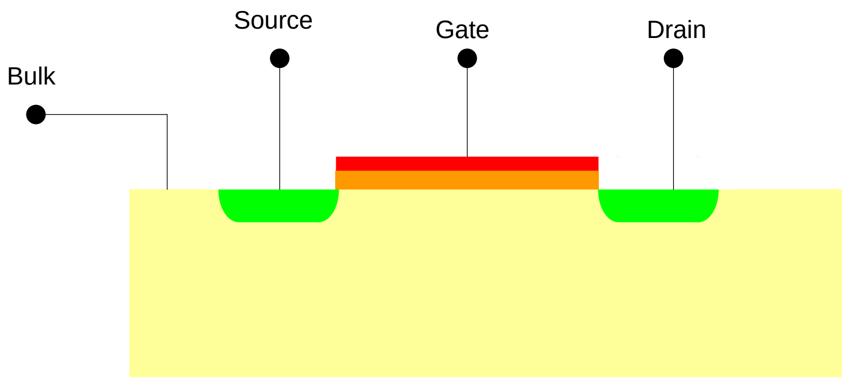
EE 330
Lecture 6

- PU and PD Networks
- Complex Logic Gates
- Pass Transistor Logic
- Improved Switch-Level Model
- Propagation Delay

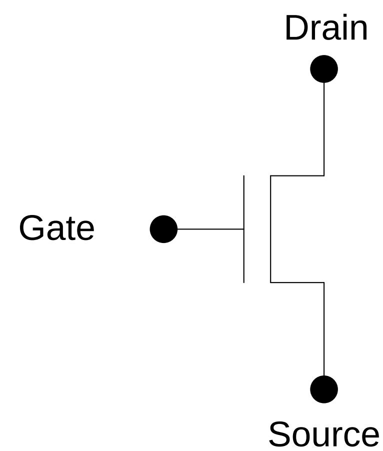
Review from Last Time

MOS Transistor

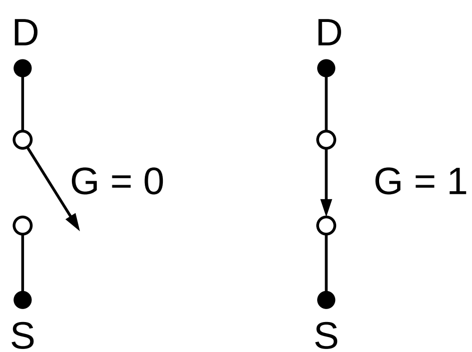
Qualitative Discussion of n-channel Operation



n-channel MOSFET



Equivalent Circuit for n-channel MOSFET



- **Source assumed connected to (or close to) ground**
- **V_{GS}=0 denoted as Boolean gate voltage G=0**
- **V_{GS}=V_{DD} denoted as Boolean gate voltage G=1**
- **Boolean G is relative to ground potential**

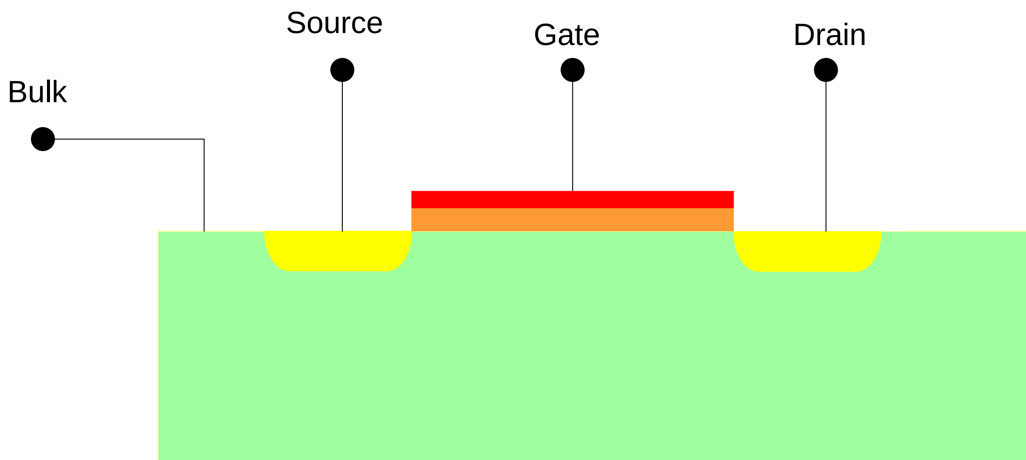
This is the first model we have for the n-channel MOSFET !

Ideal switch-level model

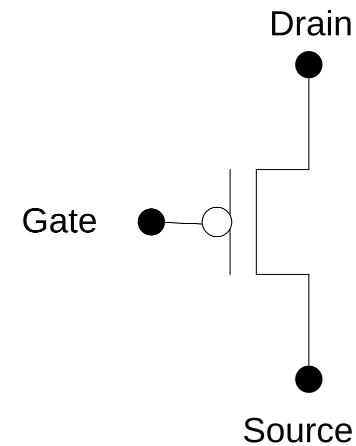
Review from Last Time

MOS Transistor

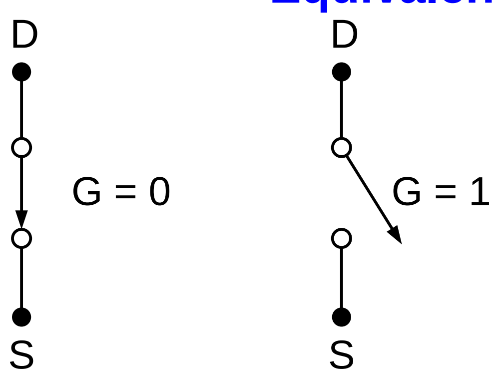
Qualitative Discussion of p-channel Operation



p-channel MOSFET



Equivalent Circuit for p-channel MOSFET

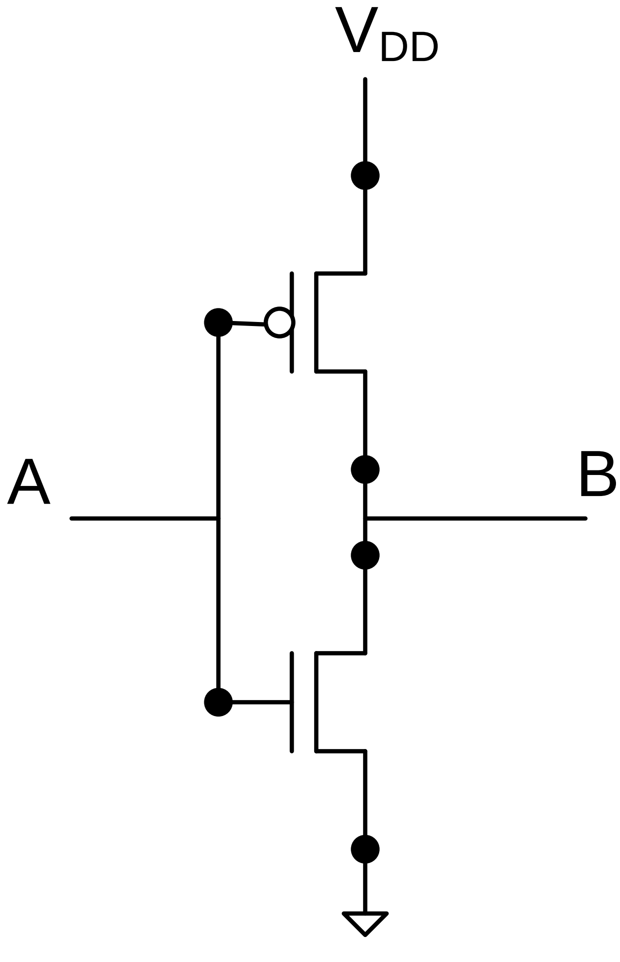


- **Source assumed connected to (or close to) positive V_{DD}**
- **V_{GS}=0 denoted as Boolean gate voltage G=1**
- **V_{GS}= -V_{DD} denoted as Boolean gate voltage G=0**
- **Boolean G is relative to ground potential**

This is the first model we have for the p-channel MOSFET !

Review from Last Time

Logic Circuits



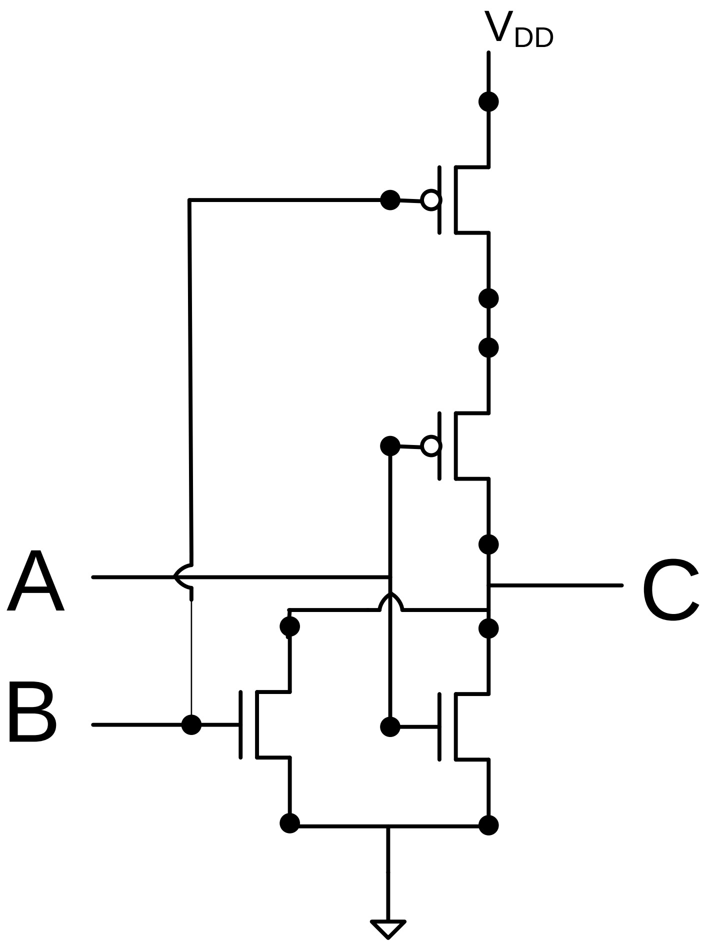
Inverter

Truth Table

A	B
0	1
1	0

Review from Last Time

Logic Circuits



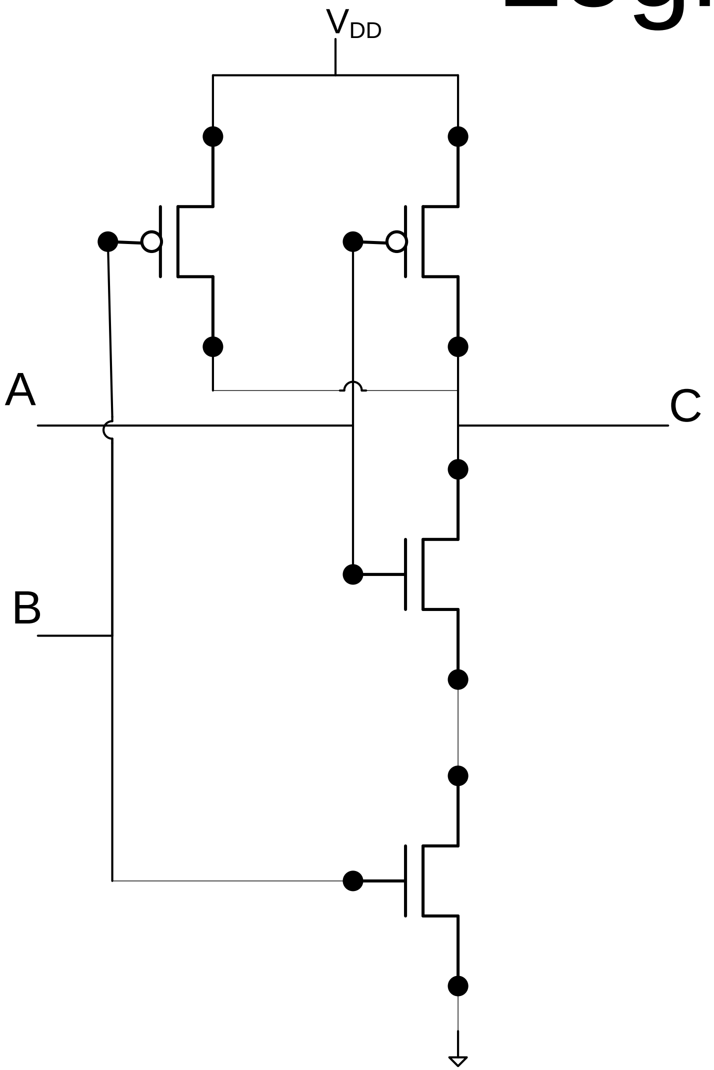
NOR Gate

Truth Table

A	B	C
0	0	1
0	1	0
1	0	0
1	1	0

Review from Last Time

Logic Circuits

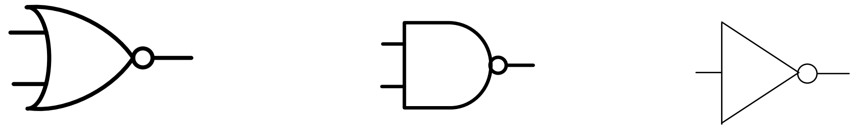


NAND Gate

Truth Table

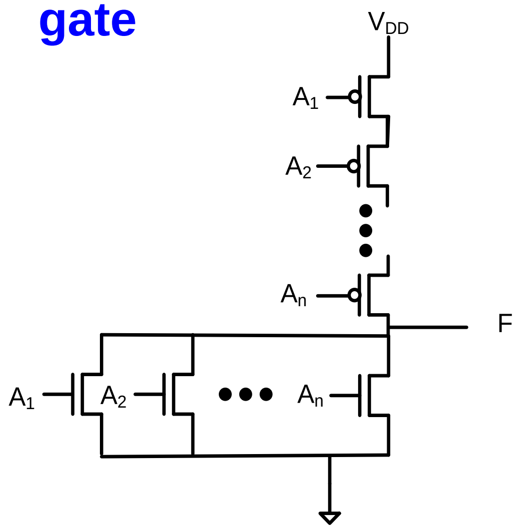
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

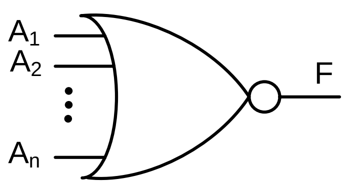
Logic Circuits



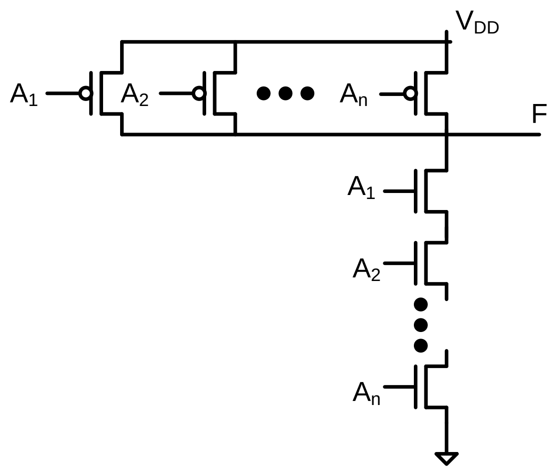
Approach can be extended to arbitrary number of inputs

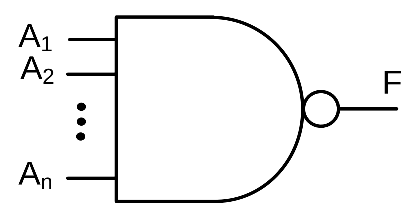
n-input NOR gate



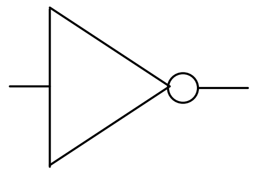


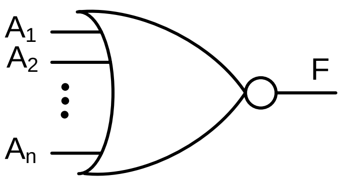
n-input NAND gate

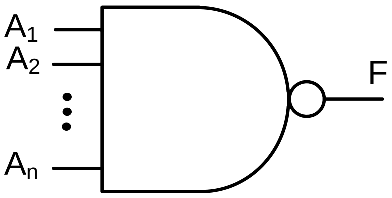




Complete Logic Family







Family of n-input NOR gates forms a complete logic family

Family of n-input NAND gates forms a complete logic family

Having both NAND and NOR gates available is a luxury

Can now implement any combinational logic function !!

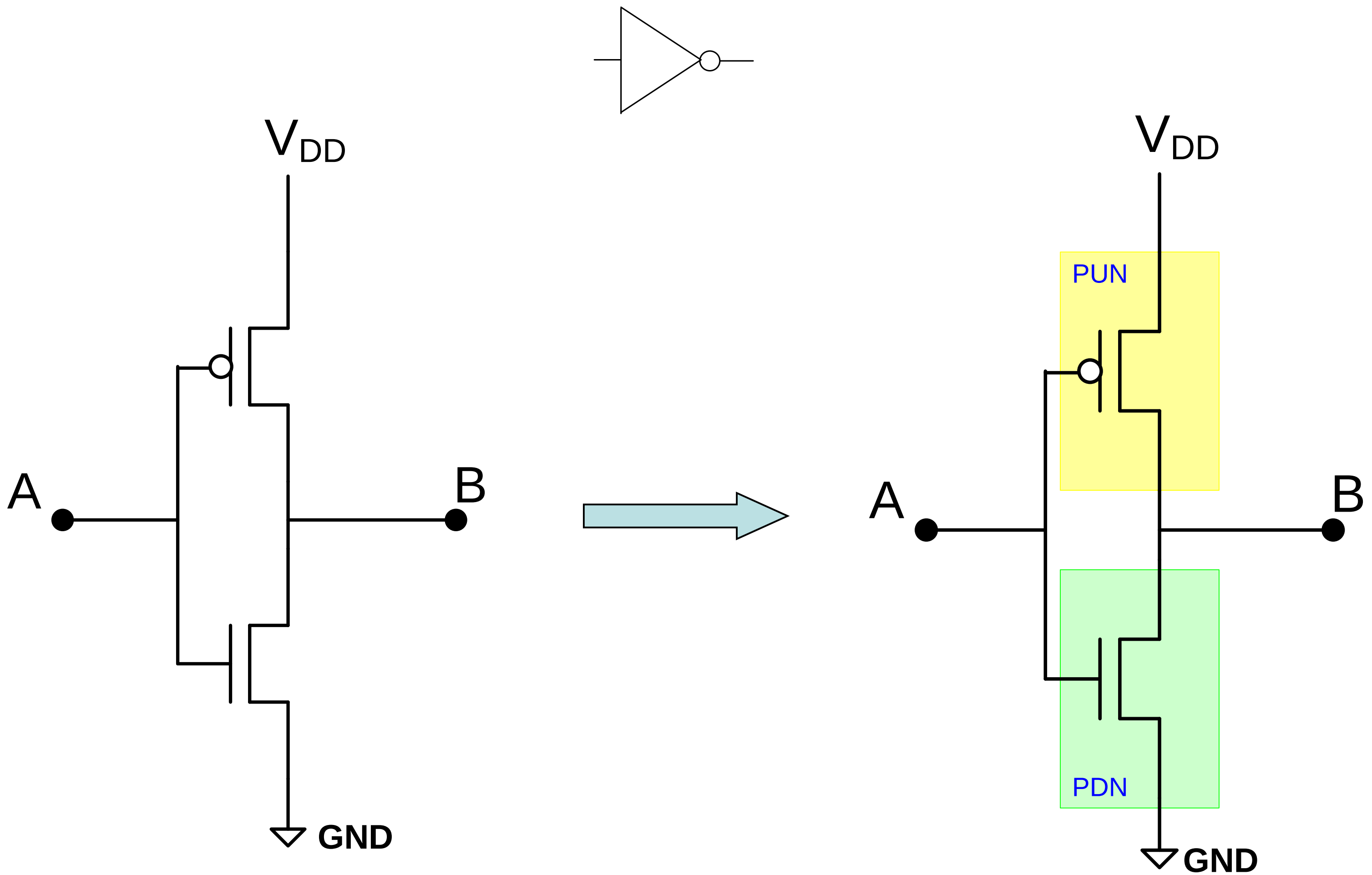
If add one flip flop, can implement any Boolean system !!

Flip flops easy to design but will discuss sequential logic systems later

Other logic circuits

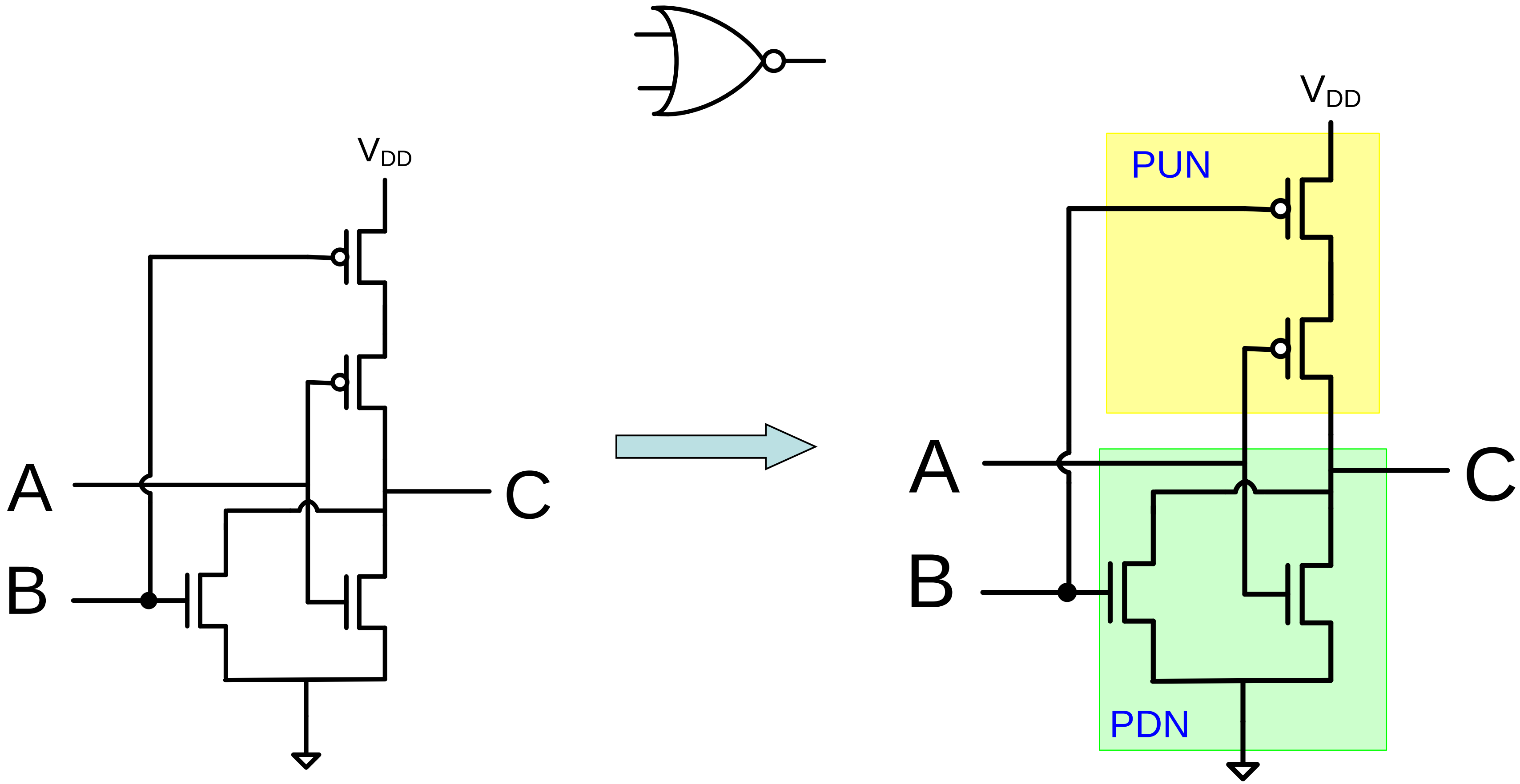
- Other methods for designing logic circuits exist
- Insight will be provided on how other logic circuits evolve
- Several different types of logic circuits are often used simultaneously in any circuit design

Pull-up and Pull-down Networks



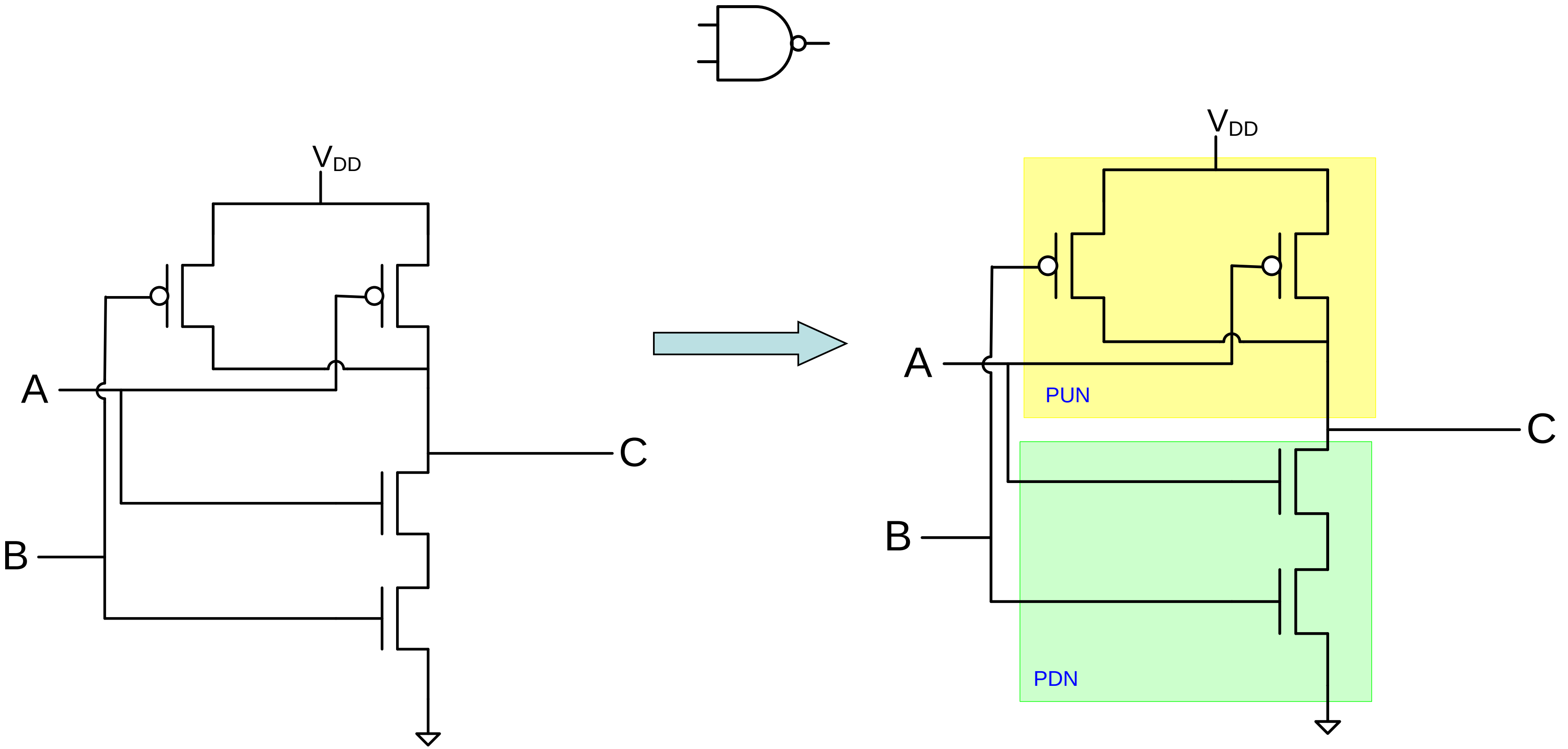
PU network comprised of p-channel device and “tries” to pull B to VDD when conducting
PD network comprised of n-channel device and “tries to pull B to GND when conducting
One and only one of these networks is conducting at the same time (to avoid contention)

Pull-up and Pull-down Networks



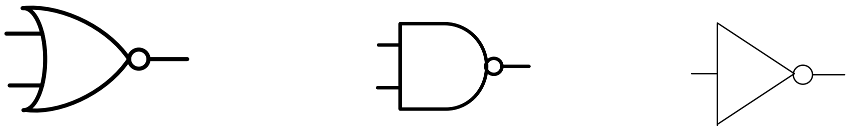
PU network comprised of p-channel devices
PD network comprised of n-channel devices
One and only one of these networks is conducting at the same time

Pull-up and Pull-down Networks

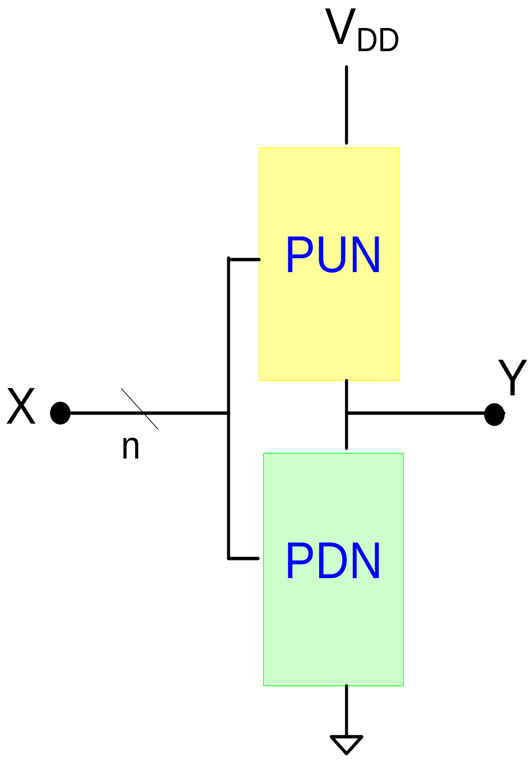


PU network comprised of p-channel devices
PD network comprised of n-channel devices
One and only one of these networks is conducting at the same time

Pull-up and Pull-down Networks



In these circuits, the PUN and PDN have the 3 interesting characteristics



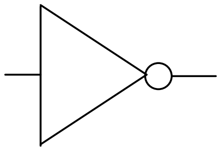
1. **PU network comprised of p-channel devices**
2. **PD network comprised of n-channel devices**
3. **One and only one of these networks is conducting at the same time**

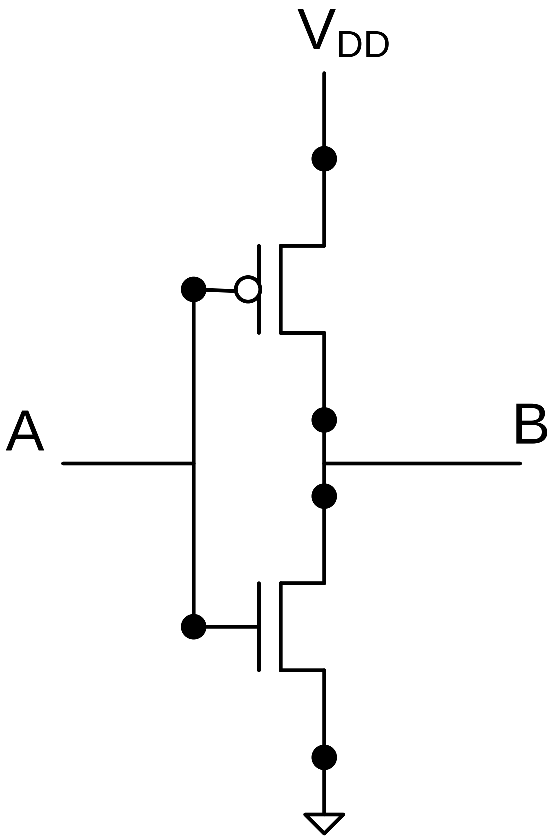
What are V_H and V_L?
What is the power dissipation?
How fast are these logic circuits?

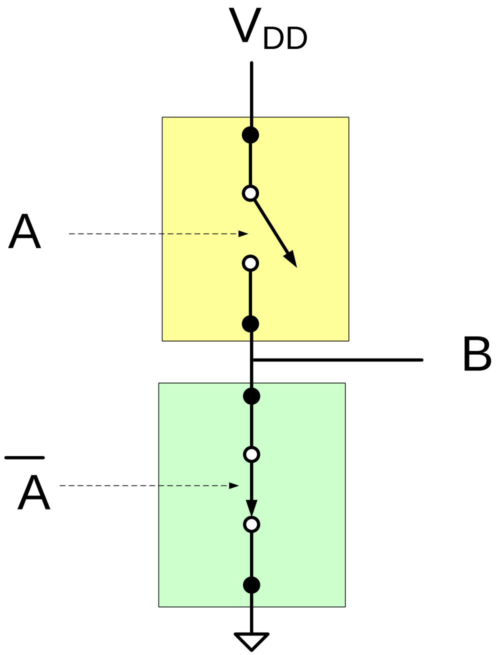
What are V_H and V_L?
What is the power dissipation?
How fast are these logic circuits?

Consider the inverter

Use switch-level model for MOS devices



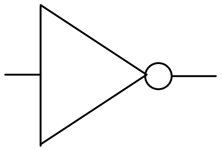


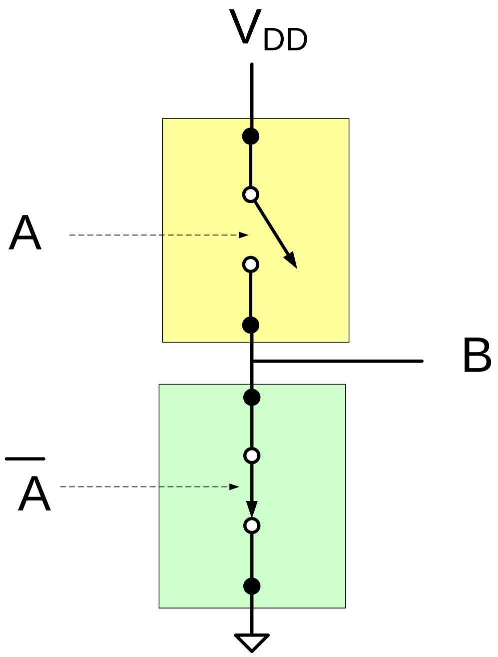


What are V_H and V_L?
What is the power dissipation?
How fast are these logic circuits?

Consider the inverter

Use switch-level model for MOS devices





$V_H = V_{DD}$

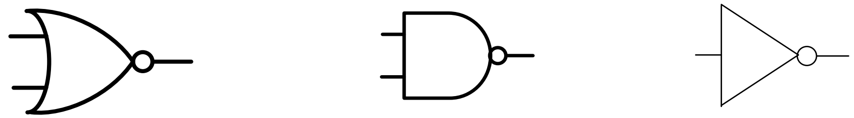
$V_L = 0$

$I_D = 0$ thus $P_H = P_L = 0$

$t_{HL} = t_{LH} = 0$

(too good to be true?)

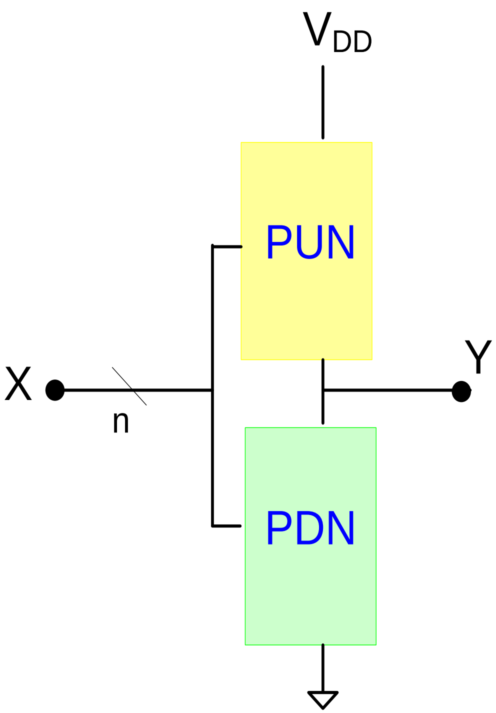
Pull-up and Pull-down Networks



For these circuits, the PUN and PDN have 3 interesting characteristics

Three key <u>characteristics</u> of these Static CMOS Gates

1. **PU network comprised of p-channel devices**
2. **PD network comprised of n-channel devices**
3. **One and only one of these networks is conducting at the same time**



Three key <u>properties</u> of these Static CMOS Gates

1. **What are V_H and V_L?**

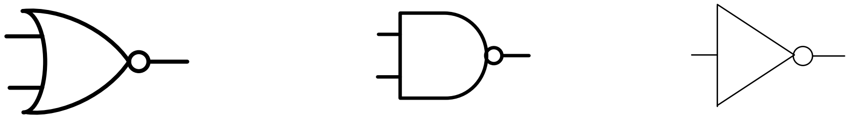
   $V_H=V_{DD}$, $V_L=0$ **(too good to be true?)**
2. **What is the power dissipation?**

   $P_H=P_L=0$ **(too good to be true?)**
3. **How fast are these logic circuits?**

   $t_{HL}=t_{LH}=0$ **(too good to be true?)**

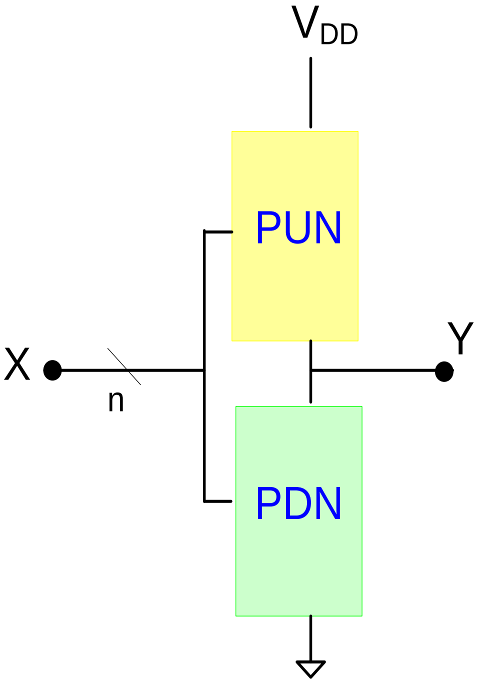
These 3 <u>properties</u> are inherent in all Boolean circuits that have these 3 <u>characteristics</u> !!!

Pull-up and Pull-down Networks



Three key characteristics of Static CMOS Gates

1. **PU network comprised of p-channel devices**
2. **PD network comprised of n-channel devices**
3. **One and only one of these networks is conducting at the same time**

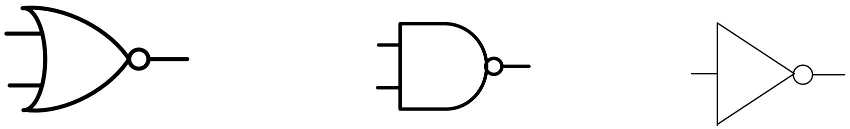


Three properties of Static CMOS Gates (based upon simple switch-level model)

1. $V_H = V_{DD}$, $V_L = 0$ (too good to be true?)
2. $P_H = P_L = 0$ (too good to be true?)
3. $t_{HL} = t_{LH} = 0$ (too good to be true?)

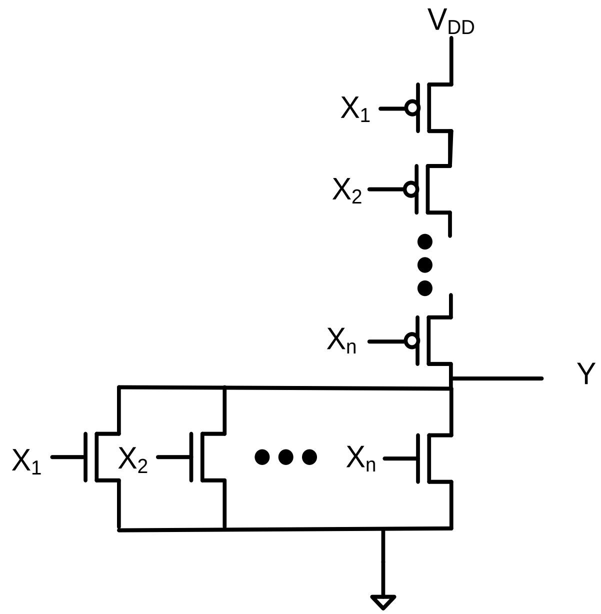
These 3 properties are inherent in Boolean circuits with these 3 characteristics

Pull-up and Pull-down Networks

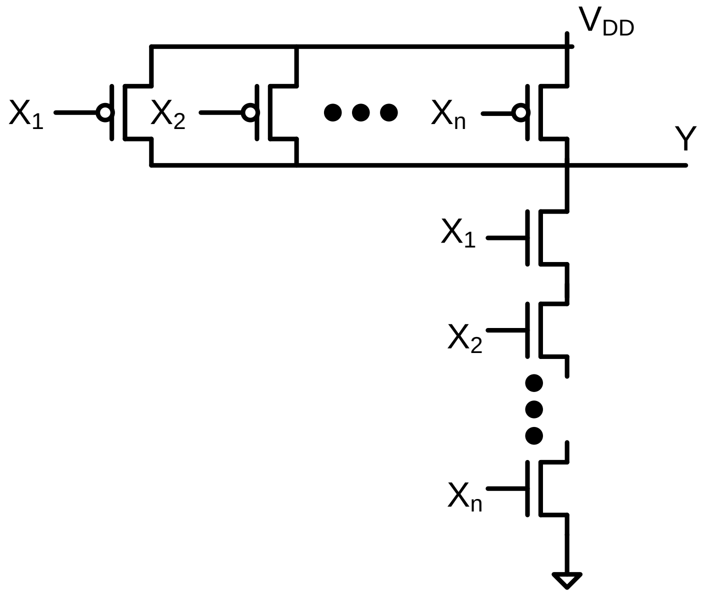


Concept can be extended to arbitrary number of inputs

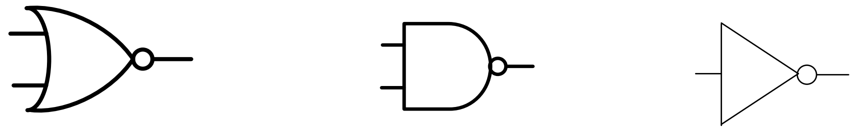
n-input NOR gate



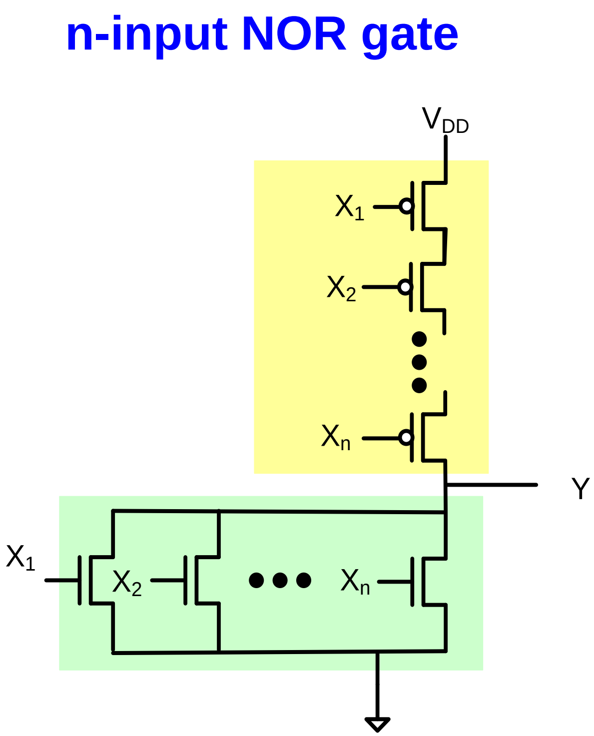
n-input NAND gate

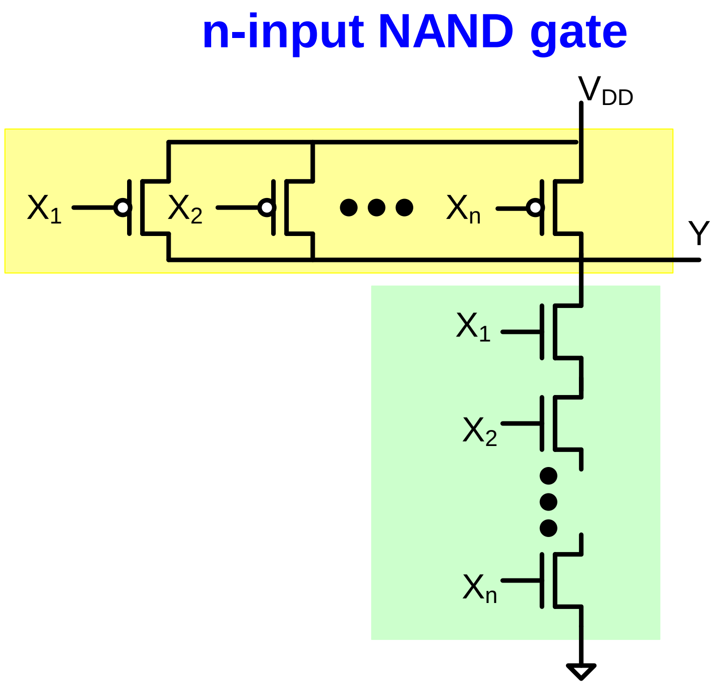


Pull-up and Pull-down Networks



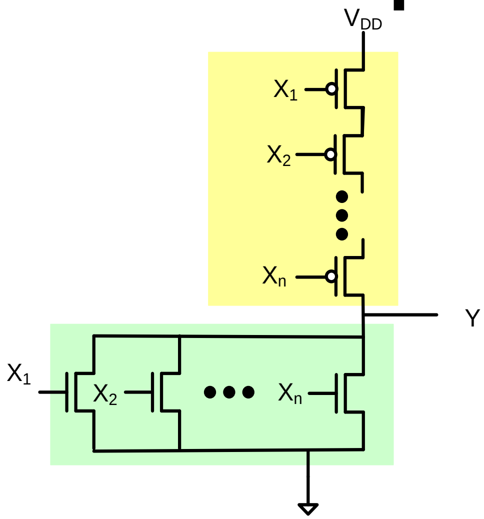
Concept can be extended to arbitrary number of inputs



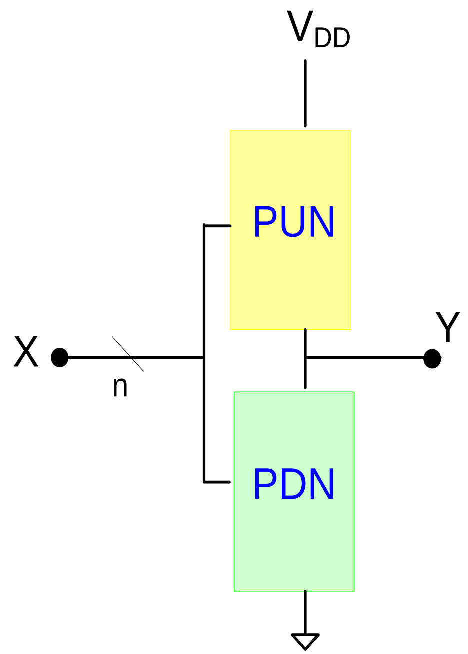


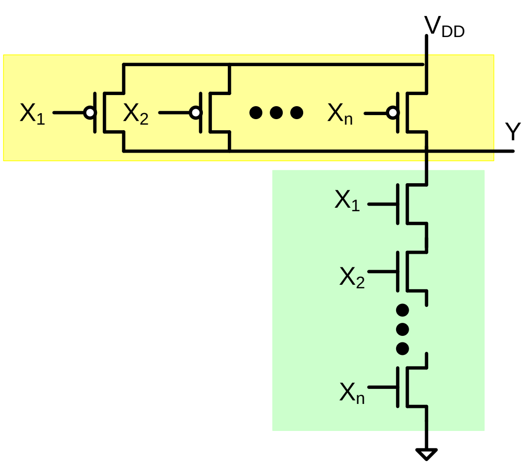
1. **PU network comprised of p-channel devices**
2. **PD network comprised of n-channel devices**
3. **One and only one of these networks is conducting at the same time**

Pull-up and Pull-down Networks



n-input NOR gate





n-input NAND gate

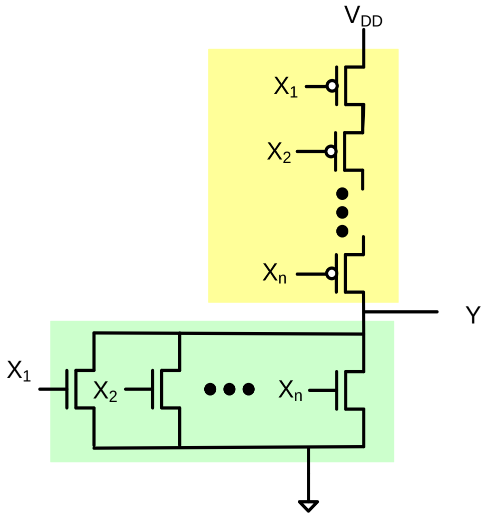
1. **PU network comprised of p-channel devices**
2. **PD network comprised of n-channel devices**
3. **One and only one of these networks is conducting at the same time**

$V_H = V_{DD}, \quad V_L = 0$

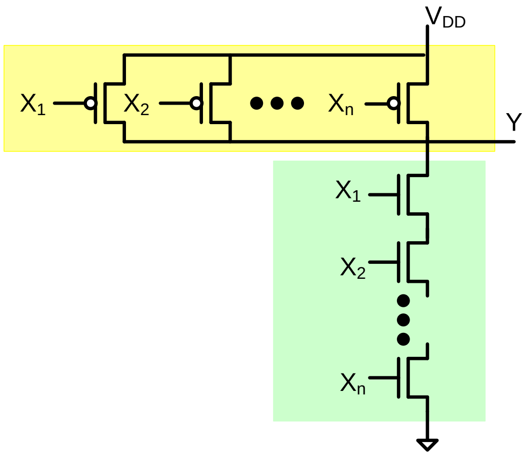
$P_H = P_L = 0$

$t_{HL} = t_{LH} = 0$

Nomenclature



n-input NOR gate



n-input NAND gate

In this class, logic circuits that are implemented by interconnecting multiple-input NAND and NOR gates will be referred to as “Static CMOS Logic”

Since the set of NAND gates is complete, any combinational logic function can be realized with the NAND circuit structures considered thus far

Since the set NOR gates is complete, any combinational logic function can be realized with the NOR circuit structures considered thus far

Many logic functions are realized with “Static CMOS Logic” and this is probably the dominant design style used today!

Circuit Structures
Circuit Design

Example 1:

How many transistors are required to realize the function

$$F = \overline{A \bullet \overline{B}} + \overline{A} \bullet C$$

in a basic CMOS process if static NAND and NOR gates are used? Assume A, B and C are available.

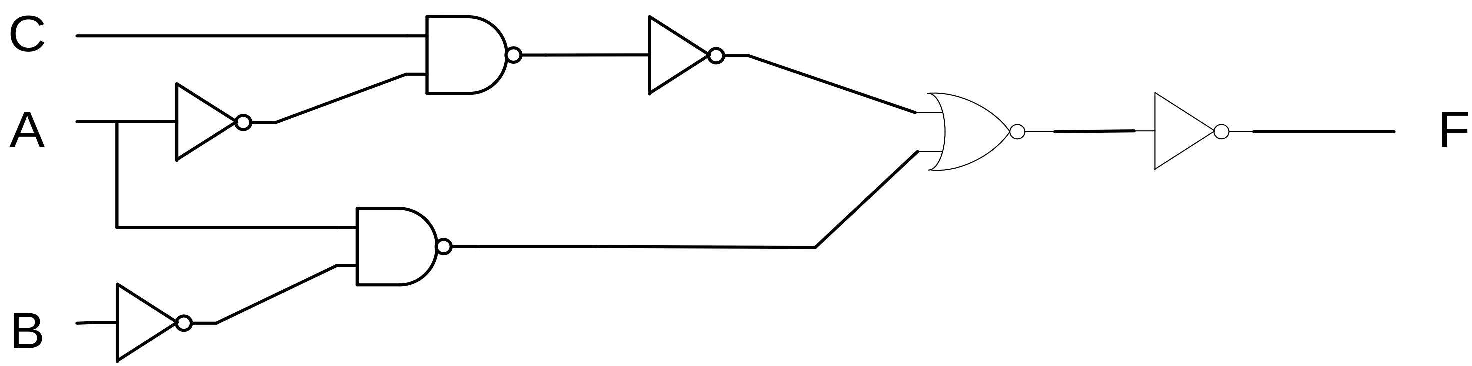
Example 1:

How many transistors are required to realize the function

$$F = \overline{A \bullet \overline{B}} + \overline{A} \bullet C$$

in a basic CMOS process if static NAND and NOR gates are used? Assume A, B and C are available.

Solution:



20 transistors and 5 levels of logic

Example 1:

How many transistors are required to realize the function

$$F = \overline{A \bullet \overline{B}} + \overline{A} \bullet C$$

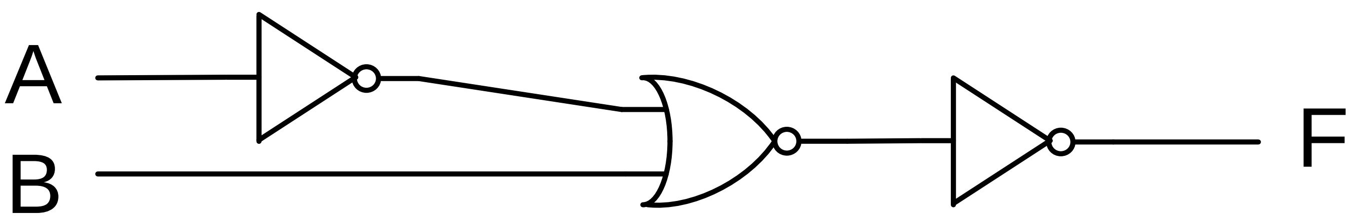
in a basic CMOS process if static NAND and NOR gates are used? Assume A, B and C are available.

Solution (alternative):

From basic Boolean Manipulations

$$F = \overline{A} + \overline{\overline{B}} + \overline{A} \bullet C = \overline{A} + B + \overline{A} \bullet C$$

$$F = \overline{A} \bullet (1 + C) + B = \overline{A} + B$$



8 transistors and 3 levels of logic

Example 1:

How many transistors are required to realize the function

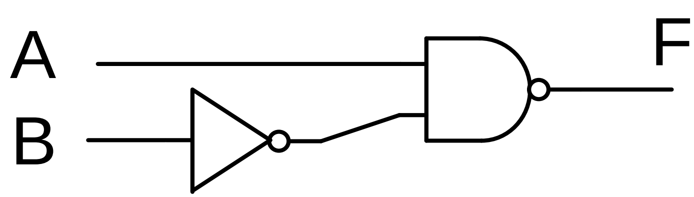
$$F = \overline{A \bullet \overline{B}} + \overline{A} \bullet C$$

in a basic CMOS process if static NAND and NOR gates are used? Assume A, B and C are available.

Solution (alternative): **From basic Boolean Manipulations**

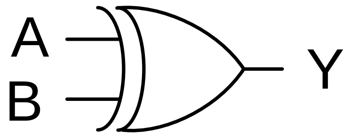
$$F = \overline{A} \bullet (1 + C) + B = \overline{A} + B$$

$$F = \overline{\overline{\overline{A} + B}} = \overline{A \bullet \overline{B}}$$



6 transistors and 2 levels of logic

Example 2: XOR Function

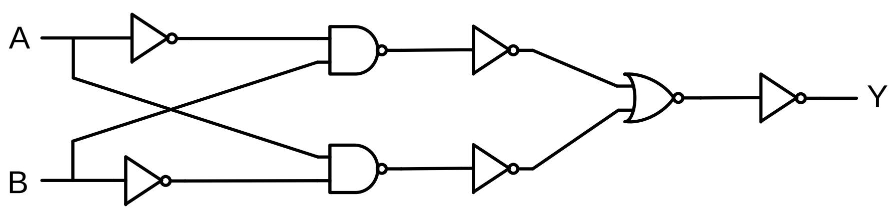


$$Y = A \oplus B$$

A widely-used 2-input Gate

Static CMOS implementation

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



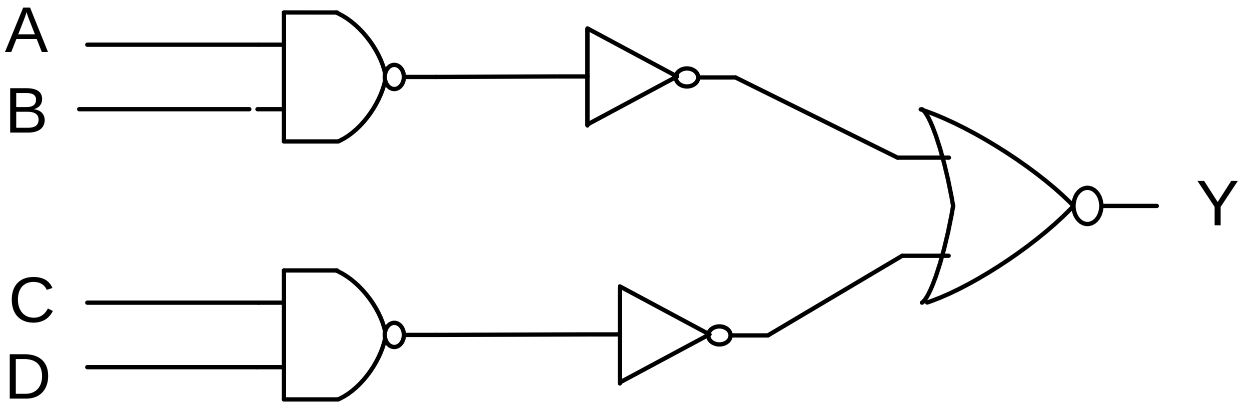
22 transistors   5 levels of logic

Delays unacceptable (will show later) and device count is too large !

Example 3:

$$\mathbf{Y} = \overline{(\mathbf{A} \bullet \mathbf{B}) + (\mathbf{C} \bullet \mathbf{D})}$$

Standard Static CMOS Implementation

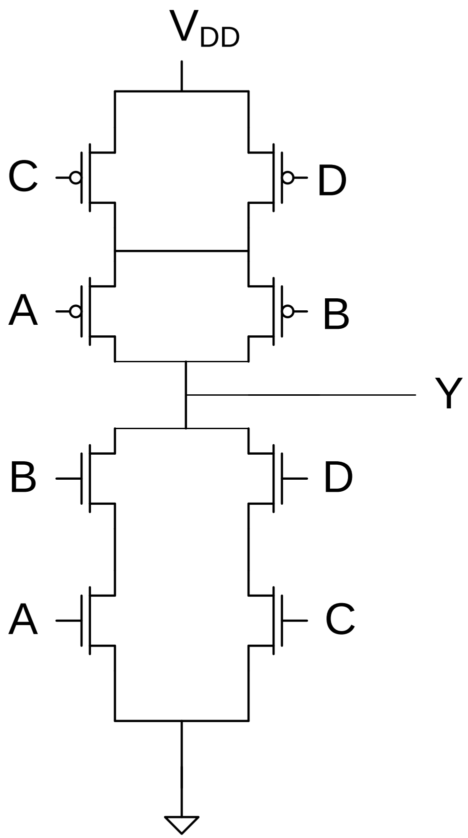


3 levels of Logic

16 Transistors if Basic CMOS Gates are Used

Can the same Boolean functionality be obtained with less transistors?

Observe:



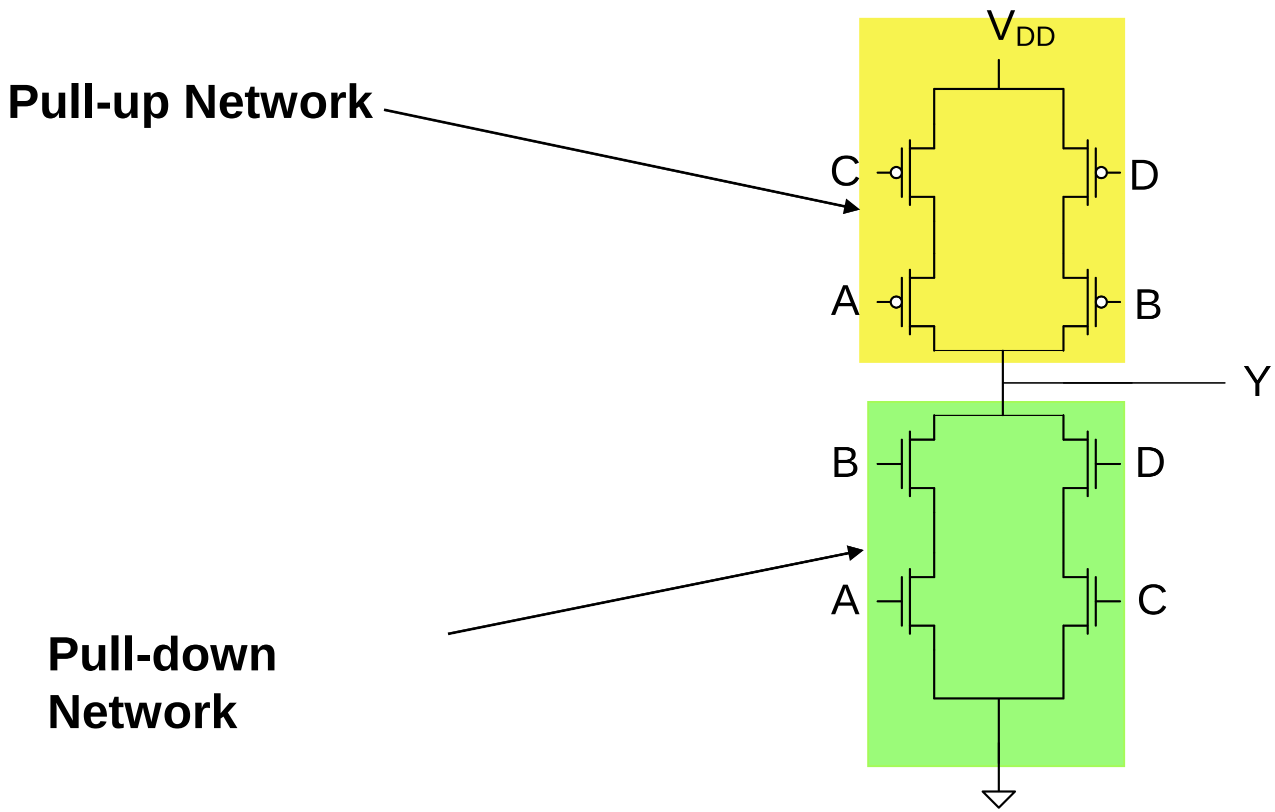
$$Y = \overline{(A \bullet B) + (C \bullet D)}$$

Significant reduction in transistor count and levels of logic for realizing same Boolean function

Termed a "Complex Logic Gate" implementation

Some authors term this a "compound gate"

Complex Logic Gates

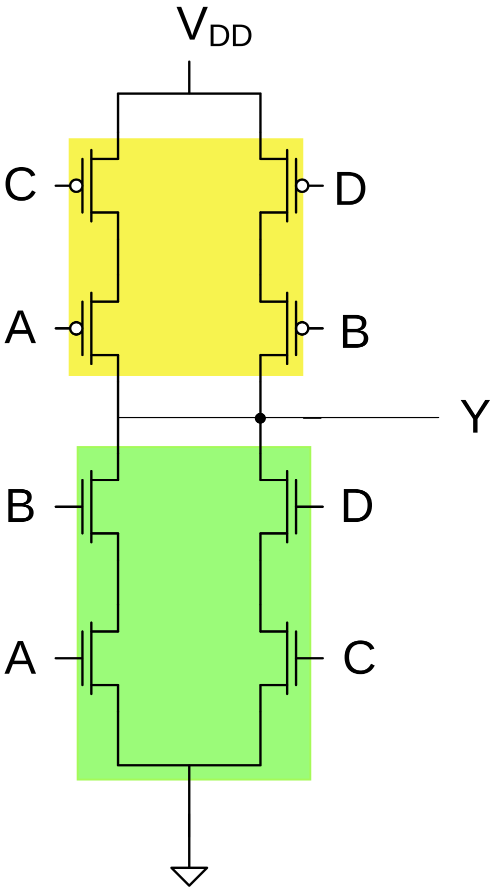


$$Y = \overline{(A \bullet B) + (C \bullet D)}$$

Complex Gates

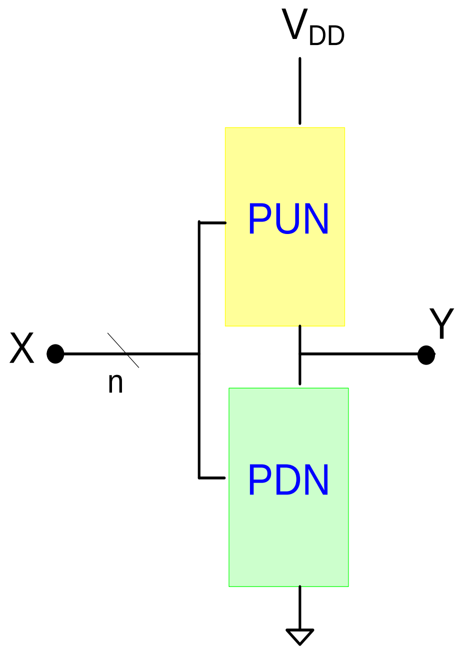
Pull up and pull down network never both conducting

One of the two networks is always conducting



Complex Gates

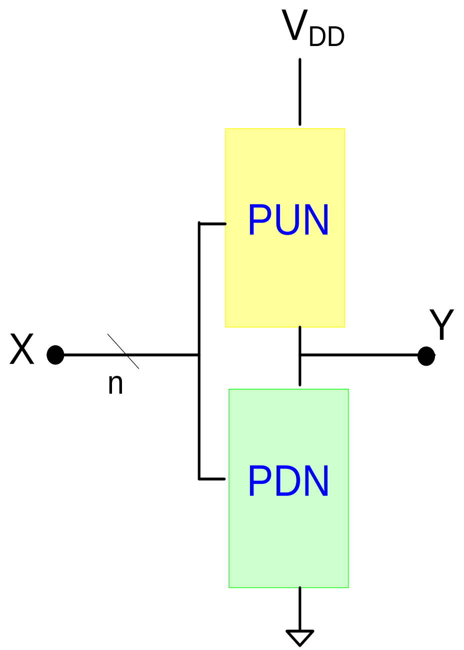
Nomenclature:



When the logic gate shown is not a multiple-input NAND or NOR gate but has Characteristics 1, 2, and 3 above, the gate will be referred to as a Complex Logic Gate

Complex Logic Gates also implement static logic functions and some authors would refer to this as Static CMOS Logic as well but we will make the distinction and refer to this as "Complex Logic Gates"

Complex Gates



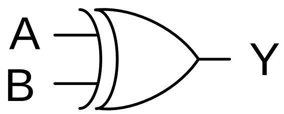
Complex Gate Design Strategy:

1. Implement $\overline{Y}$ in the PDN

2. Implement Y in the PUN (must complement the input variables since p-channel devices are used)

(Y and $\overline{Y}$ often expressed in either SOP or POS form)

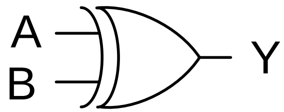
XOR in Complex Logic Gates



$$Y = A \oplus B$$

Will express $\overline{Y}$ and Y in standard SOP or POS form

XOR in Complex Logic Gates



$$Y = A \oplus B$$

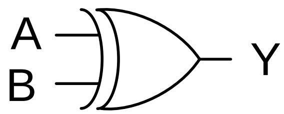
$$Y = A\bar{B} + \bar{A}B$$

$$\bar{Y} = \overline{\left(A\bar{B} + \bar{A}B\right)}$$

$$\bar{Y} = \overline{A\bar{B}} \bullet \overline{\bar{A}B}$$

$$\bar{Y} = \left(\bar{A}+B\right)\cdot\left(A+\bar{B}\right)$$

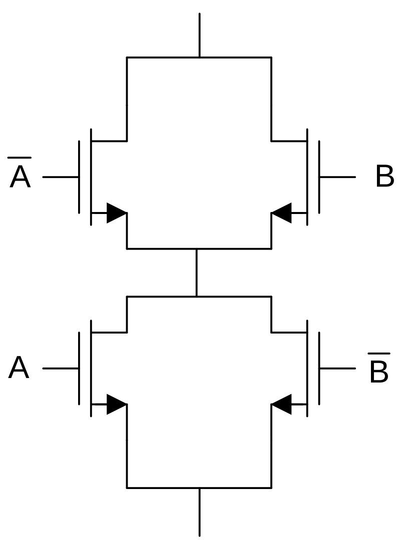
XOR in Complex Logic Gates



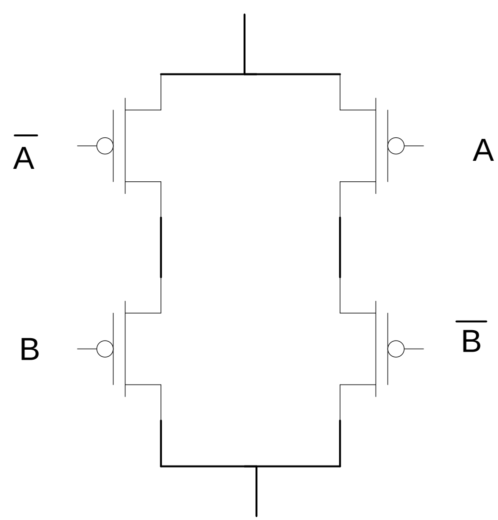
$$Y = A\overline{B} + \overline{A}B$$

$$\overline{Y} = \left(\overline{A}+B\right)\cdot\left(A+\overline{B}\right)$$

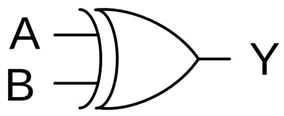
PDN



PUN

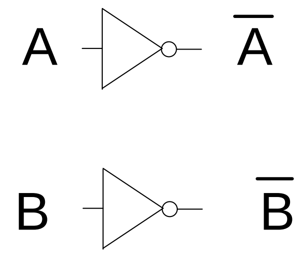


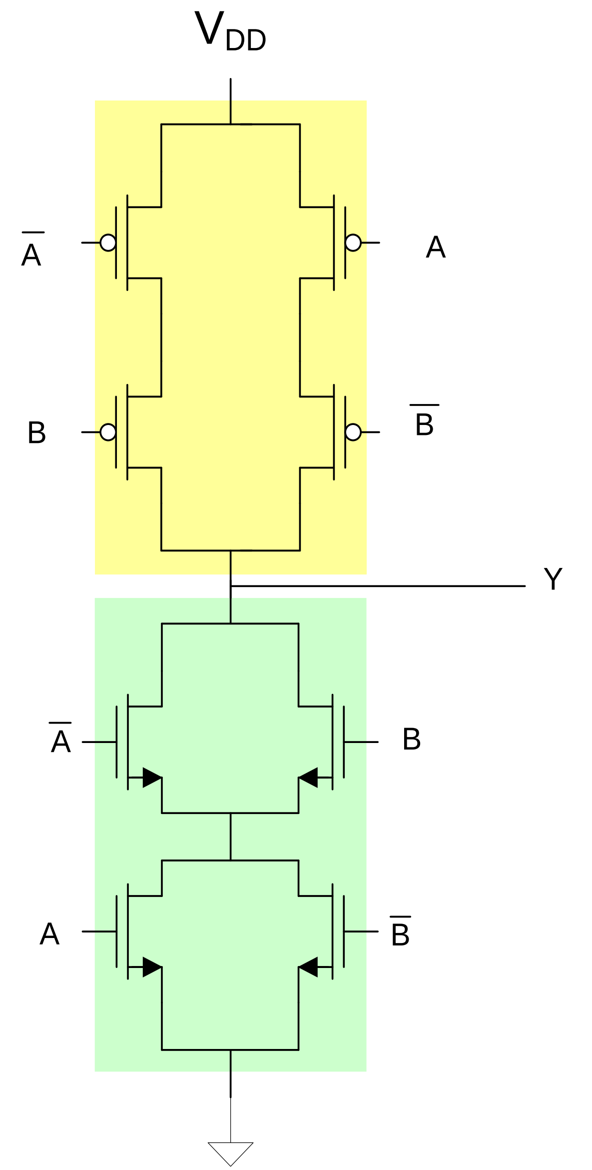
XOR in Complex Logic Gates



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

$$\overline{Y} = \left(\overline{A} + B\right) \bullet \left(A + \overline{B}\right)$$

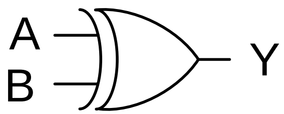




12 transistors and 2 levels of logic

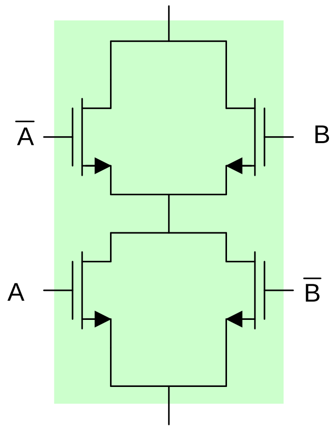
Notice a significant reduction in the number of transistors required

XOR in Complex Logic Gates



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

$$\overline{Y} = \left(\overline{A} + B\right) \bullet \left(A + \overline{B}\right)$$

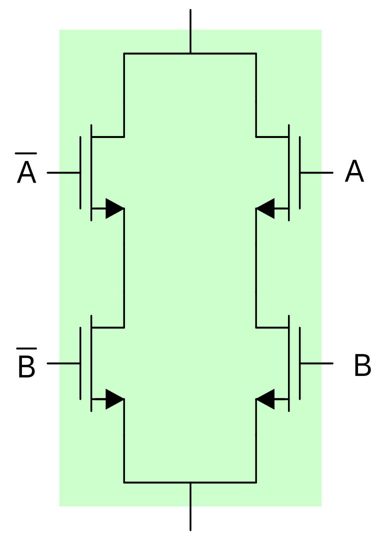


Multiple PU and PD networks can be used

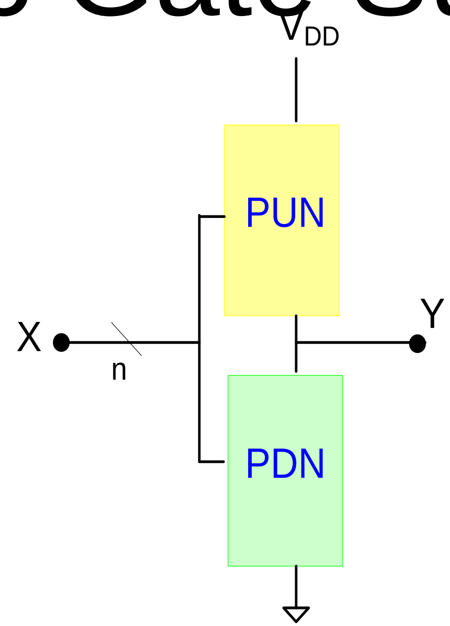
$$\overline{Y} = \left(\overline{A} + B\right) \bullet \left(A + \overline{B}\right)$$

$$= \left(\overline{A} \bullet \left(A + \overline{B}\right)\right) + \left(B \bullet \left(A + \overline{B}\right)\right)$$

$$= \left(\overline{A} \bullet \overline{B}\right) + \left(A \bullet B\right)$$



Complex Logic Gate Summary:



If PUN and PDN satisfy the characteristics:

1. **PU network comprised of p-channel device**
2. **PD network comprised of n-channel device**
3. **One and only one of these networks is conducting at the same time**

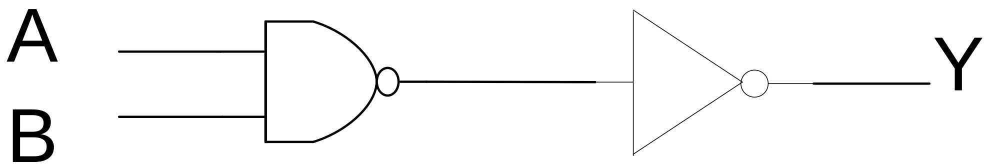
Properties of PU/PD logic of this type (with simple switch-level model):

Rail to rail logic swings
Zero static power dissipation in both Y=1 and Y=0 states
Arbitrarily fast (too good to be true? will consider again with better model)

- Pass Transistor Logic
- Improved Switch-Level Model
- Propagation Delay
- Stick Diagrams
- Technology Files

Consider $\mathbf{Y = A \bullet B}$

Standard CMOS Implementation

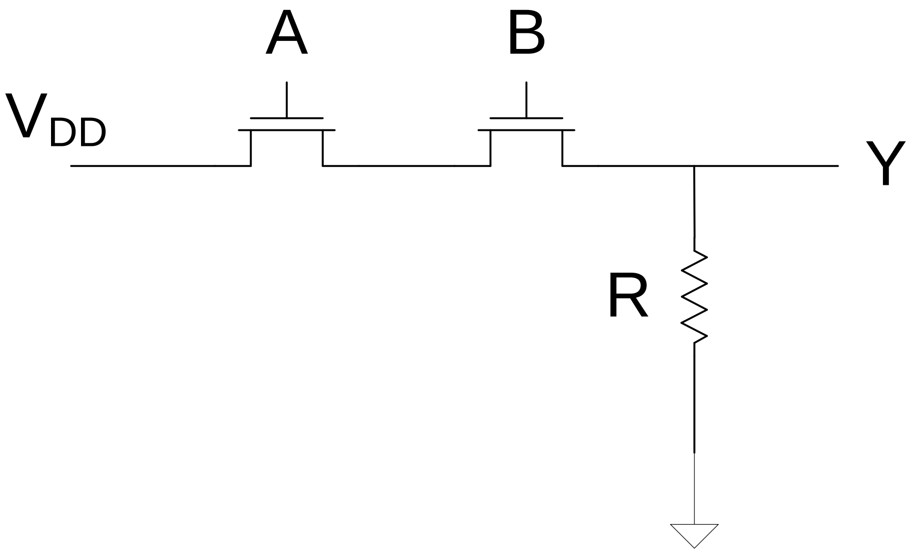


2 levels of Logic

6 Transistors if Basic CMOS Gates are Used

Basic noninverting functions generally require more complexity if basic CMOS gates are used for implementation

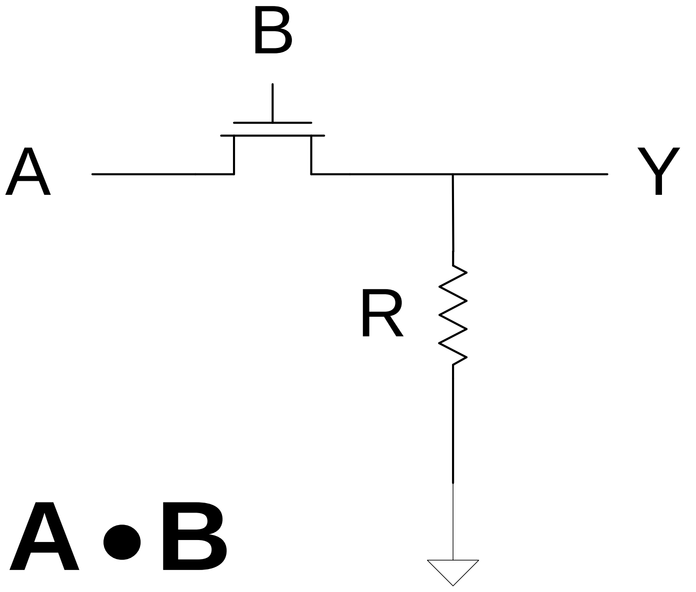
Pass Transistor Logic



$$\mathbf{Y} = \mathbf{A} \bullet \mathbf{B}$$

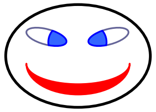
Requires only 2 transistors rather than 6 for a standard CMOS gate (and a resistor).

Pass Transistor Logic



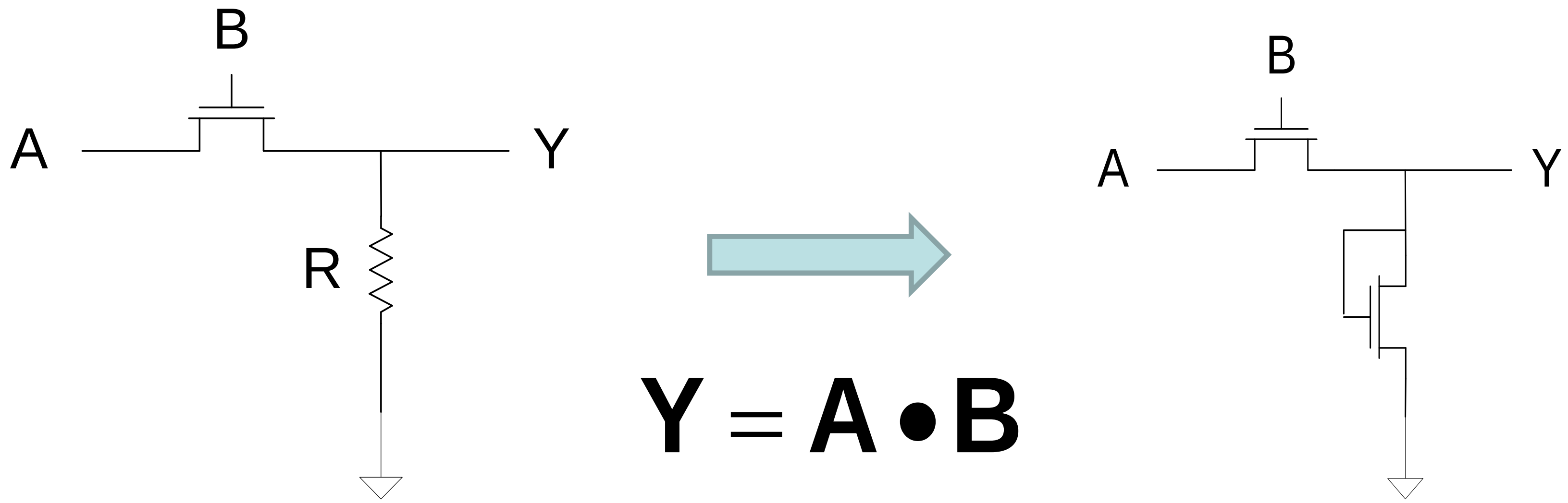
$$\mathbf{Y} = \mathbf{A} \bullet \mathbf{B}$$

Even simpler pass transistor logic implementations are possible

Requires only 1 transistor (and a resistor). 

Will see later that the area of a single practical resistor for this circuit may be comparable to that needed for hundreds or even thousands of transistors 

Pass Transistor Logic



$$\mathbf{Y = A \bullet B}$$

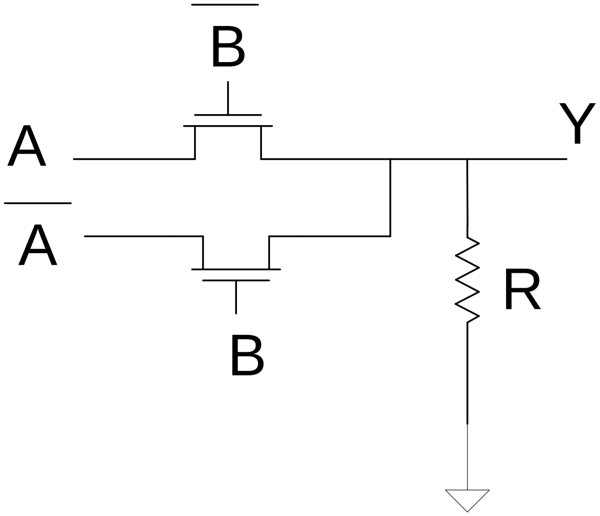
- **May be able to replace resistor with transistor** **(one of several ways shown)**

- **But high logic level can not be determined with existing device model** **(or even low logic level for circuit on right)**

- **Power dissipation can not be determined with existing device model for circuit on right**

Better device model is needed (Power? Signal Swing? Speed?)

Pass Transistor Logic

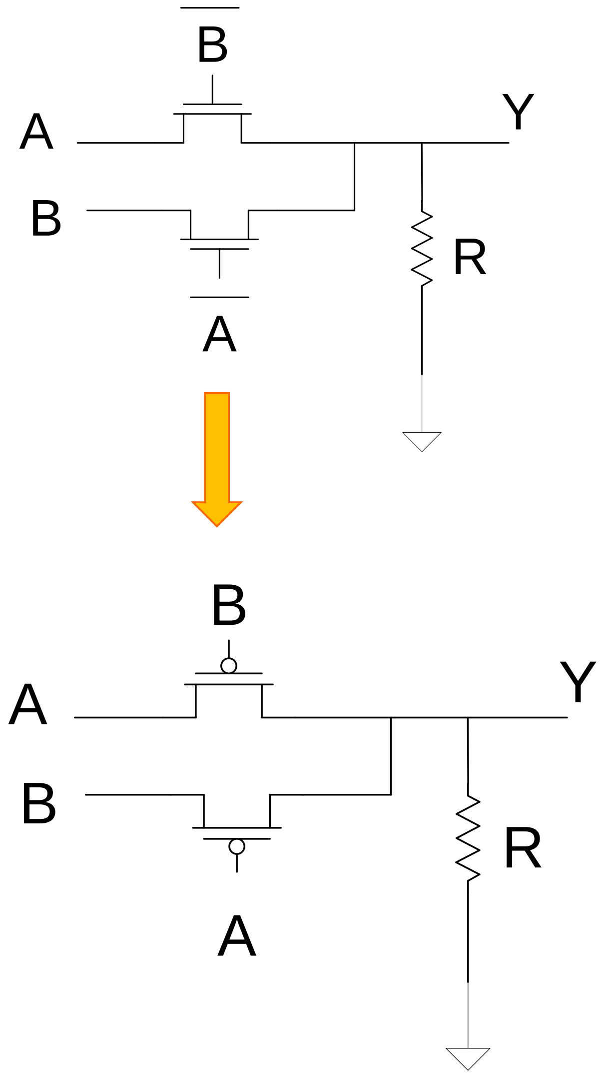


$$Y = A \oplus B$$

6 transistors, 1 resistor, two levels of logic

(the 4 transistors in the two inverters are not shown)

Pass Transistor Logic

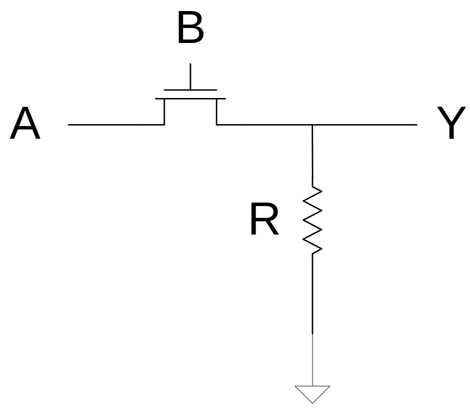


$$Y = A \oplus B$$

$$Y = A \oplus B$$

2 transistors, 1 resistor, one level of logic

Pass Transistor Logic



$$\mathbf{Y} = \mathbf{A} \bullet \mathbf{B}$$

Requires only 1 transistor (and a resistor)

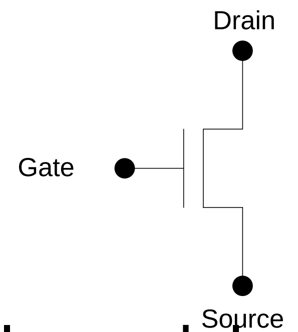
- **Pass transistor logic can offer significant reductions in complexity for some functions (particularly noninverting)**
- **Resistor may require more area than several hundred or even several thousand transistors**
- **Signal levels may not go to V_{DD} or to 0V**
- **Static power dissipation may not be zero**
- **Signals may degrade unacceptably if multiple gates are cascaded**
- **"resistor" often implemented with a transistor to reduce area but signal swing and power dissipation problems still persist**
- **Pass transistor logic <u>is</u> widely used**

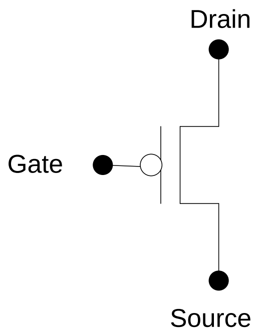
Logic Design Styles

- Several different logic design styles are often used throughout a given design (3 considered thus far)
  - Static CMOS
  - Complex Logic Gates
  - Pass Transistor Logic
- The designer has complete control over what is placed on silicon and governed only by cost and performance
- New logic design strategies have been proposed recently and others will likely emerge in the future
- The digital designer needs to be familiar with the benefits and limitations of varying logic styles to come up with a good solution for given system requirements

- Pass Transistor Logic
- Improved Switch-Level Model
- Propagation Delay
- Stick Diagrams
- Technology Files

MOSFET Modeling

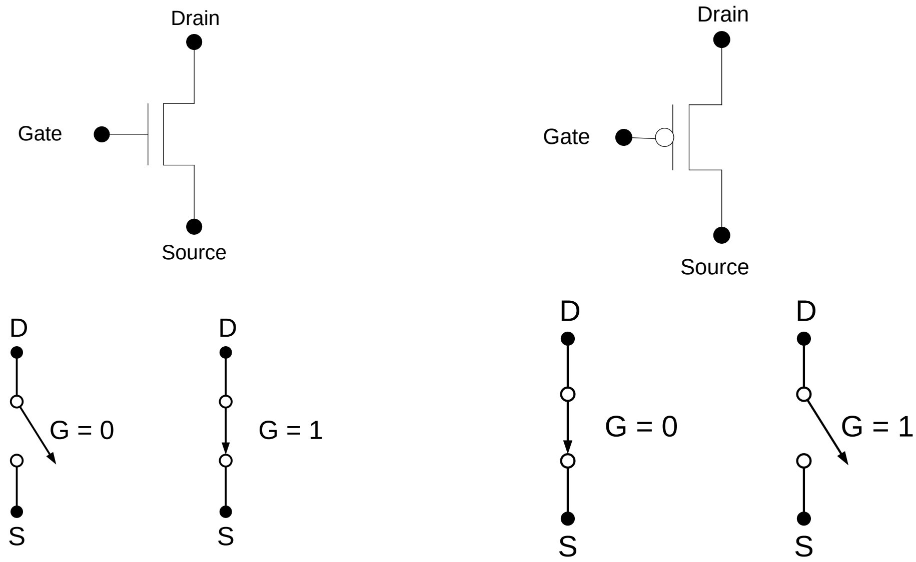




- Simple model of MOSFET was developed (termed switch-level model)
- Simple gates designed in CMOS Process were introduced
  - Some have zero power dissipation
  - Some have or appeared to have rail to rail logic voltage swings
  - All appeared to be Infinitely fast
  - Logic levels of some can not be predicted with simple model
  - Simple model is not sufficiently accurate to provide insight relating to some of these properties
- MOSFET modeling strategy
  - hierarchical model structure will be developed
  - generally use simplest model that can be justified

MOS Transistor Models

1, Switch-Level model



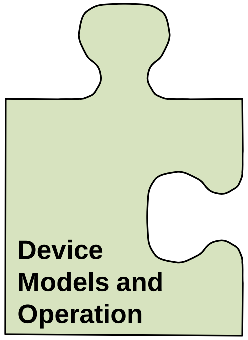
Advantages:

Simple, does not require understanding of semiconductor properties, does not depend upon process, adequate for understanding basic operation of many digital circuits

Limitations:

Does not provide timing information (surfaced when looking at static CMOS circuits, and several others that have not yet become apparent from the applications that have been considered) and can not support design of "resistor" used in Pass Transistor Logic

Improved Device Models



With the simple switch-level model, it was observed that basic static CMOS logic gates have the following three properties:

- **Rail to rail logic swings**
- **Zero static power dissipation in both Y=1 and Y=0 states**
- **Arbitrarily fast (too good to be true? will consider again with better model)**

It can be shown that the first two properties are nearly satisfied in actual fabricated circuits with p-channel/n-channel PU/PD logic but though the circuits are fast, they are observably not arbitrarily fast

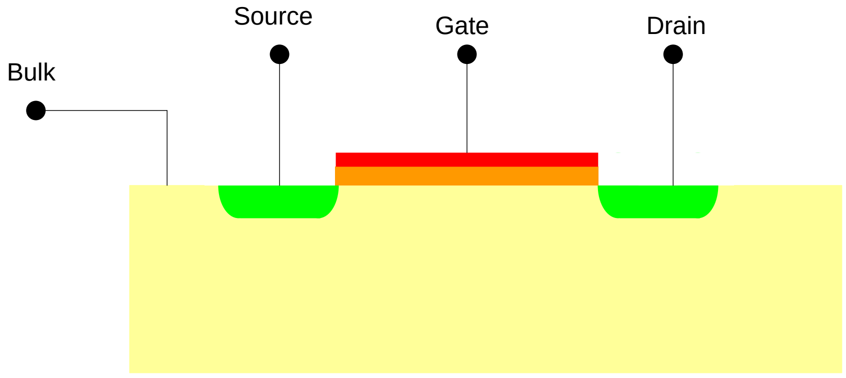
None of these properties are observed for some logic styles such as Pass Transistor Logic

Will now extend switch-level model to predict speed of basic gates in static CMOS and logic levels and power dissipation in PTL

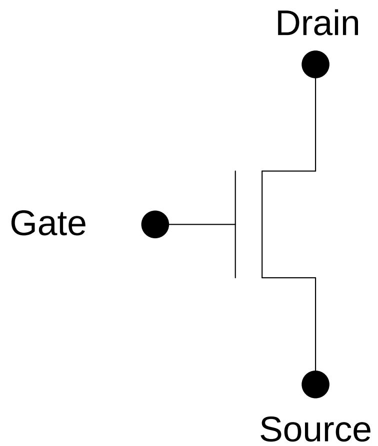
Recall

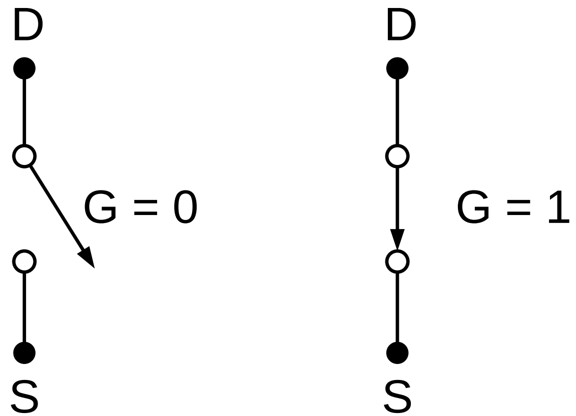
MOS Transistor

Qualitative Discussion of n-channel Operation



n-channel MOSFET

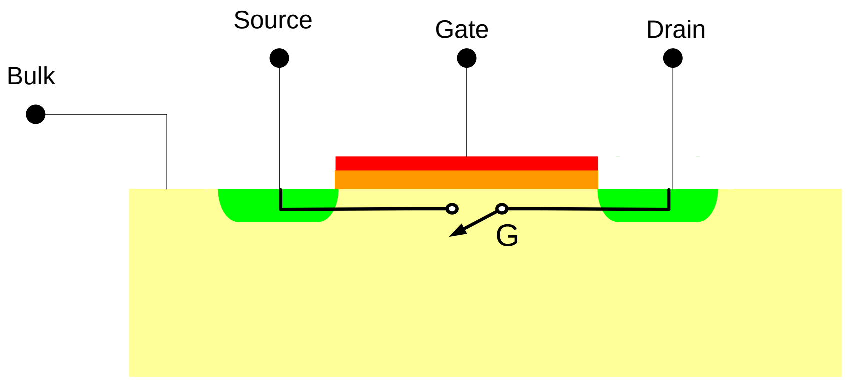




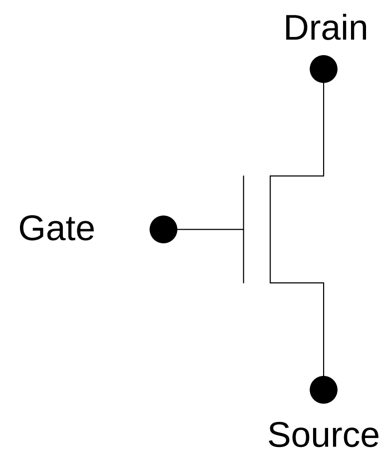
This was the first model introduced and was termed the basic switch-level mode

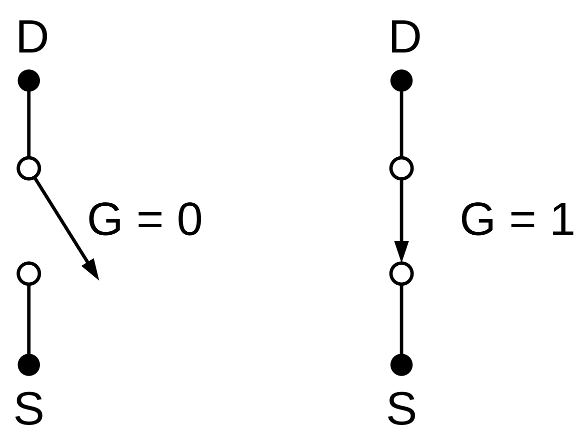
MOS Transistor

Qualitative Discussion of n-channel Operation



n-channel MOSFET

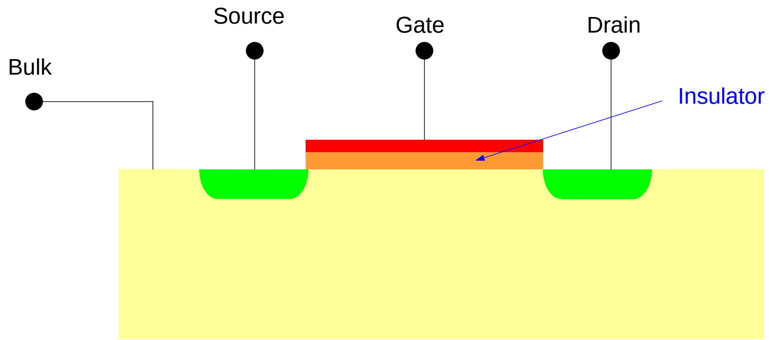




Conceptual view of basic switch-level model

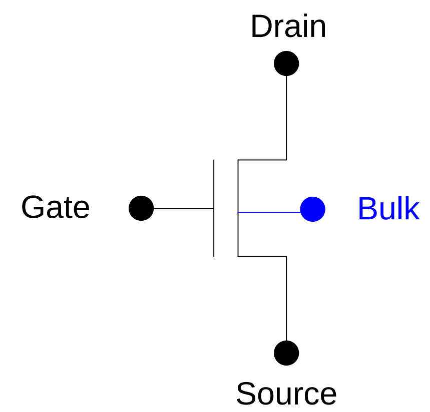
MOS Transistor

Qualitative Discussion of n-channel Operation

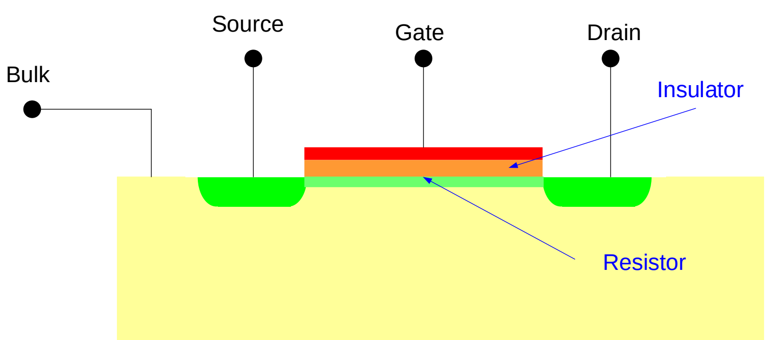


n-channel MOSFET

For V_{GS} small



MOSFET actually 4-terminal device



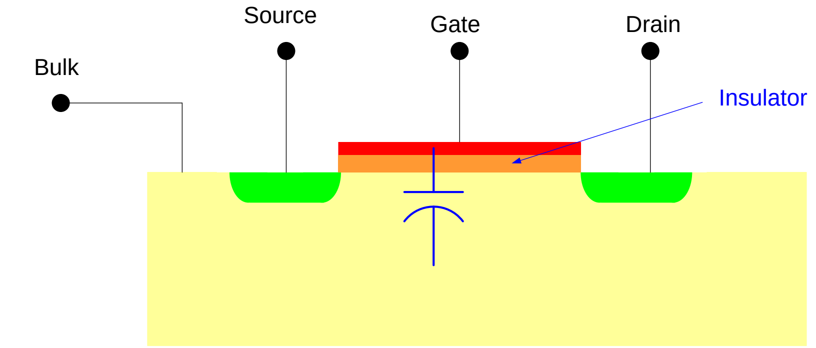
n-channel MOSFET

For V_{GS} large

- Region under gate termed the “channel”
- When “resistor” is electrically created, it is termed an “inversion region”

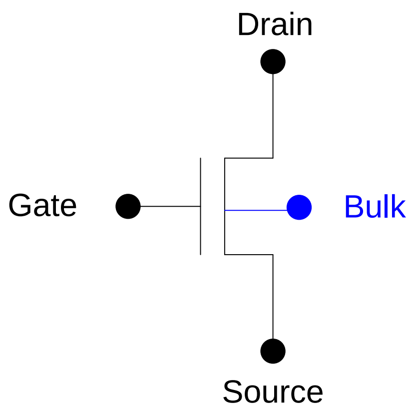
MOS Transistor

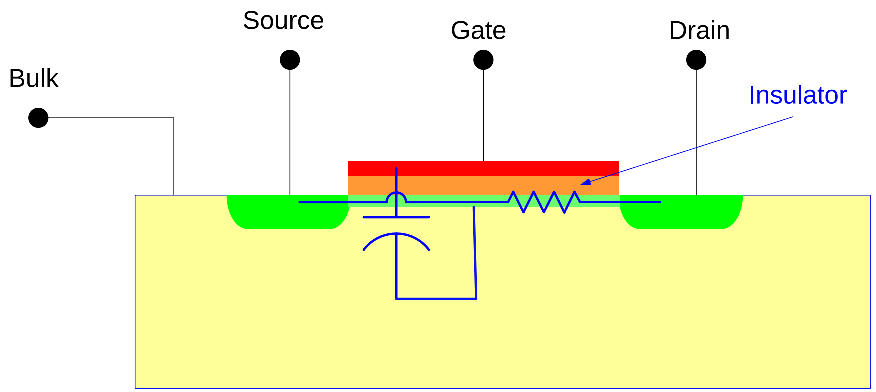
Qualitative Discussion of n-channel Operation



n-channel MOSFET

For V_{GS} small





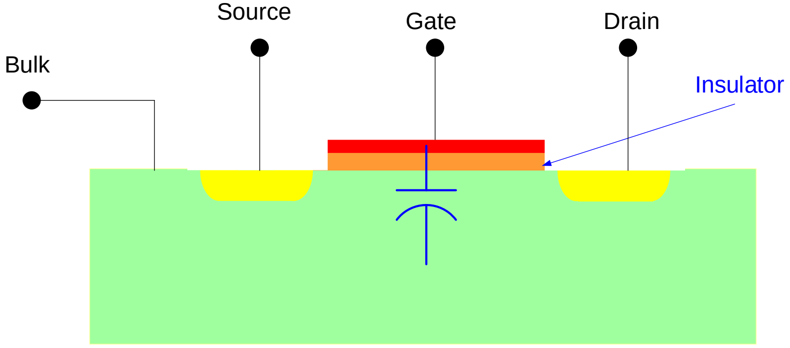
n-channel MOSFET

For V_{GS} large

- Electrically created inversion layer forms a "thin "film" resistor
- Capacitance from gate to <u>channel region </u>is distributed
- Lumped capacitance much easier to work with

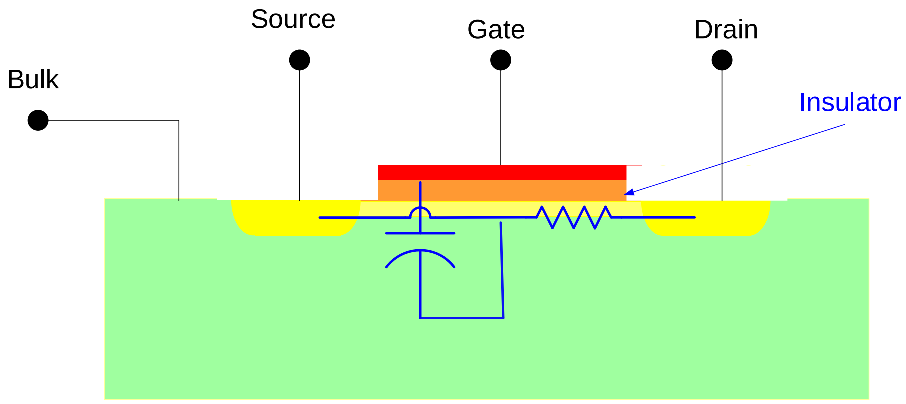
MOS Transistor

Qualitative Discussion of p-channel Operation

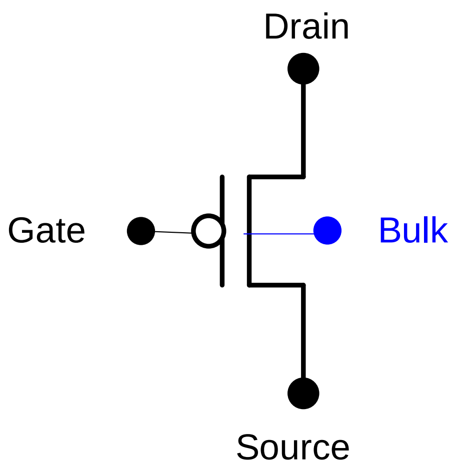


p-channel MOSFET

For $|V_{GS}|$ small



p-channel MOSFET

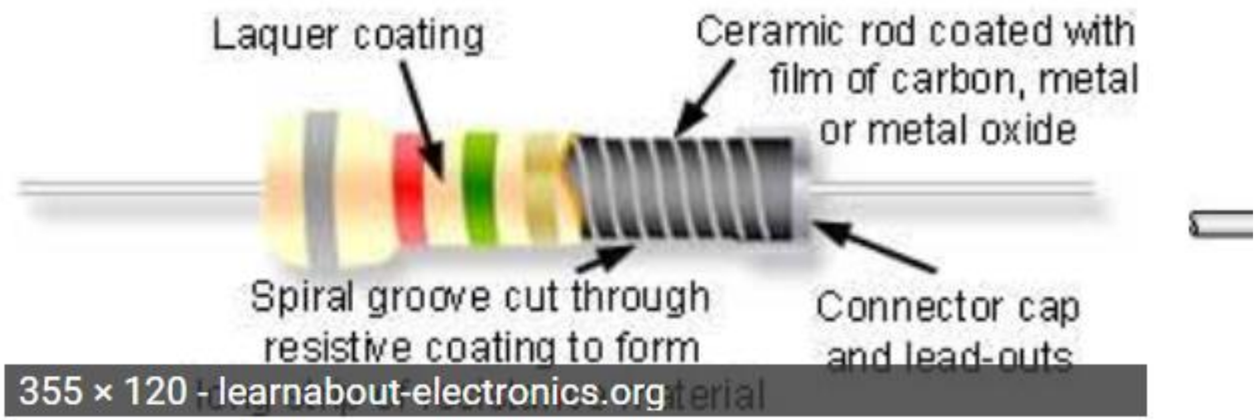


For $|V_{GS}|$ large

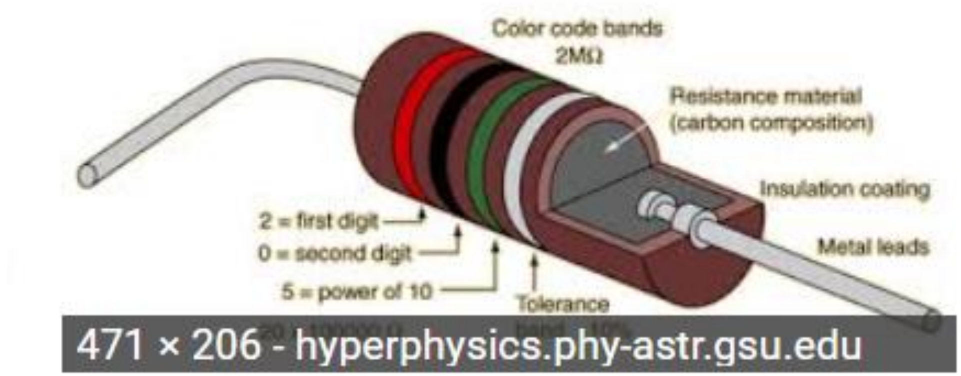
- Electrically created inversion layer forms a “thin “film” resistor
- Capacitance from gate to <u>channel region </u>is distributed
- Lumped capacitance much easier to work with

Discrete Resistors often use thin films too though not electrically created

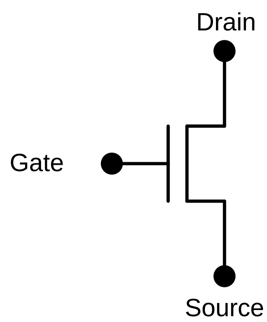
- **Thin-film spiral wound**

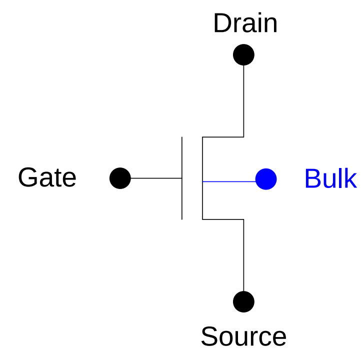


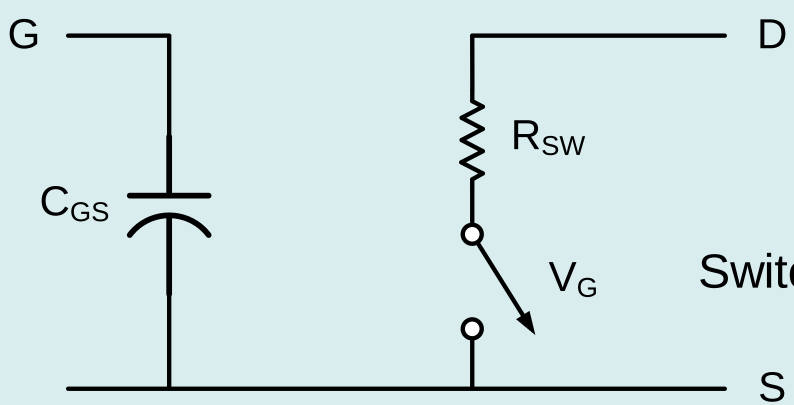
- **Carbon composition**



Improved Switch-Level Model





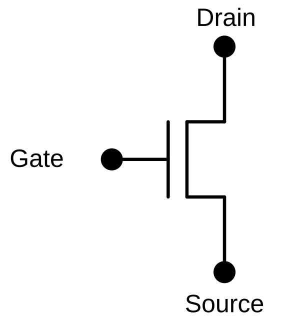


Switch closed for V_{GS}="1"

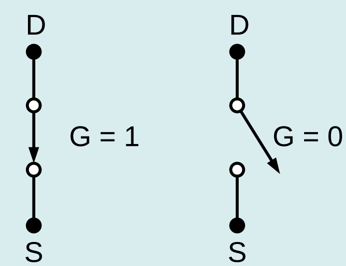
Switch-level model including gate capacitance and channel resistance

- Connect the gate capacitance to the source to create lumped model
- Still neglect bulk connection

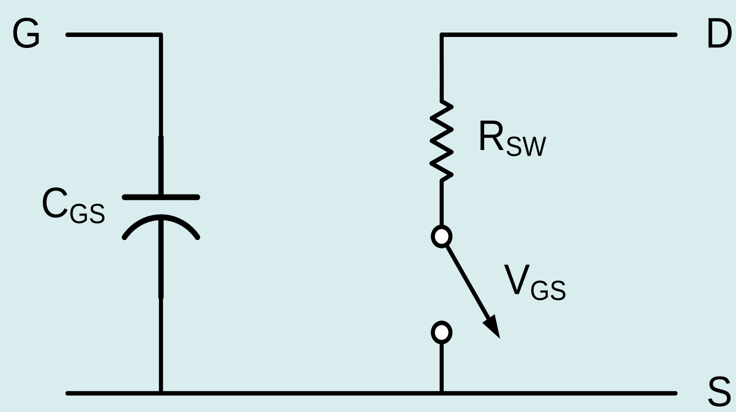
Improved Switch-Level Model



n-channel



Switch-level model

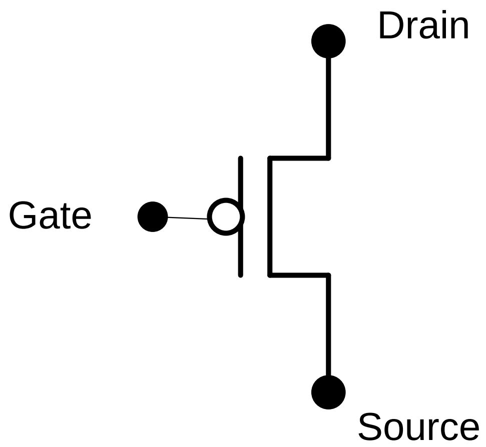


Switch closed for V_{GS}= large

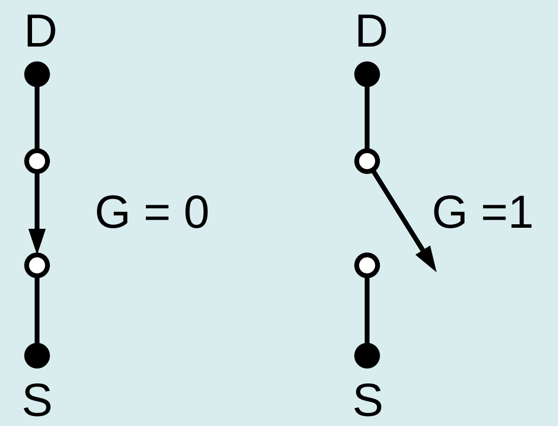
Alt: switch closed for "G = 1"

Switch-level model including gate capacitance and channel resistance

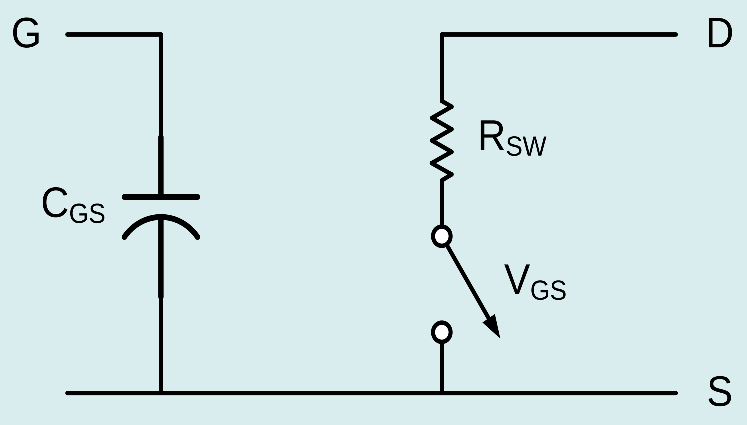
Improved Switch-Level Model



p-channel



Switch-level model

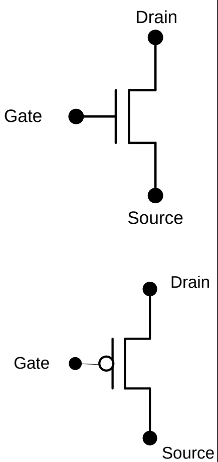


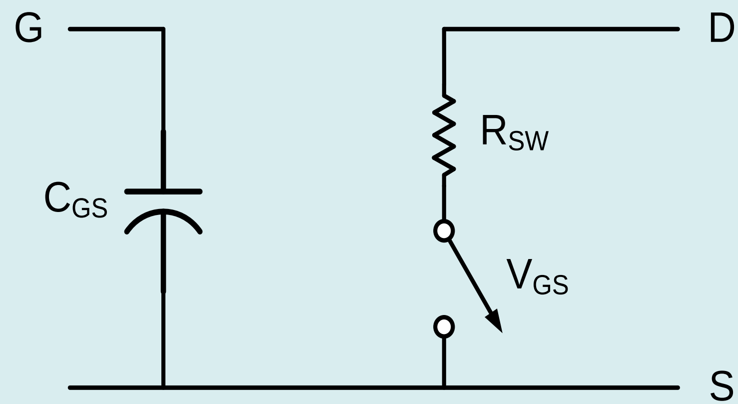
Switch closed for $|V_{GS}|$ large

Alt: If S near V_{DD}, closed for G=0

Switch-level model including gate capacitance and channel resistance

Improved Switch-Level Model





Switch closed for G =1 , n-ch
Switch closed for G =0 , p-ch

Switch-level model including gate capacitance and channel resistance

C_{GS} and R_{SW} dependent upon device sizes and process

For minimum-sized devices in a 0.5u process with V_{DD}=5V

$$\mathbf{C_{GS} \cong 1.5fF} \qquad \mathbf{R_{sw} \cong \left.\begin{matrix} 2K\Omega \;\; n-channel \\ 6K\Omega \;\; p-channel \end{matrix}\right\}}$$

Considerable emphasis will be placed upon device sizing to manage C_{GS} and R_{SW}

End of Lecture 6