Acknowledgement

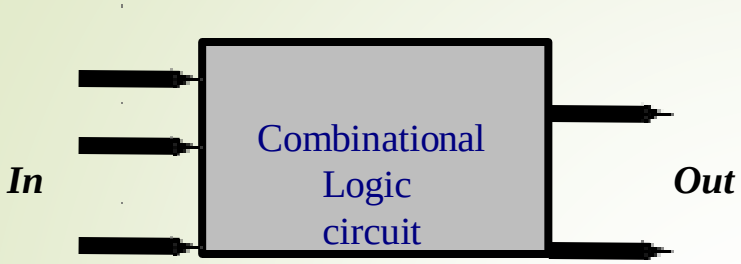
The slides used in this lecture are taken for academic purpose only.

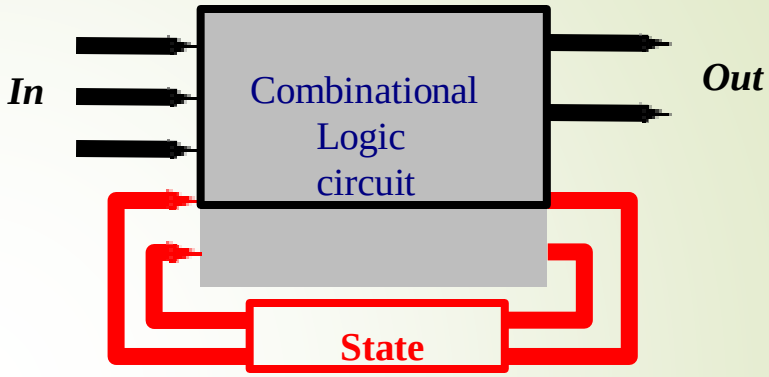
The resources are taken from:

1. CMOS VLSI Design: A Circuit and Systems Perspective, by Neil Weste, David Money Harris, 4th ed., Addison Wesley, 2011
2. CMOS Digital Integrated Circuits: Analysis and Design, Sung-Mo (Steve) Kang and Yusuf Leblebici, 2nd ed., McGraw Hill, 2003

Combinational CMOS Logic Circuits

Combinational vs. Sequential Logic





Combinational

The output is determined only by
- Current inputs

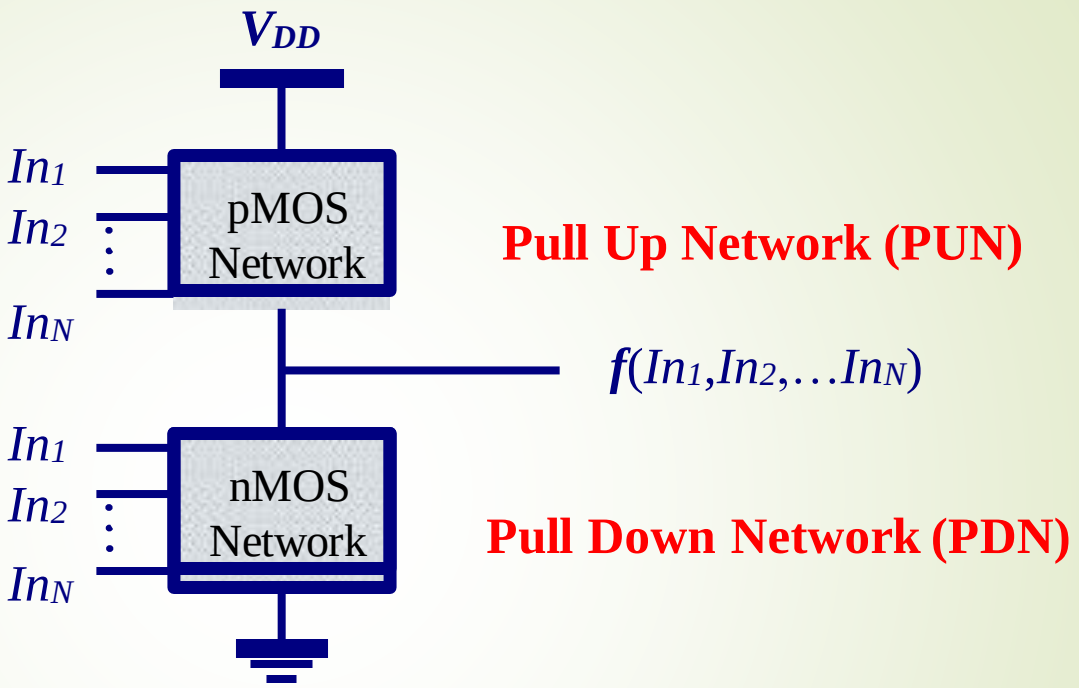
Output = *f*(*In*)

Sequential

The output is determined by
- Current inputs
- Previous inputs

Output = *f*(*In, Previous In*)

CMOS Combinational Logic

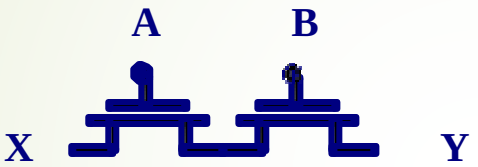


PUN and PDN are dual logic networks

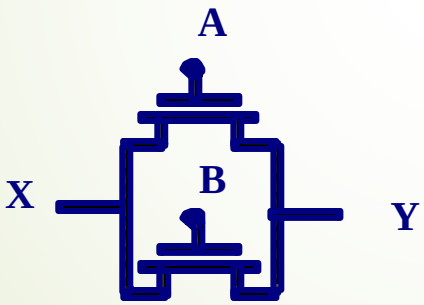
- **The complementary operation of a CMOS gate**
  - The nMOS network (PDN) is on and the pMOS network (PUN) is off
  - The pMOS network is on and the nMOS network is off.

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

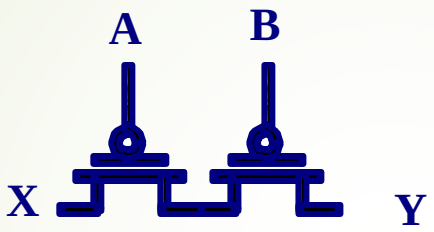


Y = X if A OR B=A+B

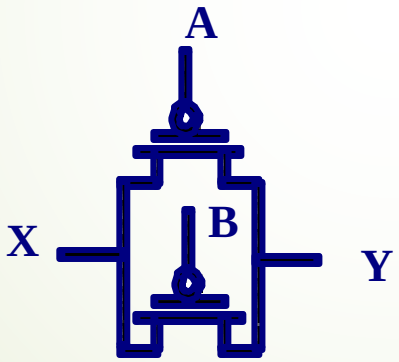
NMOS Transistors pass a “strong” 0 but a “weak” 1

PMOS Transistors Series/Parallel Connection

- PMOS switch closes when switch control input is low.



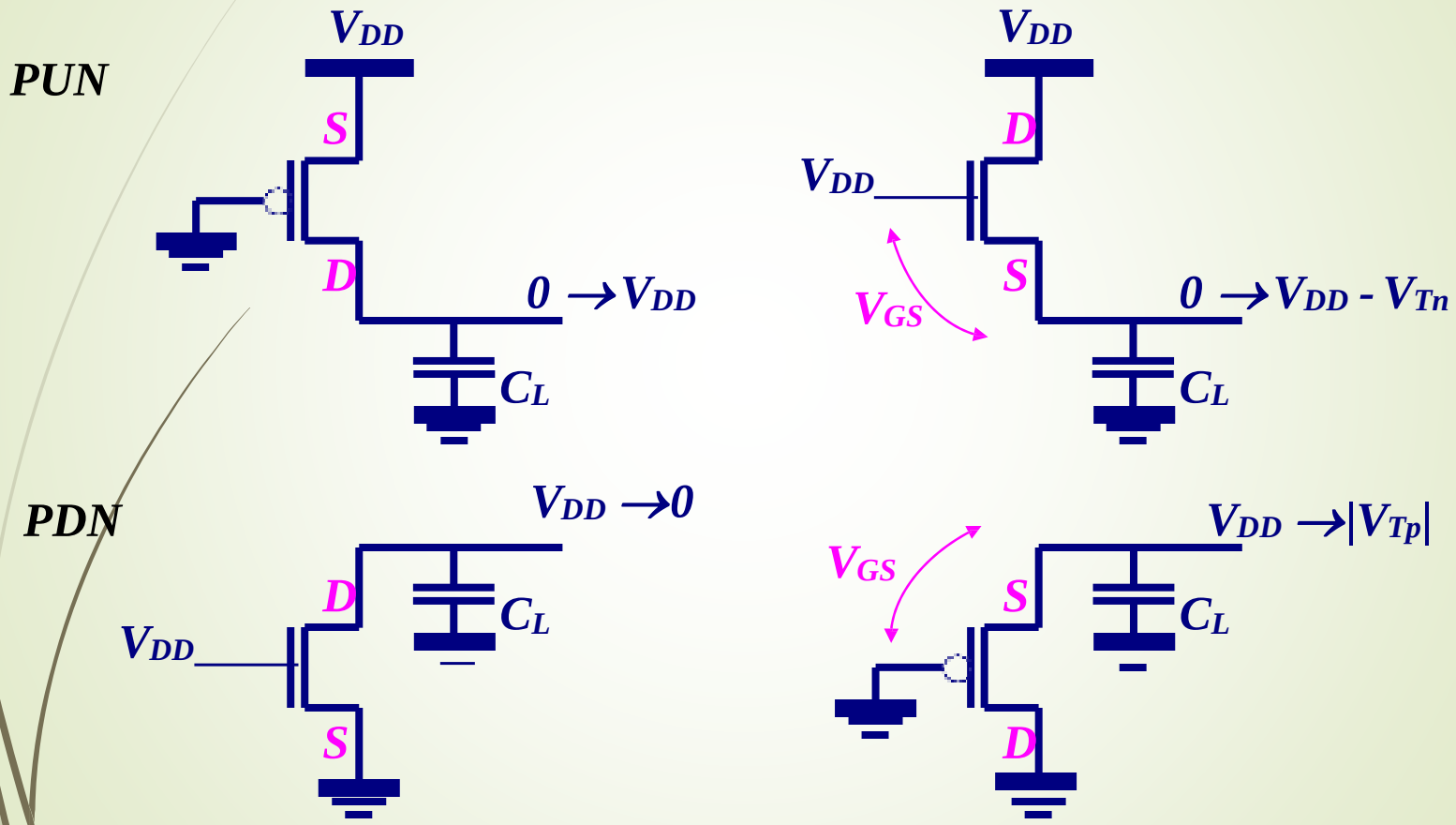
$Y = X$ if $\overline{A}$ AND $\overline{B} = \overline{A+B}$



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

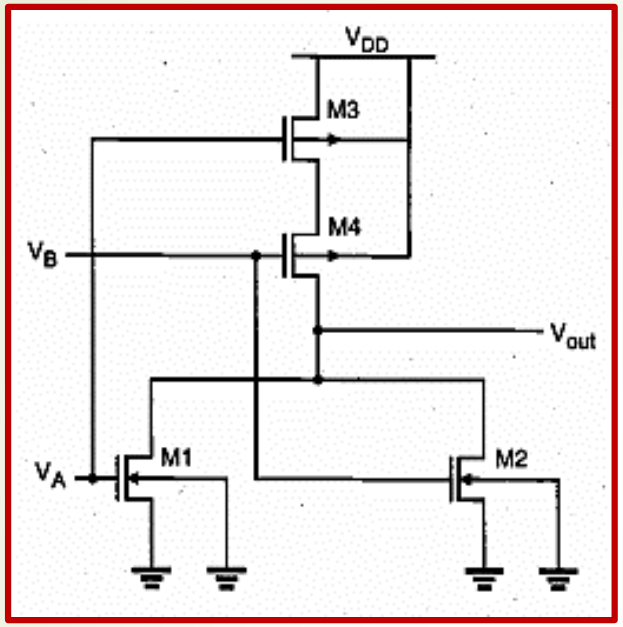
PMOS Transistors pass a "strong" 1 but a "weak" 0

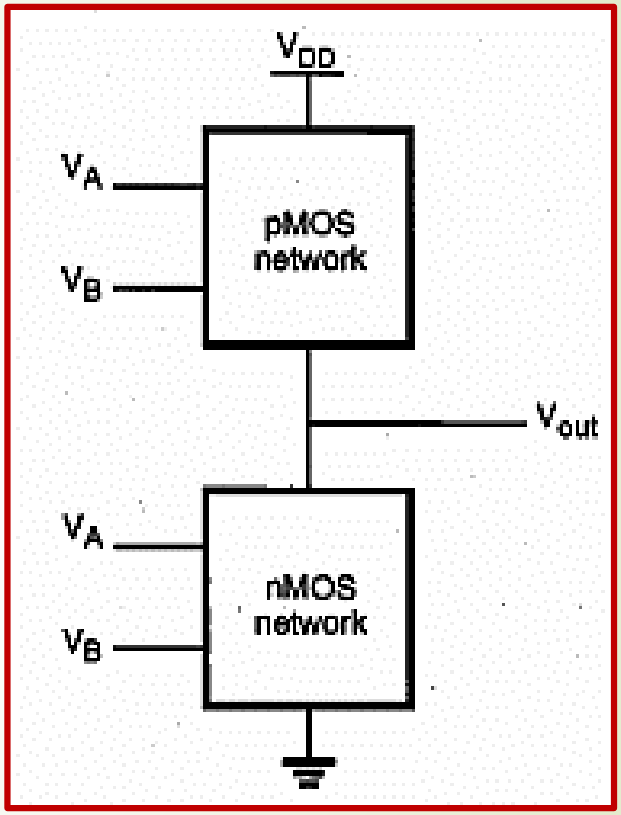
Threshold Drops

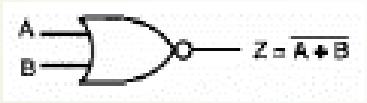


CMOS 2-Input NOR Gate

- The transistor level implementation for the NOR gate is:







V_A	V_B	V_{out}
low	low	high
low	high	low
high	low	low
high	high	lower

CMOS 2-Input NOR Gate

PMOS Pull-Up Network

- The only time the pull-up network drives the output is when we have two 0's on the inputs.
- Since the pull-up network uses PMOS transistors (0 = ON), we can say that the pull-up network is conducting if VA **AND** VB are 0.
- This implies a **series** configuration in the pull-up (PMOS) network.

NMOS Pull-Down Network

- The pull-down network drives the output continually unless VA AND VB are 0.
- Since the pull-down network uses NMOS transistors (1 = ON), we can say that the pull-down network is conducting if VA **OR** VB are 1.
- This implies a **parallel** configuration in the pull-down (NMOS) network.

CMOS 2-Input NOR Gate

- The critical voltages for an inverter Vth can be adjusted with the sizing of the transistors

- More complex logic gate can be analysed by converting it into an *equivalent inverter*

Transistors in Series

- The current flowing in series transistors needs to go through *two* channels, each with an equivalent resistance (or transconductance *k*)

- With the effective resistance doubling, the transconductance (or the ability to drive a current given an input voltage) is divided by 2

- Transistors in series with the same size can be modeled as an equivalent transistor with **keq=k/2**

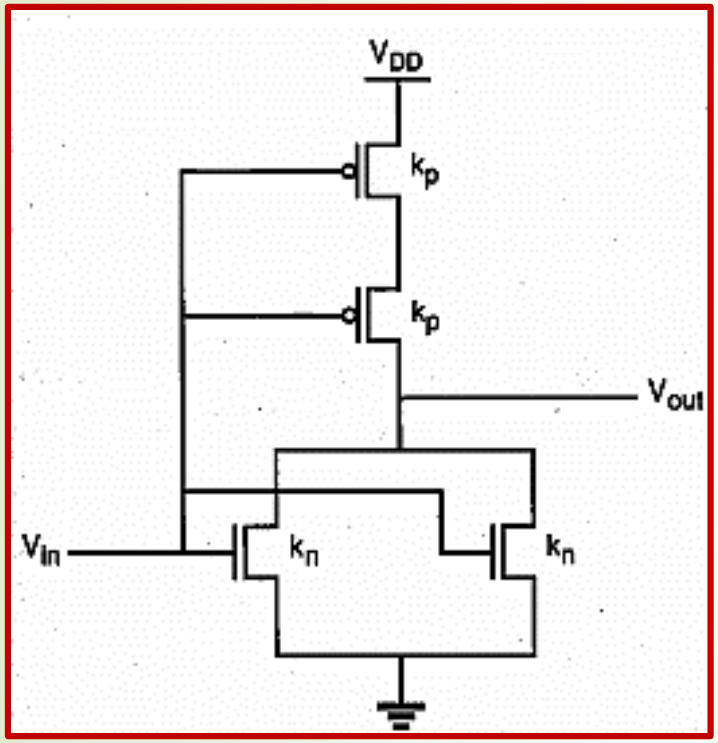
Transistors in Parallel

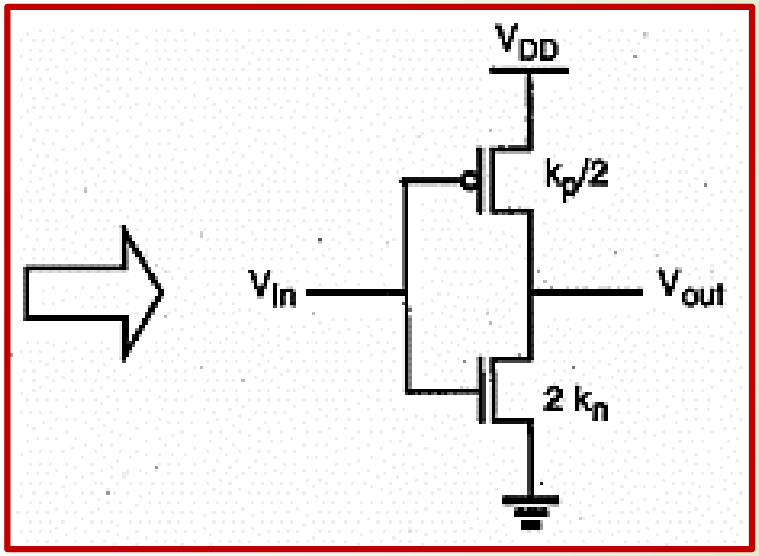
- The current flowing in parallel transistors can conduct twice the amount of current compared to a a single transistor with the same gate voltage.

- We can model this behavior with an equivalent transistor with **keq=2·k**

CMOS 2-Input NOR Gate

- We can model a 2-Input NOR gate as an equivalent inverter as follows:





- let's use representative voltages of VDD=5 V and Vth= 2.5 to illustrate the derivation

CMOS 2-Input NOR Gate

- we can derive the switching threshold by stating that:

$$V_{in} = V_{out} = V_A = V_B = V_{th}$$

- we can begin by writing the KCL equation at the V_{out} node:

$$\left|I_{D,n-network}\right| = \left|I_{D,p-network}\right|$$

- for the NMOS, since $V_{GS,n}=V_{DS,n}$, we know what both transistors are in *saturation*

- in addition, we can state that the current at the V_{out} node is the combination of both NMOS currents

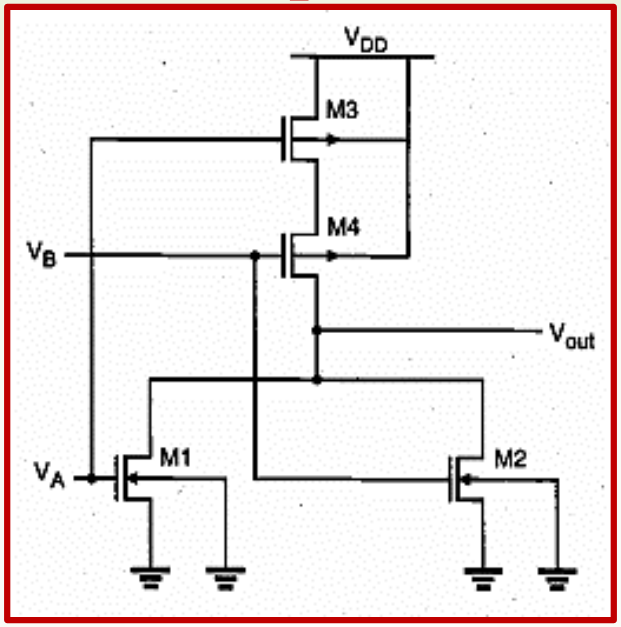
$$I_{D,n-network} = 2 \cdot I_{D,n|sat}$$

$$I_{D,n-network} = 2 \cdot \left[\frac{1}{2} \cdot k_n \cdot \left(V_{th} - V_{T,n}\right)^2\right] = k_n \cdot \left(V_{th} - V_{T,n}\right)^2$$

- rearranging this equation to get in terms of V_{th}, we have:

$$V_{th} = V_{T,n} + \sqrt{\frac{I_{D,n-network}}{k_n}}$$

CMOS 2-Input NOR Gate



- now we look at the PMOS network, which has a +2.5v drop across it.

- since both transistors are ON, we can estimate that ~1.25v drops across each transistor:

$$V_{DS,M3} = V_{DS,M4} = -1.25v$$

- looking at M3, we know that $V_{GS,p|M3}$= -2.5v and $V_{DS,p|M3}$= -1.25v so M3 is in the **linear region.**

- looking at M4, the node between M3 and M4 is estimated to be at ~3.75 (i.e., 5v – 1.25v)

- this puts $V_{DS,p|M4}$= -1.25v and $V_{GS,p|M4}$= -1.25v, which means M4 is in the ***saturation region.***

CMOS 2-Input NOR Gate

- since we know the regions of operation for M3 and M4, we can write:

$$I_{DS,3|lin} = \frac{1}{2} \cdot k_p \cdot \left[2 \cdot \left(V_{GS,p} - V_{T,p}\right) \cdot V_{DS,p} - V_{DS,p}^2\right]$$

- Since the PMOS current is expressed terms of I_{DS}, we can rewrite this as:

$$I_{DS,n} = I_{SD,p} = -I_{DS,p}$$

- we know that for M3, $V_{GS,p} = V_{th}-V_{DD}$. substituting this in and carrying the (-) through, in we get:

$$I_{SD,3|lin} = -I_{DS,3|lin} = -\frac{1}{2} \cdot k_p \cdot \left[2 \cdot \left(V_{th} - V_{DD} - V_{T,p}\right) \cdot V_{DS,p} - V_{DS,p}^2\right]$$

$$I_{SD,3|lin} = \frac{1}{2} \cdot k_p \cdot \left[2 \cdot \left(V_{DD} - V_{th} - \left|V_{T,p}\right|\right) \cdot V_{SD,p} - V_{SD,p}^2\right]$$

CMOS 2-Input NOR Gate

- M4 is in the *saturation region* so we can write the current as:

$$I_{DS,4|sat} = \frac{1}{2} \cdot k_p \cdot \left[\left(V_{GS,p} - V_{T,p} \right) \right]$$

- Again, the PMOS current can be rewritten as:

$$I_{DS,n} = I_{SD,p} = -I_{DS,p}$$

- we know that for M4:

$$V_{GS,p|M4} = V_{th} - (V_{DD} - V_{SD3})$$

$$V_{DS,p|M4} = V_{th} - (V_{DD} - V_{SD3})$$

- which gives the M4 current as:

$$I_{SD,4|sat} = -I_{DS,4|sat} = -\frac{1}{2} \cdot k_p \cdot \left(V_{th} - V_{DD} + V_{SD3} - V_{T,p} \right)^2$$

$$I_{SD,4|sat} = \frac{1}{2} \cdot k_p \cdot \left(V_{DD} - V_{th} - \left| V_{T,p} \right| - V_{SD3} \right)^2$$

CMOS 2-Input NOR Gate

- Now we can relate the drain currents knowing that $I_{D3}=I_{D4}=I_{D,n}$ giving a a 2nd equation relating V_{th} to I_D:

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

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

- combining this with our previous expression we get:

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

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

CMOS 2-Input NOR Gate

Now compare this expression with the switching threshold voltage of the CMOS inverter

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

If $k_n = k_p$ and $V_{T,n} = |V_{T,p}|$, the switching threshold of the CMOS inverter is equal to $V_{DD}/2$. Using the same parameters, the switching threshold of the NOR2 gate is

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

which is not equal to $V_{DD}/2$. For example, when $V_{DD} = 5$ V and $V_{T,n} = |V_{T,p}| = 1$ V, the switching threshold voltages of the NOR2 gate and the inverter are

$$V_{th}(\text{NOR2}) = 2 \text{ V}$$

$$V_{th}(\text{INR}) = 2.5 \text{ V}$$

CMOS 2-Input NOR Gate: Equivalent Inverter approach

- The switching threshold voltage of the NOR2 gate using the equivalent-inverter approach. When both inputs are identical, the parallel-connected nMOS transistors can be represented by a single nMOS transistor with $2k_n$. and the series-connected pMOS transistors are represented by a single pMOS transistor with $k_P/2$.

- in an equivalent inverter model, to get $V_{th}=V_{DD}/2$, we can use:

$$k_R = \frac{k_n}{k_p} = 1 = \frac{2 \cdot k_n}{k_p / 2}$$

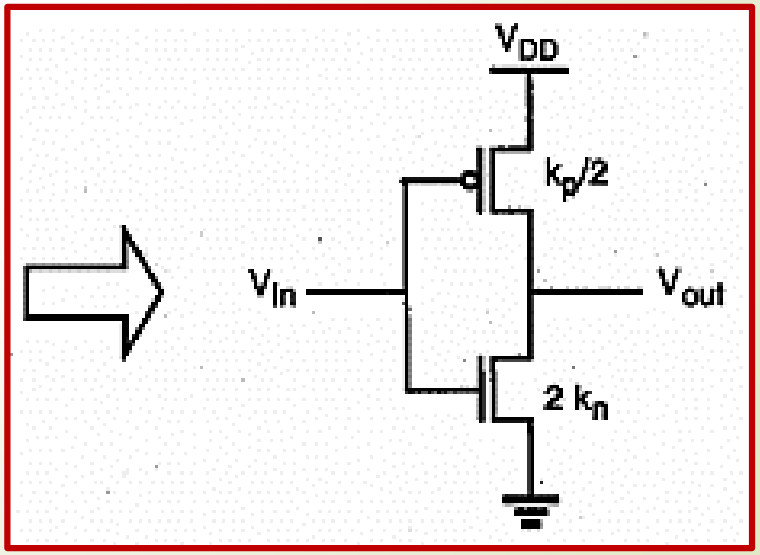
$$k_p = 4 \cdot k_n$$

- note that the PMOS series network has to be sized larger in order to overcome the voltage drop across each stage.

CMOS 2-Input NOR Gate: Equivalent Inverter approach

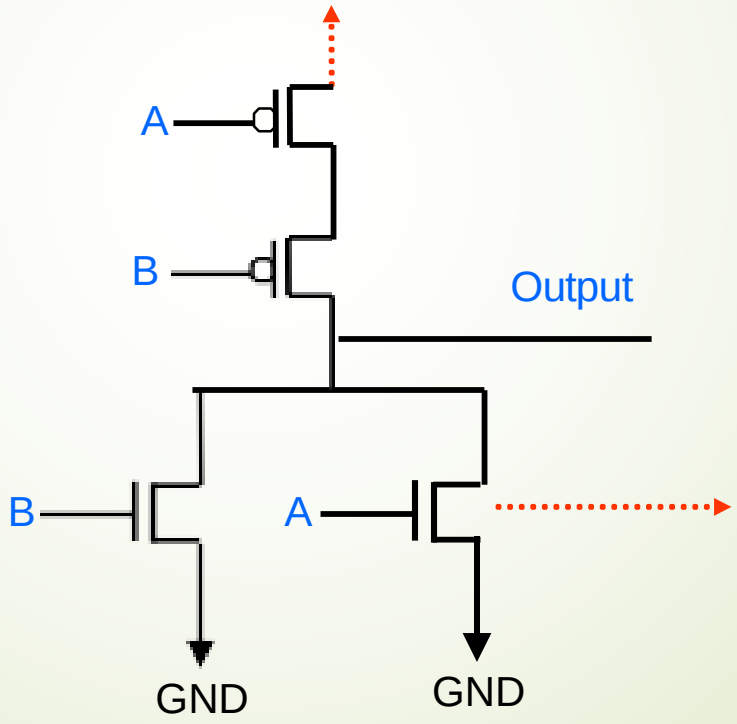
- Using the inverter switching threshold expression for the equivalent inverter circuit of NOR2 gate, we obtain

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



CMOS N-Input NOR Gate

- To expand the NOR gate to N-inputs,
  - Add more PMOS transistors in series in the Pull-up Network
  - Add more NMOS transistors in parallel in the Pull-down Network



CMOS N-Input NOR Gate

- Our V_{th} expression for an N-input NOR gate becomes:

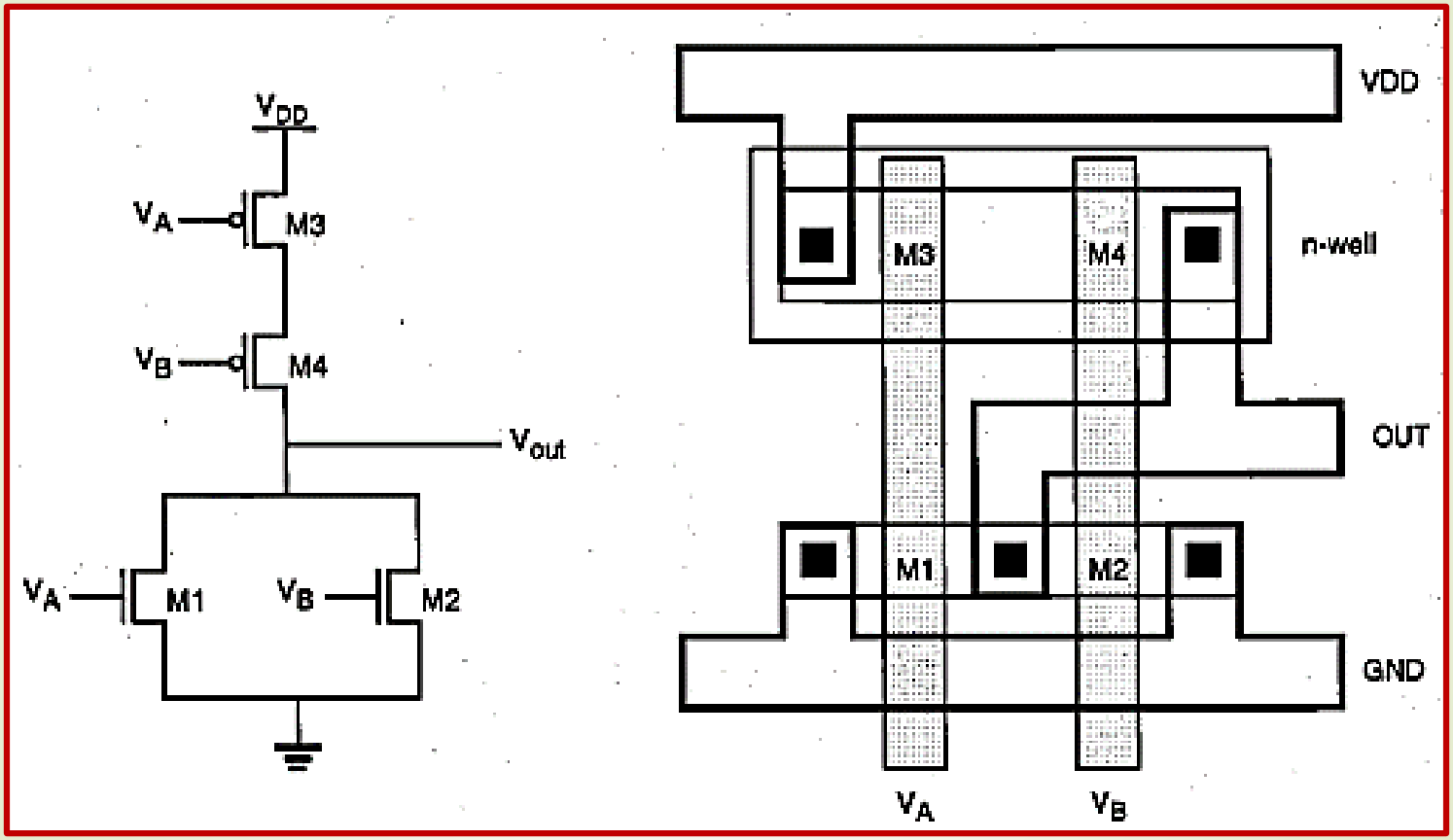
$$V_{th}(NOR) = \frac{V_{T,n} + \frac{1}{N}\sqrt{\frac{1}{k_R}}\left(V_{DD} - V_{T,p}\right)}{1 + \frac{1}{N}\sqrt{\frac{1}{k_R}}}$$

- The rule of thumb for an ideal symmetric equivalent inverter becomes:

$$k_R = \frac{k_n}{k_p} = 1 = \frac{N \cdot k_n}{k_p / N}$$

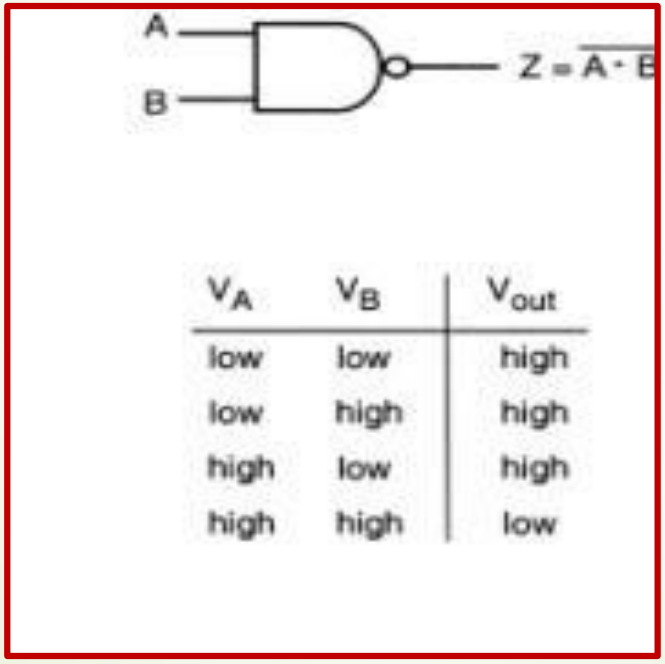
$$k_p = N^2 \cdot k_n$$

CMOS 2-Input NOR Gate Layout



CMOS 2-Input NAND Gate

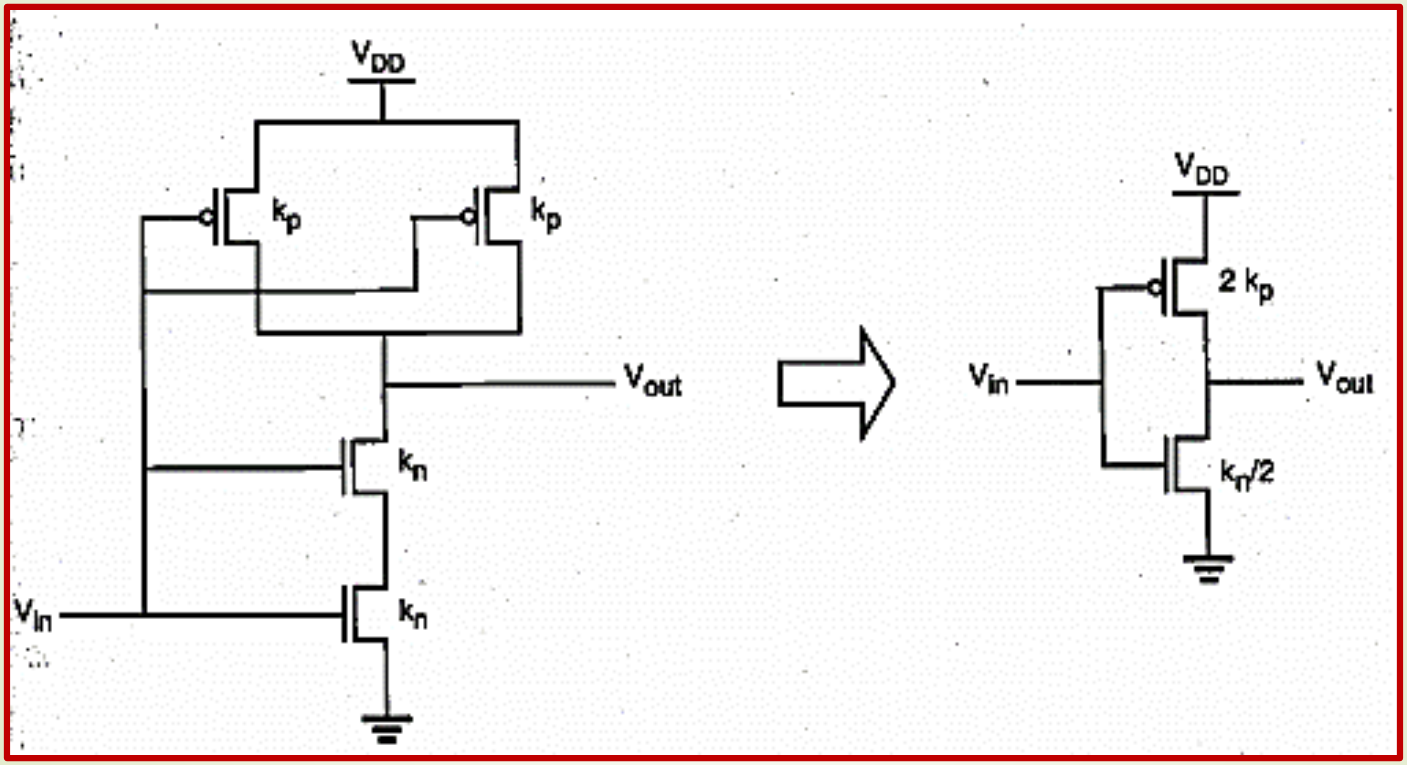
- The truth table for a 2-input NAND gate is:



V_A	V_B	V_{out}
low	low	high
low	high	high
high	low	high
high	high	low

CMOS 2-Input NAND Gate

- We can model the 2-Input NAND gate as an equivalent invert as follows:



CMOS 2-Input NAND Gate

Using an analysis similar to the one developed for the NOR2 gate, we can easily calculate the switching threshold for the CMOS NAND2 gate. Again, we will assume that the device sizes in each block are identical, with $(W/L)_{n,A} = (W/L)_{n,B}$ and $(W/L)_{p,A} = (W/L)_{p,B}$. The switching threshold for this gate is then found as

- The switching threshold for the 2-input NAND gate is:

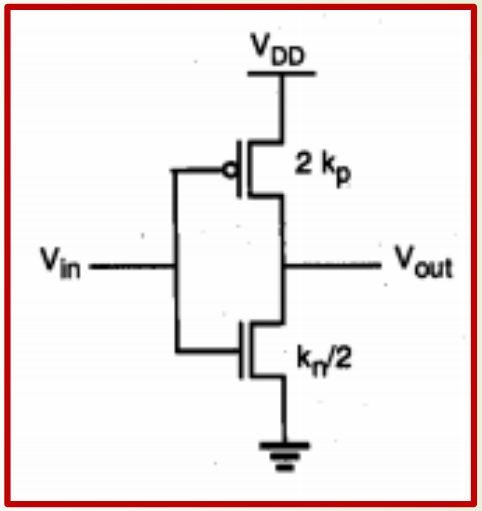
$$V_{th}(\text{NAND2}) = \frac{V_{T,n} + 2\sqrt{\frac{k_p}{k_n}}\left(V_{DD} - |V_{T,p}|\right)}{1 + 2\sqrt{\frac{k_p}{k_n}}}$$

CMOS 2-Input NAND Gate

- In an equivalent inverter model, to get $\mathbf{V_{th}=V_{DD}/2}$, we set $V_{T,n} = |V_{TP}|$:

$$k_R = \frac{k_n}{k_p} = 1 = \frac{k_n/2}{2 \cdot k_p}$$

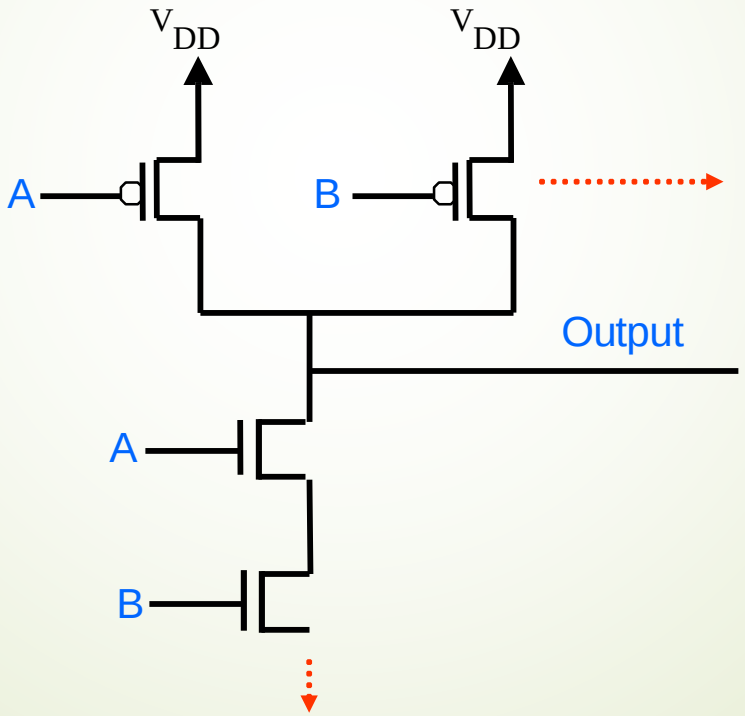
$$4 \cdot k_p = k_n$$



- The NMOS series network has to be sized larger in order to overcome the voltage drop across each series stage.

CMOS N-Input NAND Gate

- To expand the NAND gate, we
  - Add more PMOS transistors in parallel in the Pull-up Network
  - Add more NMOS transistors in series in the Pull-down Network



CMOS N-Input NAND Gate

- Our V_{th} expression for an N-input NAND gate becomes:

$$V_{th}(NAND) = \frac{V_{T,n} + N\sqrt{\frac{1}{k_R}}\left(V_{DD} - |V_{T,p}|\right)}{1 + N\sqrt{\frac{1}{k_R}}}$$

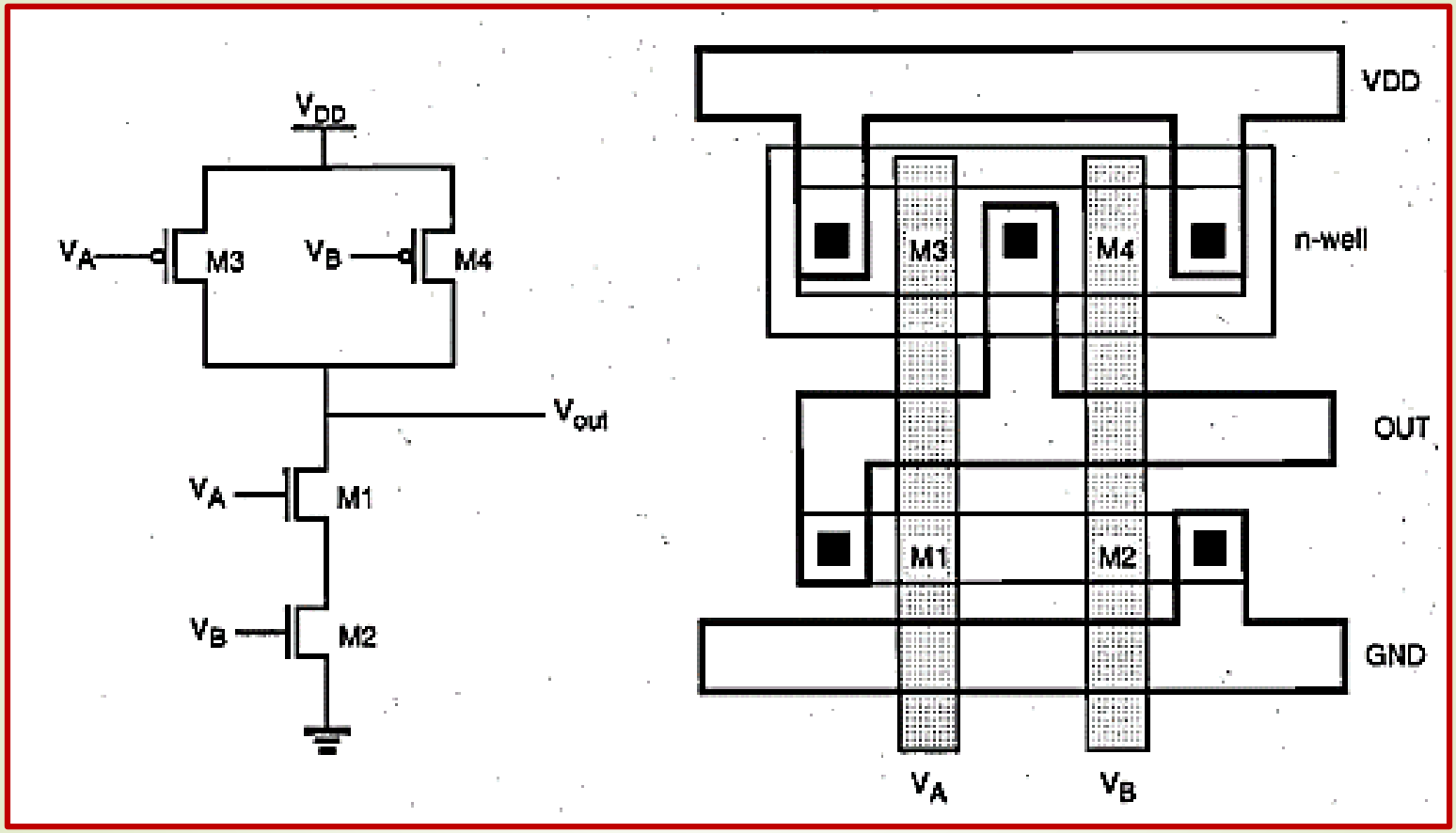
$$V_{th}(NAND) = \frac{V_{T,n} + \sqrt{\frac{N^2}{k_R}}\left(V_{DD} - |V_{T,p}|\right)}{1 + \sqrt{\frac{N^2}{k_R}}}$$

- The rule of thumb for an ideal symmetric equivalent inverter becomes:

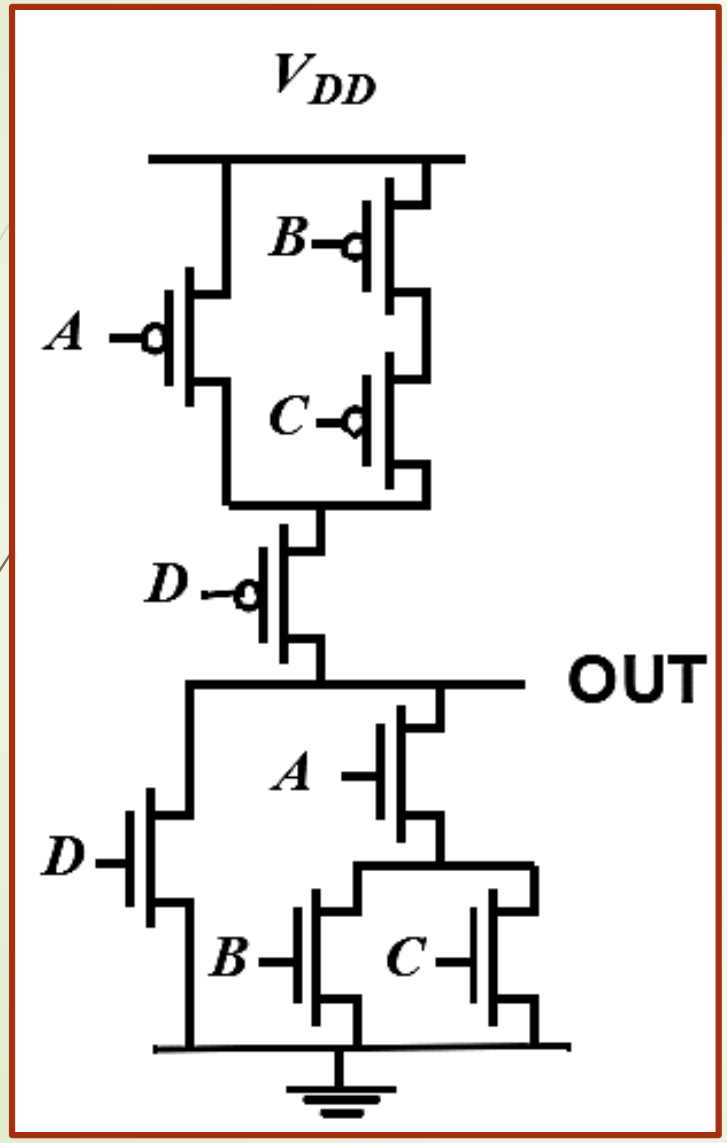
$$k_R = \frac{k_n}{k_p} = 1 = \frac{k_n / N}{N \cdot k_p}$$

$$N^2 \cdot k_p = k_n$$

CMOS 2-Input NAND Gate Layout

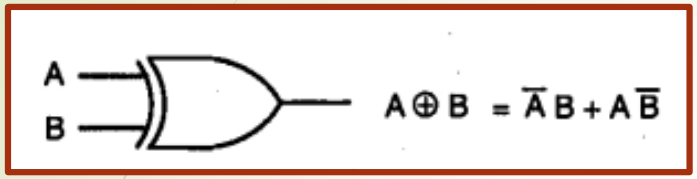


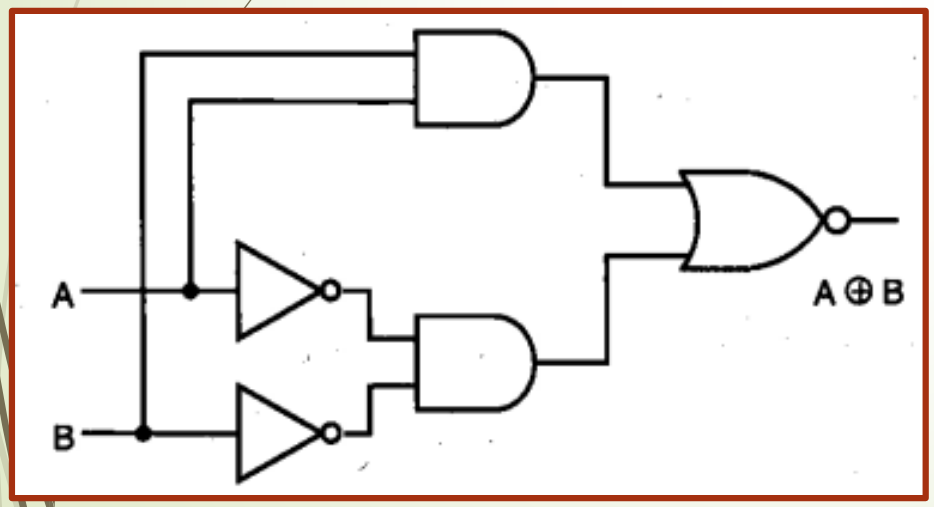
Complex CMOS Logic Circuits

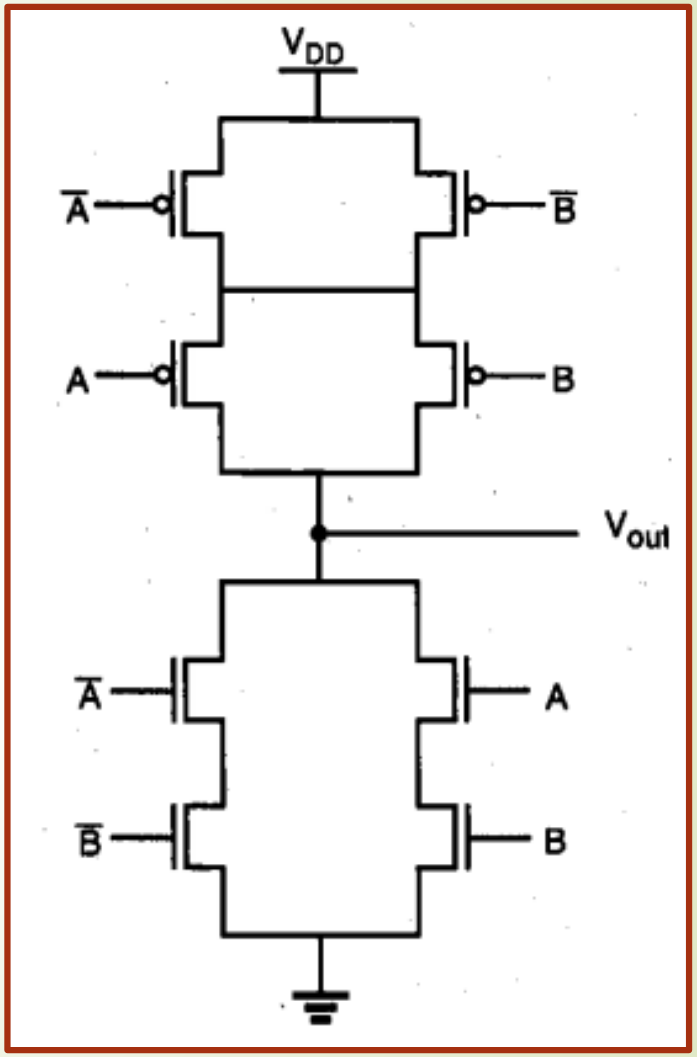


$$OUT = \overline{D + A \cdot (B+C)}$$

Full-CMOS implementation of the XOR Gate function







Complex CMOS Logic Circuits

- An example of the NMOS pull-down network synthesis is:

$$F = \overline{A \cdot (D + E) + B \cdot C}$$

  - D+E is created with two NMOS's in parallel
  - A(D+E) puts an NMOS in series with the (D+E) network
  - The entire A(D+E) network is in parallel with the BC network
  - The BC network is created with two series NMOS's

Complex CMOS Logic Circuits

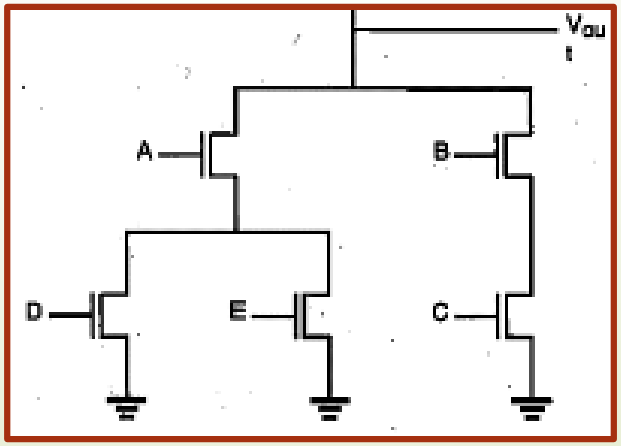
❖ Complex CMOS Logic Graphs

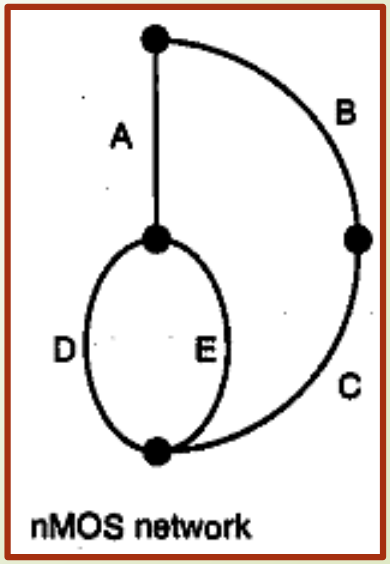
- ❖ The complementary PMOS network is created with a technique called a ***dual pull-up/down graph***
- ❖ A graphical way to create the PMOS network for a given NMOS pull-down circuit.
- ❖ First create the ***pull-down*** *graph* by representing:

  1) each NMOS transistor as an **edge** (i.e., a line), and

  2) each node as a **vertex** (i.e., a dot)

- Orient the *pull-down graph* in the same orientation as the NMOS circuit (Vout on top, VSS on bottom)





Complex CMOS Logic Circuits

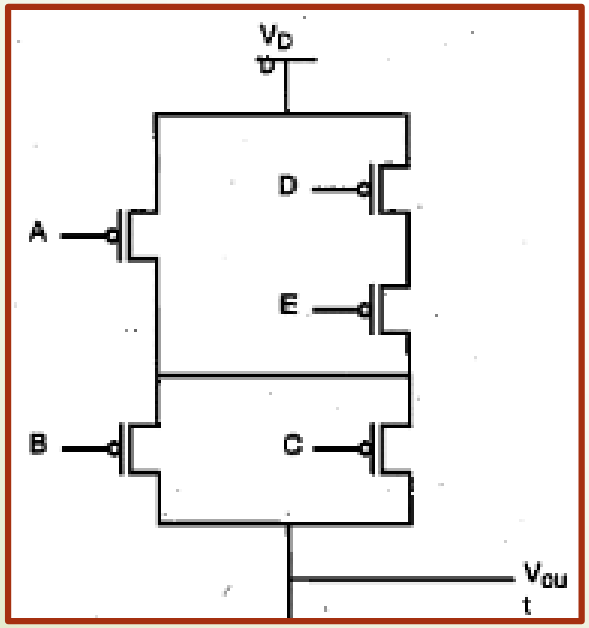
- **Complex CMOS Logic Graphs**

- Next, create the ***dual pull-up graph*** on top of the *pull-down* graph.

- Orient the *pull-up* graph with VDD on the left and Vout on the right.

- Create the *pull-up* graph using the rules:

1) a new vertex is created within each confined area of the pull-down graph
2) each vertex is connected by an edge which crosses **each** edge of the *pull-down graph*



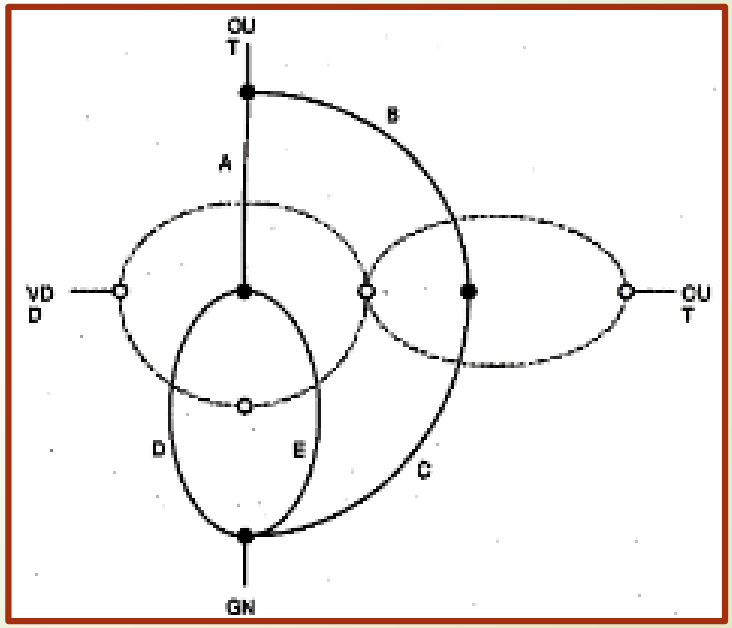


Fig: Construction of the dual pull-up graph from the pull-down graph, using the dual graph concept.

Complex CMOS Logic Circuits

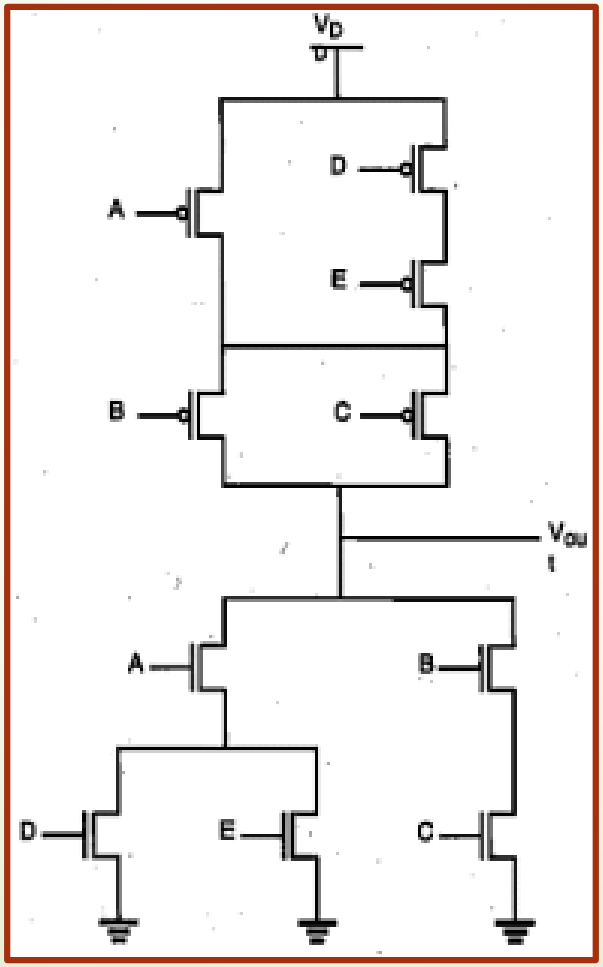
- **Complex CMOS Logic Graphs**

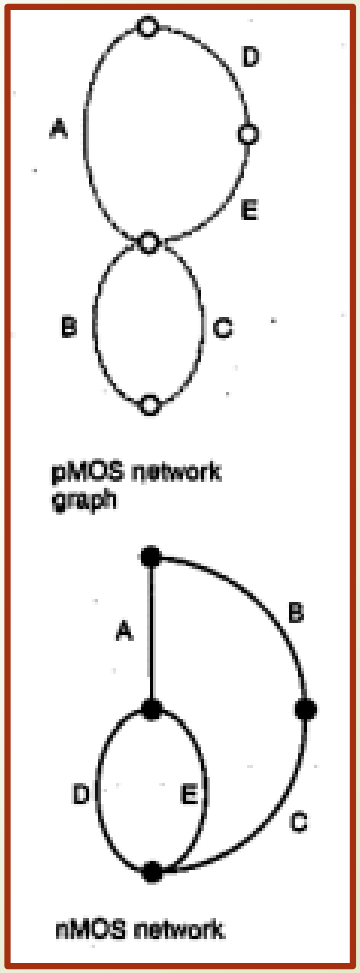
❖ We can now separate the graphs and synthesize the PMOS ***pull-up*** network

Remember that: **an edge (a line) = a transistor**

a vertex (a dot) = a node

$$F = \overline{A \cdot (D + E) + B \cdot C}$$





Complex CMOS Logic Circuits

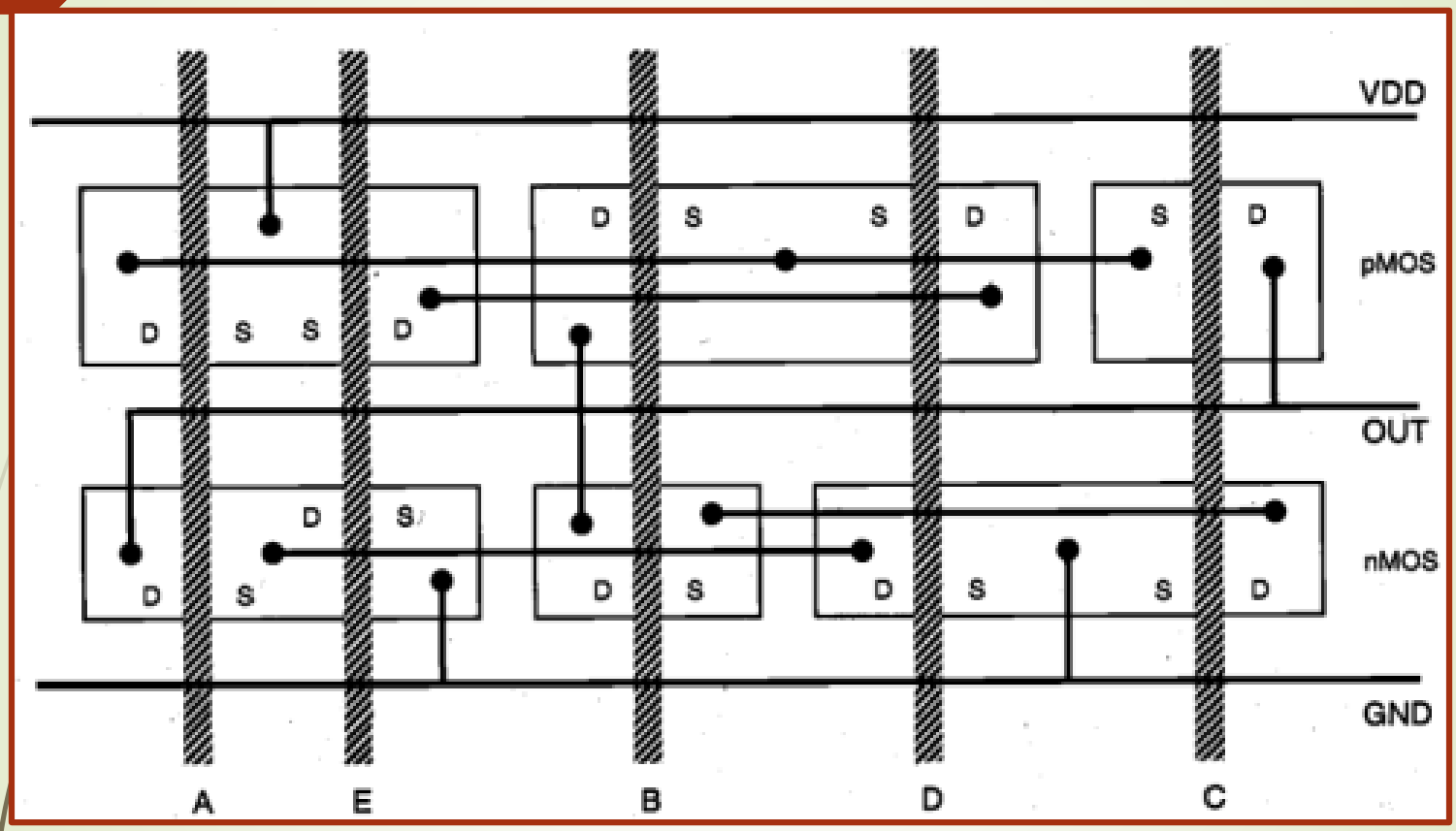
$$F = \overline{A \cdot (D + E) + B \cdot C}$$

- Above figure shows the simple construction of the dual p-net (pull-up) graph from the n-net (pull-down) graph.
- Each driver transistor in the pull-down network is represented by an edge, and each node is represented by a vertex in the pull-down graph.
- Next, a new vertex is created within each confined area in the pull-down graph, and neighboring vertices are connected by edges which cross each edge in the pull-down graph only once.
- This new graph represents the pull-up network. The resulting CMOS complex logic gate is shown in Fig.

Layout of Complex CMOS Logic Gates

- **Objective:** To construct a minimum-area layout for the complex CMOS logic gate. The stick-diagram layout is a "first attempt," using an arbitrary ordering of the polysilicon gate columns.
- The separation between the polysilicon columns must allow for one diffusion-to diffusion separation and two metal-to-diffusion contacts in between. This certainly consumes a considerable amount of extra silicon area.
- Thus to minimize the number of diffusion-area breaks for nMOS and for pMOS transistors, the separation between the polysilicon gate columns can be made smaller, which will reduce the overall horizontal dimension and, hence, the circuit layout area. The number of diffusion breaks can be minimized by changing the *ordering* of the polysilicon columns.

Stick-diagram Layout of Complex CMOS Logic Gates



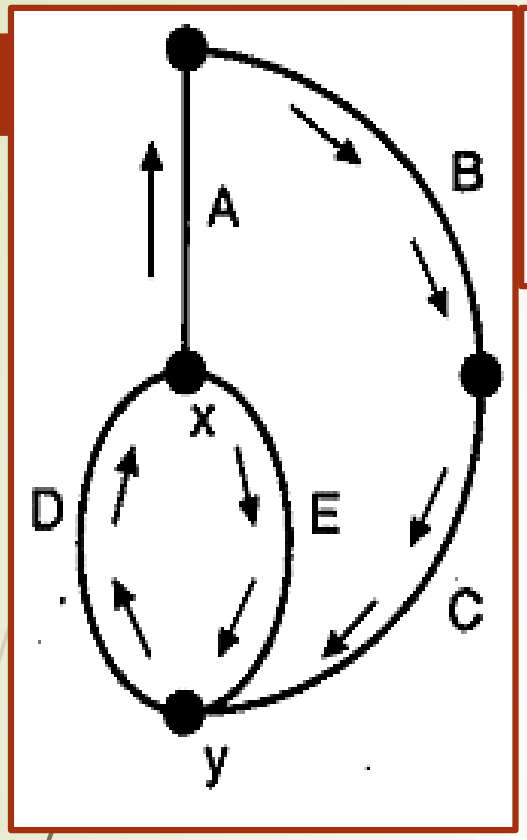
$$F = \overline{A \cdot (D + E) + B \cdot C}$$

Stick-diagram layout of the complex CMOS logic gate, with an arbitrary ordering of the polysilicon gate columns.

Layout of Complex CMOS Logic Gates
Euler Path

- A simple method for finding the optimum gate ordering is the Euler-path approach:
- **find a Euler path in the pull-down graph and a Euler path in the pull-up graph with identical ordering of input labels, i.e., find a common Euler path for both graphs. The Euler path is defined as an uninterrupted path that traverses each edge (branch) of the graph exactly once.**

Euler Path

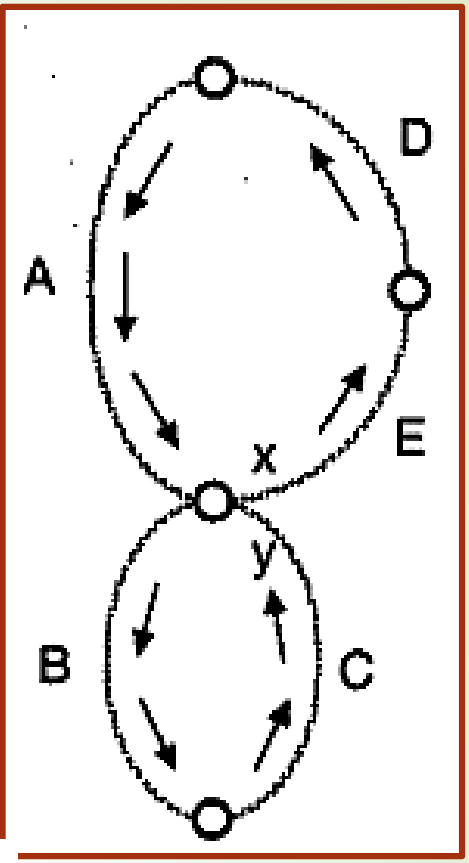


nMOS network

Common Euler path :

E - D - A - B - C

$$F = \overline{A \cdot (D + E) + B \cdot C}$$

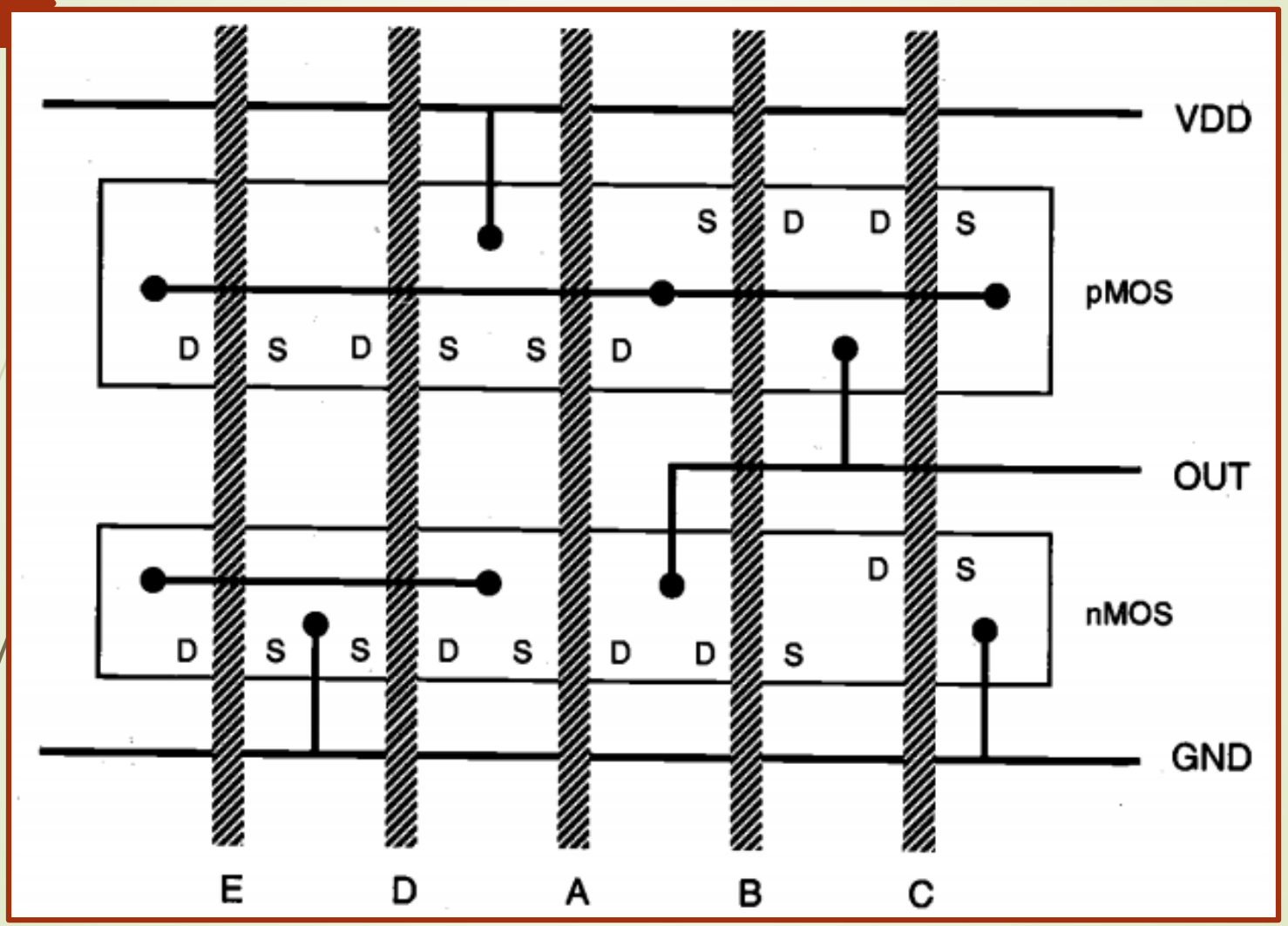


pMOS network

Figure: **Finding a common Euler path in both graphs for n-net and p-net provides a gate ordering that minimizes the number of diffusion breaks and, thus, minimizes the logic- gate layout area. In both cases, the Euler path starts at (x) and ends at (y).**

Euler Path

$$F = \overline{A \cdot (D + E) + B \cdot C}$$



Euler Path

- It is seen that there is a common sequence (E - D - A - B - C) in both graphs, i.e., a 'Euler path'. The polysilicon gate columns can be arranged according to this sequence, which results in uninterrupted p-type and n-type diffusion areas.
- In the stick diagram of the new layout shown, the polysilicon column separation Δd has to allow for only one metal-to-diffusion contact. **Advantages: more compact (smaller) layout area, simple routing of signals, and consequently, less parasitic capacitance.**

Complex CMOS Logic Circuits

AOI / OAI CMOS Logic

- We classify the common types of logic expression forms as:

AOI **AND-OR-INVERT**

- A Sum-of-Products logic expression form:

$$F = \overline{A \cdot B + B' \cdot C + C' D}$$

OAI **-OR-AND-INVERT**

- A Products-of-Sums logic expression form:

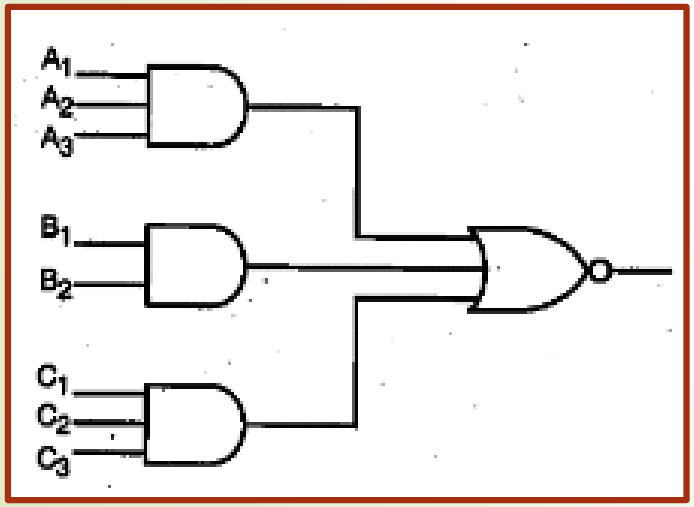
$$F = \overline{(A + B) \cdot (B' + C) \cdot (C' + D)}$$

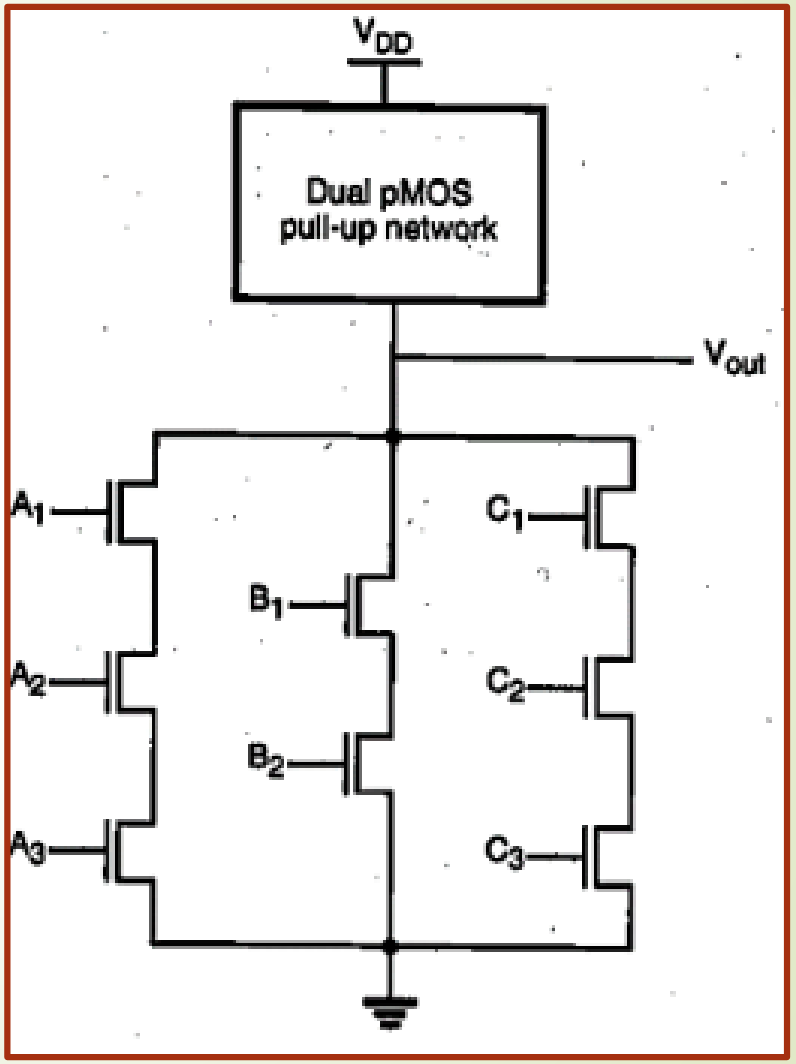
- We have the *Invert* portion in these forms so that we can directly synthesize the NMOS pull-down network.

Complex CMOS Logic Circuits

AOI / OAI CMOS Logic

AOI

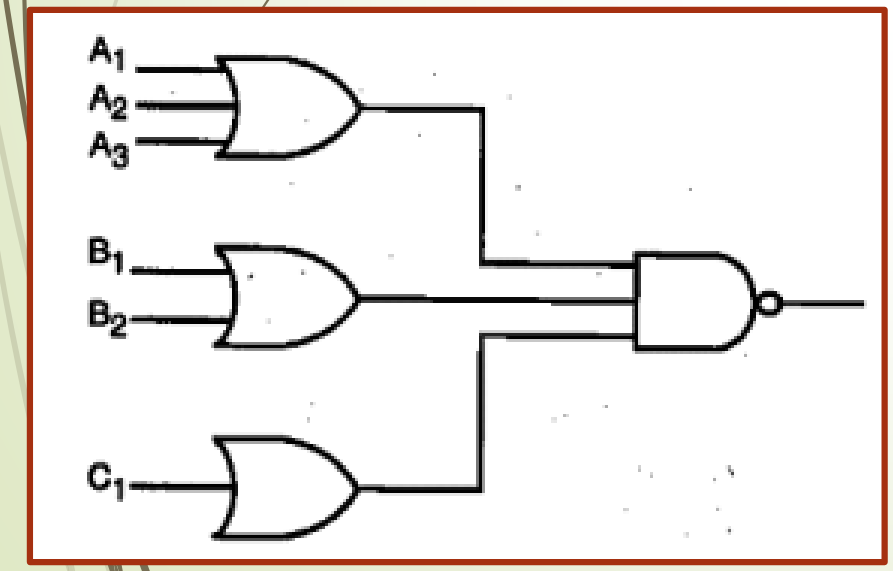


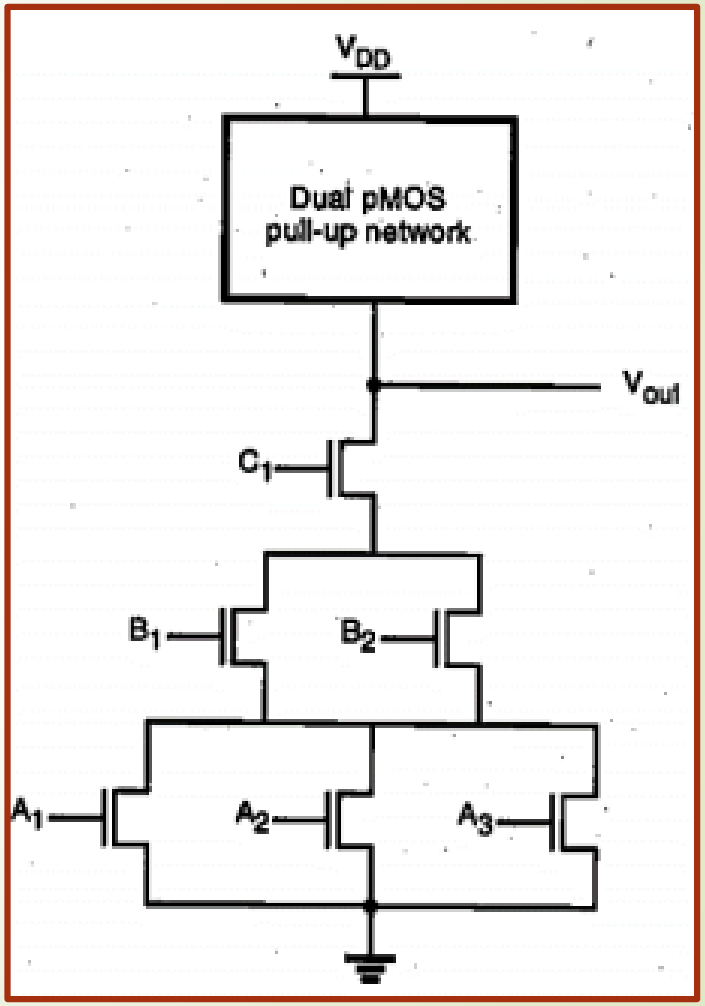


Complex CMOS Logic Circuits

AOI / OAI CMOS Logic

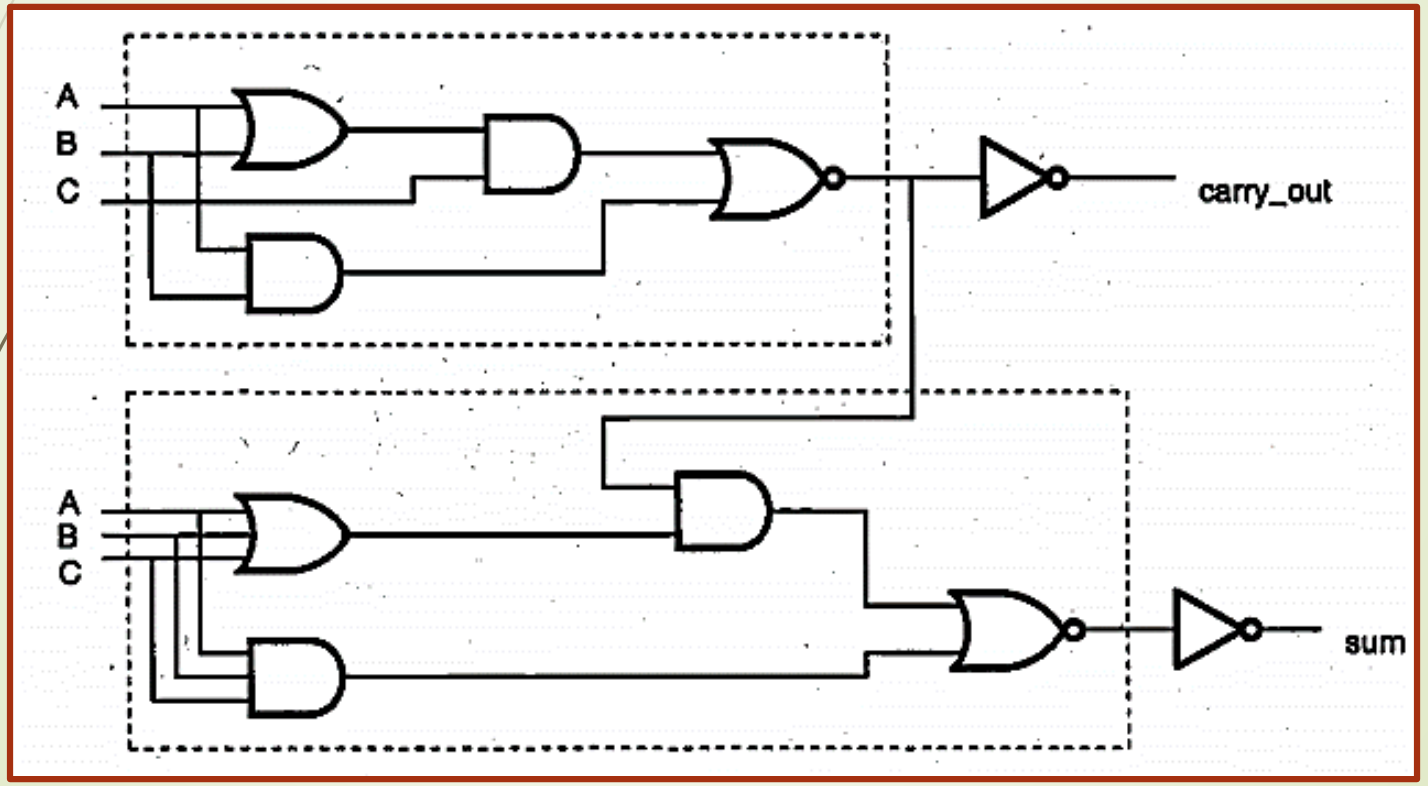
OAI



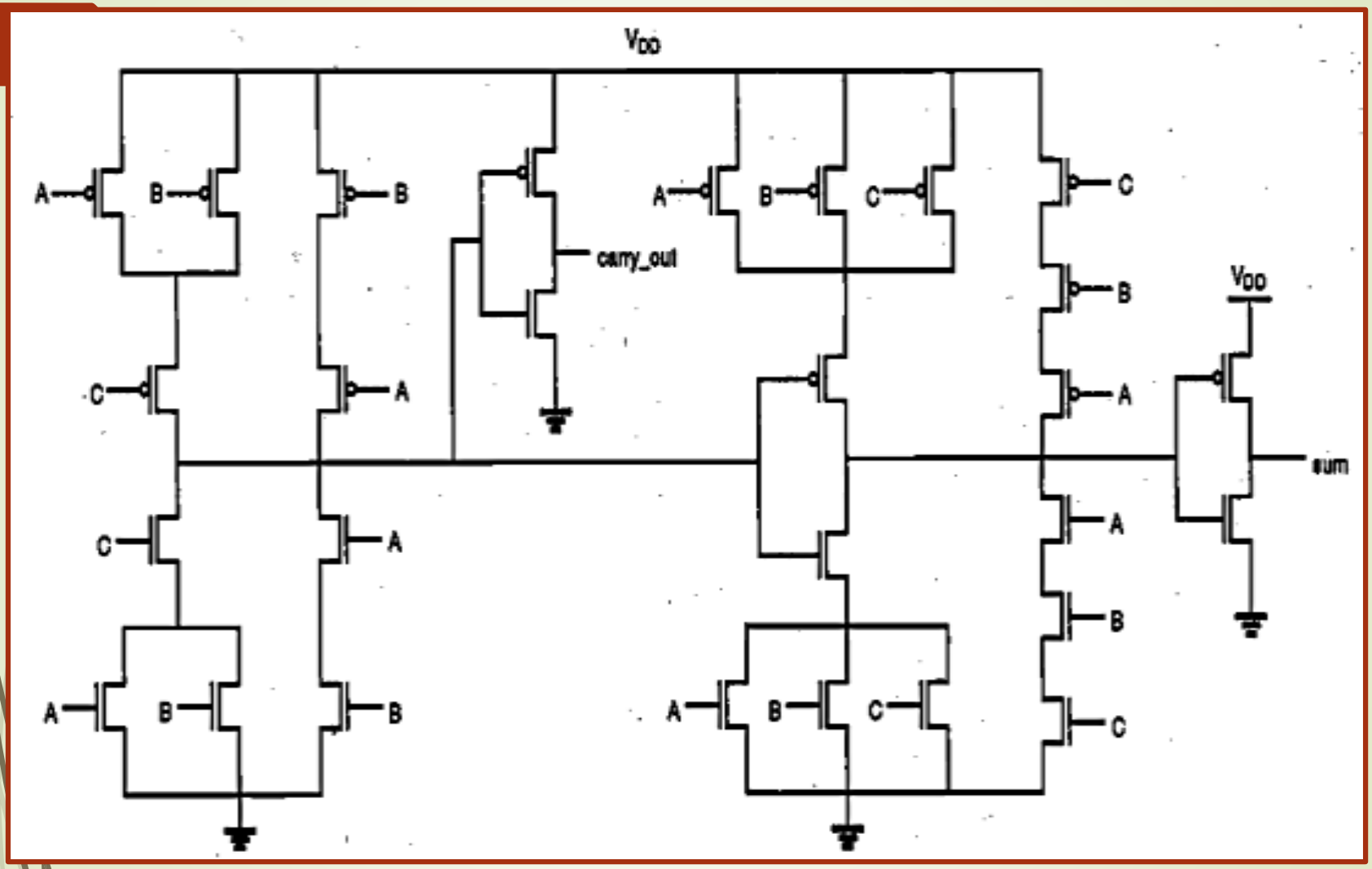


CMOS Full-Adder Circuit

$$\text{sum\_out} = A \oplus B \oplus C$$
$$= ABC + A\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + \overline{A}\,\overline{C}B$$
$$\text{carry\_out} = AB + AC + BC$$



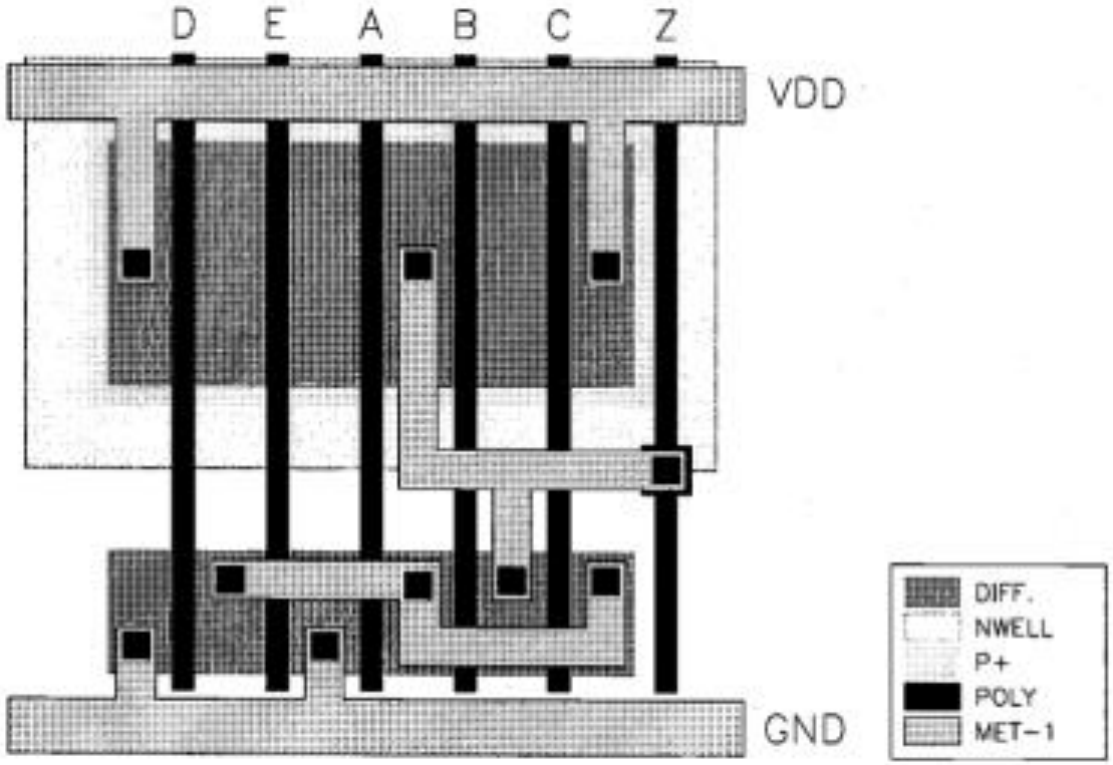
CMOS Full-Adder Circuit

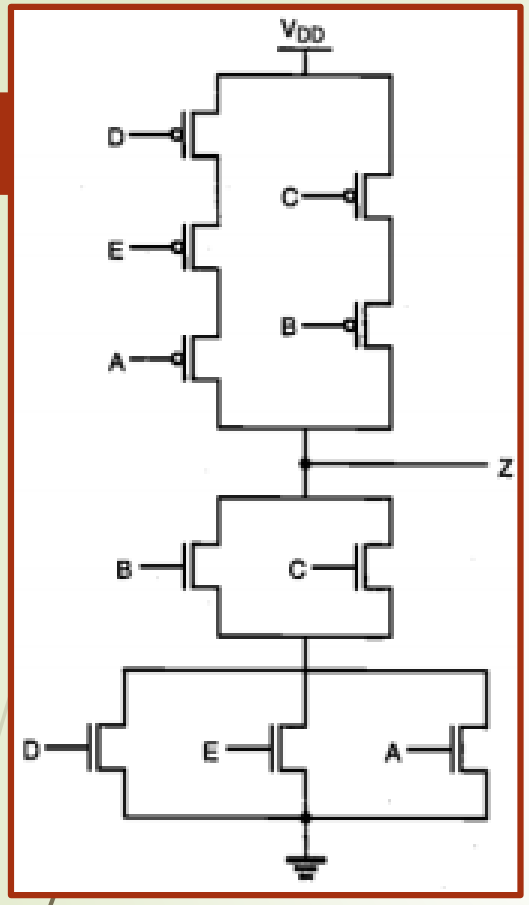


Transistor-level schematic of the one-bit full-adder circuit.

Example

The simplified layout of a CMOS complex logic circuit is given below. Draw the corresponding circuit diagram, and find an equivalent CMOS inverter circuit for simultaneous switching of all inputs, assuming that $(W/L)_p = 15$ for all pMOS transistors and $(W/L)_n = 10$ for all nMOS transistors.





Example

The Boolean function realized by this circuit is

$$Z = \overline{(D + E + A)(B + C)}$$

$$r_{DS} \propto (W/L)^{-1} \begin{cases} \text{Series Connection} \rightarrow (W/L)_{eq} = \left[\dfrac{1}{(W/L)_1} + \dfrac{1}{(W/L)_2} + \ldots \right]^{-1} \\ \text{Parallel Connection} \rightarrow (W/L)_{eq} = (W/L)_1 + (W/L)_2 + \ldots \end{cases}$$

The equivalent *(WIL)* ratios of the nMOS network and the pMOS network are determined by using the series-parallel equivalency rules:

$$\left(\frac{W}{L}\right)_{n,eq} = \frac{1}{\frac{1}{\left(\frac{W}{L}\right)_D + \left(\frac{W}{L}\right)_E + \left(\frac{W}{L}\right)_A} + \frac{1}{\left(\frac{W}{L}\right)_B + \left(\frac{W}{L}\right)_C}}$$

$$= \frac{1}{\frac{1}{30} + \frac{1}{20}} = 12$$

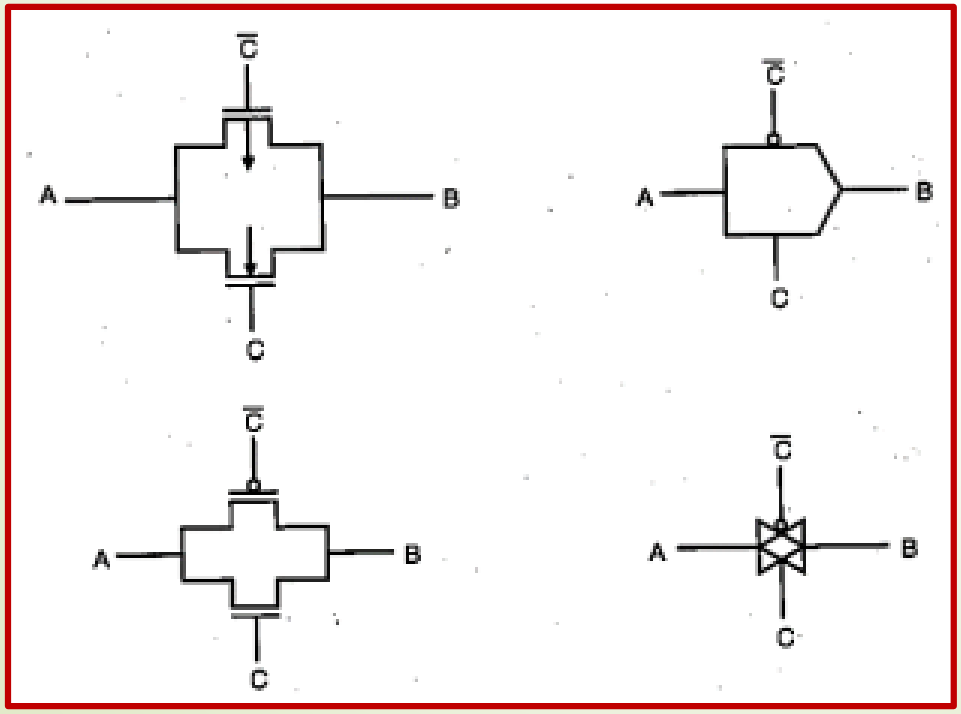
$$\left(\frac{W}{L}\right)_{p,eq} = \frac{1}{\frac{1}{\left(\frac{W}{L}\right)_D} + \frac{1}{\left(\frac{W}{L}\right)_E} + \frac{1}{\left(\frac{W}{L}\right)_A}} + \frac{1}{\frac{1}{\left(\frac{W}{L}\right)_B} + \frac{1}{\left(\frac{W}{L}\right)_C}}$$

$$= \frac{1}{\frac{1}{15} + \frac{1}{15} + \frac{1}{15}} + \frac{1}{\frac{1}{15} + \frac{1}{15}} = 12.5$$

Complex CMOS Logic Circuits

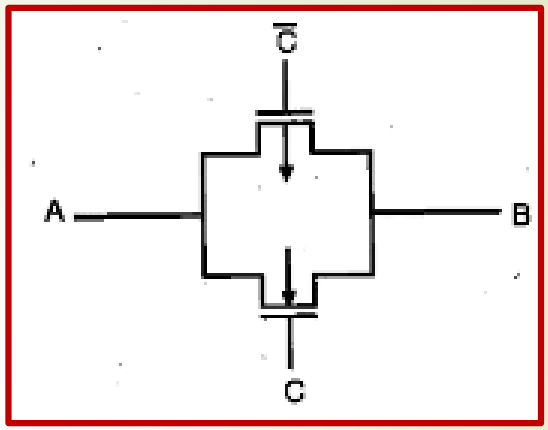
❖ Transmission Gate (Pass Gate)

- ❖ A Transmission Gate (T-gate or TG or *pass gate*) is a bi-directional switch made up of an NMOS and PMOS is parallel.
- ❖ a control signal is connected to the gate of the NMOS (C) and its complement is sent to the gate of the PMOS (C')
- ❖ The T-gate is a bidirectional switch between A and B which is controlled by C



Complex CMOS Logic Circuits

❖ Transmission Gate (Pass Gate)



❖ When the control signal C is HIGH : (VDD)

Both transistors are turned on

A low resistance path exists between A and B

- When the control signal is LOW (0 V)

Both transistors are off

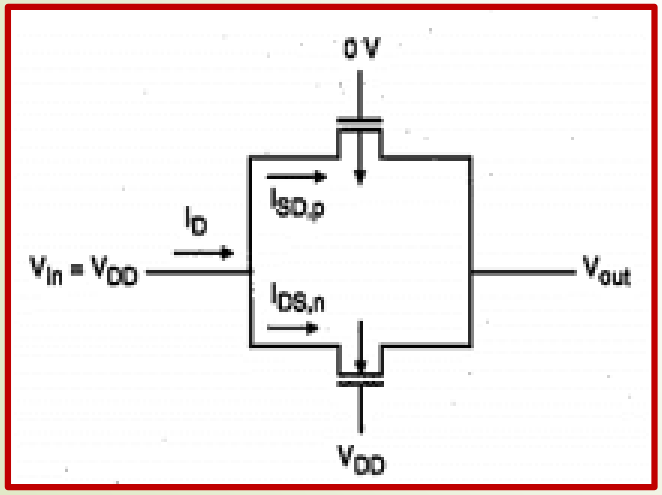
The T-gate looks like an open circuit

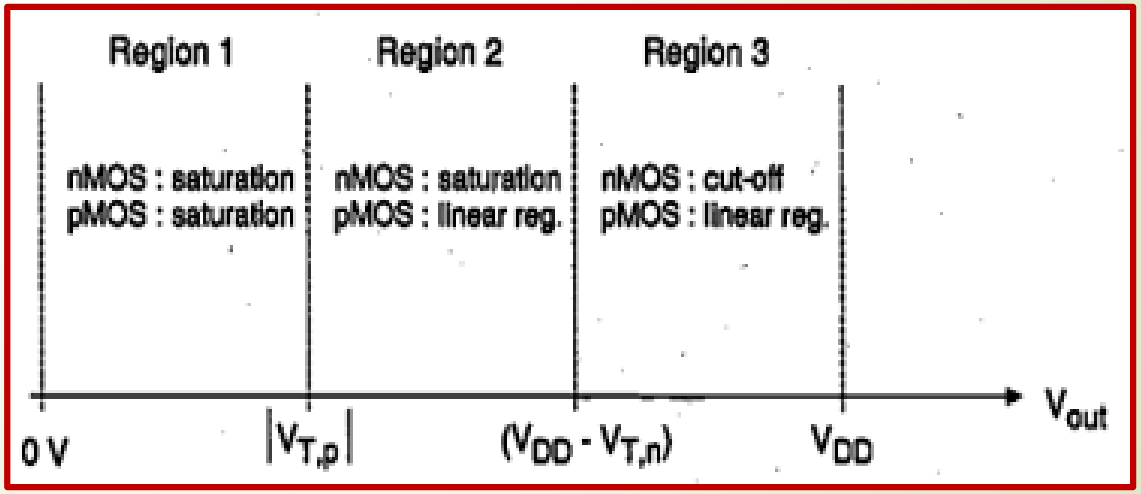
❖ This type of operation is commonly used in bus situations where only one gate can drive the bus line at the same time

❖ T-gates are put on the output of each gate on the bus. The circuit that drives will use a T-gate to connect to the bus with a low impedance path. All other circuits that aren't driving will switch their T-gates to be a high-impedance.

Complex CMOS Logic Circuits

- **Transmission Gate (Pass Gate)**
  - When the T-gate is on, the regions of operation of the transistors will depend on Vin and Vout
  - let's say we drive Vin=VDD and initially Vout = 0 V
  - As Vout moves from 0V to VDD, the regions of operation for the transistors are as follows:





Bias conditions and operating regions of the CMOS transmission gate as function of the output voltage

Transmission Gate (Pass Gate)

It can be seen from Fig. that the drain-to-source and the gate-to-source voltages of the nMOS transistor are

$$V_{DS,n} = V_{DD} - V_{out}$$
$$V_{GS,n} = V_{DD} - V_{out}$$

Thus, the nMOS transistor will be turned off for $V_{out} > V_{DD} - V_{T,n}$ and will operate in the saturation mode for $V_{out} < V_{DD} - V_{T,n}$. The V_{DS} and V_{GS} voltages of the pMOS transistor are

$$V_{DS,p} = V_{out} - V_{DD}$$
$$V_{GS,p} = -V_{DD}$$

Consequently, the pMOS transistor is in saturation for $V_{out} < |V_{T,p}|$, and it operates in the linear region for $V_{out} > |V_{T,p}|$. Note that, unlike the nMOS transistor, the pMOS transistor remains turned on, regardless of the output voltage level V_{out}.

Transmission Gate (Pass Gate)

Total current flowing through the transmission gate is the sum of the NMOS drain current and the PMOS drain current.

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

At this point, we may devise an *equivalent resistance* for each transistor in this structure,

$$R_{eq,n} = \frac{V_{DD} - V_{out}}{I_{DS,n}}$$

$$R_{eq,p} = \frac{V_{DD} - V_{out}}{I_{SD,p}}$$

The total equivalent resistance of the CMOS TG will then be the parallel equivalent of these two resistances, $R_{eq,n}$ and $R_{eq,p}$. Now, we will calculate the equivalent resistance values for the three operating regions of the transmission gate.

Transmission Gate (Pass Gate)

Region 1:

Here, the output voltage is smaller than the absolute value of the pMOS transistor threshold voltage, i.e., $V_{out} < |V_{T,p}|$. According to Fig. , both transistors are in saturation. We obtain the equivalent resistance of both devices as

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

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

Note that the source-to-substrate voltage of the nMOS transistor is equal to the output voltage V_{out}, while the source-to-substrate voltage of the pMOS transistor is equal to zero. Thus, we have to take into account the substrate-bias effect for the nMOS transistor in our calculations.

Transmission Gate (Pass Gate)

Region 2:

In this region, $|V_{T,p}| < V_{out} < (V_{DD} - V_{T,n})$. Thus, the pMOS transistor now operates in the linear region, while the nMOS transistor continues to operate in saturation.

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

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

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

Transmission Gate (Pass Gate)

Region 3:

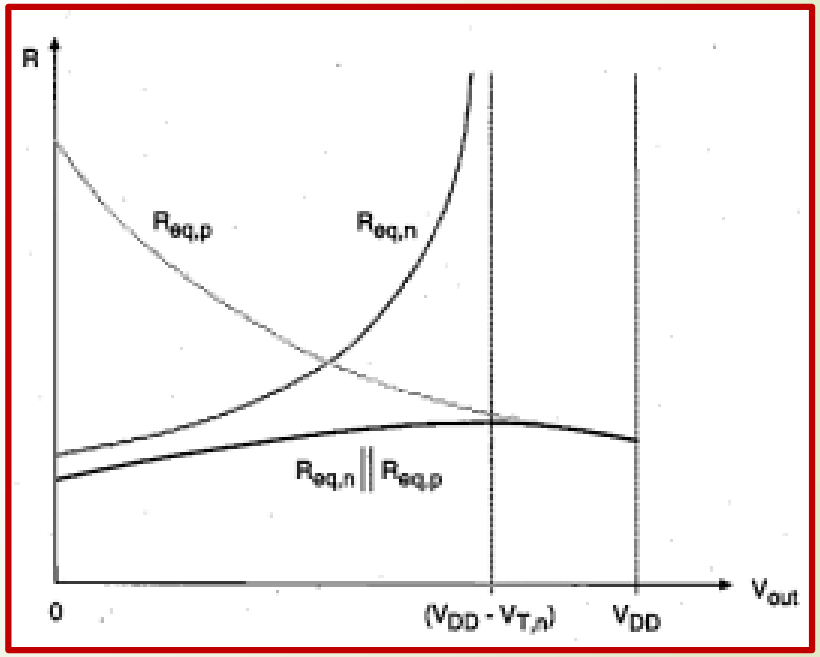
Here, the output voltage is $V_{out} > (V_{DD} - V_{T,n})$. Consequently, the nMOS transistor will be turned off, which results in an open-circuit equivalent. The pMOS transistor will continue to operate in the linear region.

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

Combining the equivalent resistance values found for the three operating regions, we can now plot the total resistance of the CMOS transmission gate as a function of the output voltage V_{out},

Transmission Gate (Pass Gate)

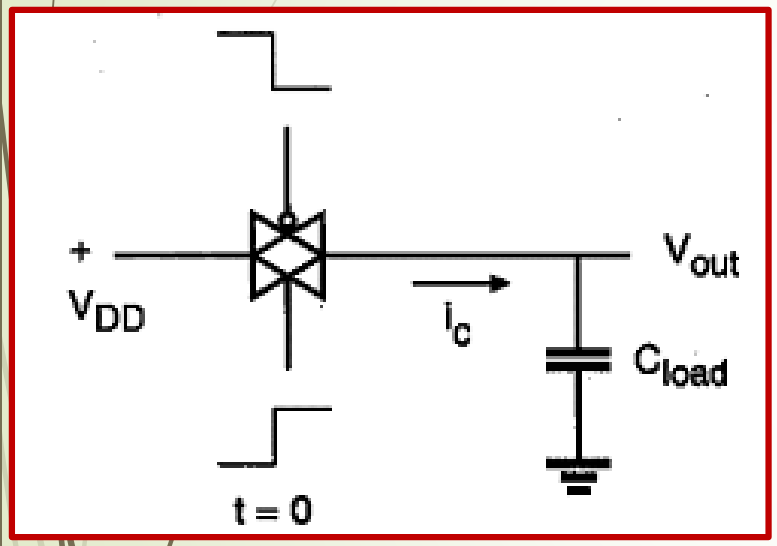
It can be seen that the total equivalent resistance of the TG remains relatively constant, i.e., its value is almost independent of the output voltage, whereas the individual equivalent resistances of both the nMOS and the pMOS transistors are strongly dependent on V_{out}. This property of the CMOS TG is naturally quite desirable. A CMOS pass gate which is turned on by a logic-high control signal can be replaced by its simple equivalent resistance for dynamic analysis,

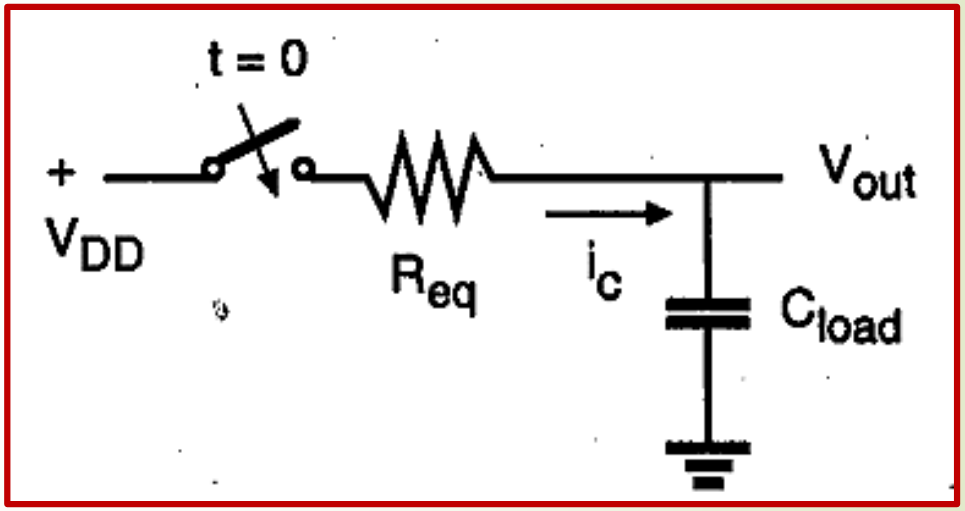


Equivalent resistance of the CMOS transmission gate plotted as a function of the output voltage

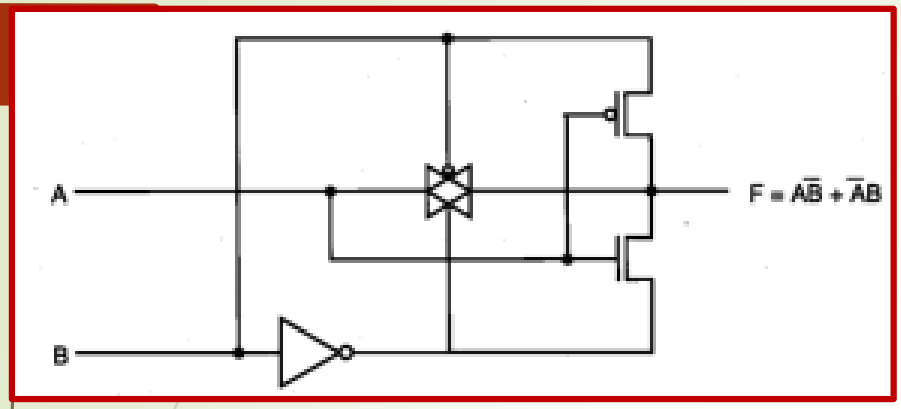
Transmission Gate (Pass Gate)

Resistor equivalent Circuit

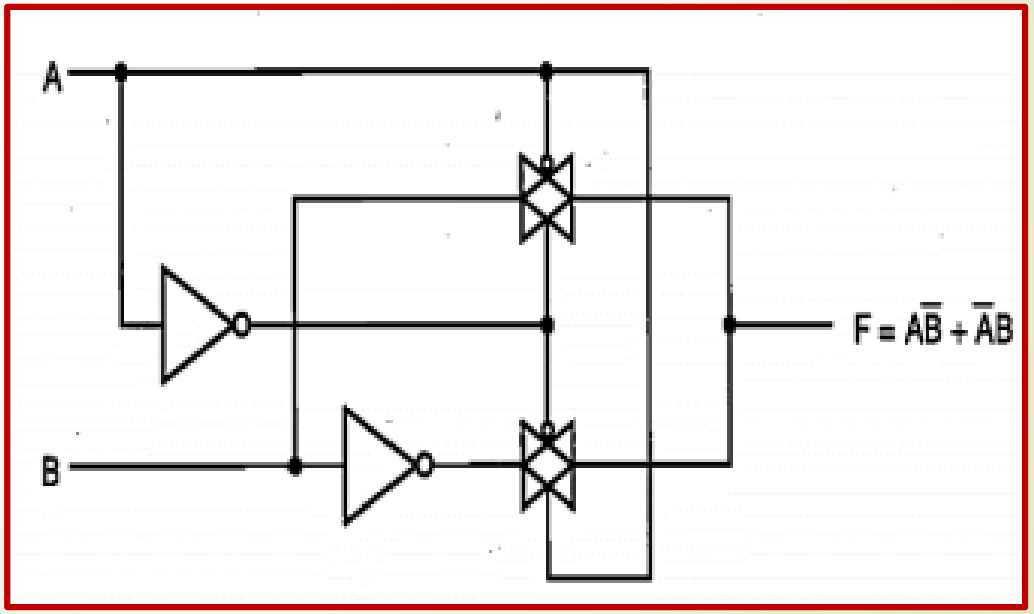




Transmission Gate (Pass Gate) Applications

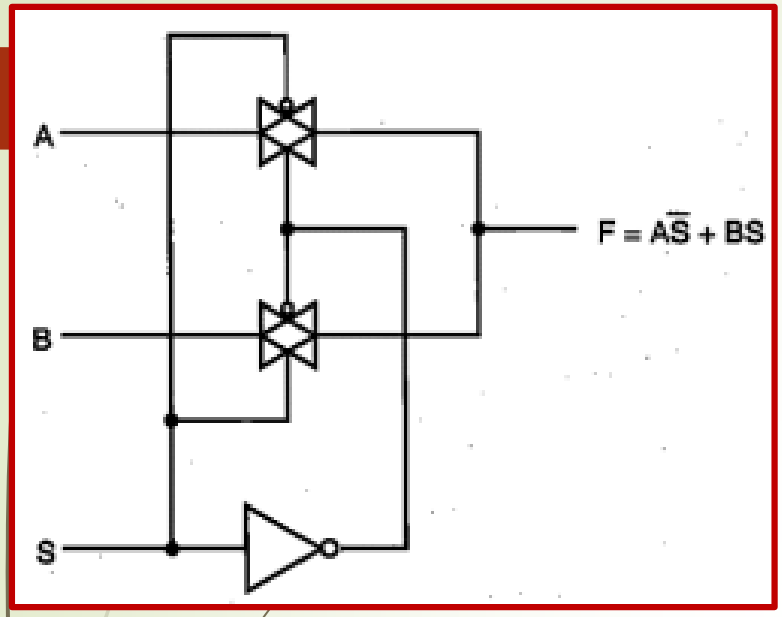


XOR gate using Six Transistor CMOS Transmission gates

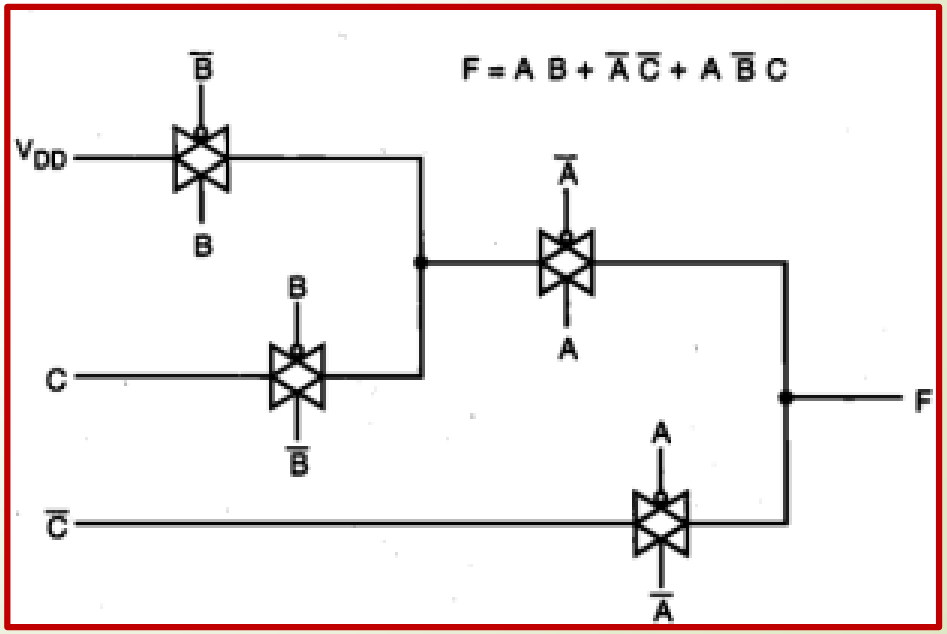


XOR gate using Eight Transistor CMOS Transmission gates

Transmission Gate (Pass Gate) Applications



Two Input Multiplexer using
Two CMOS Transmission gates



Three variable Boolean function using
CMOS Transmission gates