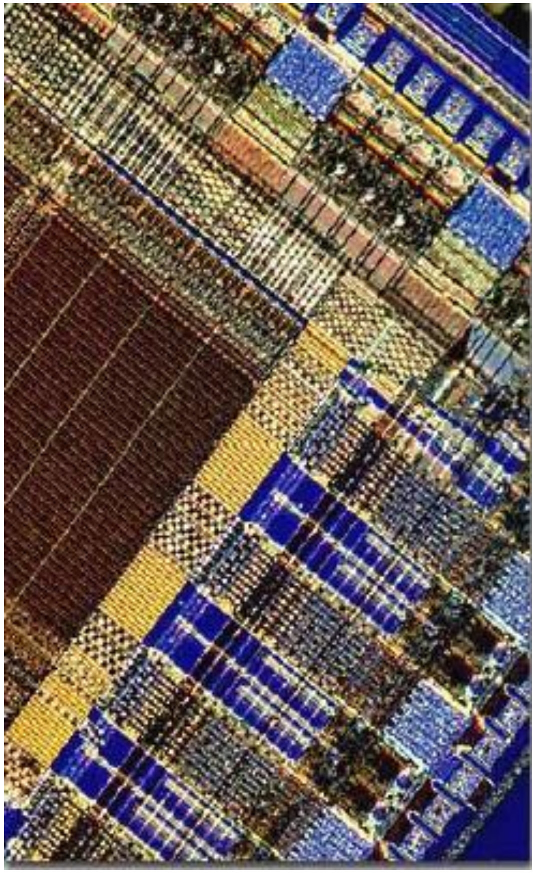
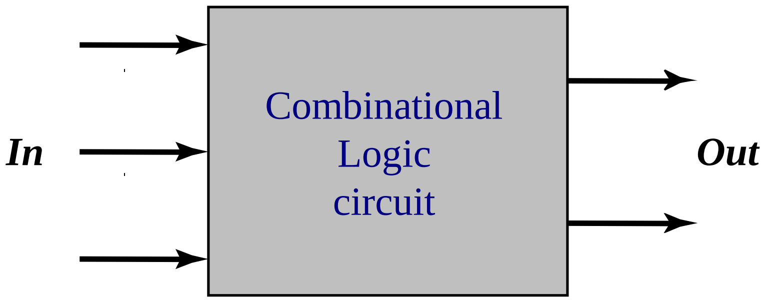
CMOS Digital Integrated Circuits

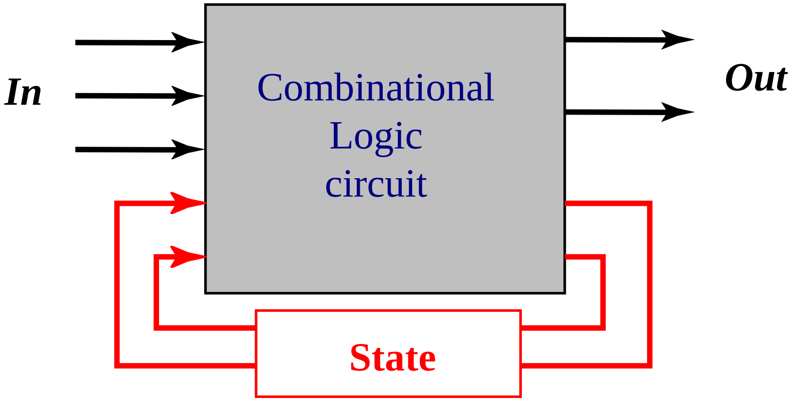


Lec 10

Combinational CMOS Logic Circuits

Combinational vs. Sequential Logic





Combinational

The output is determined only by

- Current inputs

Output = *f*(*In*)

Sequential

The output is determined by

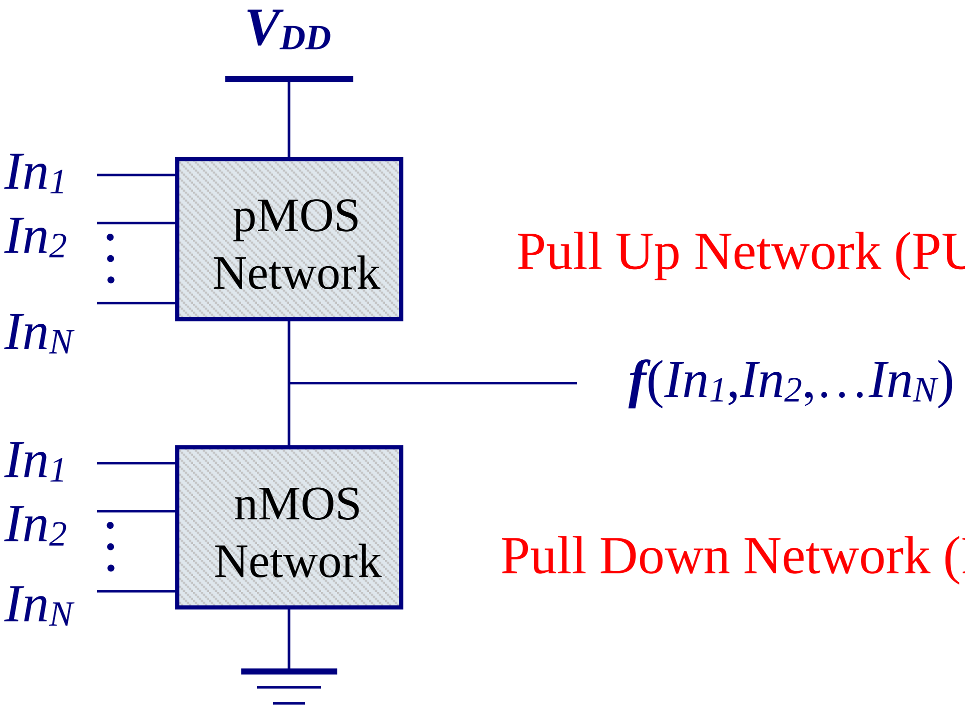
- Current inputs
- Previous inputs

Output = *f(In, Previous In)*

Static CMOS Circuit

- At every point in time (except during the switching transients) each gate output is connected to either ***V_{DD}*** or ***V_{SS}*** via a low-resistive path
- The outputs of the gates assume at all times the value of the Boolean function, implemented by the circuit (ignoring, once again, the transient effects during switching periods).
- This is contrasted to the ***dynamic*** circuit class, which relies on temporary storages of signal values on the capacitance of high impedance circuit nodes.

Static CMOS



Pull Up Network (PUN)

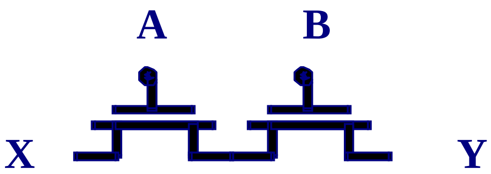
Pull Down Network (PDN)

PUN and PDN are dual logic networks

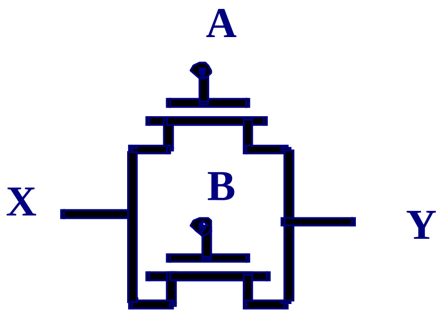
- **The complementary operation of a CMOS gate**
  - » The nMOS network (PDN) is on and the pMOS network (PUN) is off
  - » The pMOS network is on and the nMOS network is off.

NMOS Transistors
Series/Parallel Connection

- Transistors can be thought as a switch controlled by its gate signal
- NMOS switch closes when switch control input is high



Y = X if A and B=AB

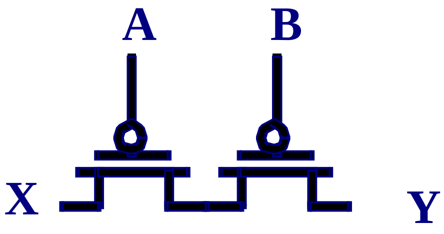


Y = X if A OR B=A+B

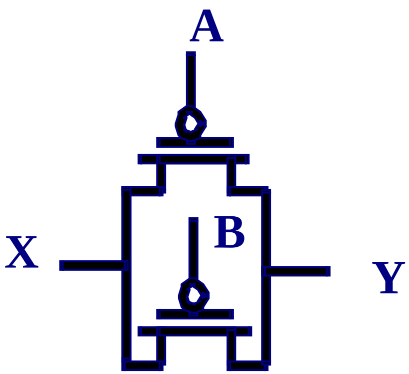
NMOS Transistors pass a “strong” 0 but a “weak” 1

PMOS Transistors
Series/Parallel Connection

- PMOS switch closes when switch control input is low



Y = X if $\overline{A}$ AND $\overline{B}$ = $\overline{A+B}$



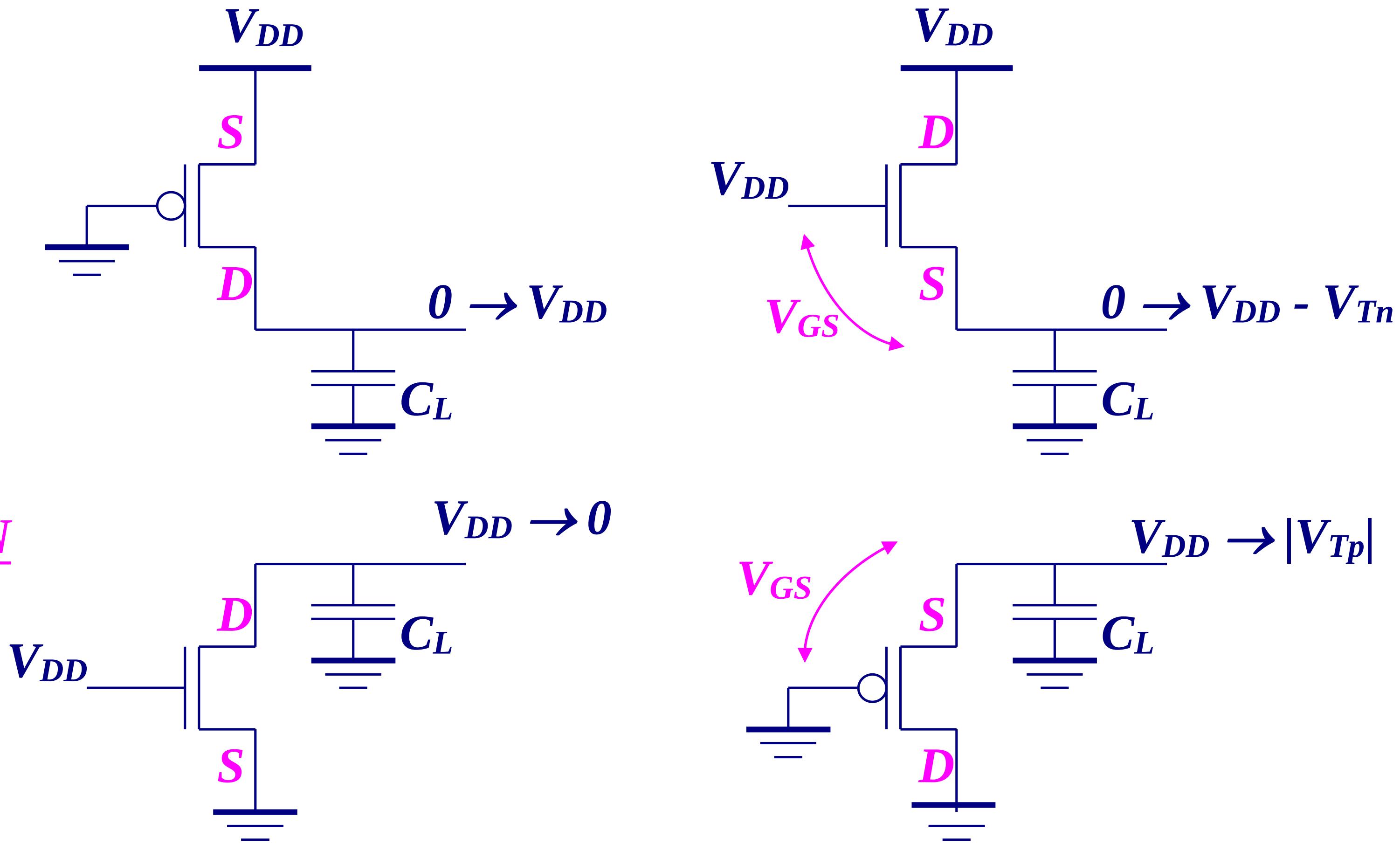
Y = X if $\overline{A}$ OR $\overline{B}$ = $\overline{AB}$

PMOS Transistors pass a “strong” 1 but a “weak” 0

Threshold Drops

PUN

PDN



CMOS Logic Style

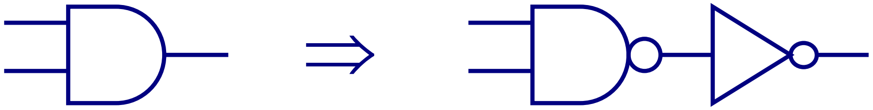
- PUN is the **DUAL** of PDN

  (can be shown using DeMorgan's Theorem's)

$$\overline{A+B} = \overline{A}\,\overline{B}$$

$$\overline{AB} = \overline{A} + \overline{B}$$

- The complementary gate is inverting

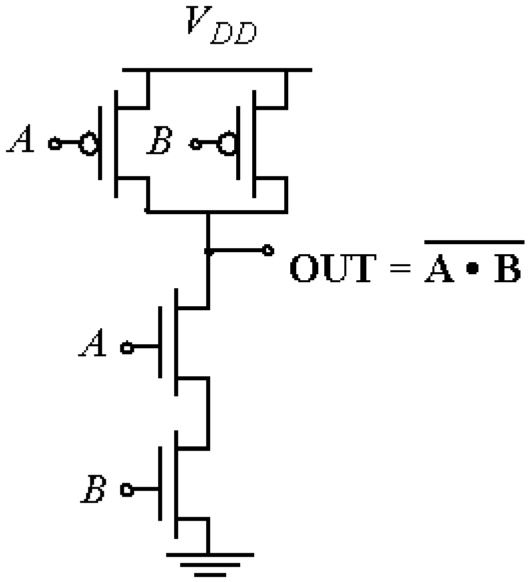


AND = NAND + INV

Example Gate: NAND

A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

Truth Table of a 2 input NAND gate



PDN: G = A B ⇒ Conduction to GND

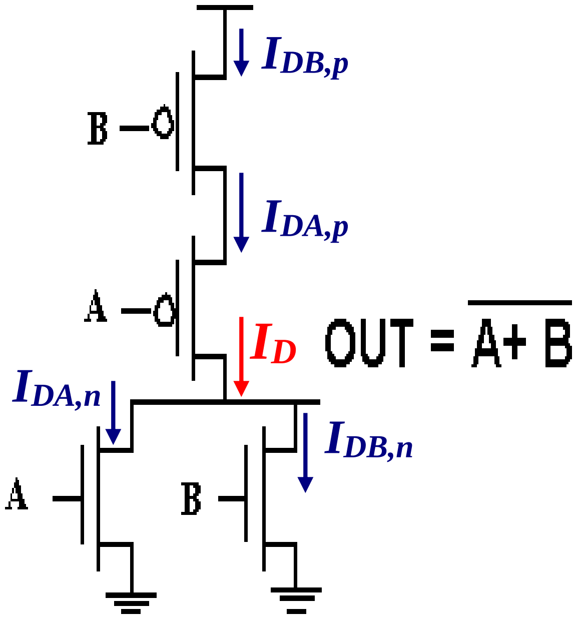
PUN: F= $\overline{A} + \overline{B} = \overline{AB}$ ⇒ Conduction to V_{DD}

$$\overline{G(In_1, In_2, In_3, \ldots)} \equiv F(\overline{In_1}, \overline{In_2}, \overline{In_3}, \ldots)$$

CMOS NOR2
Two-Input NOR Gate

A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0

Truth Table of a 2 input NOR gate



$OUT = \overline{A + B}$

CMOS NOR2
Threshold Calculation (*1/3*)

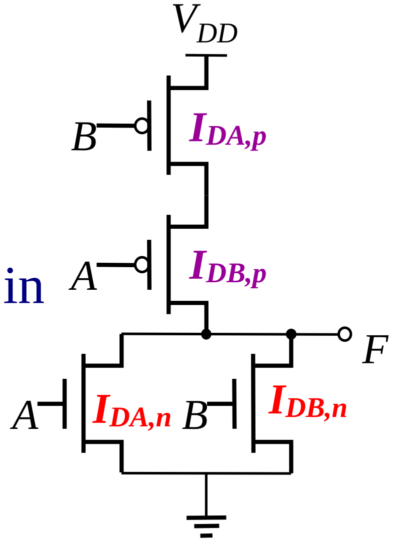
- **Basic Assumptions**
  - » Both input ***A*** and ***B*** switch simultaneously ($V_A = V_B$)
  - » The device sizes in each block are identical. $(W/L)_{n,A} = (W/L)_{n,B}$, and $(W/L)_{p,A} = (W/L)_{p,B}$
  - » The substrate-bias effect for the PMOS is neglected

V_{th} Calculation

- By definition, $V_A = V_B = V_{out} = V_{th}$. The two NMOS transistors are saturated because $V_{GS} = V_{DS}$,

$$I_D = I_{DA,n} + I_{DB,n} = k_n(V_{th}-V_{T,n})^2$$

$$\Rightarrow V_{th} = V_{T,n} + \sqrt{I_D / k_n}$$



- PMOS-B operates in the linear region, and PMOS-A is in saturation for $V_{in} = V_{out}$,

$$I_{DB,p} = \frac{k_p}{2}\left[2\left(V_{DD} - V_{th} - |V_{T,p}|\right)V_{SDB,p} - V_{SDB,p}^2\right]$$

$$I_{DA,p} = \frac{k_p}{2}\left(V_{DD} - V_{th} - |V_{T,p}| - V_{SDB,p}\right)^2$$

CMOS NOR2
Threshold Calculation (2/3)

Since $\boldsymbol{I_{DA,p} = I_{DB,p} = I_D}$, we have

$$V_{DD} - V_{th} - \left|V_{T,p}\right| = 2\sqrt{I_D / k_p}$$

- Combine the above equations, we obtain

$$V_{th}(NOR2) = \frac{V_{T,n} + \frac{1}{2}\sqrt{\frac{k_p}{k_n}}\left(V_{DD} - \left|V_{T,p}\right|\right)}{1 + \frac{1}{2}\sqrt{\frac{k_p}{k_n}}}$$

which is different with the expression of $\boldsymbol{V_{th}(INV)}$

$$V_{th}(INV) = \frac{V_{T,n} + \sqrt{\frac{k_p}{k_n}}\left(V_{DD} - \left|V_{T,p}\right|\right)}{1 + \sqrt{\frac{k_p}{k_n}}}$$

CMOS NOR2 Threshold Calculation (3/3)

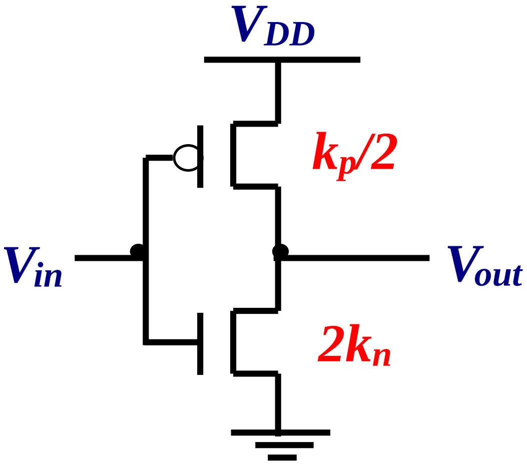
- If $k_n = k_p$ and $V_{T,n} = |V_{T,p}|$, $V_{th}(INV) = V_{DD}/2$. However,

$$V_{th}(NOR2) = \frac{V_{DD} + V_{T,n}}{3}$$

Equivalent-Inverter Approach (both inputs are identical)

» The parallel connected nMOS transistors can be represented by a nMOS transistor with $2k_n$.

» The series connected pMOS transistors can be represented by a pMOS transistor with $k_p/2$.



CMOS NOR2
Equivalent-Inverter Approach

- Therefore

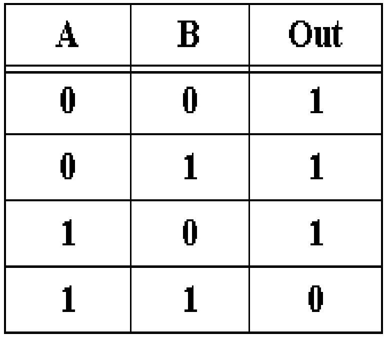
$$V_{th}(NOR2) = \frac{V_{T,n} + \sqrt{\dfrac{k_p}{4k_n}}\left(V_{DD} - \left|V_{T,p}\right|\right)}{1 + \sqrt{\dfrac{k_p}{4k_n}}}$$

- To obtain a switching threshold voltage of $\boldsymbol{V_{DD}/2}$ for simultaneous switching, we have to set $\boldsymbol{V_{T,n} = |V_{T,p}|}$ and $\boldsymbol{k_p=4k_n}$

Parasitic Capacitances and Simplified Equivalent Circuit: See Fig. 7.12 in Kang and Leblebici.

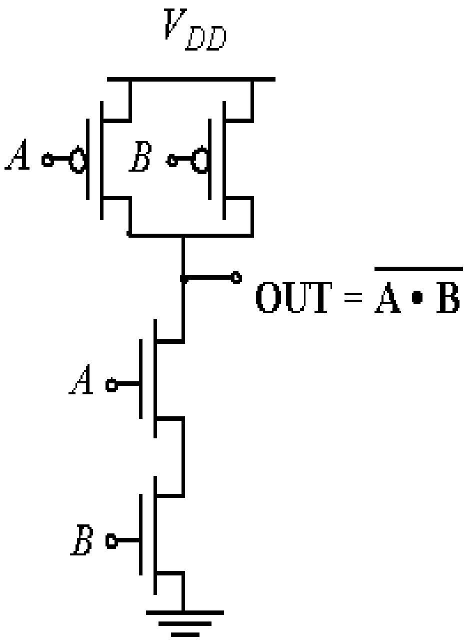
» The total lumped load capacitance is assumed to be equal to the sum of all internal capacitances in the worst case.

CMOS NAND2
Two-Input NAND Gate



A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

Truth Table of a 2 input NAND gate



PDN: G = A B ⇒ Conduction to GND

PUN: F= $\bar{A} + \bar{B} = \overline{AB}$ ⇒ Conduction to V_{DD}

$$\overline{G(In_1, In_2, In_3, \ldots)} \equiv F(\overline{In_1}, \overline{In_2}, \overline{In_3}, \ldots)$$

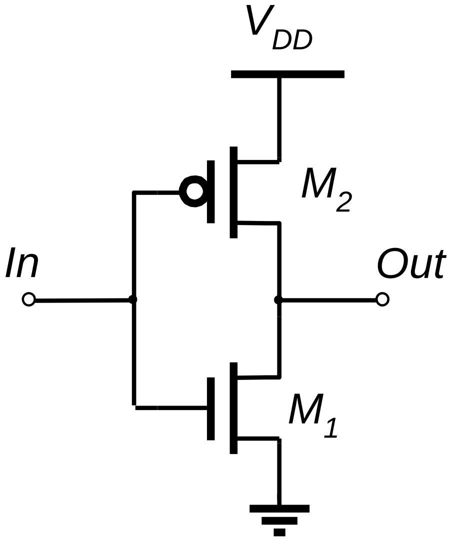
CMOS NAND2
Threshold Calculation

- Assume the device sizes in each block are identical, $(W/L)_{n,A} = (W/L)_{n,B}$, and $(W/L)_{p,A} = (W/L)_{p,B}$, and by the similar analysis to the one developed for the NOR2 gate, we have

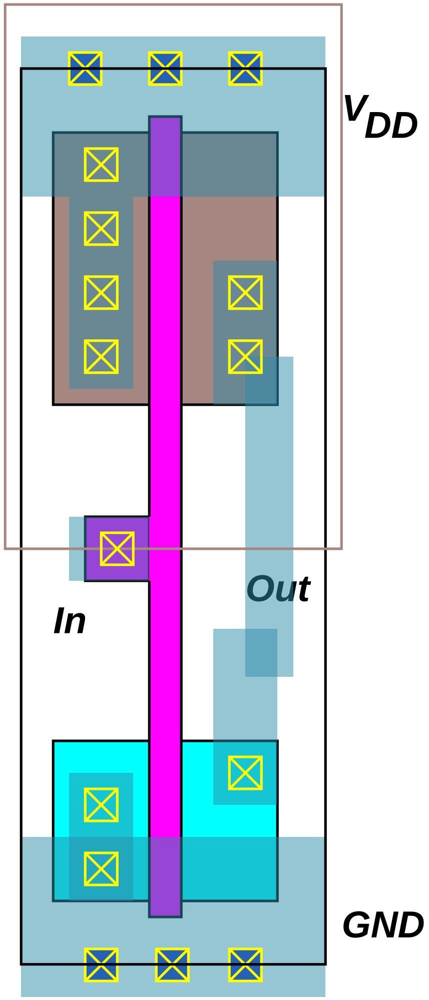
$$V_{th}(NAND2) = \frac{V_{T,n} + 2\sqrt{\frac{k_p}{k_n}}\left(V_{DD} - \left|V_{T,p}\right|\right)}{1 + 2\sqrt{\frac{k_p}{k_n}}}$$

- To obtain a switching threshold voltage of $V_{DD}/2$ for simultaneous switching, we have to set $V_{T,n} = |V_{T,p}|$ and $k_n = 4k_p$

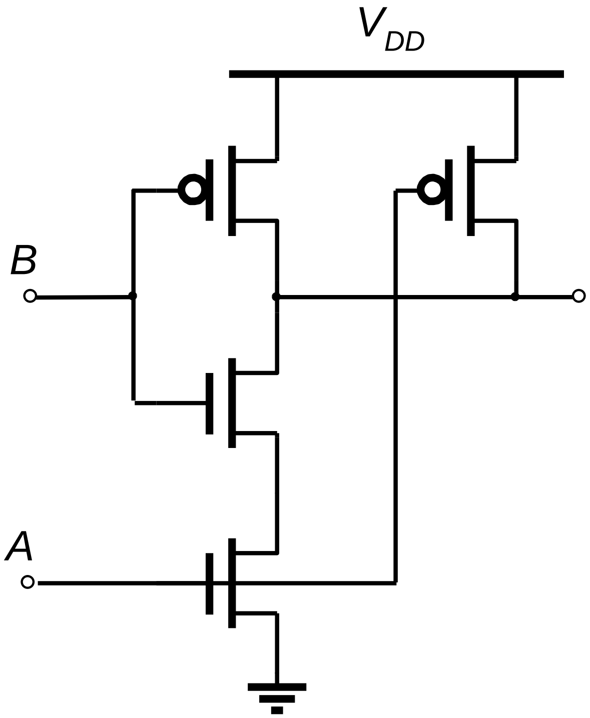
Layout of Simple CMOS Logic Gates (1/2)



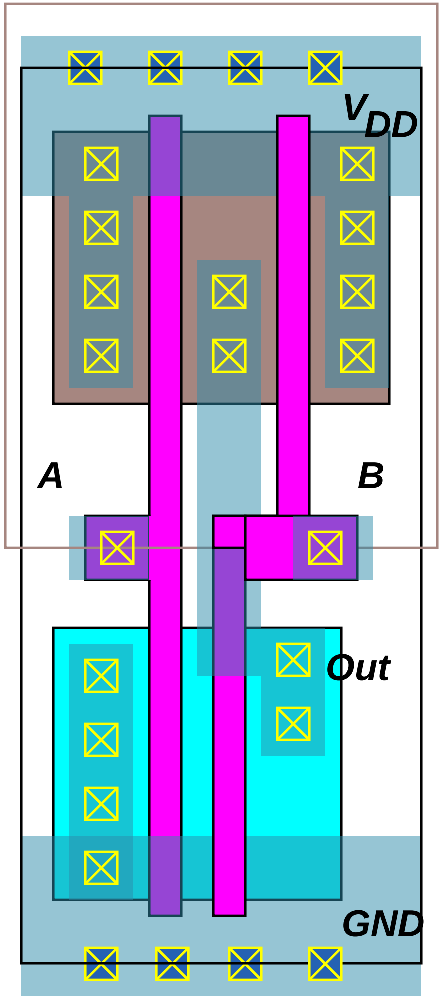
Inverter



Layout of Simple CMOS Logic Gates (2/2)



2-input NAND gate



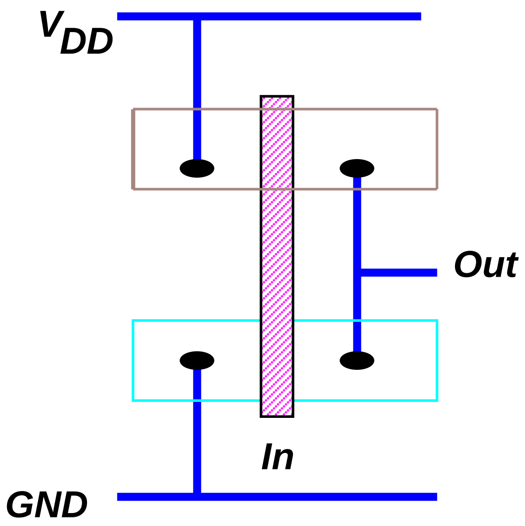
Stick Diagram (*1/2*)

- Does not contain any information of dimensions.
- Represent relative positions of transistors

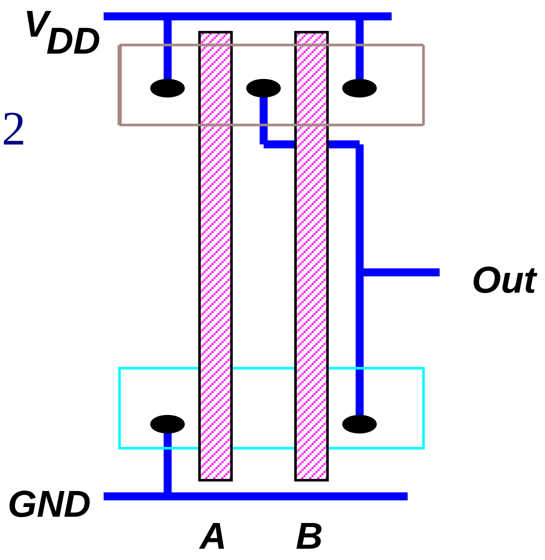
Basic Elements

- » **Rectangle**: Diffusion Area
- » **Solid Line**: Metal Connection
- » **Circle**: Contact
- » **Cross-Hatched Strip**: Polysilicon

INV

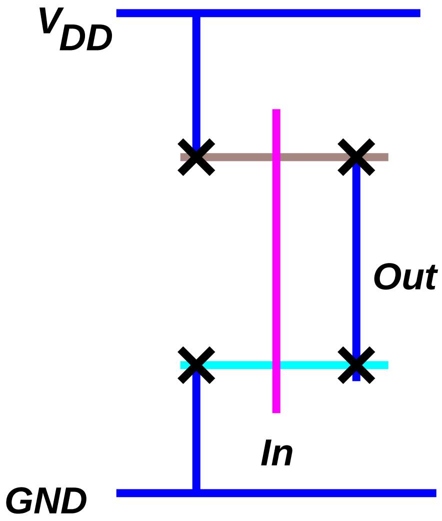


NAND2

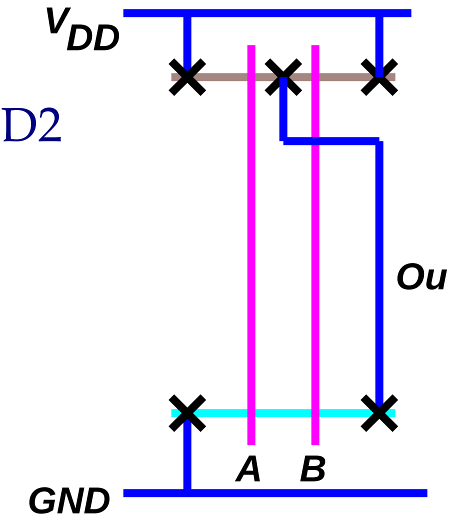


Stick Diagram (2/2)

INV



NAND2



Complex CMOS Gates Functional Design (*1/3*)

- **OR** operations are performed by **parallel-connected drivers**.
- **AND** operations are performed by **series-connected drivers**.
- **Inversion** is provided by **the nature of MOS circuit operation**.

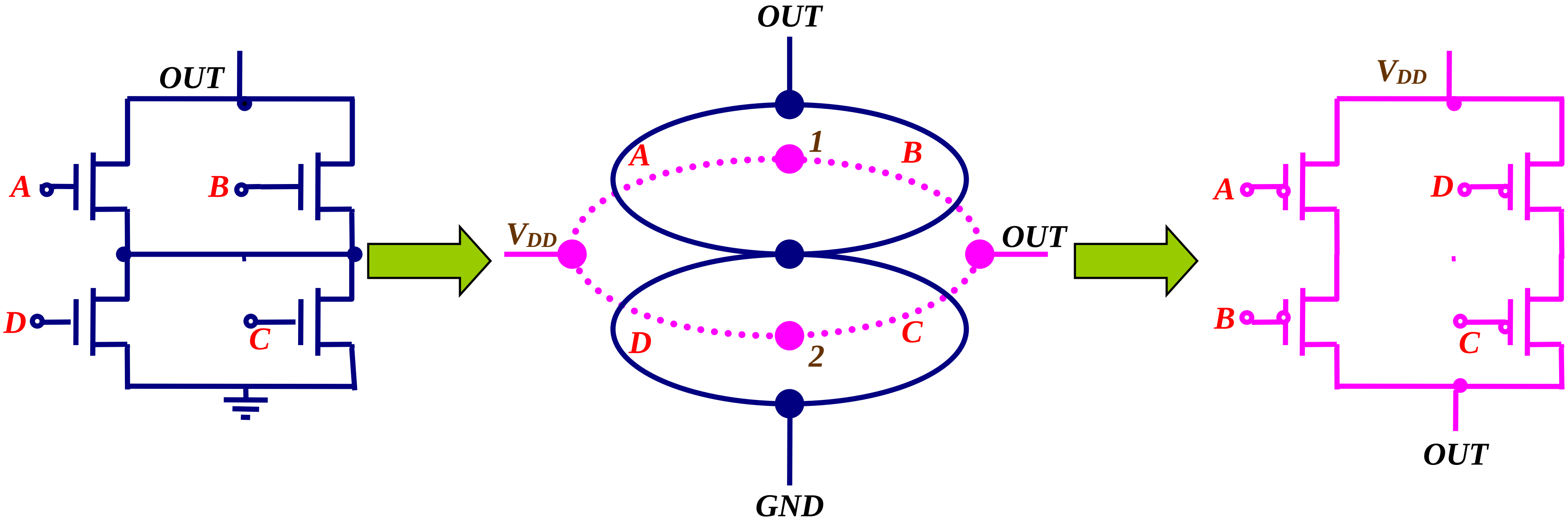
- The realization of pull-down network is based on the same basic design principle examined earlier.
- The pMOS pull-up network must be the ***dual network*** of the nMOS pull-down network.
- One method systematically derives the pull-up network directly form the pull-down network. This method constructs the ***dual graph*** of the network. The pull-down network graph has nodes for circuit nodes and arcs for nFETs with the each arc labeled with the literal on the input to the corresponding nFET.

Complex CMOS Gates
Functional Design (2/3)

- To construct a graph and pull-up network from a pull-down network
  - Insert a node in each of the enclosed areas within the pull-down network graph.
  - Place two nodes outside of the network separated by arcs from GND and OUT.
  - Connect pairs of new nodes by drawing an arc through each arc in the pull-down circuit that lies between the corresponding pairs of areas.
  - Draw the resulting pull-up network with a pFET for each of the new arcs labeled with the same literal as on the nFET from which it came.
- The justification
  - The complement of a Boolean expression can be obtained by taking its dual, replacing ANDs with ORs and ORs with ANDs and complementing the variables,
  - The graphical dual corresponds directly to the algebraic dual.
  - Complementation of the variables takes place automatically because each nFETs is replaced with a pFET.

Complex CMOS Gates
Functional Design (3/3)

- This method is illustrated by the generation of the pull-up from the pull-down shown.



- On the dual graph, which of the two side nodes is labeled ***V_{DD}*** or ***OUT*** is functionally arbitrary. The selection may, however, affect the location of capacitances, and hence, the performance.

Complex CMOS Gates
Device Sizing in Complex Gates (*1/4*)

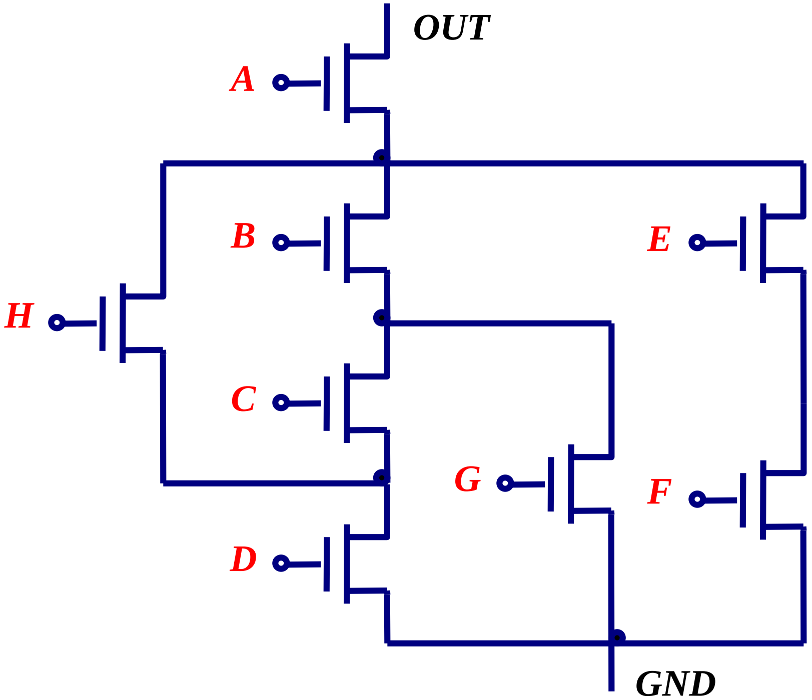
- Method used for sizing **NAND** and **NOR** gates also applies to complex gates
- Most easily transferred by examining all possible paths from **OUT** to **GND** (and from V_{DD} to **OUT**)
- Suppose that we are dealing with CMOS and the sized inverter devices use minimum channel lengths and widths W_n and W_p.
- For the pull-down network:
  1. Find the length n_{max} of the longest paths between **OUT** and through **GND** the network. Make the width of the nFETs on these paths $n_{max}W_n$.

     In this algorithm, a path is a series of FETs that does not contain any complementary pair of literals such as X and $\bar{X}$.
  2. For next longest paths through the circuit between **OUT** and **GND** consisting of nFETs not yet sized, repeat Step 1.
  3. Repeat Step 2 until there are no full paths consisting of unsized nFETS

Complex CMOS Gates
Device Sizing in Complex Gates (2/4)

4. For each longest partial path in the circuit consisting of unsized nFETs, based on the **longest** path between **OUT** and **GND** on which it lies, find the equivalent W_{eq} required for the partial path.
5. Repeat Step 1 for each longest partial path from Step 4 with **OUT** and **GND** replaced the endpoints of the partial path. Make the widths of devices on the path equal to $n_{max}W_{eq}$ where n_{max} is the number of FETs on the partial path.
6. Repeat 4 and 5 for newly generated longest partial paths until all devices are sized.

Complex CMOS Gates
Device Sizing in Complex Gates (3/4)



- This can be illustrated for the example above. L_n =0.5μ, W_n=5 μ, in the inverter.

  1. A longest path through the network from **OUT** to **GND** is ***A-B-C-D*** with n_{max}=4. Thus, the widths W_A, W_B, W_C, W_D are 4×5=20 μ. This is the only longest path we can find from

Complex CMOS Gates
Device Sizing in Complex Gates (*4/4*)

OUT to **GND** without passing through a sized device.

2. ***H*** and ***G*** are partial path. But it is important that they are considered as part of a longest between **OUT** and **GND** for evaluation. Thus, a "split" partial path consisting of ***H*** and ***G*** must be considered. Based on the evaluation segments, W_{eq} =2W_n=10μ. Thus, W_H and W_G are 1×10=10 μ.
3. The longest remaining partial path in the circuit is ***E-F*** with n_{max} = 2. Since this path is in series with ***A*** with width 4W_n=20 μ, it needs to have an equivalent width of W_{eq} determined from:

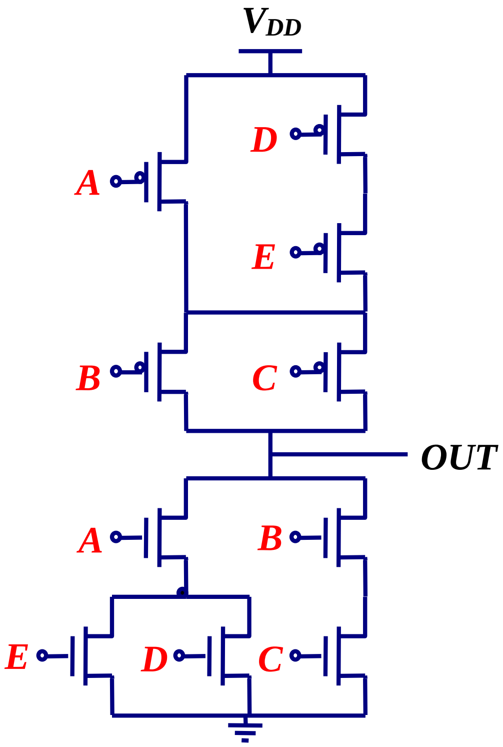
$$\frac{1}{W_n}=\frac{1}{4W_n}+\frac{1}{W_{eq}}\Rightarrow\frac{1}{5}=\frac{1}{20}+\frac{1}{W_{eq}}$$

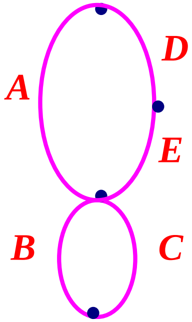
W_{eq} = 20/3 μ and the widths W_E and W_F are 2×20/3 μ=40/3 μ.

Since all devices are sized, we are finished.

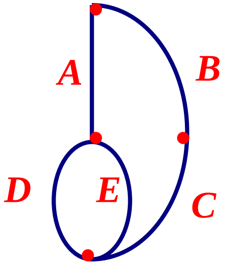
Complex CMOS Gates
Layout of Complex Gates (*1/4*)

- **Goal:** Given a complex CMOS logic gate, how to find a minimum-area layout.





pMOS network



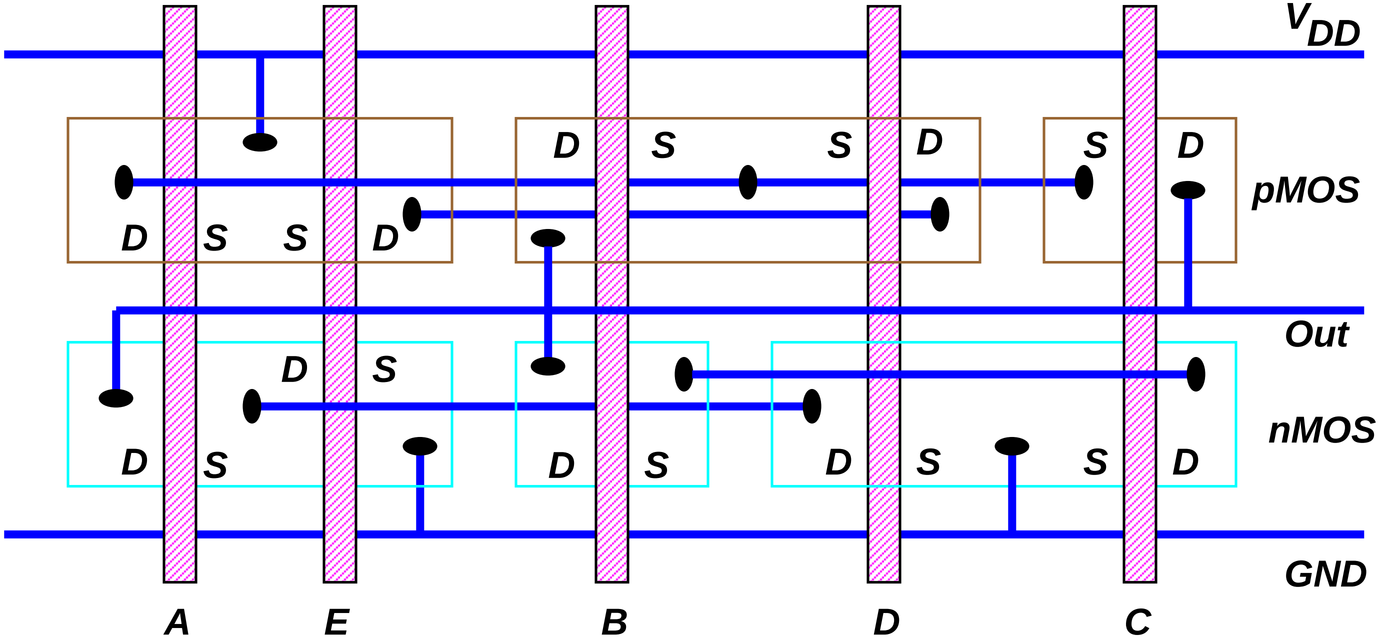
nMOS network

Complex CMOS Gates
Layout of Complex Gates (2/4)

Arbitrary ordering of the polysilicon columns:

» The separation between the polysilicon columns must allow for one diffusion-to-diffusion separation and two metal-to-diffusion contacts in between

⇒ Consume a considerable amount of extra silicon area

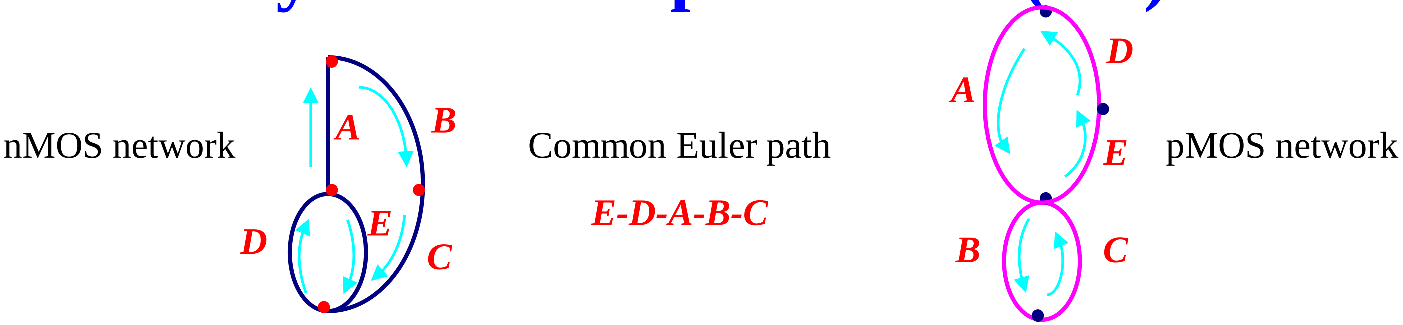


Complex CMOS Gates
Layout of Complex Gates (3/4)

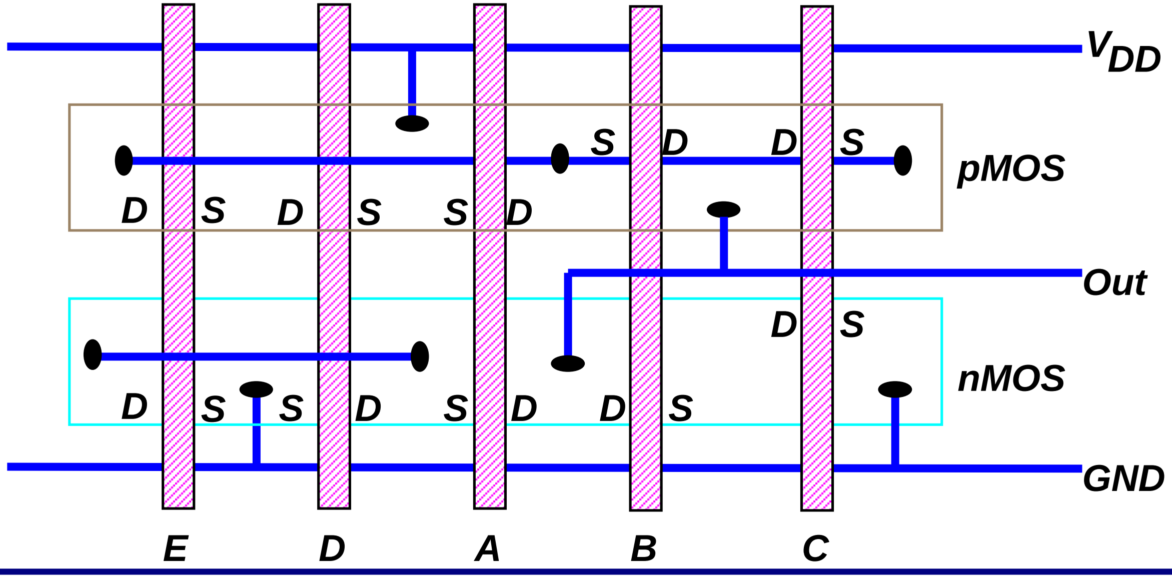
Euler Path Approach

- Objective: To order the inputs such that the diffusion breaks between input polysilicon strips is minimized, thereby reducing the width of the layout.
- Definition: An ***Euler path*** is an uninterrupted path that traverses each gate of the graph exactly once.
- Approach:
  - » Draw the graph for the NMOS and PMOS networks.
  - » Find a common Euler path through both of the graphs.
    - Note that nodes with an odd number of attached edges must be at the end points of the Euler path.
    - **Some circuits may not have Euler paths** – ***Do Euler paths for parts of the circuit in such cases.*** A circuit constructed using the dual graph method is more likely to have an Euler path.
  - » Order the transistor pairs in the layout in the order of the path from *left to right* or *right to left.*

Complex CMOS Gates
Layout of Complex Gates (4/4)



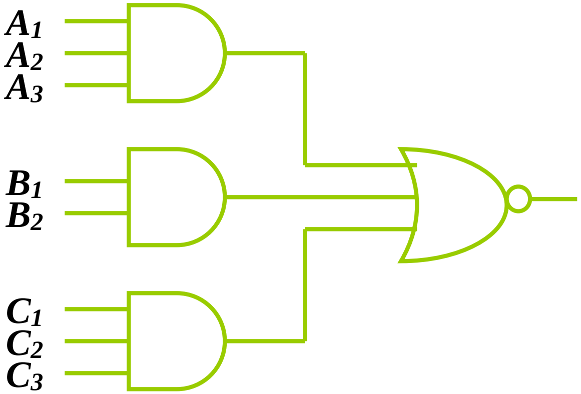
- Euler path successful: Order: ***E-D-A-B-C***
- Do the symbolic layout (stick diagram)
  - » More compact, simple routing of signals, and consequently, *less parasitic capacitance*

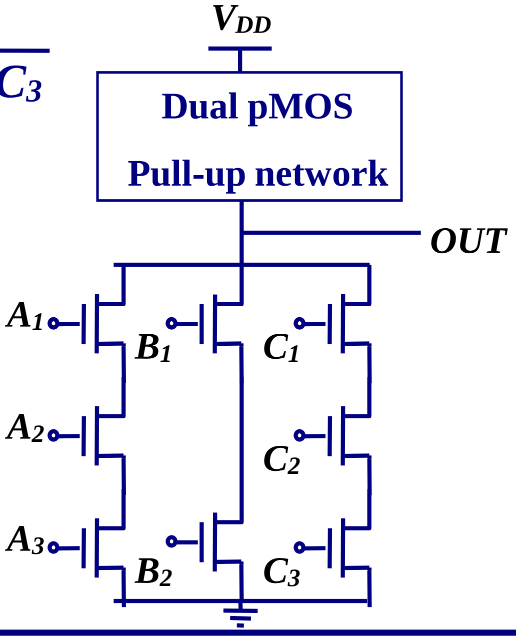


Complex CMOS Gates
AOI Gates

- **AOI** (AND-OR-INVERT): Enable the sum-of-products realization of a Boolean function in one logic gate.
  - » The pull-down network consists of parallel branches of series-connected nMOS driver transistors.
  - » The corresponding pull-up network can be found using the dual-graph concept.

Example: $\boldsymbol{OUT} = \overline{\boldsymbol{A_1A_2A_3+B_1B_2+C_1C_2C_3}}$

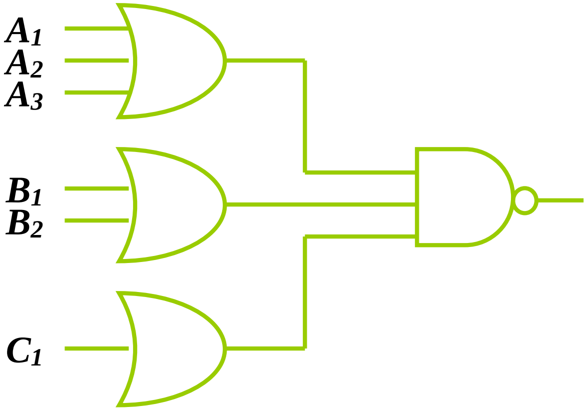


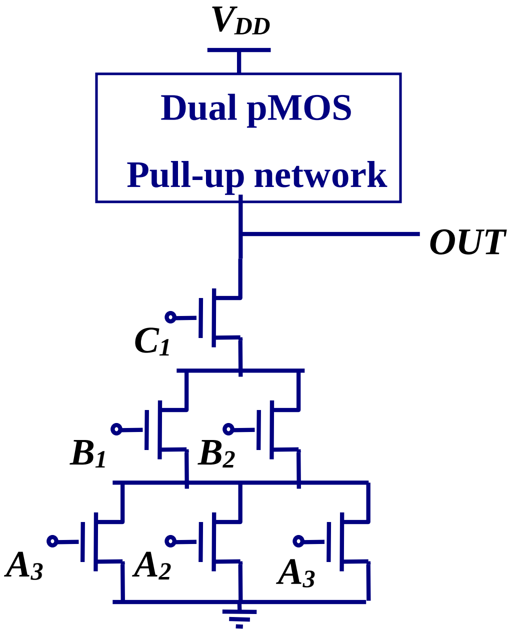


Complex CMOS Gates
OAI Gates

- **OAI** (OR-AND-INVERT): Enable the product-of-sums realization of a Boolean function in one logic gate.
  - » The pull-down network consists of series branches of parallel-connected nMOS driver transistors.
  - » The corresponding pull-up network can be found using the dual-graph concept.

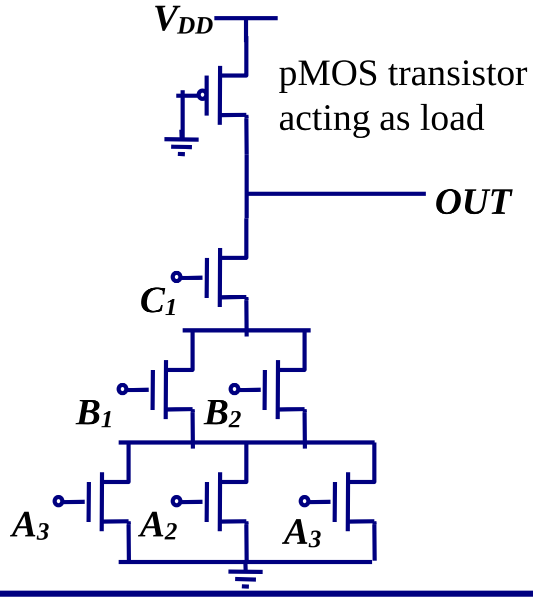
$$OUT = \overline{(A_1+A_2+A_3)(B_1+B_2)\ C_1}$$





Complex CMOS Gates
Pseudo-NMOS

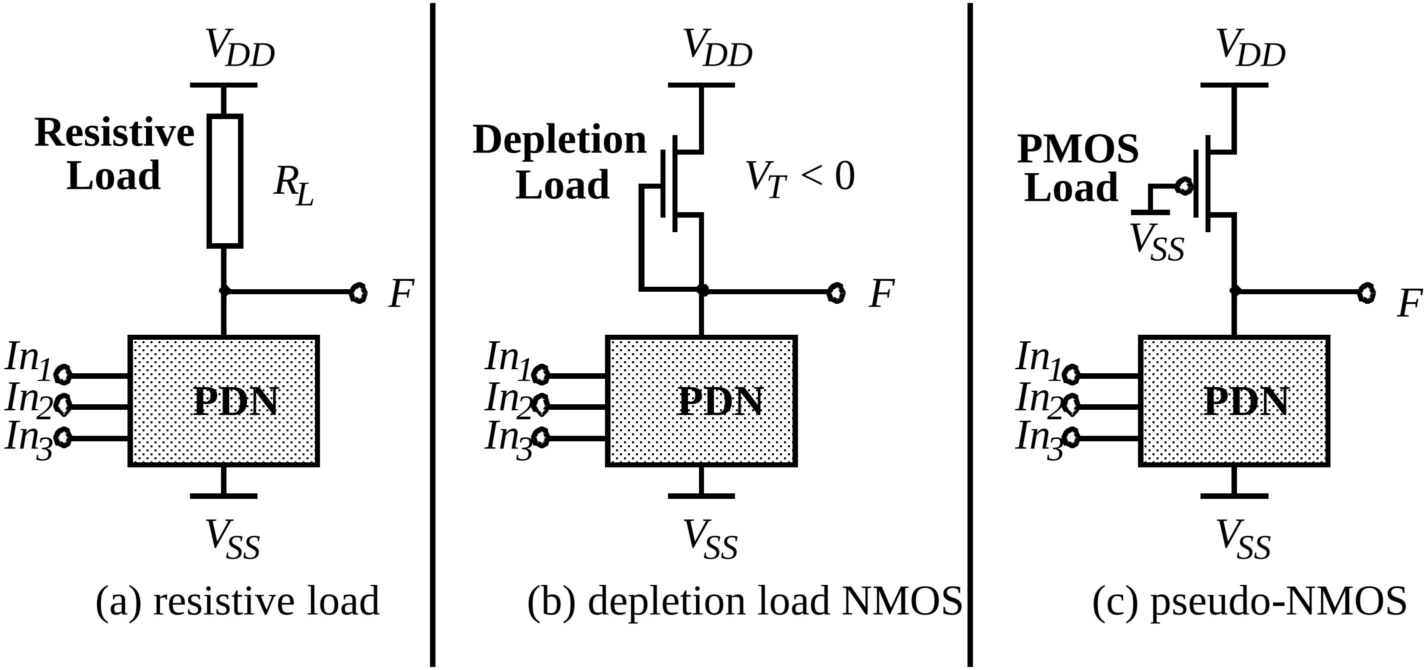
- In Pseudo-NMOS, the PMOS network is replaced by a single pFET with its gate attached to ***GND***. This provides a fixed load such as on NMOS circuits, hence called *pseudo-NMOS.*
- **Advantage:** Eliminate the PMOS network and hence reduce area.
- **Disadvantages:**
  - » Back to ratioed design and $\boldsymbol{V_{OL}}$ problems as in NMOS since PFET is always ON.
  - » “Non-zero” static power dissipation.



pMOS transistor acting as load

Ratioed Logic (*1/2*)

- **Ratioless Logic:** The logic levels are not dependent upon the relative device sizes.
- **Ratioed Logic:** The logic levels are determined by the relative dimensions of composing transistors



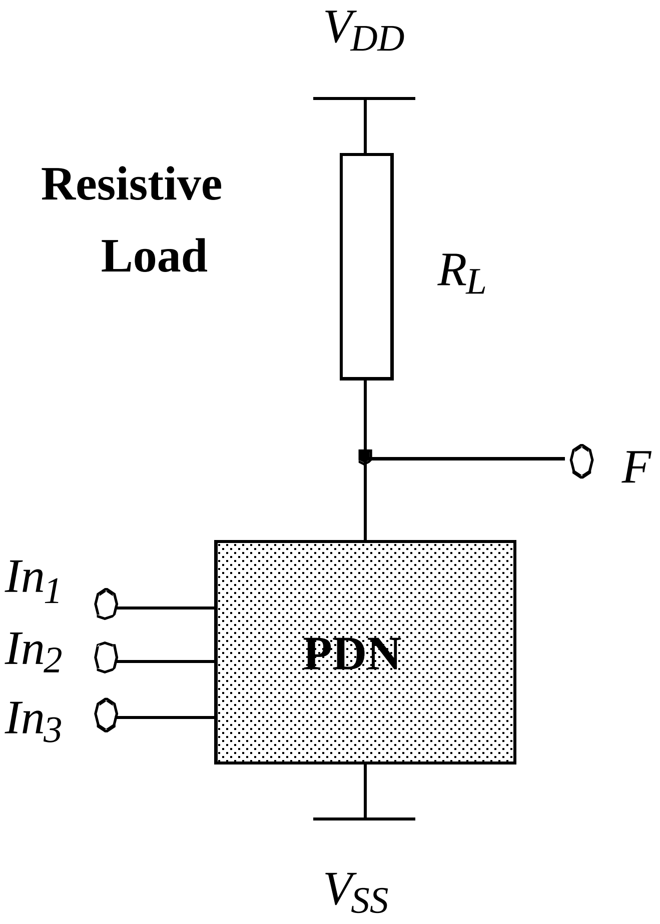
(a) resistive load

(b) depletion load NMOS

(c) pseudo-NMOS

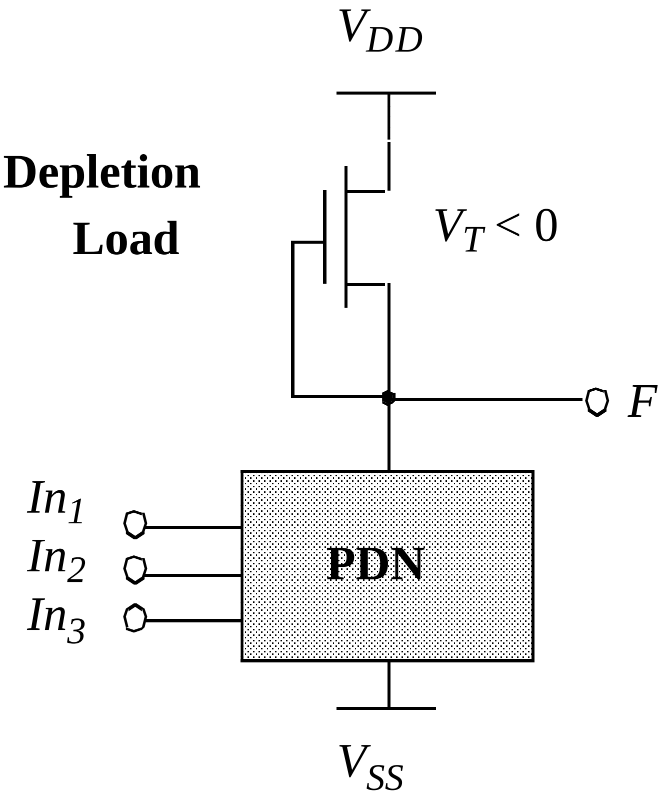
Goal: To reduce the number of devices over complementary CMOS

Ratioed Logic (2/2)

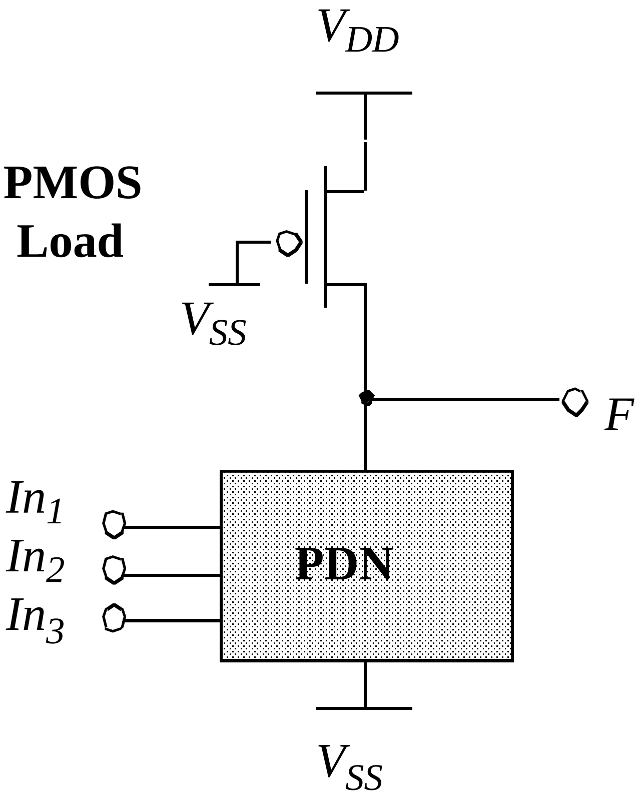


- N transistors + Load
- $V_{OH} = V_{DD}$
- $V_{OL} = \dfrac{R_{PN}}{R_{PN} + R_L} V_{DD}$
- Assymetrical response
- Static power consumption
- $t_{pL} = 0.69\ R_L C_L$

Active Loads

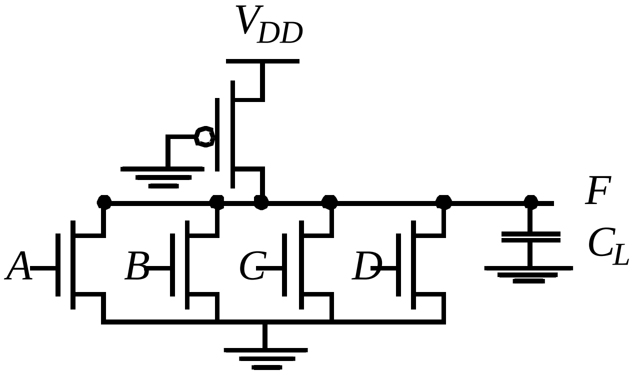


depletion load NMOS



pseudo-NMOS

Pseudo-NMOS



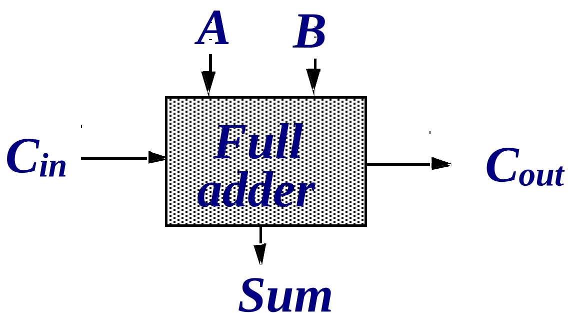
V_{OH} = V_{DD} (similar to complementary CMOS)

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

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

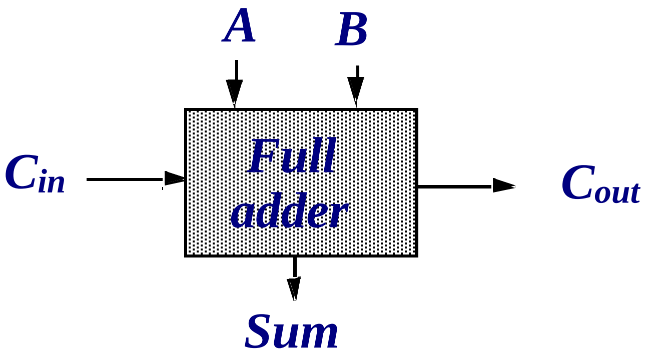
Smaller area and load but Static power dissipation!!!

CMOS Full-Adder Circuit



A	B	C_{in}	S	C_{out}	*Carry status*
0	0	0	0	0	delete
0	0	1	1	0	delete
0	1	0	1	0	propagate
0	1	1	0	1	propagate
1	0	0	1	0	propagate
1	0	1	0	1	propagate
1	1	0	0	1	generate
1	1	1	1	1	generate

CMOS Full-Adder Circuit
The Binary Adder



$$Sum = A \oplus B \oplus C_{in}$$

$$= ABC_{in} + A\overline{B}\,\overline{C}_{in} + \overline{A}\,\overline{B}C_{in} + \overline{A}B\overline{C}_{in}$$

$$= ABC + (A+B+C)\overline{C}_{out}$$

at least two of ***A***, ***B***, and ***C*** are zeros

$$C_{out} = AB + BC_{in} + AC_{in}$$

CMOS Full-Adder Circuit
Express Sum and Carry as a Function of P, G, D

- Define three new variable which ONLY depend on ***A***, ***B***

  Generate (***G***) = ***AB***

  Propagate (***P***) = $A \oplus B$

  Delete (***D***)= $\overline{A}\,\overline{B}$

$$C_{out}(G,P) = G + PC_{in}$$

$$Sum(G,P) = P \oplus C_{in}$$

- Can also derive expressions for ***S*** and C_{out} based on ***D*** and ***P***.

  G = 1: Ensure that the carry bit will be ***generated***

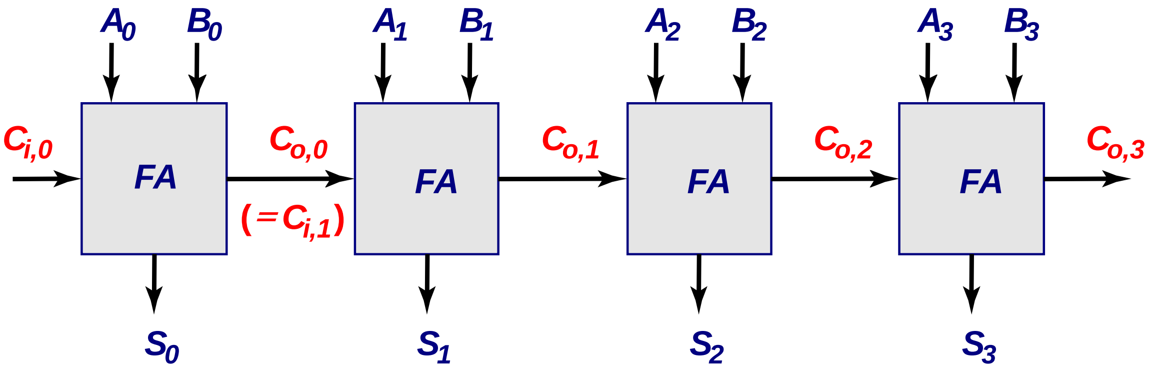
  D = 1: Ensure that the carry bit will be ***deleted***

  P = 1: Guarantee that an incoming carry will be ***propagated*** to C_{out}

- Note that ***G***, ***P*** and ***D*** are only functions of ***A*** and ***B*** and are not dependent on C_{in}

CMOS Full-Adder Circuit
The Ripple-Carry Adder

- The ***N***-bit adder is constructed by cascading ***N*** full-adder circuits.
- The carry bit ***ripples*** from one stage to the other.
- The delay through the circuit depends upon the number of logic stages which need to be traversed, and is a function of the applied signals.



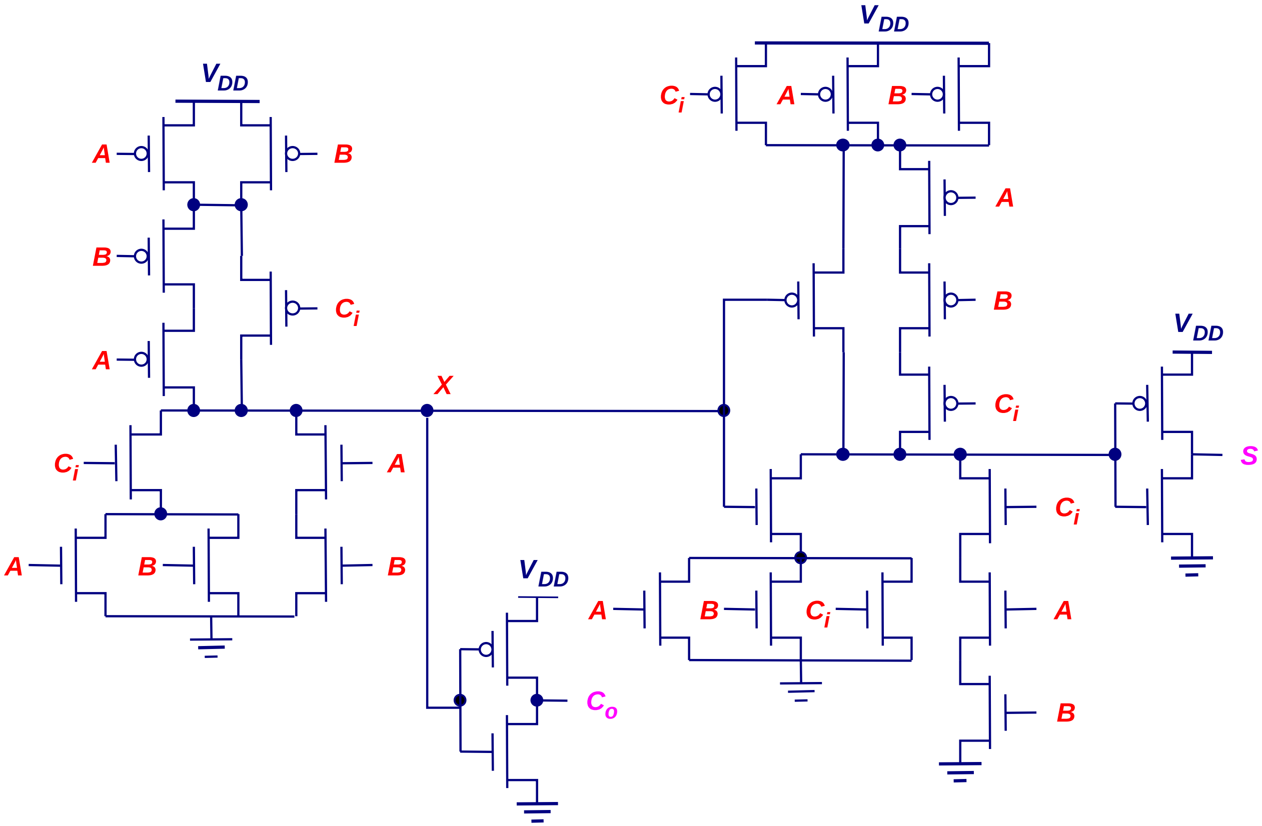
Worst case delay linear with the number of bits

$$\tau_p = O(N)$$

$$\tau_{adder} \approx (N-1)\tau_{carry} + \tau_{sum}$$

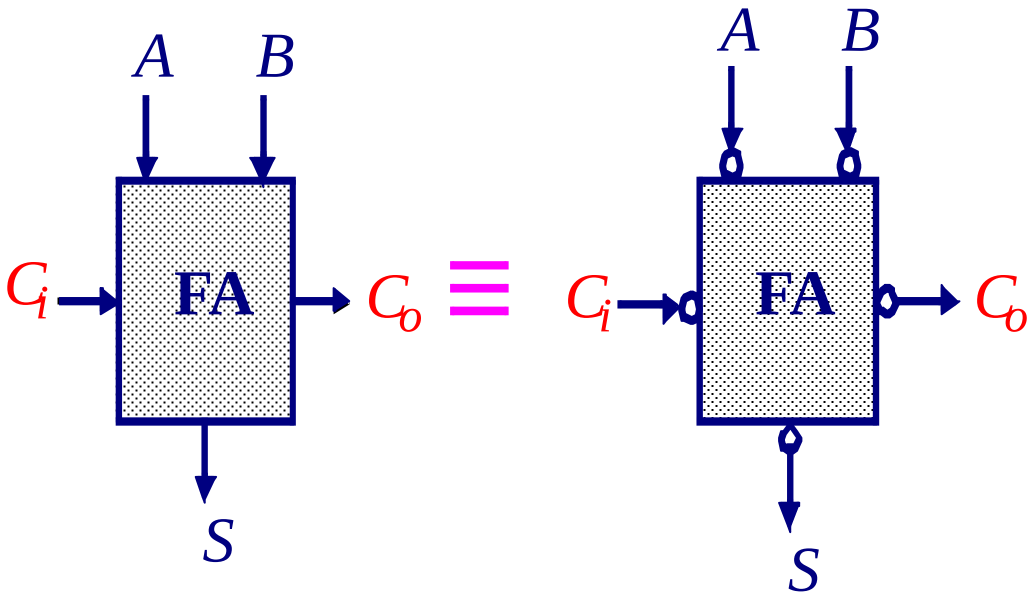
Goal: Make the fastest possible carry path circuit

CMOS Full-Adder Circuit
Transistor-Level of One-Bit Full-Adder Circuit



28 transistors

CMOS Full-Adder Circuit
Inversion Property

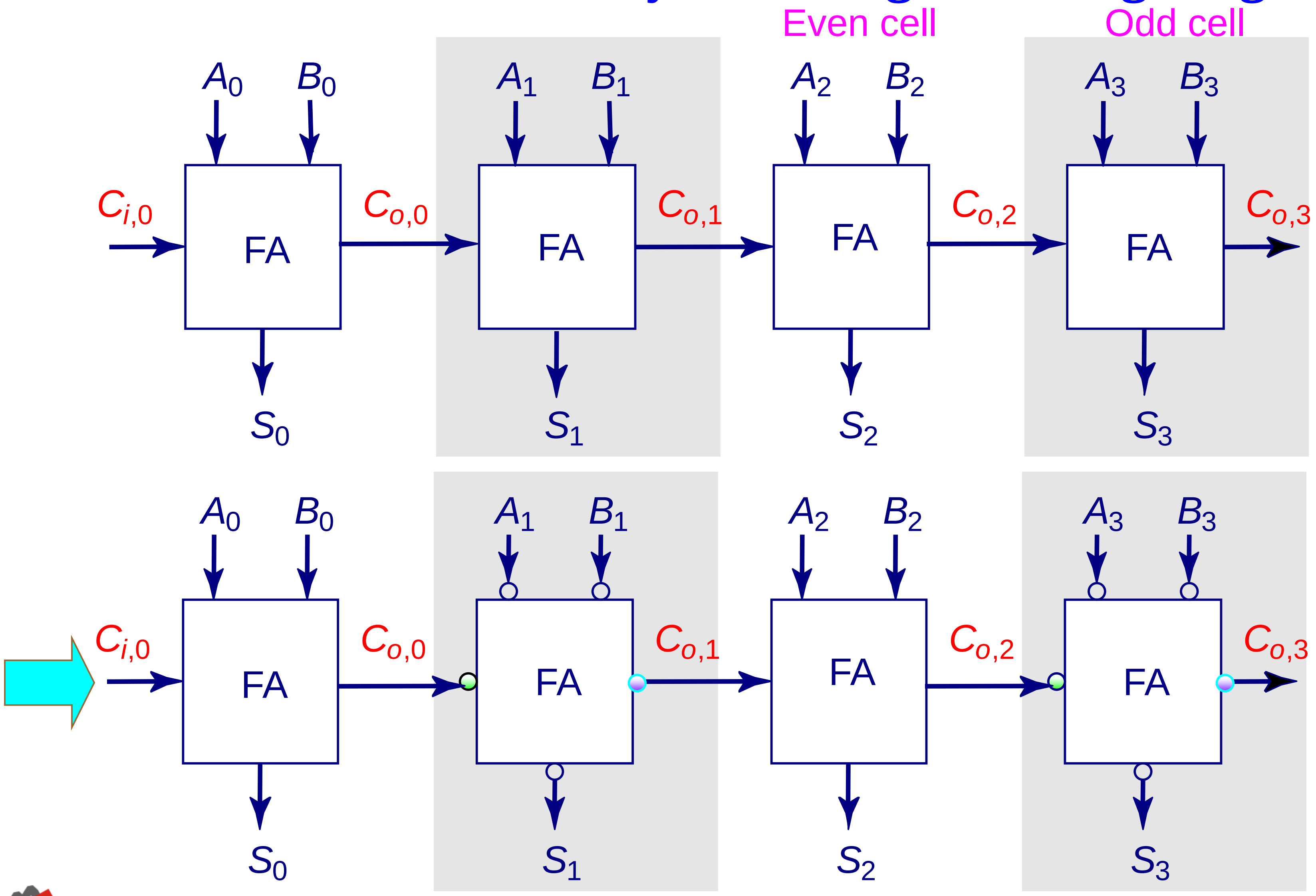


$$\overline{S}(A,B,C_i) = S(\overline{A},\overline{B},\overline{C_i})$$

$$\overline{C_o}(A,B,C_i) = C_o(\overline{A},\overline{B},\overline{C_i})$$

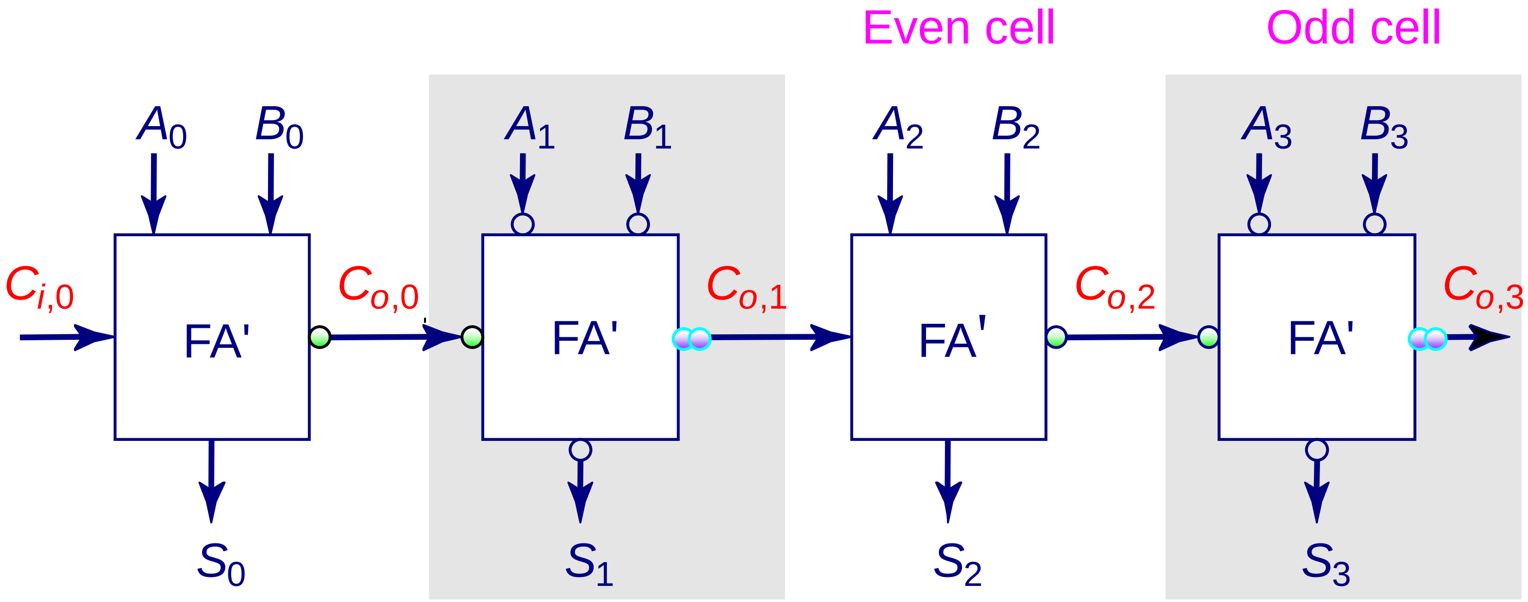
CMOS Full-Adder Circuit

Minimize Critical Path by Reducing Inverting Stages (*1*/2)



CMOS Full-Adder Circuit

Minimize Critical Path by Reducing Inverting Stages (2/2)



*FA' is a full adder without the inverter in the carry path.

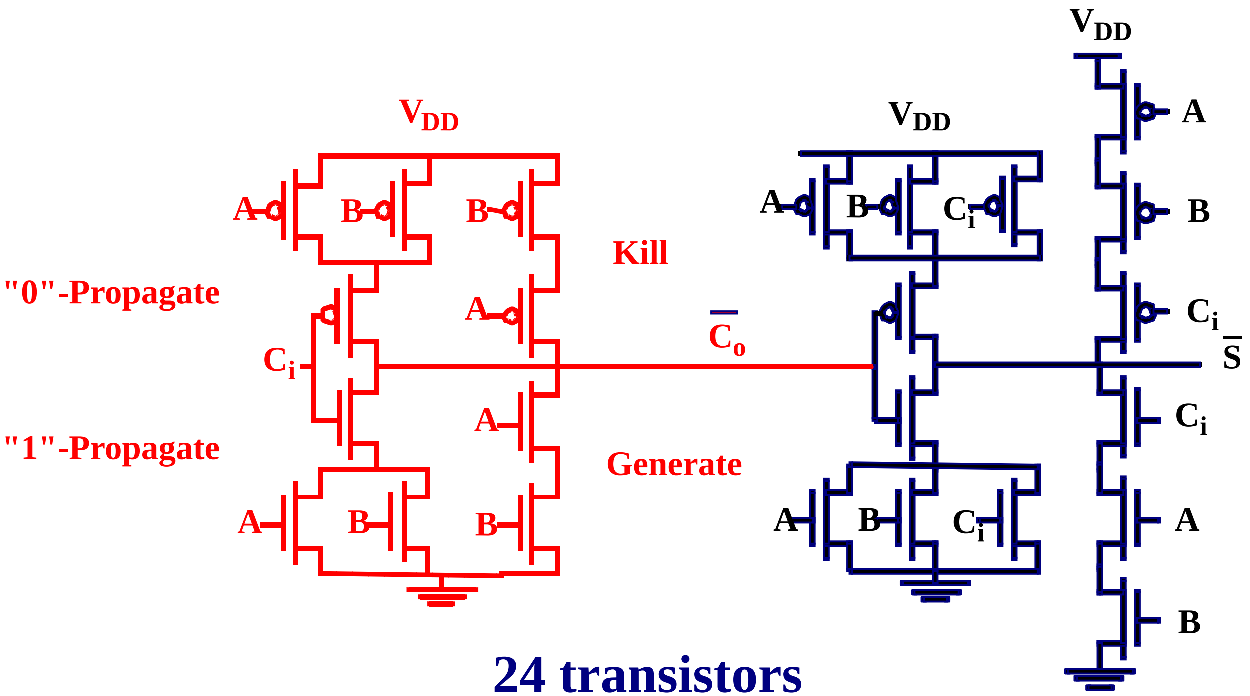
Exploit Inversion Property

- The number of inverting stages in the carry path is reduced.
- The only disadvantage is that it need different cells for the even and old slices.

CMOS Full-Adder Circuit
A Better Structure: The Mirror Adder (*1/3*)

- **Carry Generation Circuitry**
  - » Carry-inverting gate is eliminated
  - » **PDN** and **PUN** networks are not dual
    - ***D*** or ***G*** is high ⇒ $\overline{C_0}$ is set to ***V**_**DD***_ or ***GND***
    - ***P*** is high ⇒ the incoming carry is propagated to $\overline{C_0}$

$$C_{out}(G,P) = G+PC_{in}$$
$$Sum(G,P) = P\oplus C_{in}$$



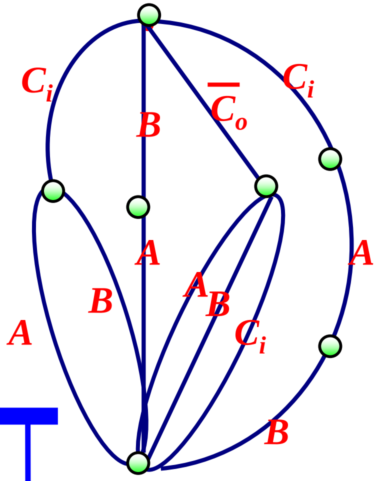
24 transistors

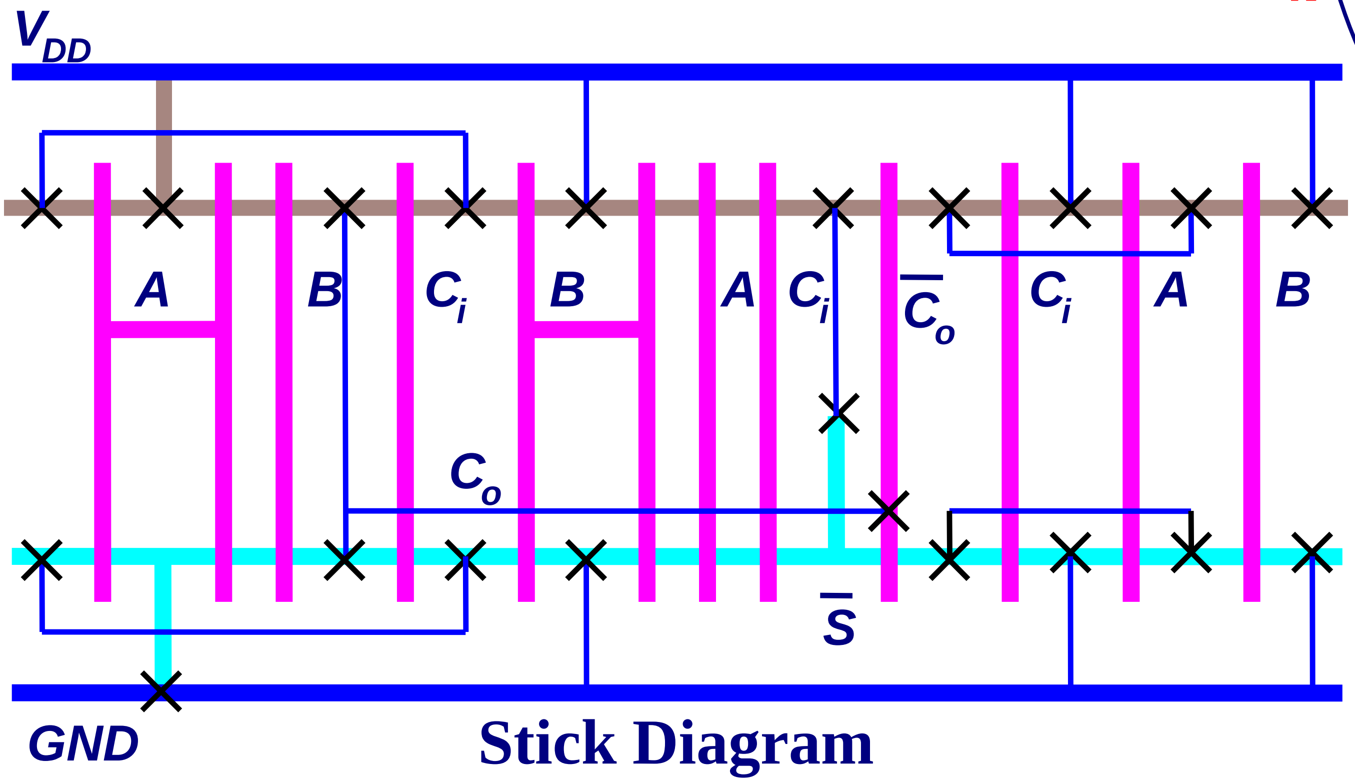
CMOS Full-Adder Circuit The Mirror Adder (2/3)

- Only need 24 transistors.
- NMOS and PMOS chains are completely symmetrical. This guarantees identical rising and falling time if the NMOS and PMOS devices are properly sized.
- A ***maximum*** of ***two series transistors*** can be observed in the carry generation circuitry.
- The critical issue is to minimize the capacitance at node $\overline{C_0}$.
- Capacitance at node $\overline{C_0}$
  - 4 diffusion capacitances
  - 2 internal gate capacitances
  - 6 gate capacitances in the connecting adder cell

  ⇒ A total 12 gate capacitances (Assume $C_{diffusion} \approx C_{gate}$)
- The transistors connected to C_i are placed closest to the output.
- Only the transistors in the carry stage have to be optimized for speed. All transistors in the sum gate can be minimum-size.

CMOS Full-Adder Circuit
The Mirror Adder (3/3)

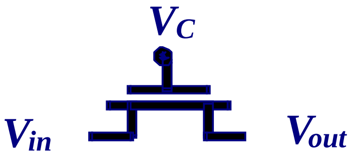


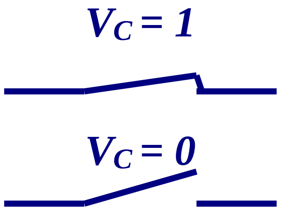


Stick Diagram

Pass Transistors

- The **pass transistor** is an **nFET** used as a switch-like element to connect logic and storage.





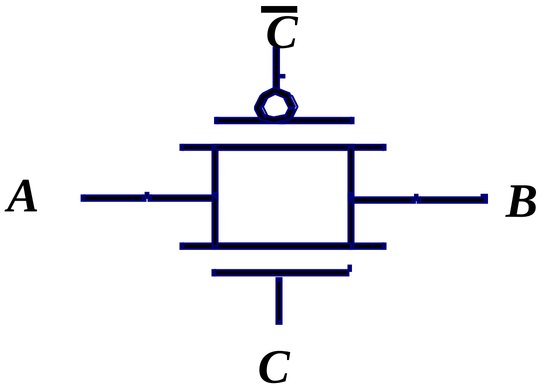
- Used in **NMOS**; sometimes used in **CMOS** to reduce cost.
- The voltage on the gate, V_C, determines whether the pass transistor is “open” or “closed” as a switch.
  - » If $V_C = H$, it is “closed” and connects V_{out} to V_{in}.
  - » If $V_C = L$, it is “open” and V_{out} is not connected to V_{in}.
- Consider $V_{in} = L$ and $V_{in} = H$ with $V_C = H$. With $V_{in} = L$, the pass transistor is much like a pull-down transistor in an inverter or **NAND** gate. So V_{out}, likewise, becomes L. But, for $V_{in} = H$, the output becomes the effective source of the **FET**. When $V_{GS} = V_{DD} - V_{OUT} = V_{Tn}$, the **nFET** cuts off. The **H** level is $V_{OUT} = V_{DD} - V_{Tn}$.

Transmission Gates (Pass Gates) (*1/2*)

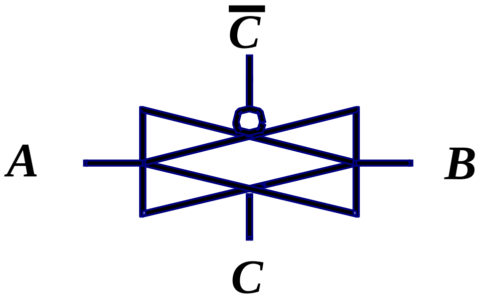
- With body effect, for V_{DD} = ***5***V, the value on V_{out} can be around ***3.0*** to ***3.5*** V. This reduced level diminishes NM_H and the current drive for the gate or gates driven by the pass transistor.
- For both **NMOS** and **CMOS**, the lack of current drive slows circuit operation and NM_H can be particularly problematic. As a consequence, in **CMOS**, a **pFET** is added to form a ***transmission gate***.

Transmission Gates

- Symbols:



Circuit



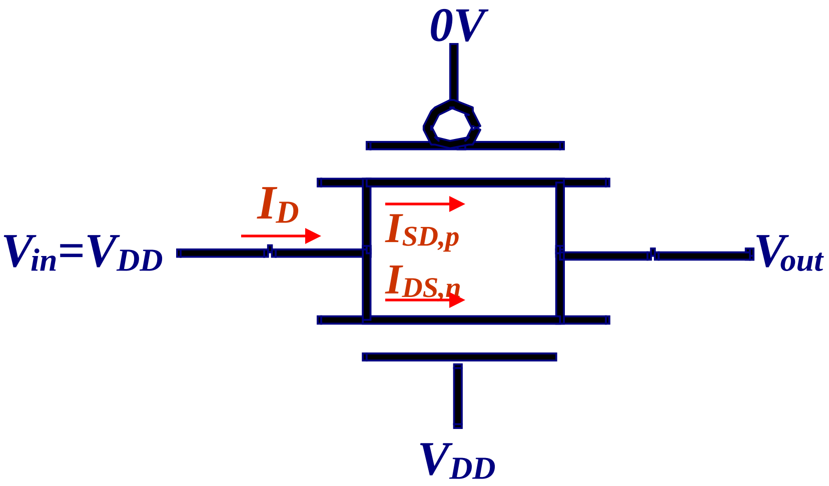
Popular Usage

Transmission Gates (2/2)

- **Operation**
  - » **C** is logic high ⇒ Both transistors are turned on and provide a low-resistance current path between nodes ***A*** and ***B***.
  - » **C** is logic low ⇒ Both transistors will be off, and the path between nodes ***A*** and ***B*** will be open circuit. This condition is called the *high-impedance state*.
- With the parallel **pFET** added, it can transfer a full V_{DD} from ***A*** to ***B*** (or ***B*** to ***A***). It can also charge driven capacitance faster.
- The substrates of **NMOS** and **PMOS** are connected to ***ground*** and V_{DD}, respectively. Therefore, the substrate-bias effect must be taken into account.

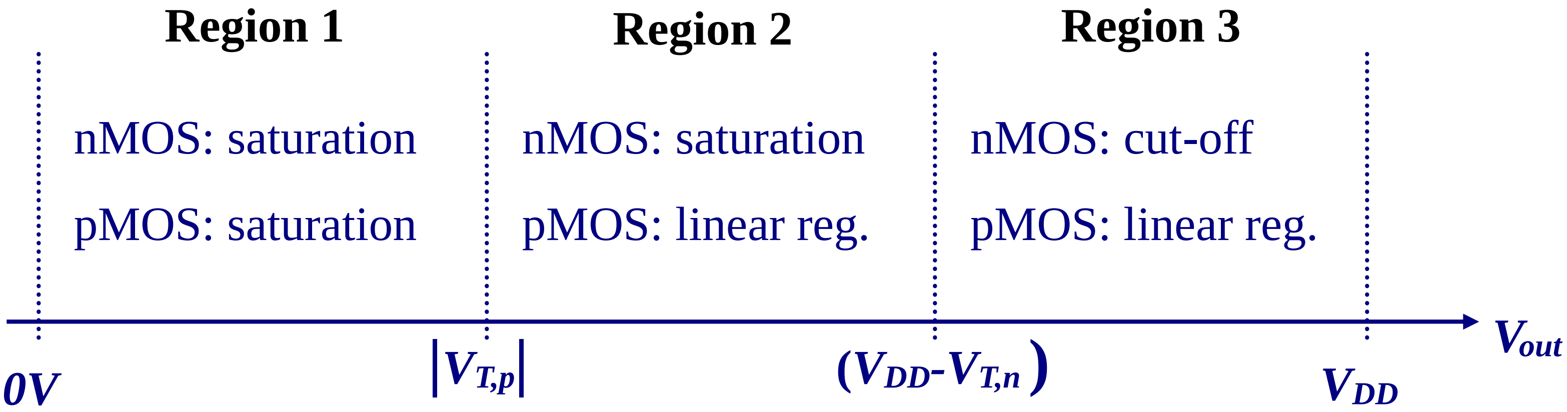
Transmission Gates
DC Analysis (1/3)

- $V_{in} = V_{DD}$, $V_C = V_{DD}$, and node B is connected to a capacitor, which represents capacitive loading of the subsequent logic stages.



- The **nMOS** transistor, $V_{DS,n}=V_{DD}–V_{out}$, and $V_{GS,n}=V_{DD}–V_{out}$. Thus,
  - » Turn off: If $V_{out} > V_{DD} – V_{T,n}$
  - » Saturation: If $V_{out} < V_{DD} – V_{T,n}$
- The **pMOS** transistor, $V_{DS,p}=V_{out}–V_{DD}$, and $V_{GS,p}= –V_{DD}$. Thus,
  - » Saturation: If $V_{out} < |V_{T,p}|$
  - » Linear: If $V_{out} > |V_{T,p}|$

Transmission Gates
DC Analysis (2/3)



Region 1	Region 2	Region 3
nMOS: saturation	nMOS: saturation	nMOS: cut-off
pMOS: saturation	pMOS: linear reg.	pMOS: linear reg.

- The current flowing through the transmission gate is equal to

$$I_D = I_{DS,n} + I_{SD,p}$$

- The *equivalent resistance* for each transistor can be represented as

$$R_{eq,n} = (V_{DD}-V_{out})/I_{DS,n}$$

$$R_{eq,p} = (V_{DD}-V_{out})/\ I_{DS,p}$$

and

$$R_{eq} = R_{eq,n} \parallel R_{eq,p}$$

Transmission Gates
DC Analysis (3/3)

The values of $R_{eq,n}$ and $R_{eq,p}$

- **Region 1**

$$R_{eq,n} = \frac{2(V_{DD} - V_{out})}{k_n\left(V_{DD} - V_{out} - V_{T,n}\right)^2}$$

$$R_{eq,p} = \frac{2(V_{DD} - V_{out})}{k_p\left(V_{DD} - |V_{T,p}|\right)^2}$$

- **Region 2**

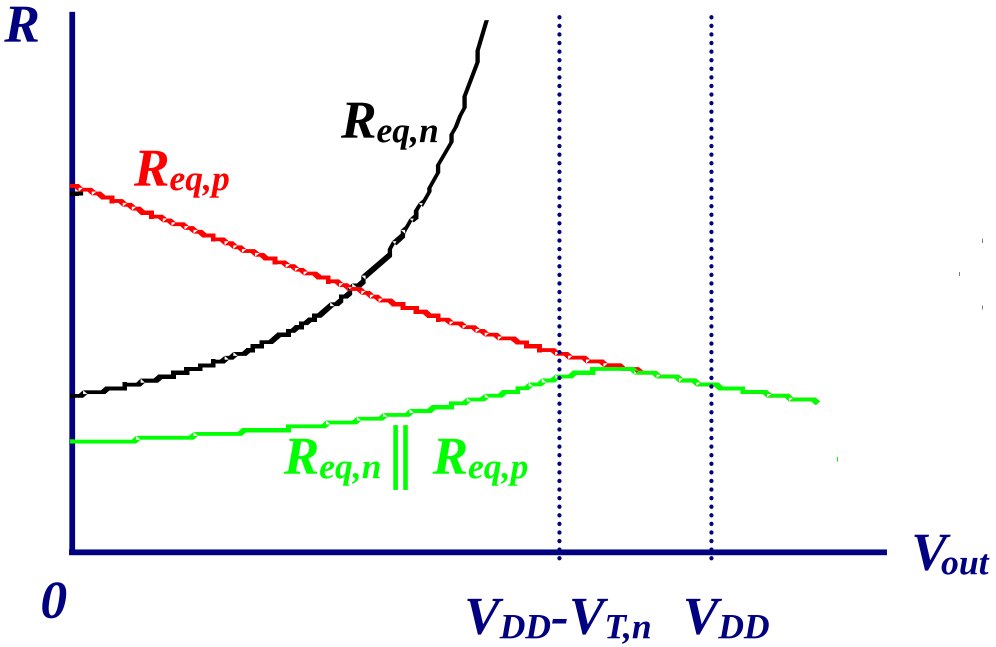
$$R_{eq,n} = \frac{2(V_{DD} - V_{out})}{k_n\left(V_{DD} - V_{out} - V_{T,n}\right)^2}$$

$$R_{eq,p} = \frac{2}{k_p\left[2(V_{DD} - |V_{T,p}|) - (V_{DD} - V_{out})\right]}$$

- **Region 3**

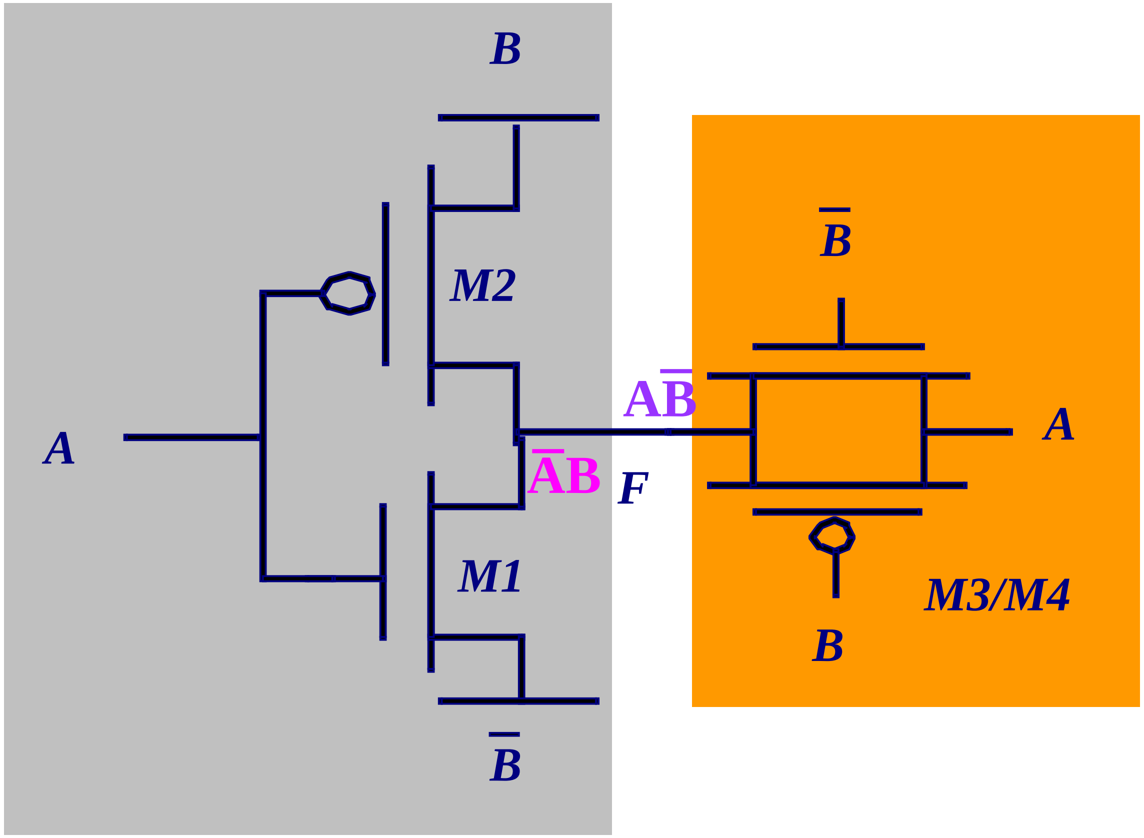
$$R_{eq,p} = \frac{2}{k_p\left[2(V_{DD} - |V_{T,p}|) - (V_{DD} - V_{out})\right]}$$

Resistance of Transmission Gate



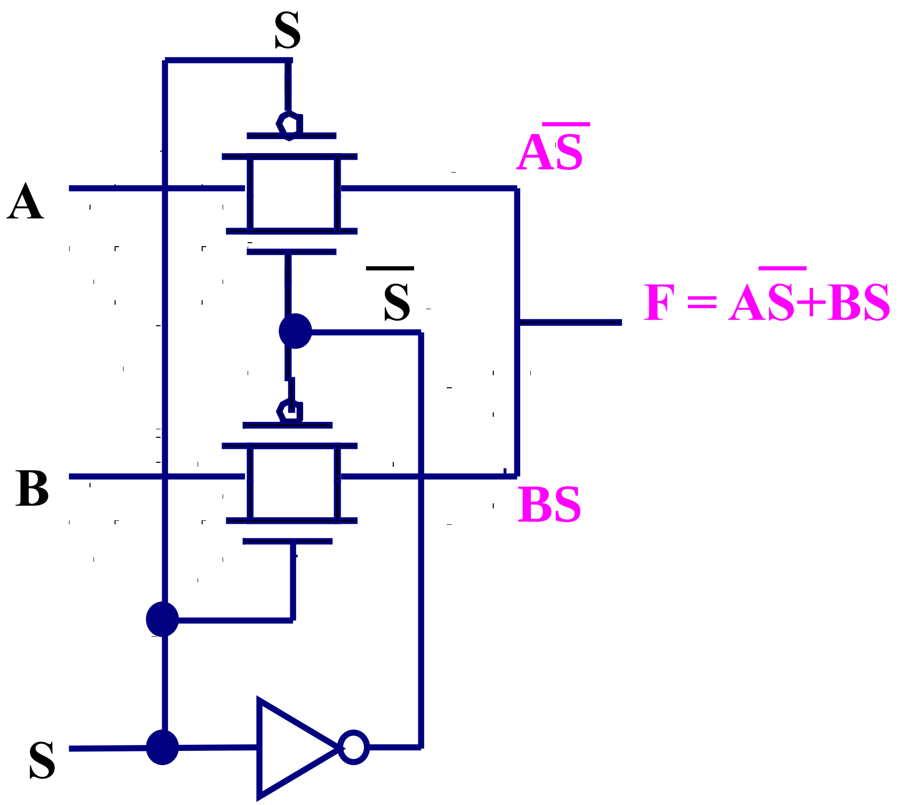
- The parallel combination of the pFET and the nFET result in an equivalent resistance that is **roughly constant**. This constant value, R_{eq}, can be used in series with an ideal switch controlled by $\overline{C}$ and C to model the transmission gate. See p. 311 of the text book.
- The implementation of CMOS transmission gates in logic circuit design usually results in compact circuit structures which may even **require a smaller number of transistors**.

Applications of Transmission Gate
Example: XOR



Only need 6 transistors

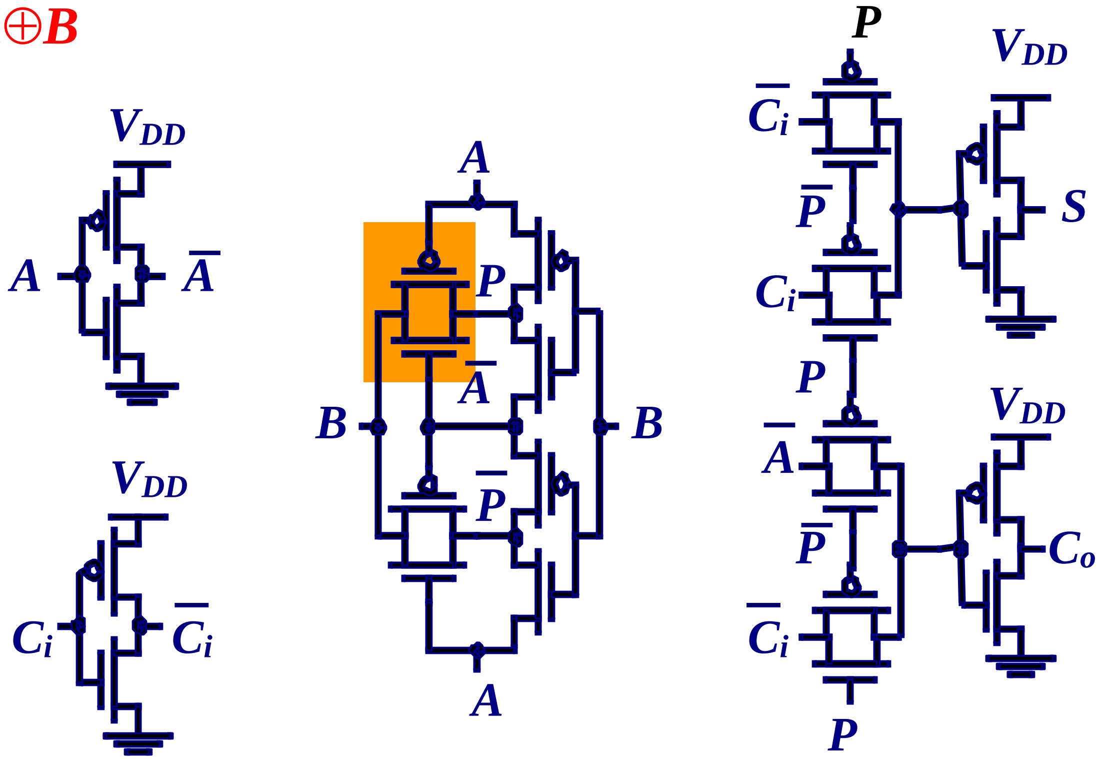
Applications of Transmission Gate
Example: Multiplexer



Applications of Transmission Gate
Examples: Transmission Gate Full Adder

Generate (*G*) = *AB*
Propagate (*P*) = *A*⊕*B*

$C_{out}(G,P) = G+PC_{in}$
$Sum(G,P) = P \oplus C_{in}$



Setup

Sum Generation

Carry Generation

Similar delays for sum and carry