the enhancement-type NMOS load transistor of the first stage is active, so 1st inverter's output attains its valid output logic level
- When φ_2 is active, the input pass transistor of the next stage is turned on and the logic level is transferred to the next stage



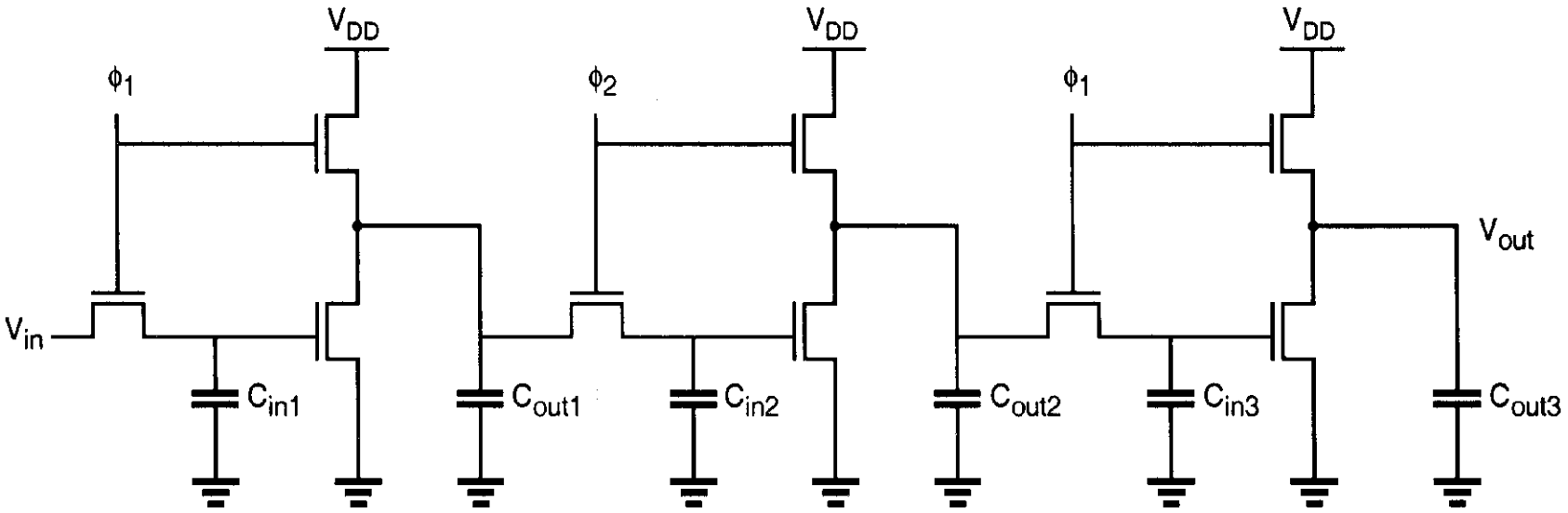
Ratioless Dynamic Logic (Cont.)

- If the output level across C_{out1} is logic-high at the end of the active φ_1 phase, this voltage level is transferred to C_{in2} via charge sharing over the pass transistor during the active φ_2 phase
- Note: logic high at output is one threshold voltage lower than V_{DD}
- To correctly transfer logic high level after charge sharing, the ratio of C_{out} /C_{in} must be made large during circuit design



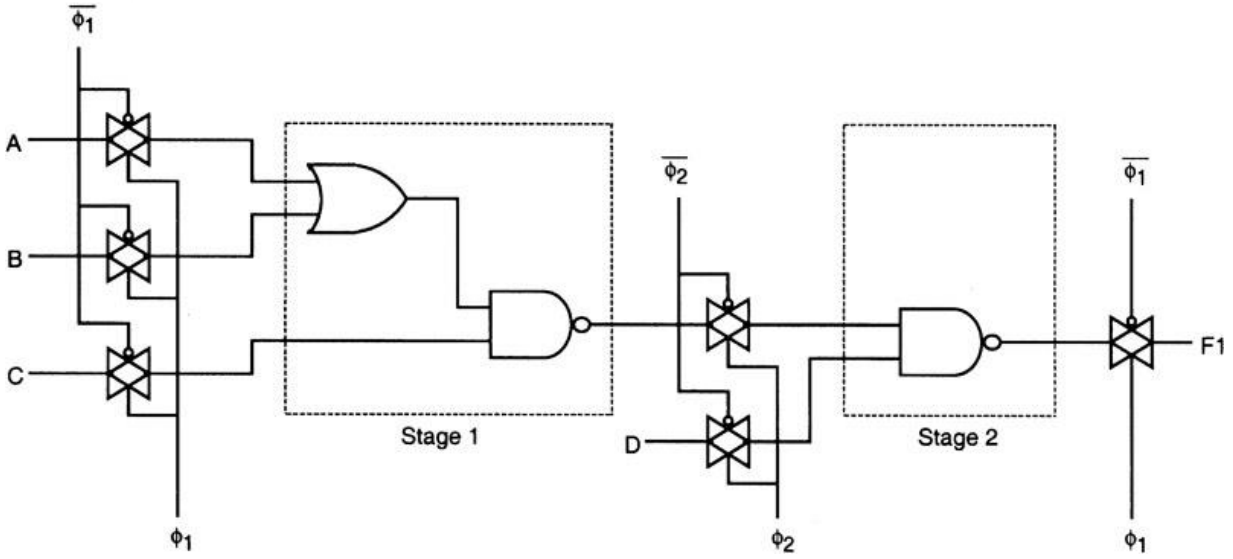
Ratioless Dynamic Logic (Cont.)

- If the output level of the first stage is logic-low at the end of the active φ_1 phase, then C_{out1} will be completely drained to a voltage of V_{OL}=0 V when φ_1 turns off
- Since valid logic-low level V_{OL}=0 can be achieved regardless of the driver-to-load ratio, this circuit arrangement is called ratioless dynamic logic



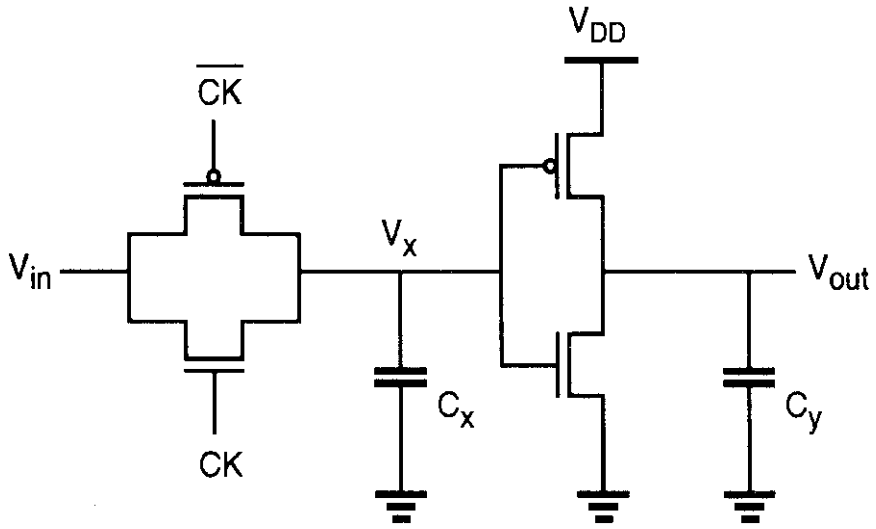
Dynamic CMOS Transmission Gate (TG) Logic

- The basic two-phase synchronous logic circuit principle (in which in which individual logic blocks are cascaded via clock-controlled switches) can also be adopted to CMOS structures
- Static CMOS gates are used for implementing the logic blocks, and CMOS TGs are used for transferring the output levels of one stage to the inputs of the next stage (as switches)
- Each TG is controlled by the clock signal and its complement (4 clock signals are generated and routed throughout the circuit)



Dynamic CMOS TG Shift Register

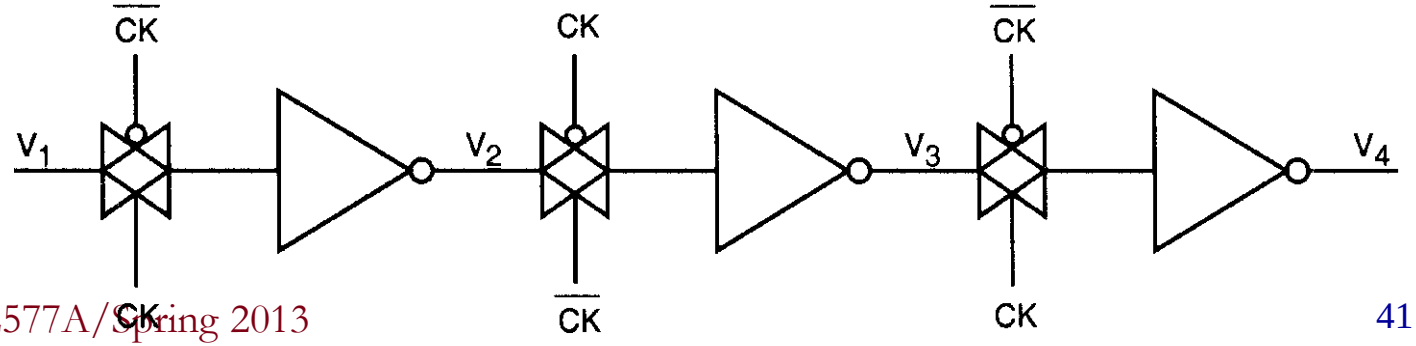
- When CK=1, V_{in} is transferred onto C_x via the transmission gate
- When CK=0, TG turns off and the voltage level across C_x can be preserved until the next cycle
- TG advantages:
  1. Smaller transfer time compared to those for NMOS-only switches , because of TG's low on-resistance
  2. No threshold voltage drop across the CMOS TG



Basic building block of a CMOS TG dynamic shift register

TG Dynamic Shift Register (Cont.)

- Following figure shows a single-phase CMOS shift register, which is built by cascading identical units of previous figure
- TGs of the odd-stages would conduct when CK=1, while the transmission gates of the even-numbered stages are off
- Note: CK and CK' do not constitute a truly nonoverlapping signal pair, since clock voltage has finite rise/fall times, and the skew between them (if CK' is built by inverting CK), therefore a true two phase clocking with two nonoverlapping φ_1 and φ_2 and their complements is preferred over single phase clocking in dynamic CMOS TG logic

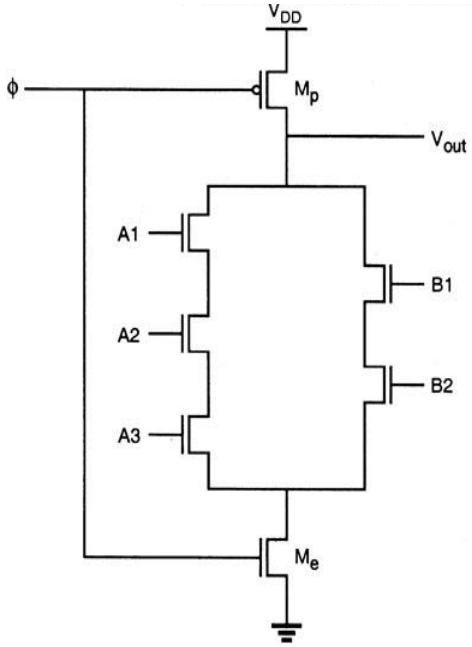


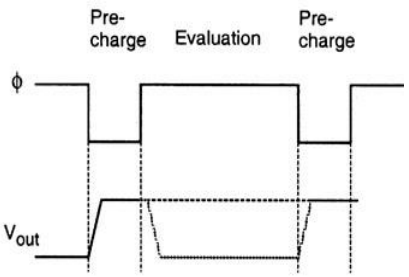
Dynamic CMOS Logic

- This technique significantly reduces the number of transistors used to implement any logic function (compared to CMOS) with zero static power consumption (ignoring leakage)
- The circuit operation is based on first *precharging* the output node capacitance, and subsequently, *evaluating* the output level according to the applied inputs
- Both operations are scheduled by a single clock signal

Dynamic CMOS logic gate implementing the complex Boolean function

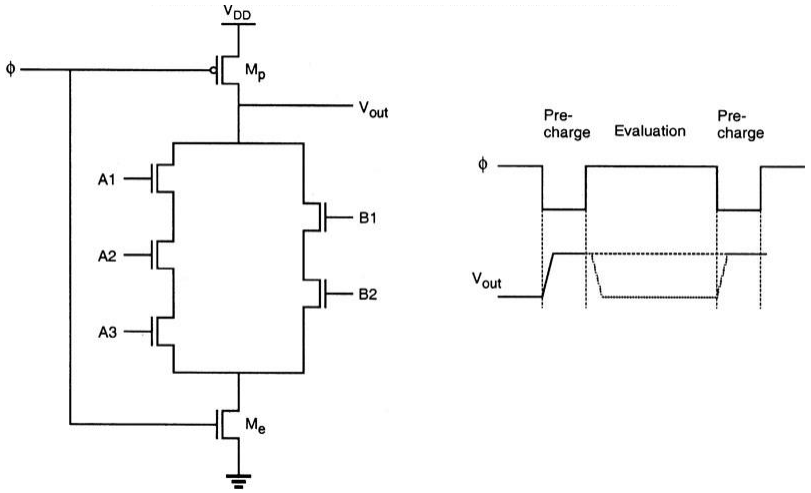
$$F=(A_1A_2A_3+B_1B_2)'$$





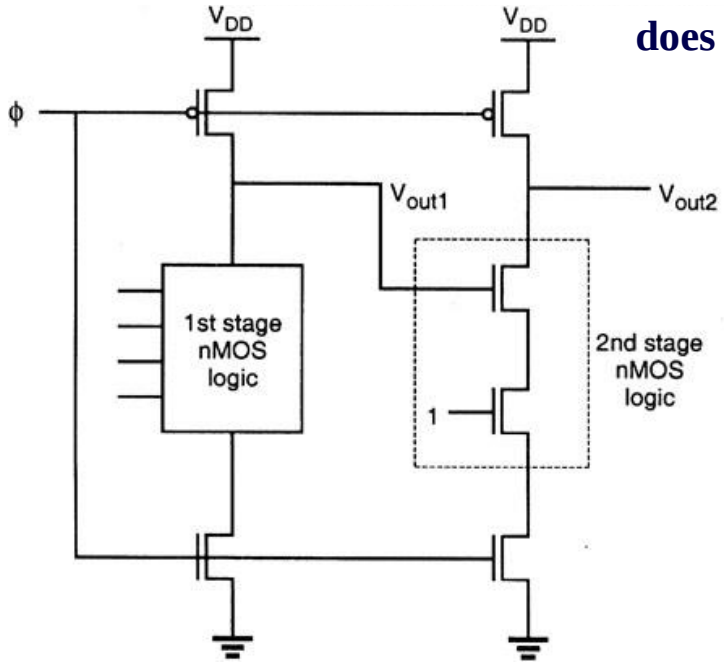
Dynamic CMOS Logic (Cont.)

- When the clock signal is low (precharge phase), M_p is conducting, while M_e is off. The parasitic output capacitance is charged up to $V_{out}=V_{DD}$
- When the clock signal becomes high (evaluate phase), M_p turns off and M_e turns on. The output node voltage now depends on the input voltage levels
- The operation of single-stage dynamic CMOS logic gate is straightforward
- For practical multi-stage applications, however, the dynamic CMOS gate presents a significant problem (shown next)

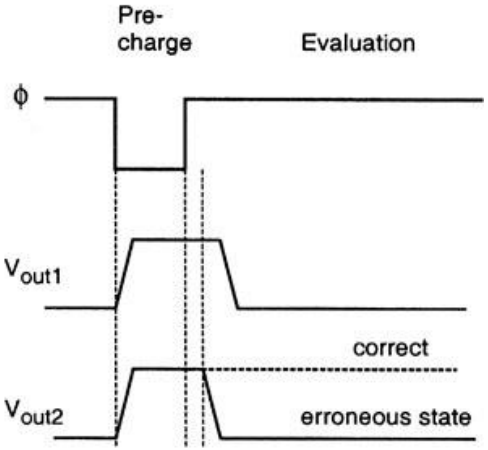


Cascading Dynamic CMOS Logic

- Consider the two-stage cascaded structure shown below
  - The output of the first dynamic CMOS stage drives one of the inputs of the second dynamic CMOS stage, which is assumed to be a two-input NAND gate



We are going to show that this structure does not work



Cascading Dynamic CMOS Logic (Cont.)

- During the precharge phase, V_{out1} and V_{out2} are pulled up. Also the external inputs are applied during this phase
- Let's assume that the input variables of the first stage are such that V_{out1} will drop to logic "0" during the evaluation phase

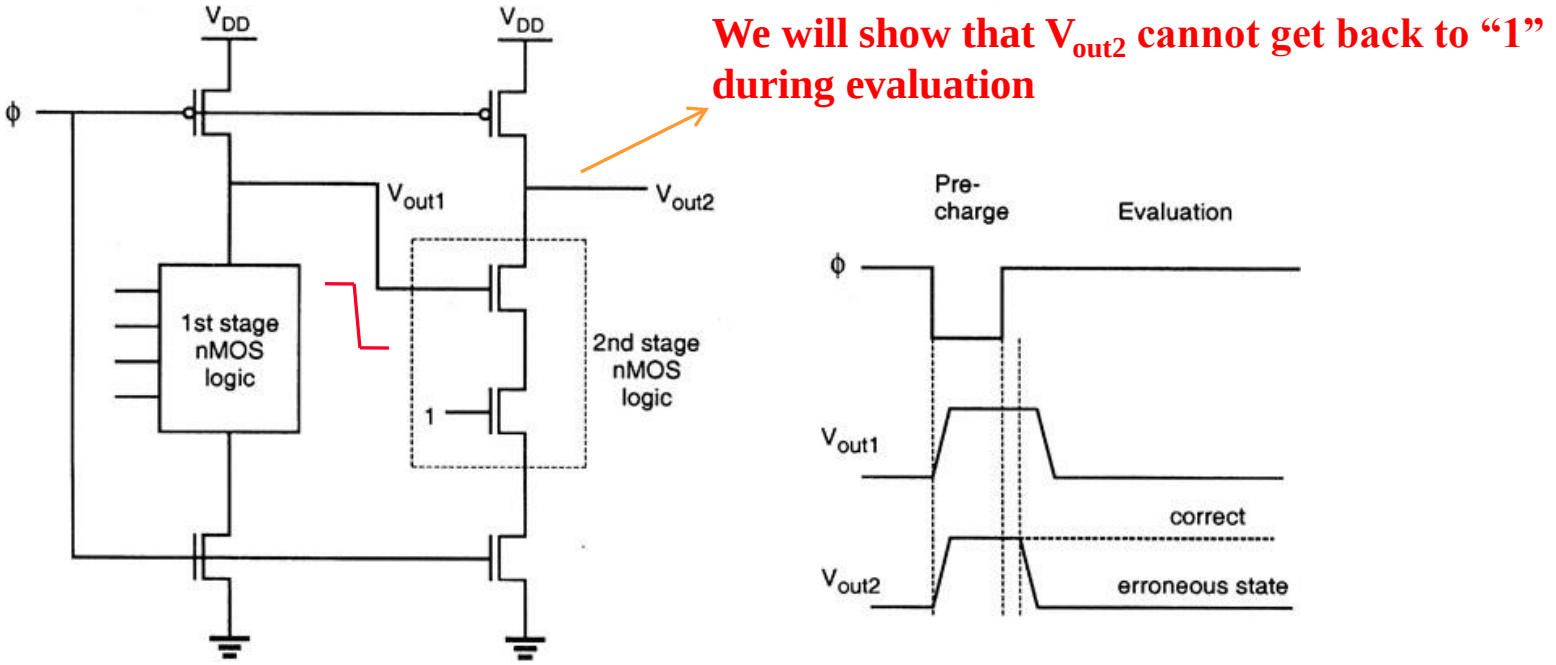


Illustration of the cascading problem in dynamic CMOS logic

Cascading Dynamic CMOS Logic (Cont.)

- When the evaluation phase begins, V_{out1} and V_{out2} are logic-high
- V_{out1} eventually drops to its correct logic level after a certain time delay
- Since the evaluation in the second stage is done concurrently, starting with the high value of V_{out1} at the beginning of the evaluation phase, V_{out2} at the end of the evaluation phase may be erroneously low
- Although V_{out1} finally settles to its correct value, after the stored charge is drained, the correction of V_{out2} is not possible

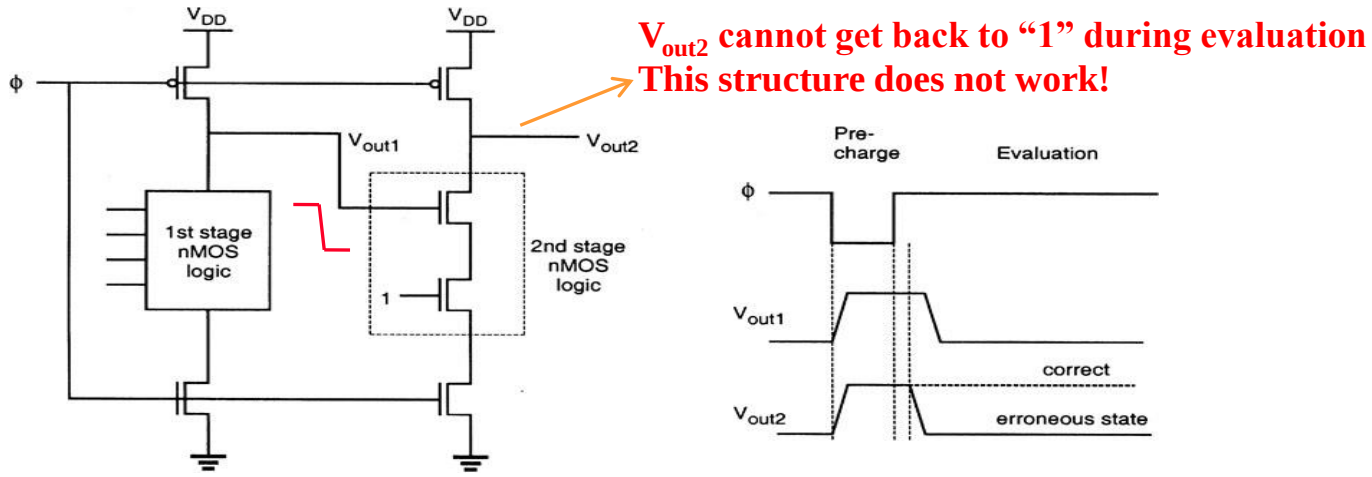
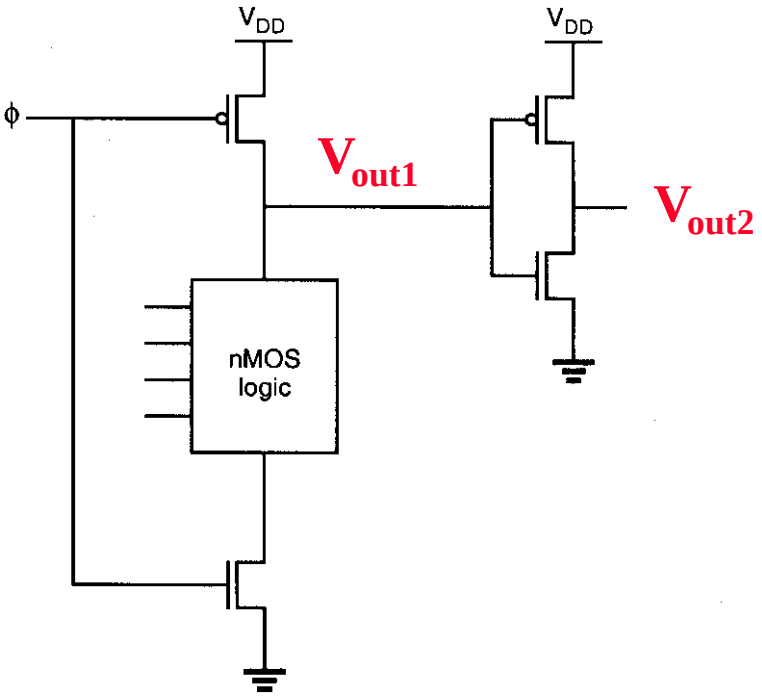


Illustration of the cascading problem in dynamic CMOS logic

Domino CMOS Logic

- A dynamic CMOS logic stage must be cascaded with a static CMOS inverter stage
  - During the precharge phase (CK=0), the output node of the dynamic CMOS stage, V_{out1}, is precharged to a high logic level, and the output of the CMOS inverter, V_{out2}, becomes low
  - During the evaluation phase, V_{out1} is either discharged to a low level or remains high, therefore V_{out2} stays at low or charges to high (0->1 transition)
  - 0 and 0->1 for V_{out2} will not result in error for the next stage

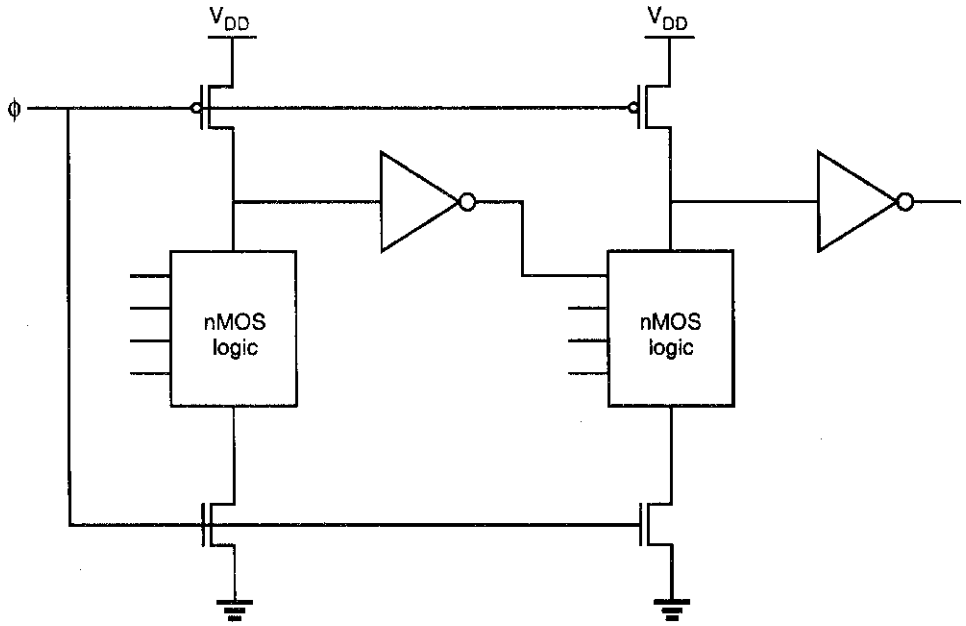


Generalized circuit diagram of a domino CMOS logic gate

Cascading Domino CMOS Logic

- The domino structure solves the problem in cascading conventional dynamic CMOS stages by removing the possibility of 1-to-0 transition at the inputs of the next NMOS logic

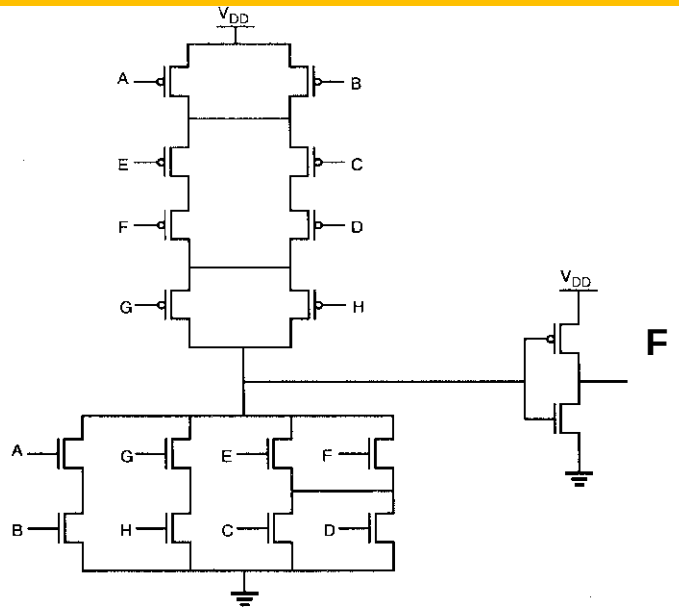
This structure works fine!



Cascaded domino CMOS logic gates

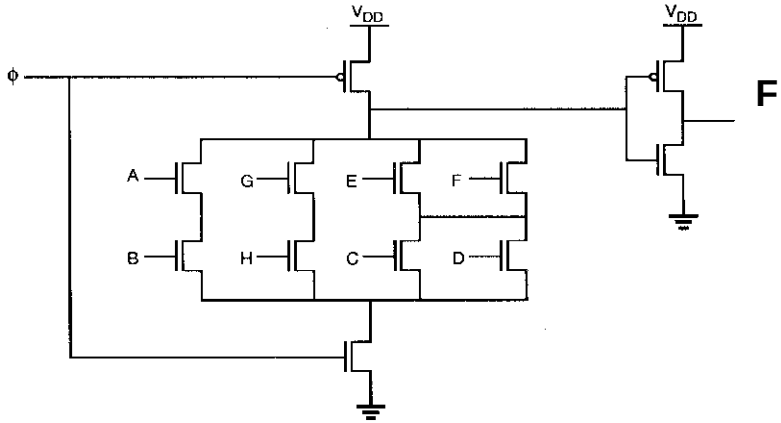
Domino CMOS Logic (Cont.)

- Domino CMOS logic gates allow a significant reduction in the number of the transistors required to realize any complex Boolean function



An 8-input complex logic gate, realized using conventional CMOS logic

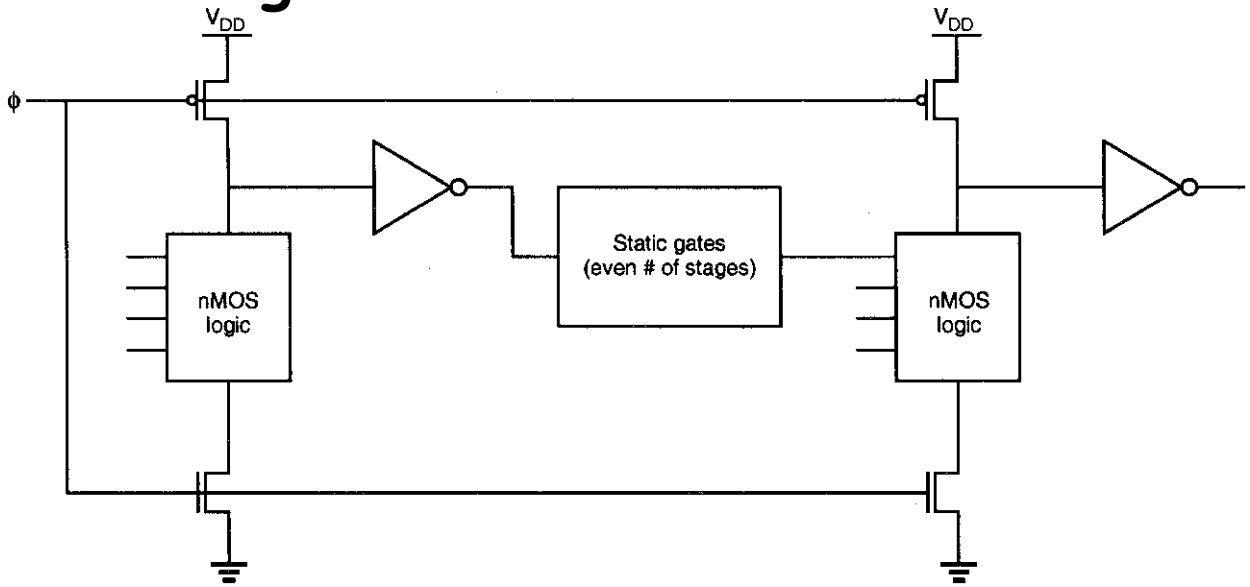
$$F = AB + (C+D)(E+F) + GH$$



Domino CMOS logic

How to Utilize Static CMOS with Domino

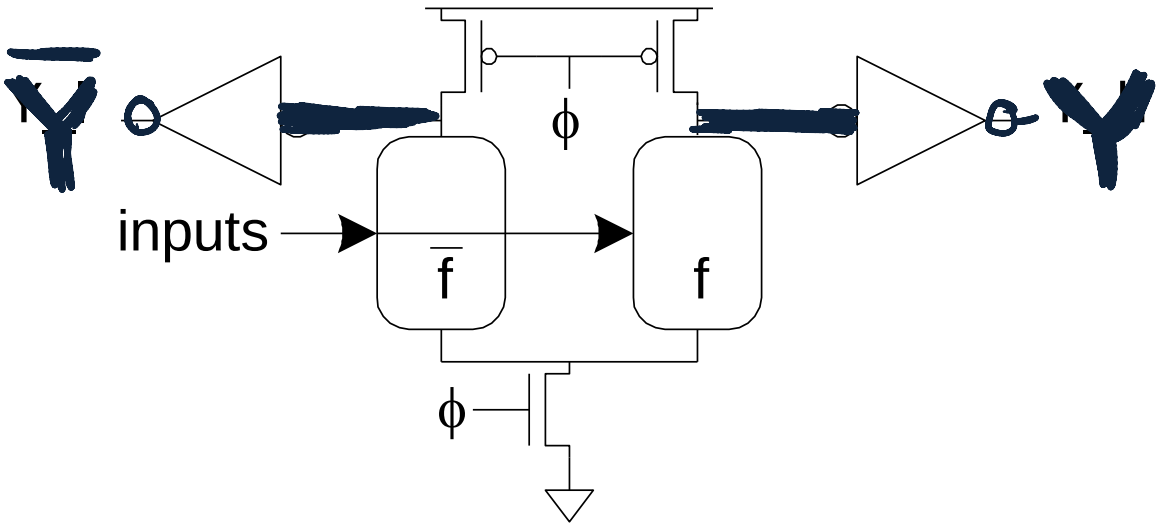
- Conventional static CMOS logic gates can be used together with domino CMOS gates in a cascaded configuration
- Limitation: The number of inverting static logic stages in cascade must be even, so that the inputs of the next domino CMOS stage experience only 0 to 1 transitions during the evaluation



Cascading domino CMOS logic gates with static CMOS logic gates

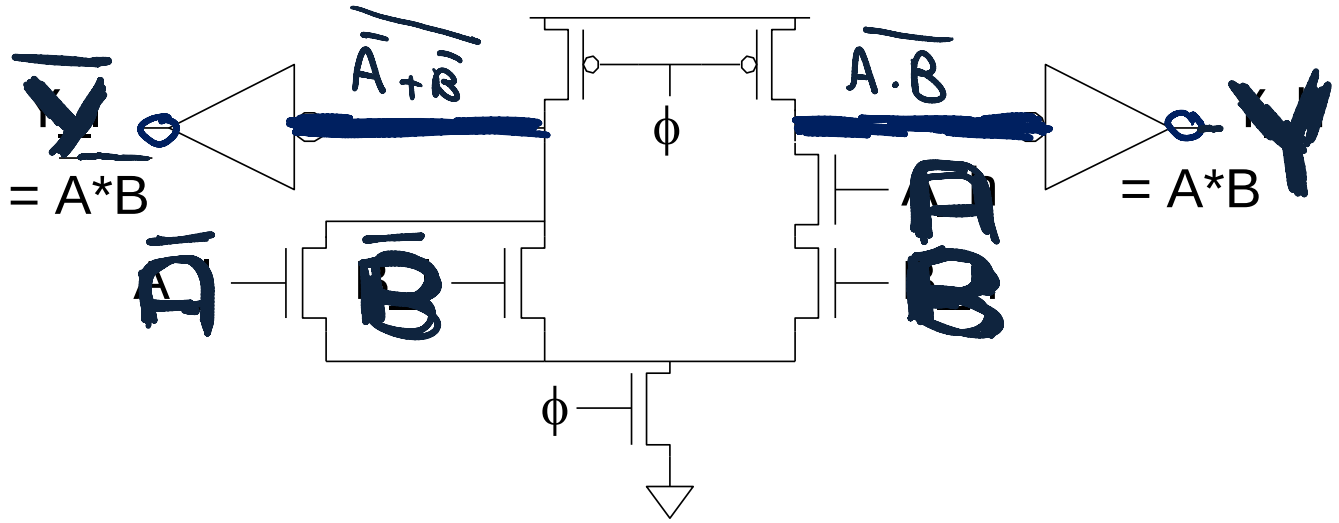
How to Do Inverting Functions in Domino (Dual-Rail Domino)

- Even inversion implies that domino only performs non-inverting 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



Dual-Rail Domino Example: AND/NAND

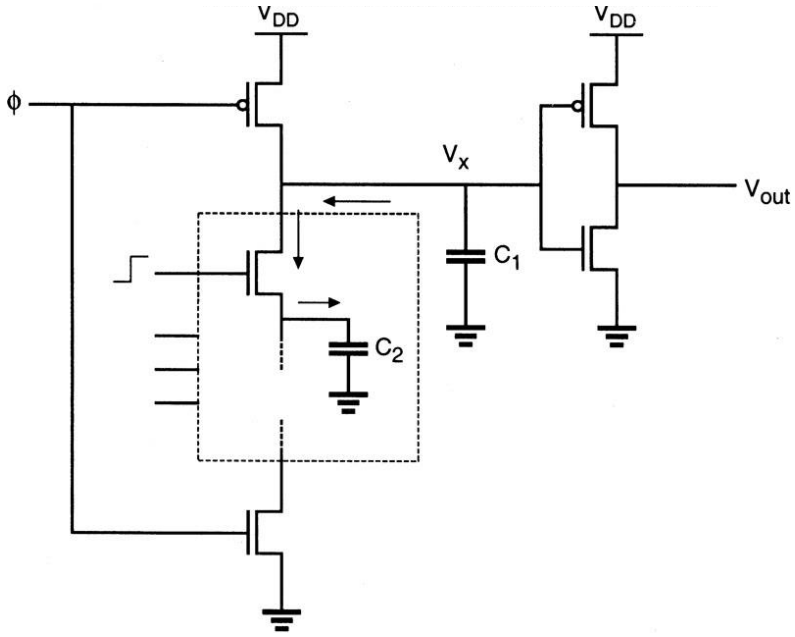
- Given A_h, A_l, B_h, B_l
- Compute Y_h = A . B, Y_l = ~(A . B)
- Pulldown networks are conduction complements



- Disadvantage: If you want to build NOR structure (which is efficiently realizable in Domino), you will also have to build the NAND structure (which is quite inefficient in Domino)

Charge Sharing in Domino

- Consider the following circuit:
  - C_2 is comparable in size to C_1
  - All inputs are initially low and voltage across C_2 is 0 V
  - During precharge phase, C_1 is charged up to V_{DD}
  - During evaluation phase, if the input signal of the uppermost NMOS transistor switches from low to high, the charge initially stored in C_1 will now be shared by C_2, leading to *charge-sharing* phenomenon

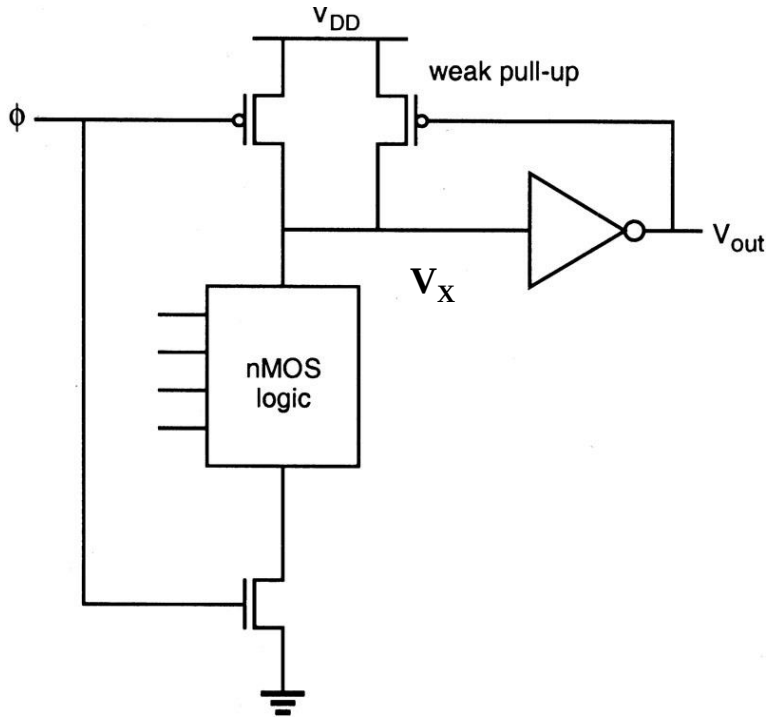


Charge sharing between C_1 and C_2 during the evaluation cycle may reduce the output voltage level

- Output node voltage after charge sharing: $\frac{V_{DD}}{1+\frac{C_2}{C_1}}$

Preventing the Charge Sharing Problem

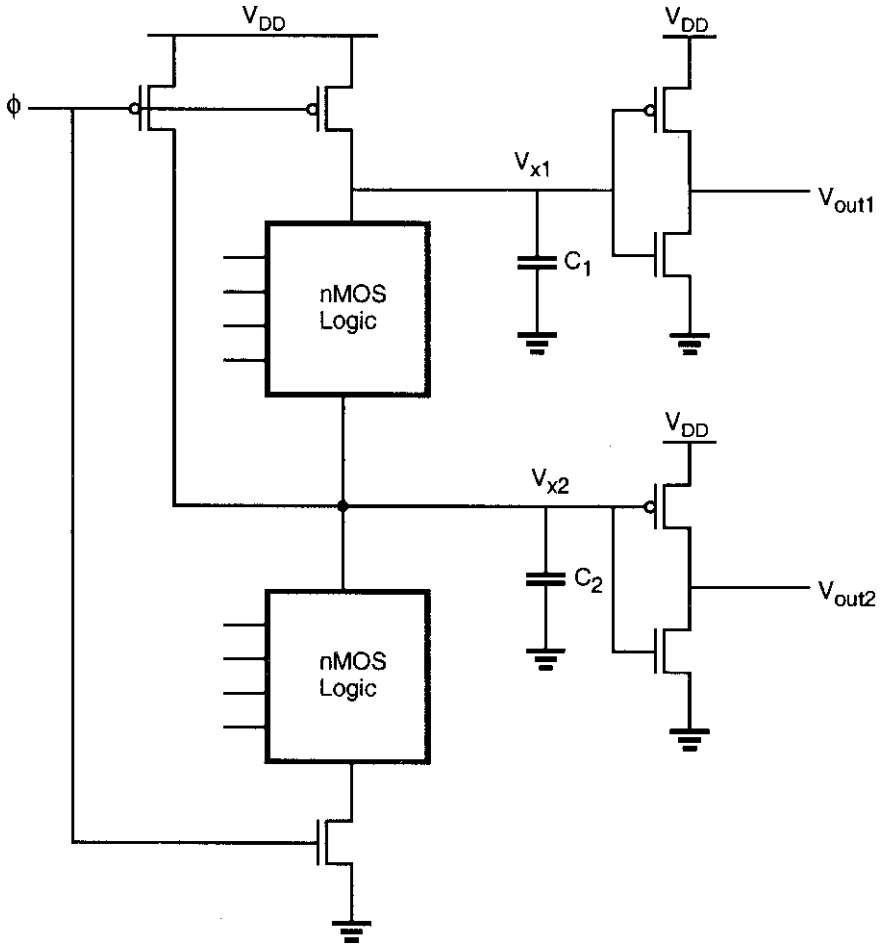
- Due to charge sharing $V_X = V_{DD}/2$ if $C_1 = C_2$ which may result in erroneous V_{out} (if $V_M > V_{DD}/2$)
- To prevent erroneous output levels due to charge sharing in domino CMOS gates:
  - One may add a weak pMOS pull-up device (small W/L) to the dynamic CMOS stage output, which essentially forces a high V_X unless there is a strong pull-down path between V_X and the ground
- Note: pMOS is off when V_{out} is high
- This also helps avoid value loss due to leakage currents



A weak PMOS pull-up device in a feedback loop can be used to prevent the loss of output voltage level due to charge sharing

Preventing the Charge Sharing Problem (Cont.)

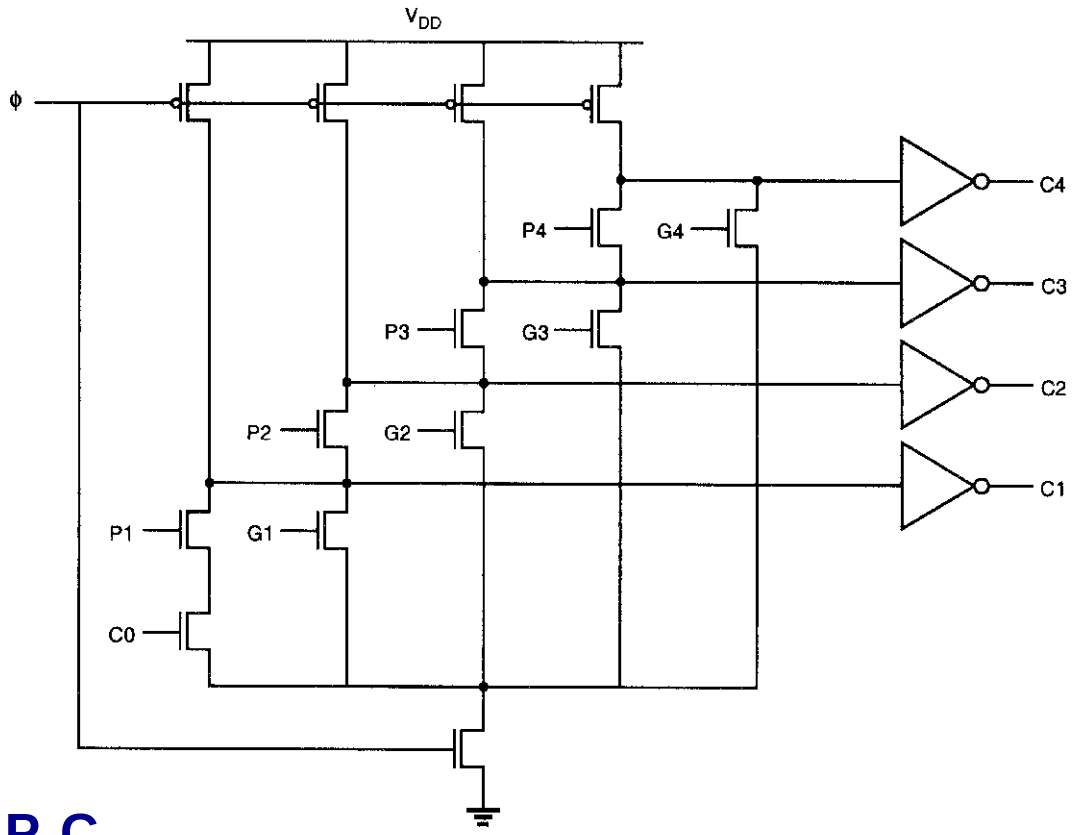
- Another solution is to use separate pMOS transistors to precharge all intermediate nodes in the NMOS pull-down tree which have a large parasitic capacitance
- The use of multiple precharge transistors also enables us to use the precharged intermediate nodes as resources for additional outputs



Precharging of internal nodes to prevent charge sharing also allows implementation of multiple-output domino CMOS structures

Multiple-Output Domino

- Figure shows the realization of four Boolean functions of nine variables, using a single domino CMOS logic gate



- Compare this with alternatives 1) four separate standard CMOS logic gates or 2) four separate single-output domino CMOS circuits, both need larger number of transistors

- The four functions are:

$C_1=G_1+P_1C_0$

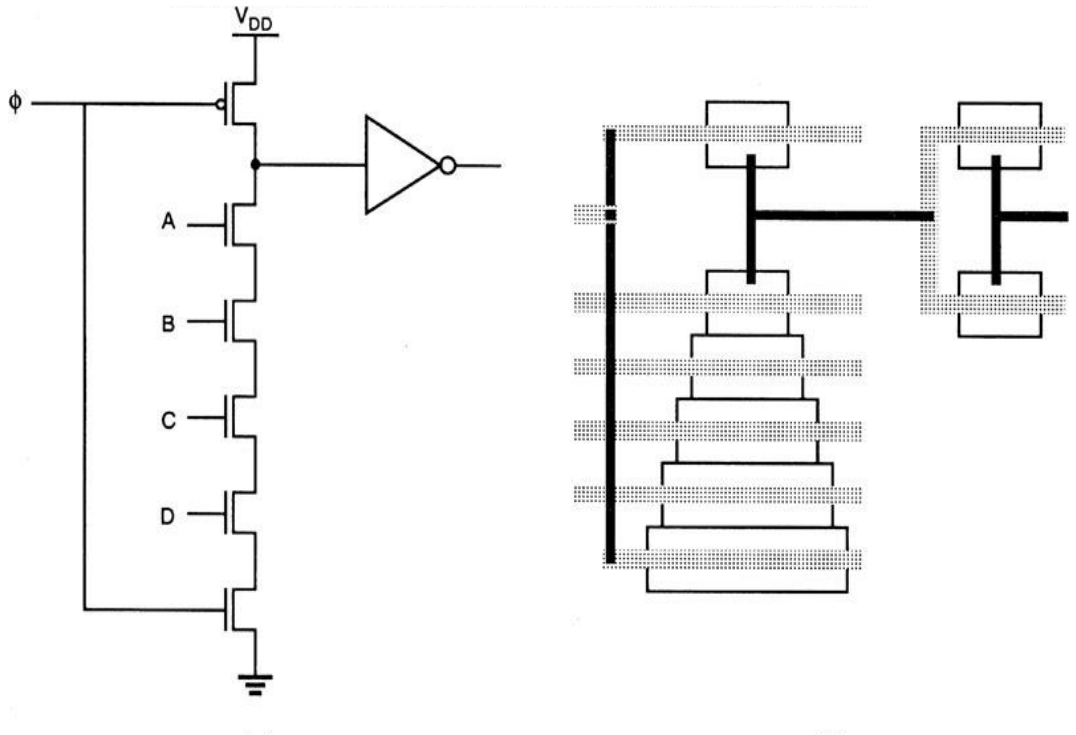
$C_2=G_2+P_2G_1+P_2P_1C_0$

$C_3=G_3+P_3G_2+P_3P_2G_1+P_3P_2P_1C_0$

$C_4=G_4+P_4G_3+P_4P_3G_2+P_4P_3P_2G_1+P_4P_3P_2P_1C_0$

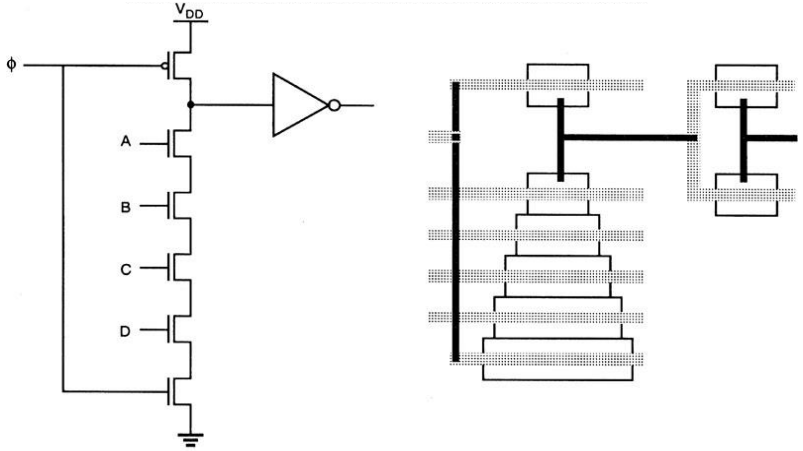
Transistor Sizing for Domino

- Adjusting the NMOS transistor sizes in the pull-down path can reduce the discharging time
- Recall that the best performance is obtained with a graded sizing of nMOS transistors in series structures. The same idea is applicable for the case of dynamic logic: The bottom-most transistor should be sized the biggest



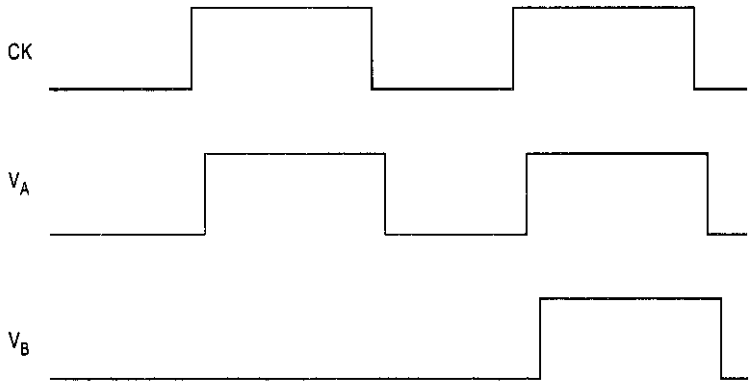
Transistor Sizing for Domino

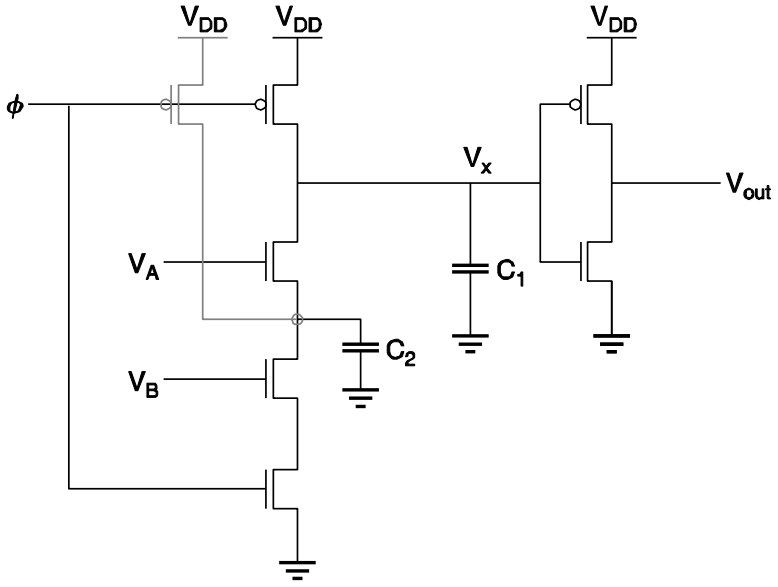
- C_L : load capacitance of the domino CMOS gate
- C_1 : parasitic capacitance of the intermediate node closest to output
  - Assume a pull-down chain of N series connected nMOS transistors
  - (shown by Shoji) If $C_L < (N-1)C_1/2$, then the overall delay time can be reduced by decreasing the size of the nMOS transistor closest to the output node. The result can be iteratively applied to other transistors in the pull-down chain which leads to graded sizing of all nMOS devices

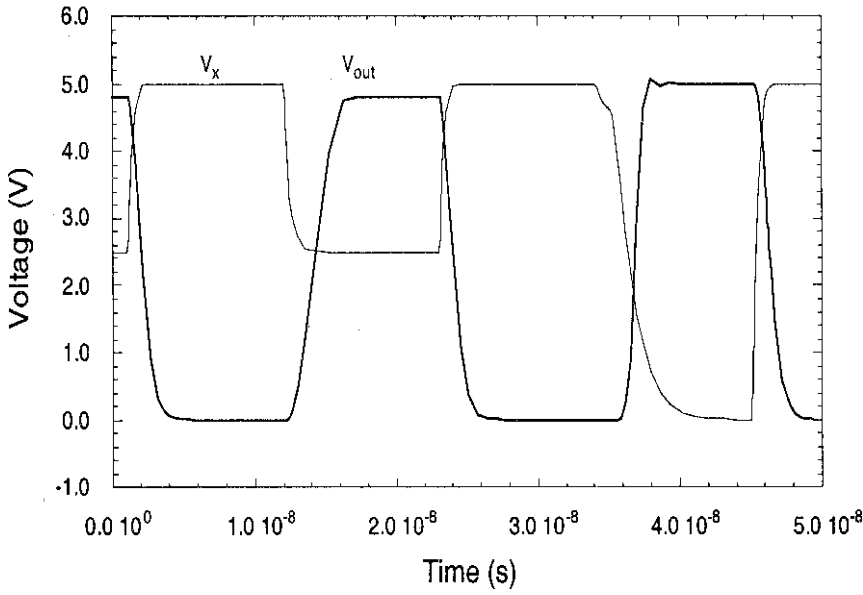


Example 9.4

- Consider the domino CMOS NAND2 gate, where $C_1=C_2=0.05$ pF
- First, the operation of the circuit with one pMOS transistor is tested: since $C_1=C_2$, we expect that the charge-sharing phenomenon will cause erroneous output values

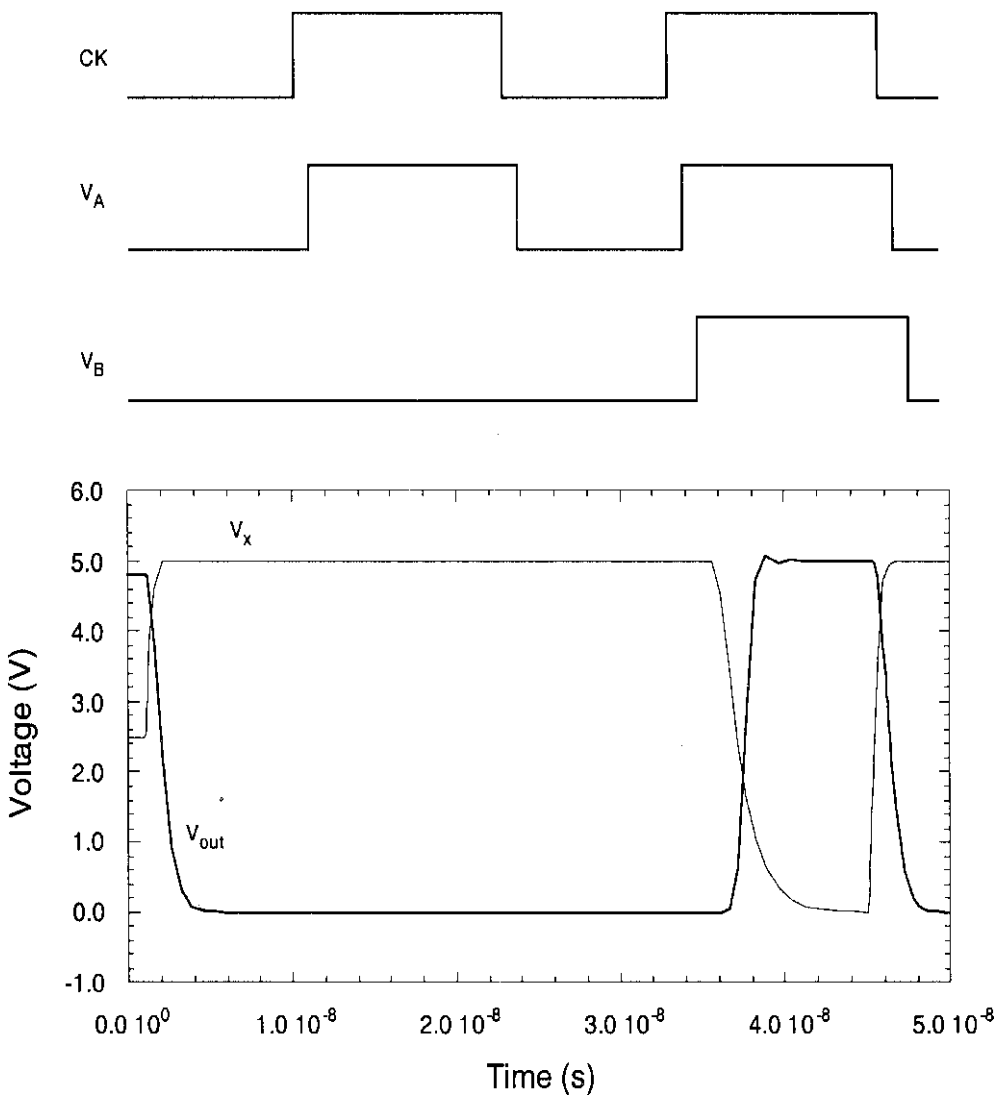






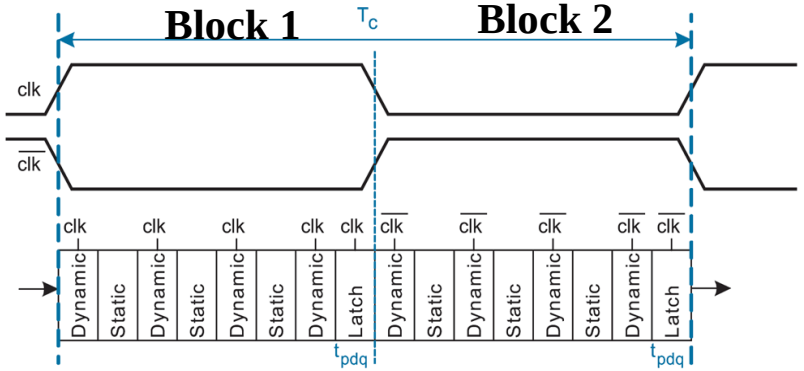
Example 9.4 (Cont.)

- Now consider the case where an additional pMOS precharge transistor is connected between V_{DD} and the intermediate node
- Both pMOS transistors conduct during the precharge phase, and charge up the node capacitance to the same voltage level. Consequently, charge sharing can no longer cause a logic error at the output node



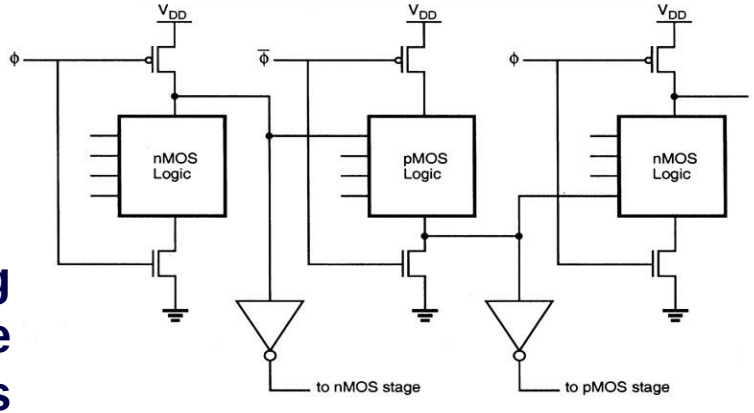
[Optional] Pipelined Domino

- While the clock is high, the first block of logic evaluates and the second precharges The opposite occurs when clock is low
- With this ping-pong approach, the precharge time does not appear in the critical path
- The latches hold the result of the one-half cycle while that half-cycle precharges and the next evaluates
- The data must arrive at the 1st half-cycle latch a setup time before the clock falls
- It propagates through the latch, so the overhead of each latch is the **maximum** of its setup time and its D-to-Q propagation delay!

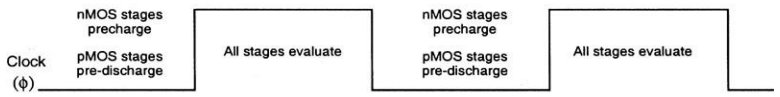


NORA CMOS Logic (NP-Domino Logic)

- NORA includes dynamic logic stages that also use pMOS transistors
- The pMOS logic is controlled by the inverted clock signal
- When the clock signal is low, the output nodes of nMOS logic are precharged to V_{DD} through pMOS precharge devices whereas output nodes of pMOS logic blocks are pre-discharged to 0V through the nMOS discharge transistor
- When the clock signal makes a low-to high transition, all cascaded nMOS and pMOS logic stages evaluate one after another

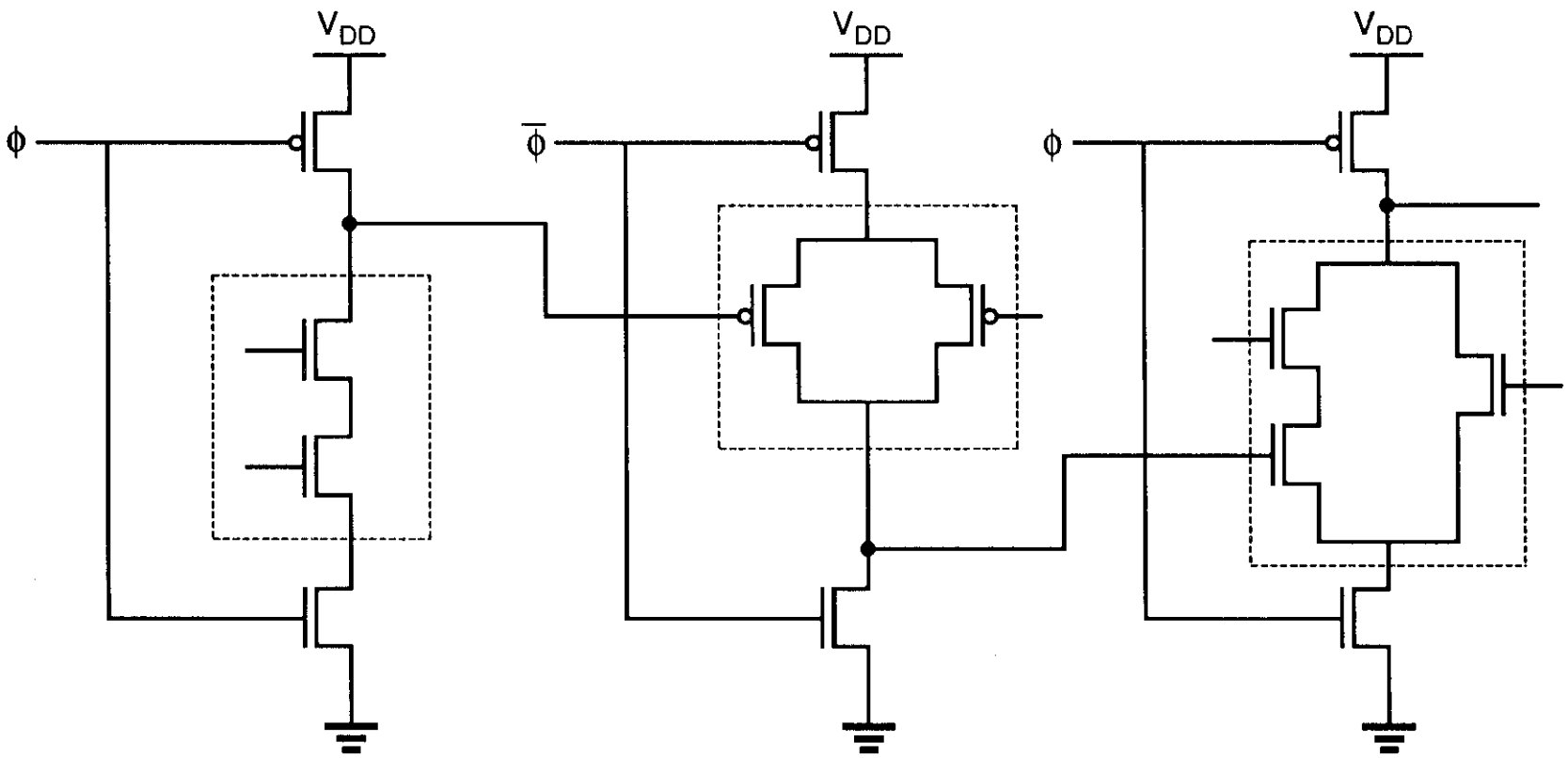


NORA CMOS logic consisting of alternating NMOS and PMOS stages, and the scheduling of precharge/evaluation phases



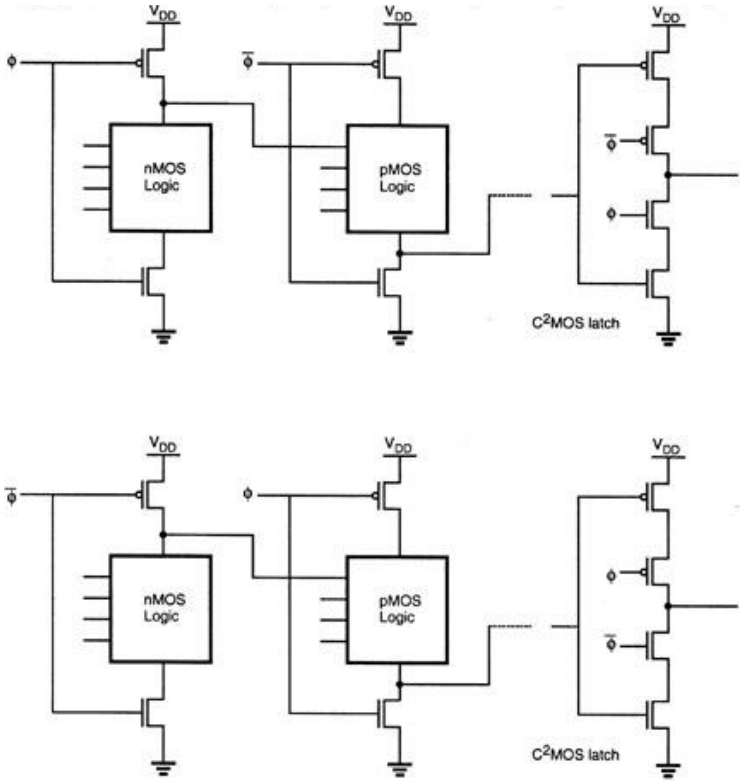
NP-Domino Logic – Example

- A simple NORA CMOS circuit example:

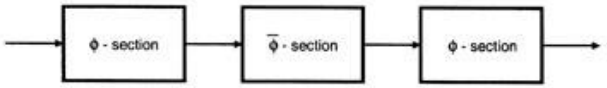


NP-Domino Logic (Cont.)

- Advantages:
  - A static CMOS inverter is not required at the output of every dynamic logic stage
  - Compatible with domino CMOS logic
  - It allows pipelined system architecture

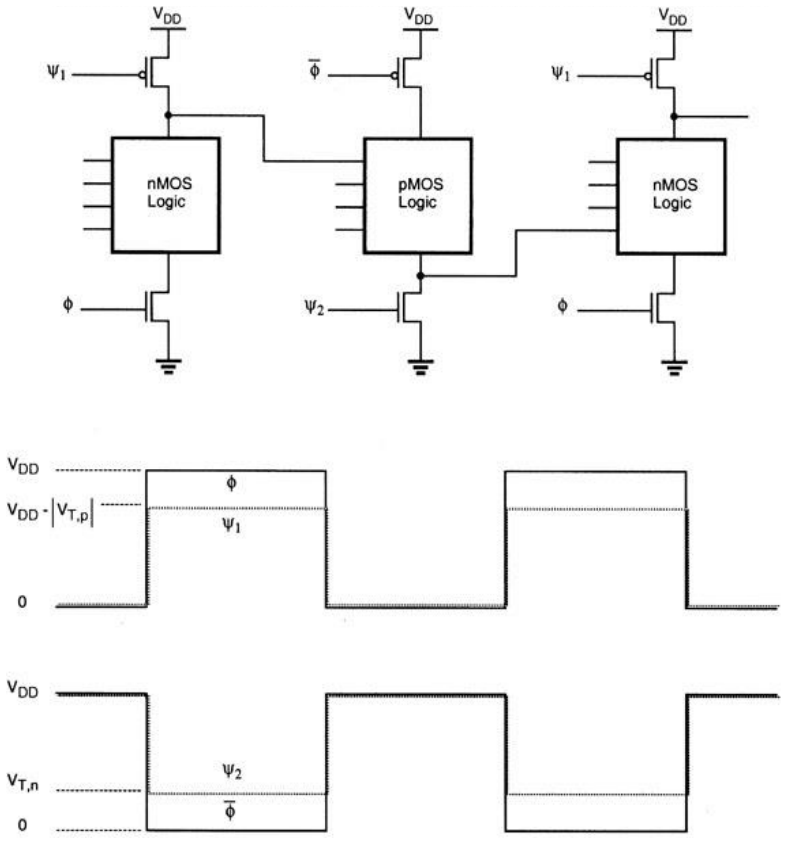


(a) NORA CMOS Φ-section; evaluation occurs during Φ=1
(b) NORA CMOS Φ-section; evaluation occurs during Φ=0
(c) A pipelined NORA CMOS system
Note: C^2MOS means Clocked CMOS



Zipper CMOS circuits

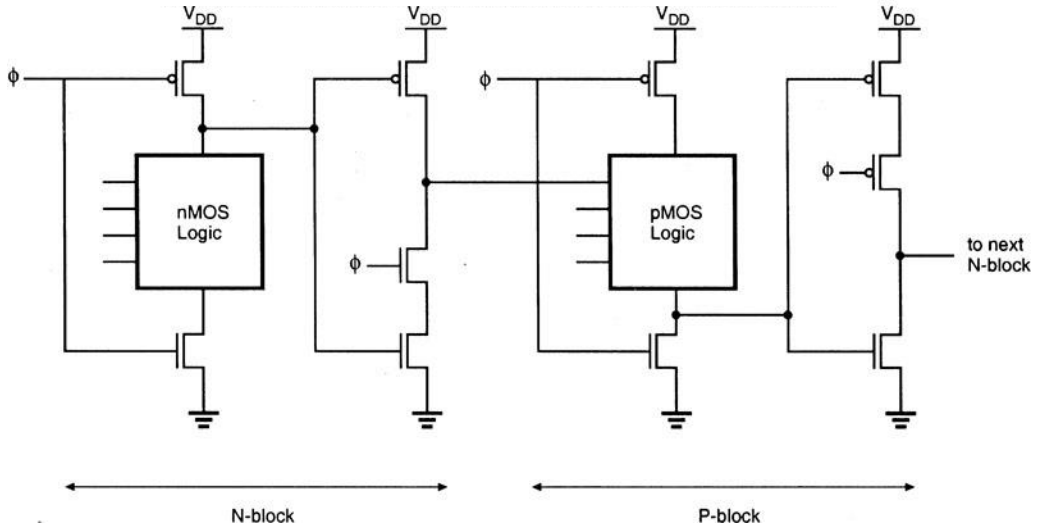
- Zipper technique is to overcome the dynamic charge sharing and soft-node leakage problems
- Identical to NORA CMOS, but requires the generation of slightly different clock signals for the precharge (discharge) transistors and for the pull-down (pull-up) transistors
- The clock signals allow the precharge (discharge) transistors to remain in weak conduction or near cut-off during the evaluation phase, thus compensating for the charge leakage and charge-sharing problems



General circuit structure and the clock signals of Zipper CMOS

True Signal-Phase clock (TSPC) dynamic CMOS

- TSPC uses one clock signal which is never inverted hence no clock skew problem exists
- This results in higher clock frequency



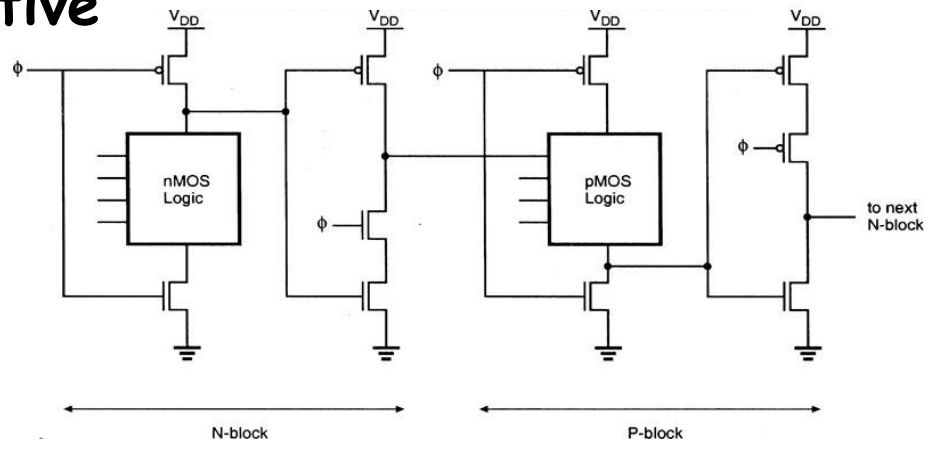
A pipelined true single-phase clock CMOS system

TSPC dynamic CMOS (Cont.)

- Consider the circuit shown before:
  - When the clock signal is low, the output node of the N-block is being precharged to V_{DD}
  - When the clock signal switches from low to high, the logic stage output is evaluated and the output latch generates a valid output level
  - The P-block pre-discharges when the clock is high, and evaluates when the clock is low
- Compared to NORA CMOS, we need two more transistors per stage, but the ability to operate with a true single-phase clock signal is attractive

  from system design

  point of view



[Optional] TSPC dynamic CMOS (Cont.)

- This DFF can work with clock frequency of 500MHz
- With relatively simple design, low transistor count, and high speed, TSPC-based circuits very favorable alternative to conventional CMOS circuits