Chapter 1
Introduction to CMOS Circuit Design

Jin-Fu Li
Advanced Reliable Systems (ARES) Lab.
Department of Electrical Engineering
National Central University
Jhongli, Taiwan

Outline

□ Introduction

□ MOS Transistor Switches

□ CMOS Logic

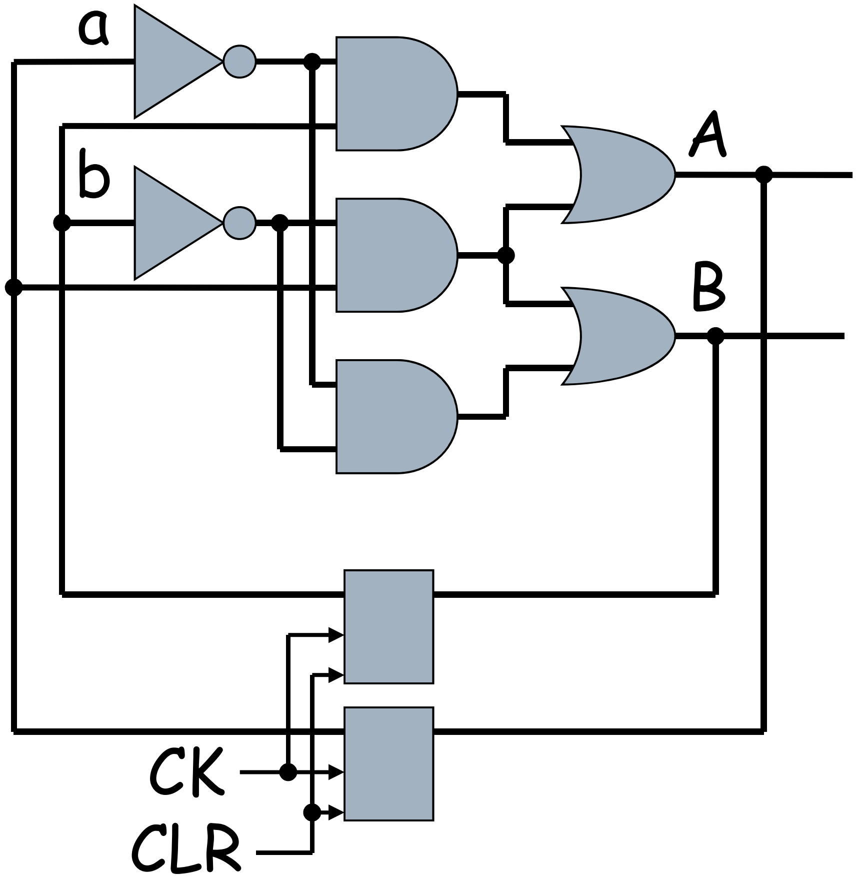
□ Circuit and System Representation

Binary Counter

Present state		Next state	
a	b	A	B
0	0	0	1
0	1	1	0
1	0	1	1
1	1	0	0

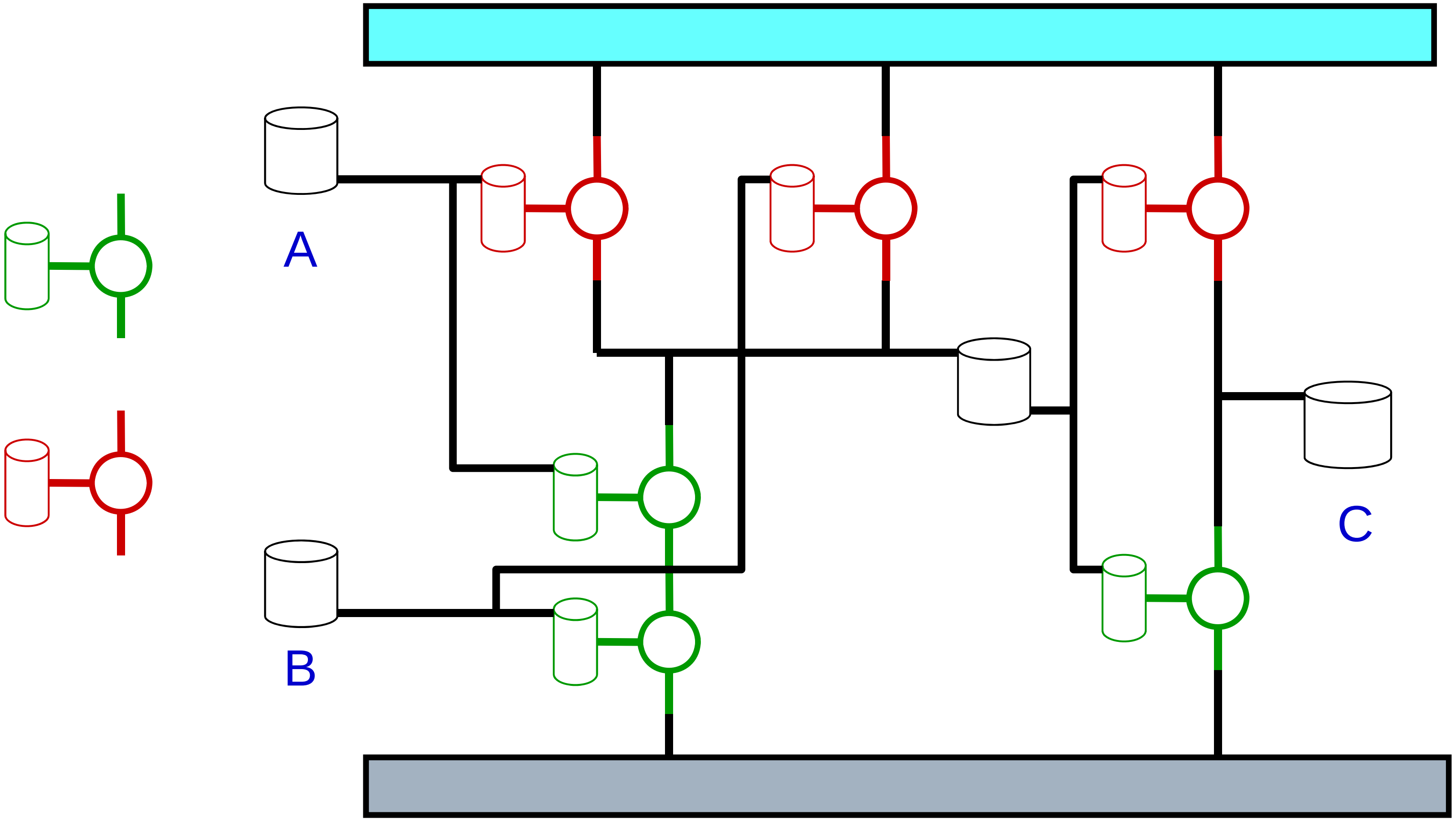
$A = a'b + ab'$

$B = a'b' + ab'$



Source: Prof. V. D. Agrawal

1-bit Multiplier



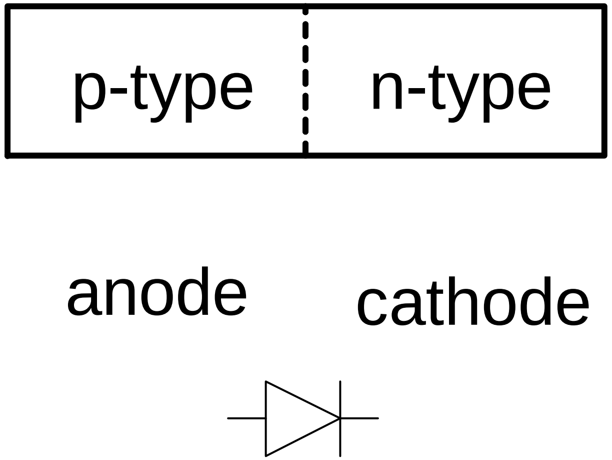
C=AxB

Switch: MOSFET

- MOSFETs are basic electronic devices used to direct and control logic signals in IC design
  - **MOSFET**: **M**etal-**O**xide-**S**emiconductor **F**ield-**E**ffect **T**ransistor
  - N-type MOS (NMOS) and P-type MOS (PMOS)
  - Voltage-controlled switches
- A MOSFET has four terminals: gate, source, drain, and substrate (body)
- Complementary MOS (CMOS)
  - Using two types of MOSFETs to create logic networks
  - NMOS & PMOS

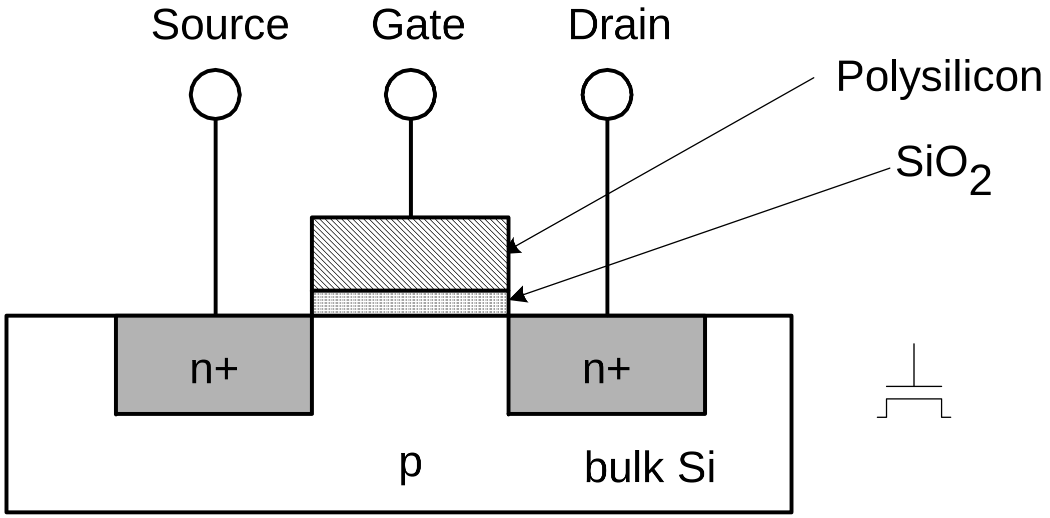
P-N Junctions

- A junction between p-type and n-type semiconductor forms a diode.
- Current flows only in one direction



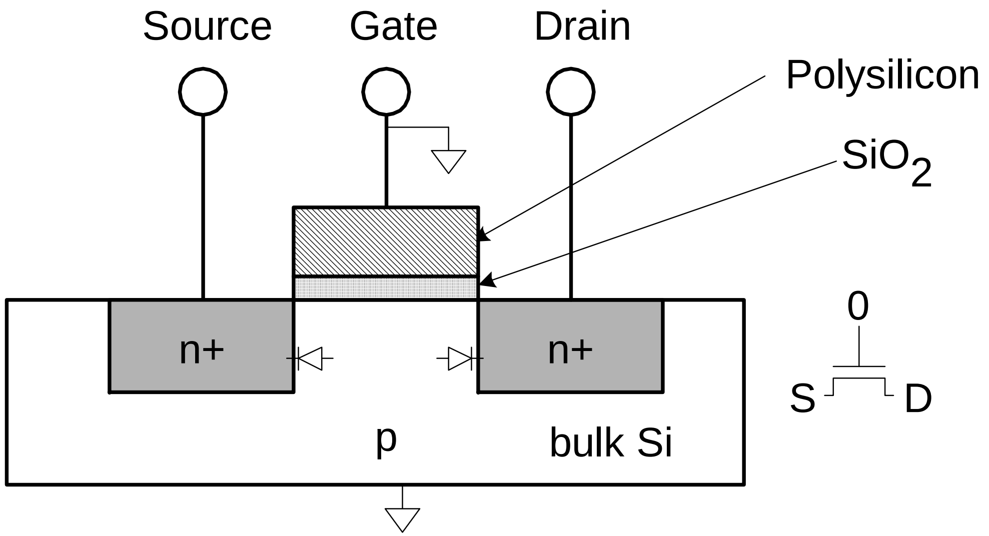
NMOS Transistor

- Four terminals: gate, source, drain, body
- Gate-oxide-body stack looks like a capacitor
  - Gate and body are conductors
  - SiO_2 (oxide) is a very good insulator
  - Called metal-oxide-semiconductor (MOS) capacitor
  - Even though gate is no longer made of metal



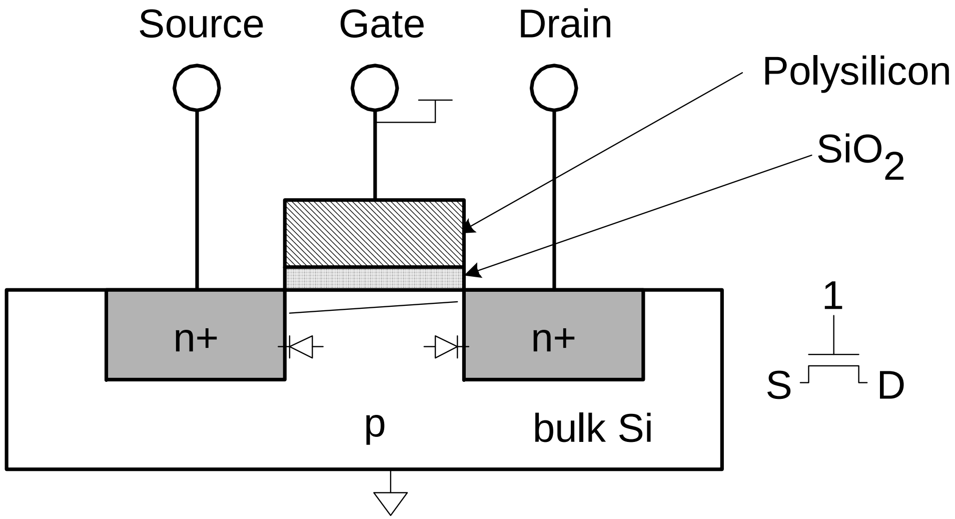
NMOS Operations

- □ Body is commonly tied to ground (0 V)
- □ When the gate is at a low voltage:
  - P-type body is at low voltage
  - Source-body and drain-body diodes are OFF
  - No current flows, transistor is OFF



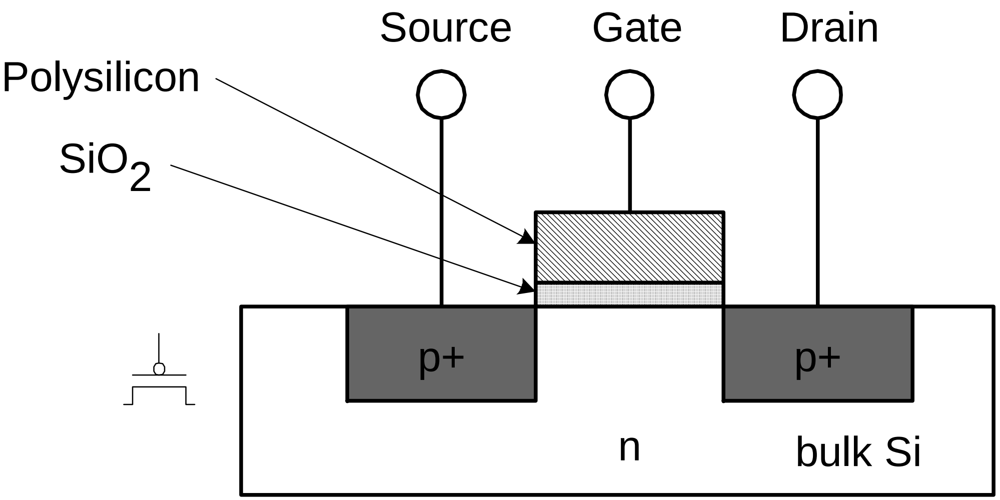
NMOS Operations (Cont.)

- When the gate is at a high voltage:
  - Positive charge on gate of MOS capacitor
  - Negative charge attracted to body
  - Inverts a channel under gate to n-type
  - Now current can flow through n-type silicon from source through channel to drain, transistor is ON



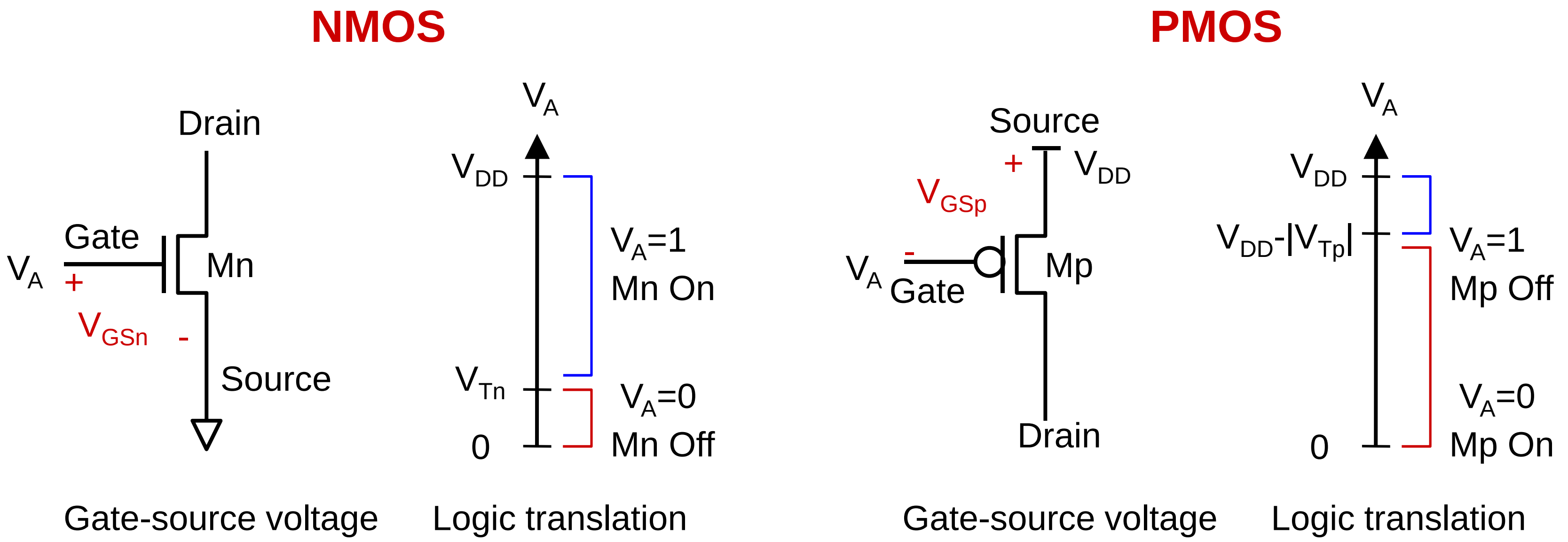
PMOS Operations

- Similar, but doping and voltages reversed
  - Body tied to high voltage (V_{DD})
  - Gate low: transistor ON
  - Gate high: transistor OFF
  - Bubble indicates inverted behavior



Threshold Voltage

- □ Every MOS transistor has a characterizing parameter called the threshold voltage V_T
- □ The specific value of V_T is established during the manufacturing process
- □ Threshold voltage of an NMOS and a PMOS


NMOS

PMOS

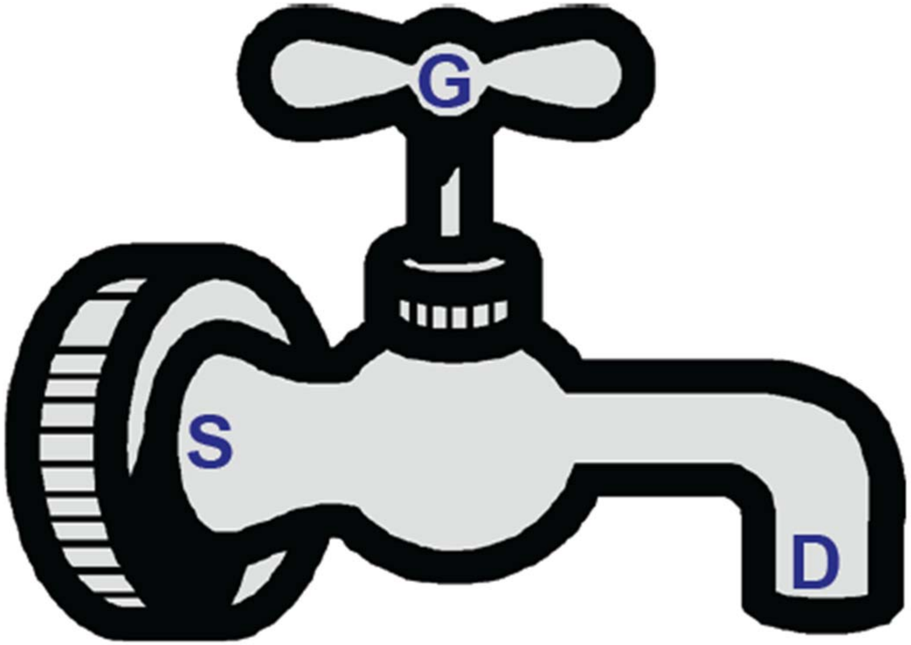
Gate-source voltage

Logic translation

Gate-source voltage

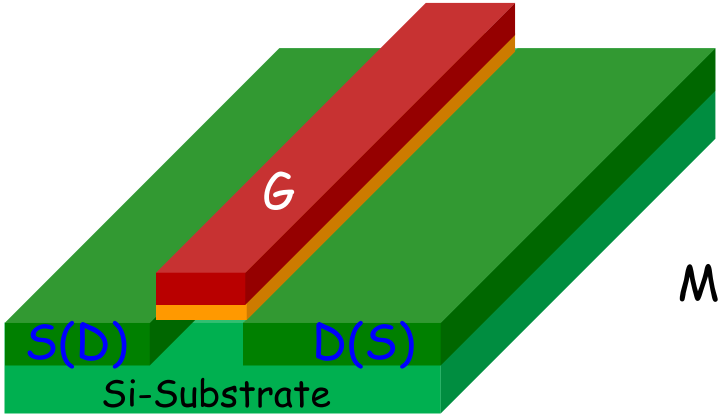
Logic translation

MOS Transistor is Like a Tap…

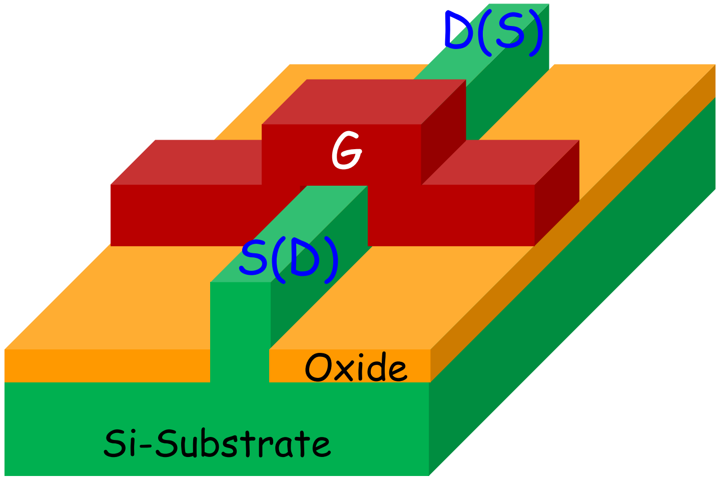


Source: Prof. Banerjee, ECE, UCSB

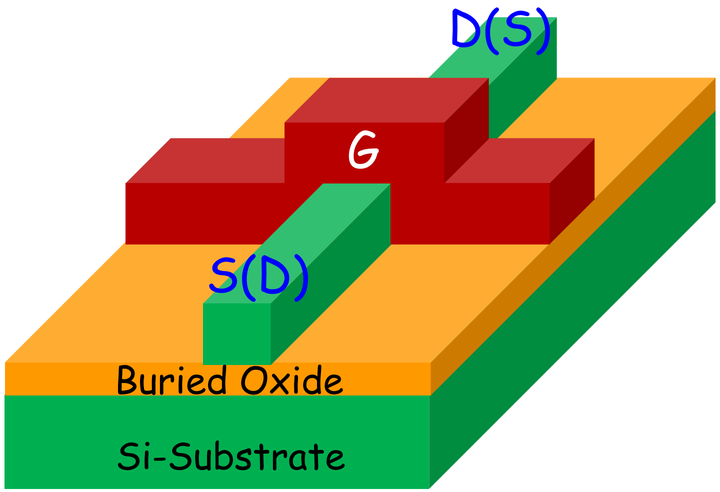
MOSFET & FinFET



MOSFET



Bulk FinFET

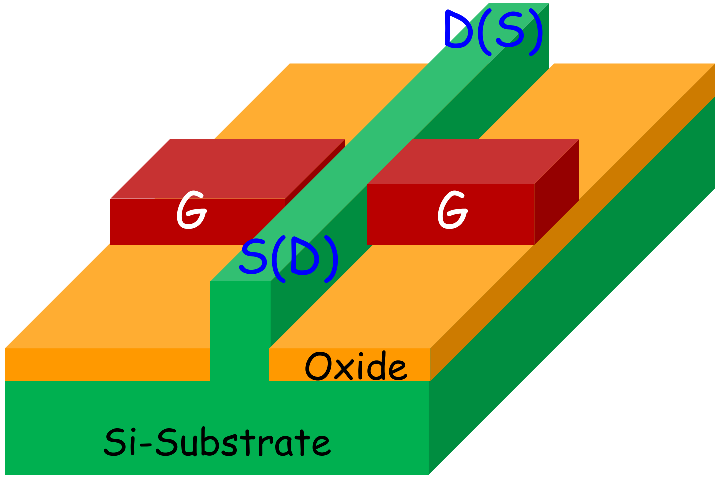


SOI FinFET

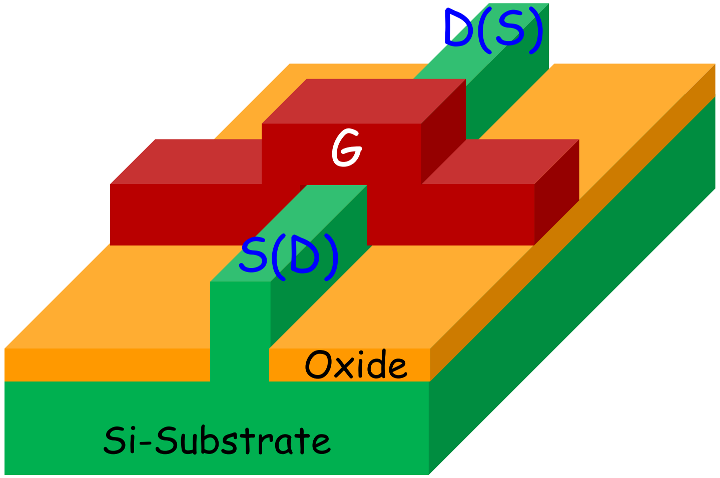
IG & SG FinFETs

□ According to the gate structure, FinFET can be classified as

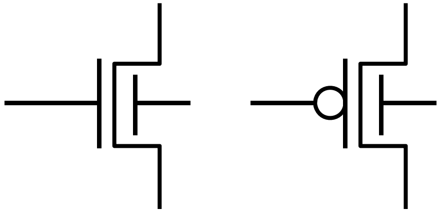
- Independent-Gate (IG) FinFET
- Short-Gate (SG) FinFET

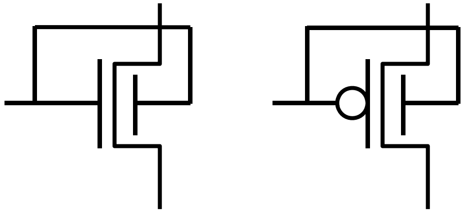


IG FinFET



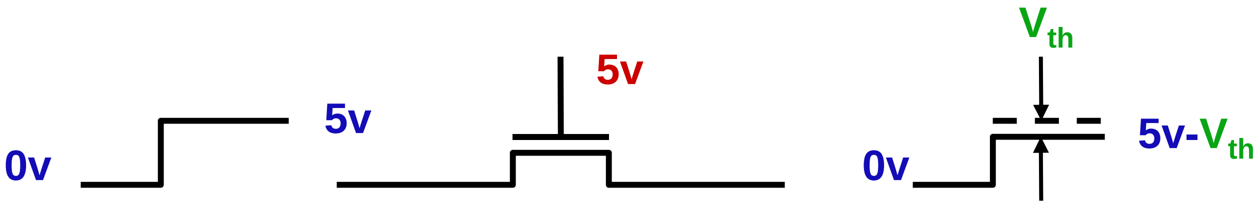
SG FinFET



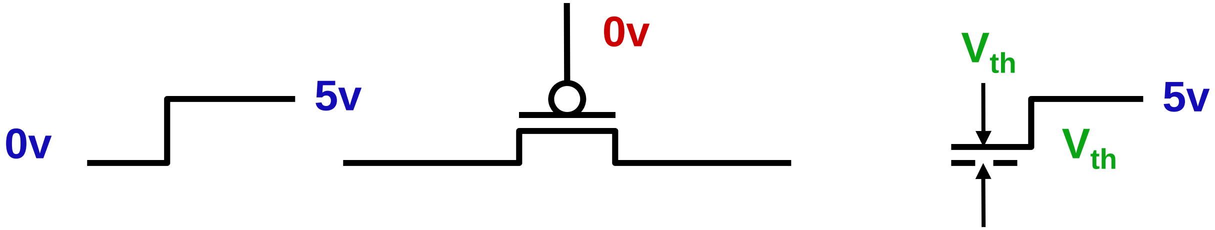


MOS Switches

- □ NMOS symbol and characteristics



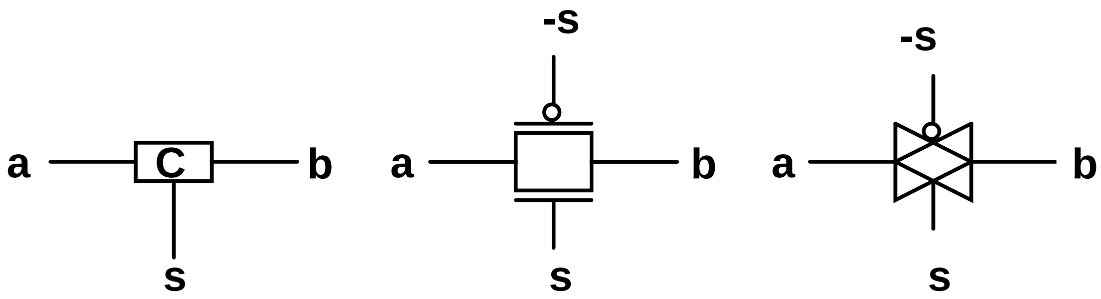
- □ PMOS symbol and characteristics



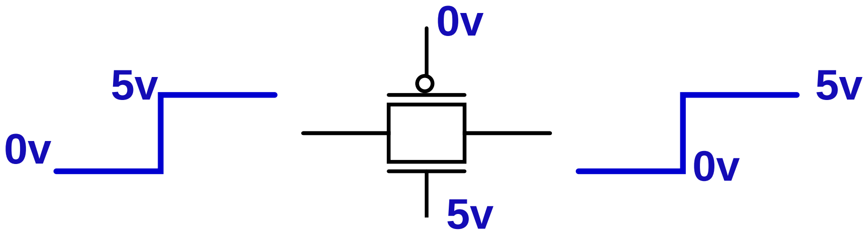
CMOS Switch

- A complementary CMOS switch
  - Transmission gate

Symbols

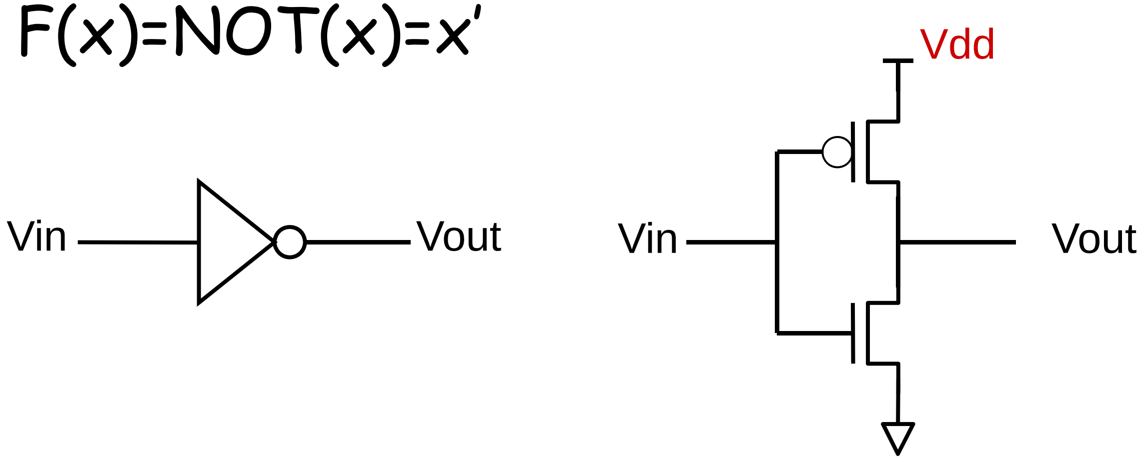


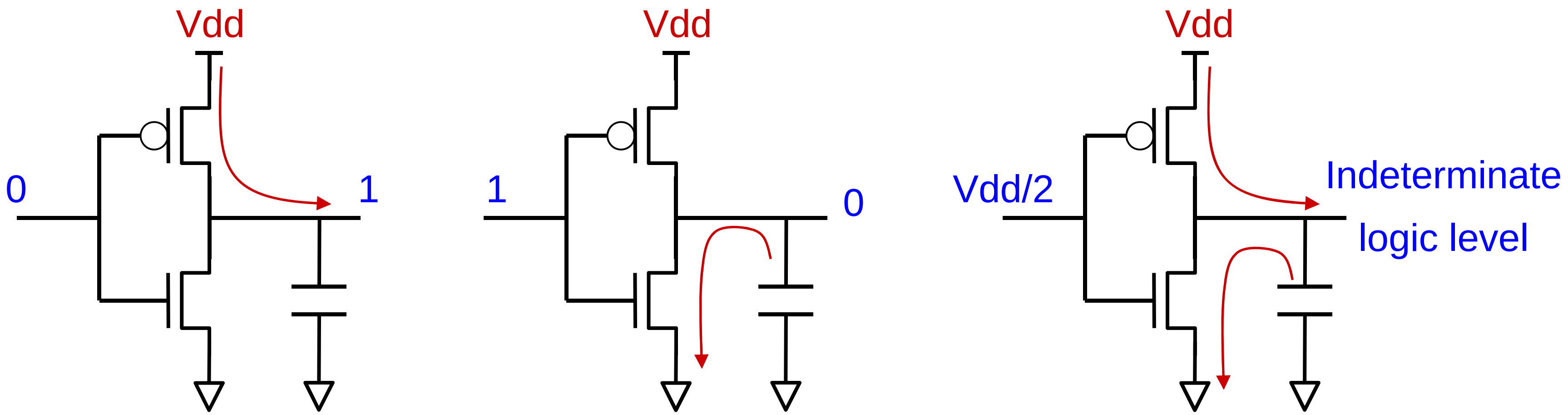
Characteristics



CMOS Logic-Inverter

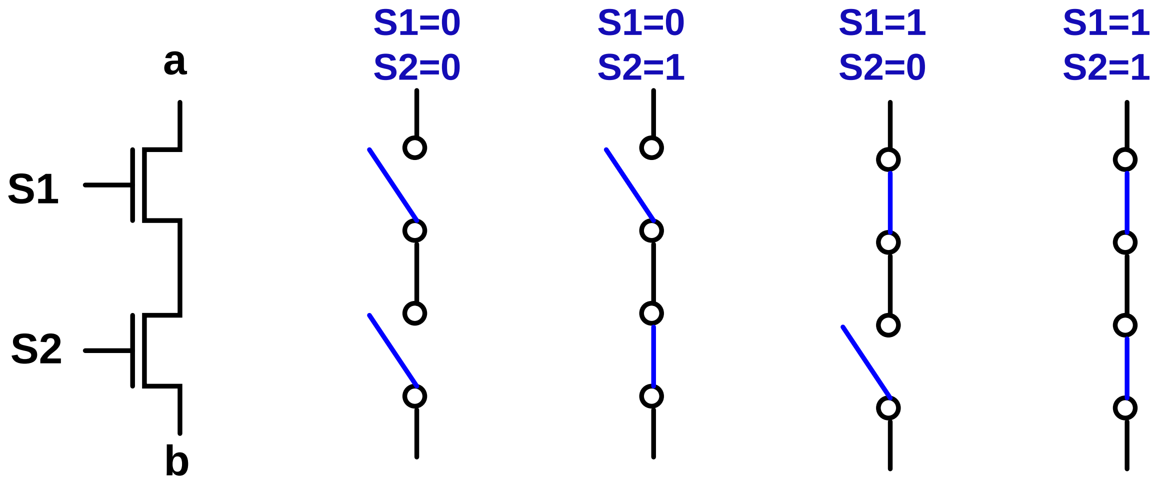
- The NOT or INVERT function is often considered the simplest Boolean operation
  - F(x)=NOT(x)=x'



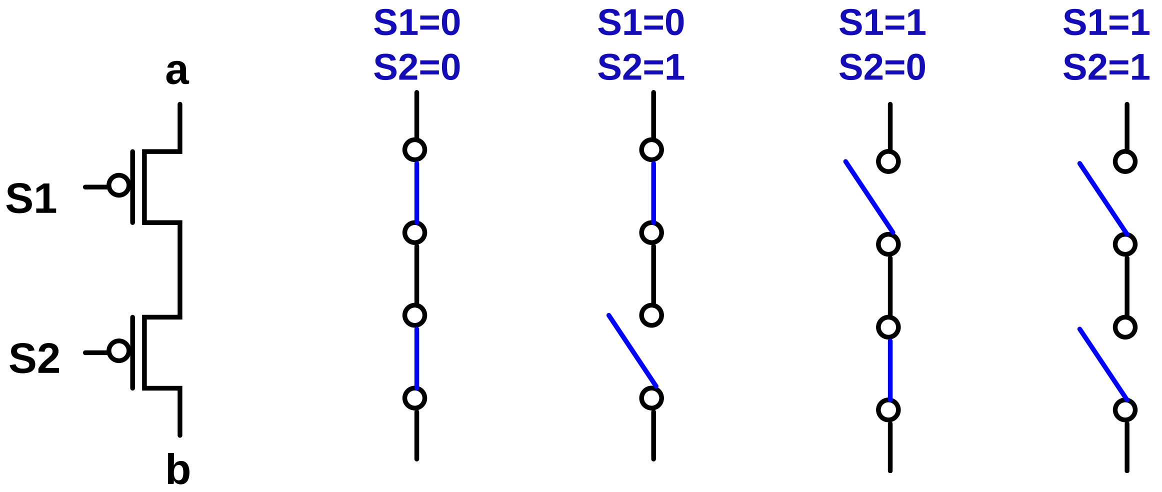


Combinational Logic

□ Serial structure



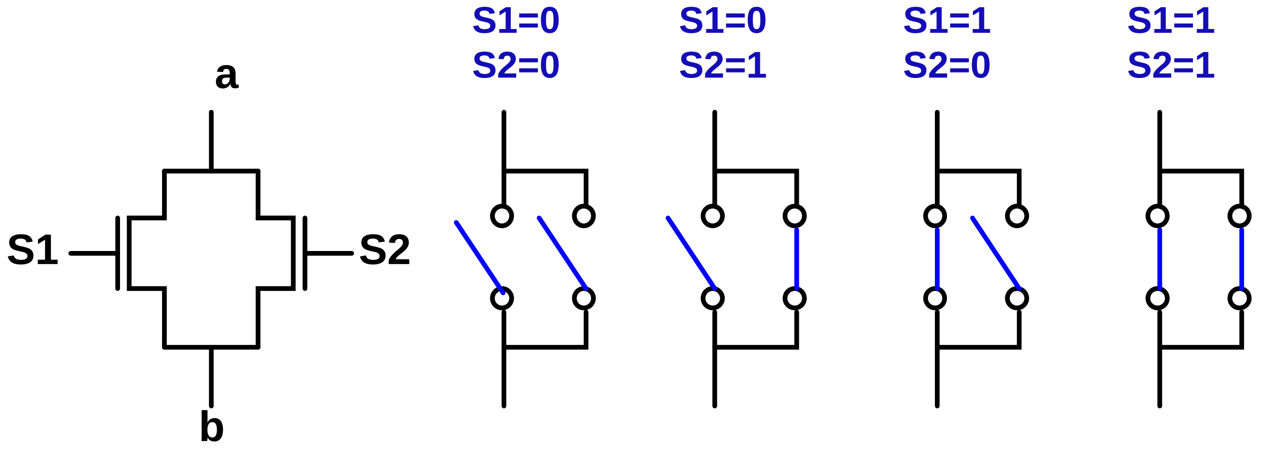
		S1	
		0	1
S2	0	a!=b	a!=b
	1	a!=b	a=b



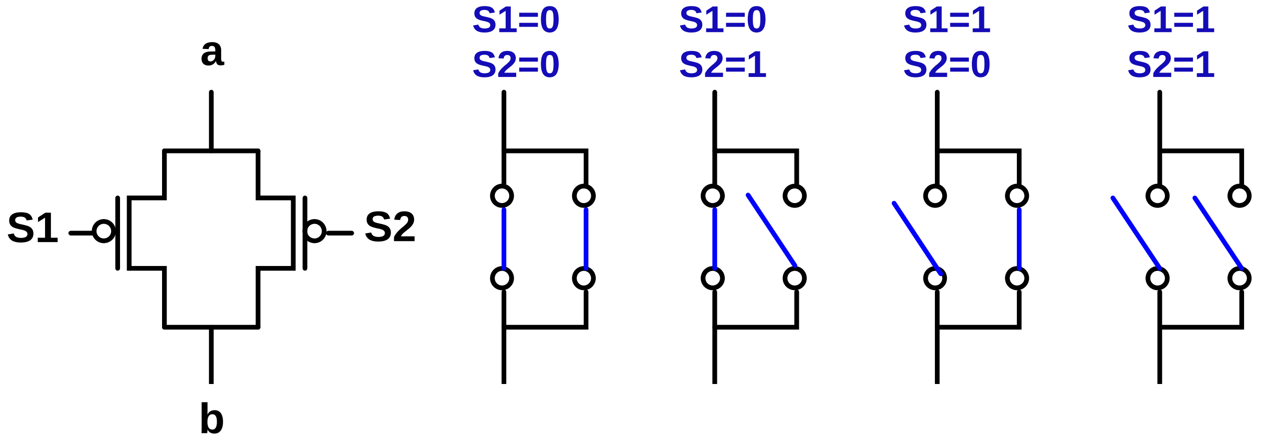
		S1	
		0	1
S2	0	a=b	a!=b
	1	a!=b	a!=b

Combinational Logic

□ Parallel structure

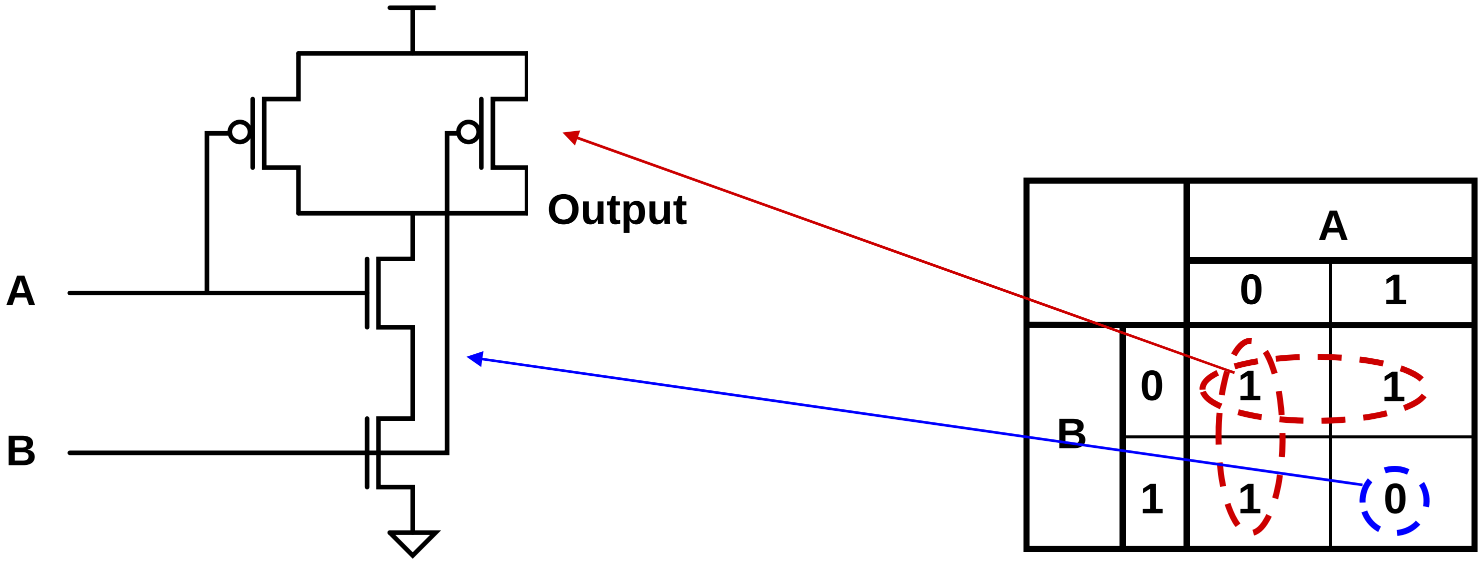


		S1	
		0	1
S2	0	a!=b	a=b
	1	a=b	a=b

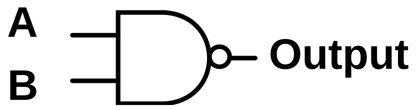


		S1	
		0	1
S2	0	a=b	a=b
	1	a=b	a!=b

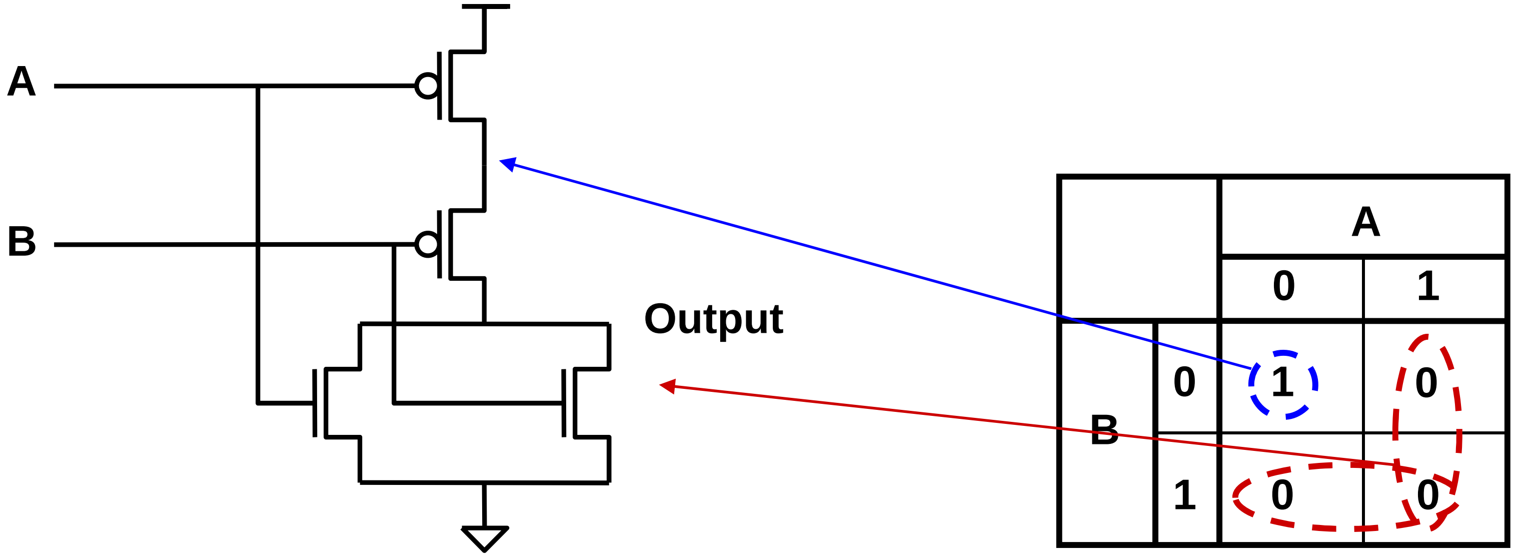
NAND Gate



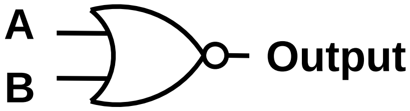
		A	
		0	1
B	0	1	1
	1	1	0



NOR Gate

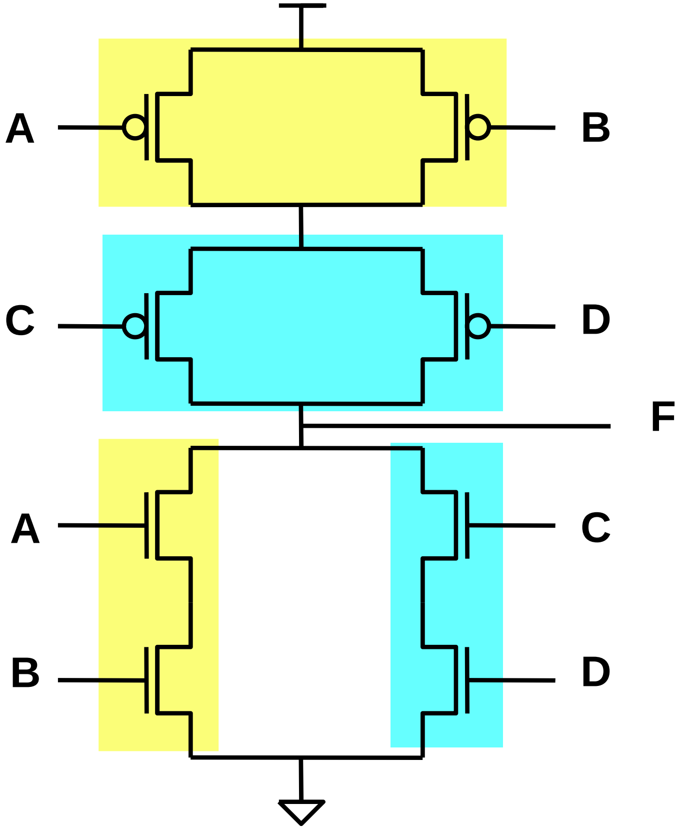


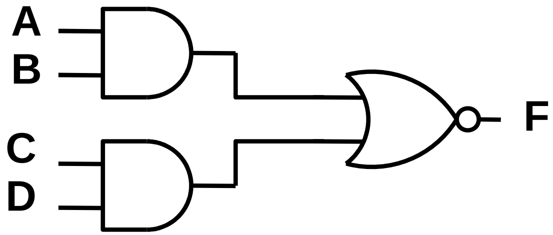
		A	
		0	1
B	0	1	0
	1	0	0



Compound Gate

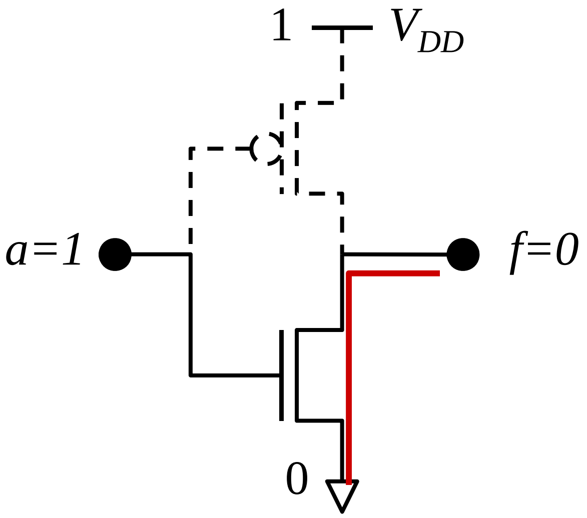
- □ $F = \overline{((AB)+(CD))}$





Structured Logic Design

- CMOS logic gates are intrinsically inverting
  - The output always produces a NOT operation acting on the input variables
- For example, the inverter shown below illustrates this property



Structured Logic Design

- The inverting nature of CMOS logic circuits allows us to construct logic circuits for AOI and OAI expressions using a structured approach
- AOI logic function
  - Implements the operations in the order AND then OR then NOT
  - E.g., $g(a,b,c,d) = \overline{a.b + c.d}$
- OAI logic function
  - Implements the operations in the order OR then AND then NOT
  - E.g., $g(a,b,c,d) = \overline{(a+b)\cdot(c+d)}$

Structured Logic Design

- Behaviors of nMOS and pMOS groups
  - Parallel-connected nMOS
    - OR-NOT operations
  - Parallel-connected pMOS
    - AND-NOT operations
  - Series-connected nMOS
    - AND-NOT operations
  - Series-connected pMOS
    - OR-NOT operations
- Consequently, wired groups of nMOS and pMOS are logical **duals** of another

Dual Property

- □ If an NMOS group yields a function of the form

$$g = \overline{a \cdot (b + c)}$$

then an identically wired PMOS array gives the dual function

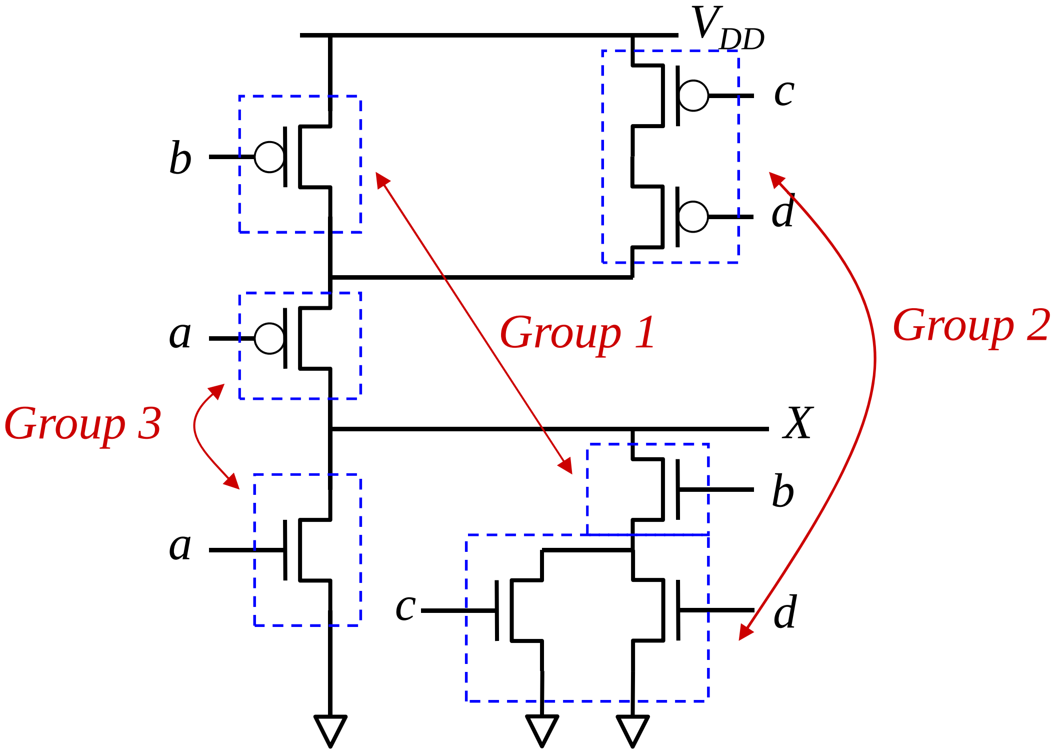
$$G = \overline{a + (b \cdot c)}$$

where the AND and OR operations have been interchanged

- □ This is an interesting property of NMOS-PMOS logic that can be exploited in some CMOS designs

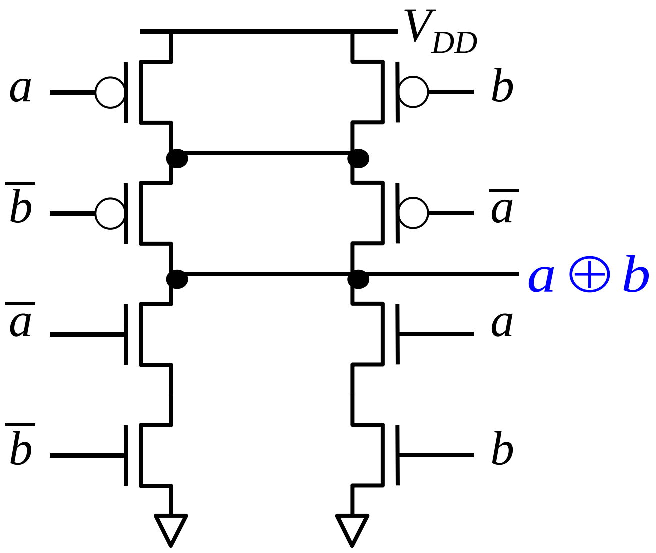
An Example of Structured Design

- □ $X = \overline{a + b \cdot (c + d)}$

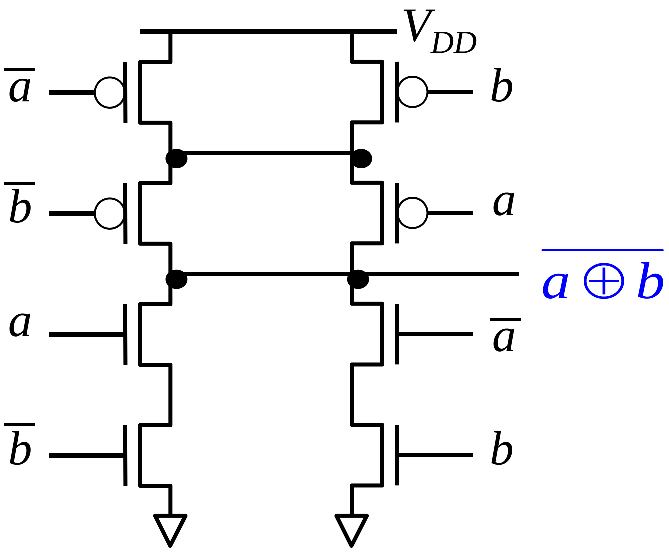


An Example of XOR Gate

- Boolean equation of the two input XOR gate
  - $a \oplus b = \overline{a} \cdot b + a \cdot \overline{b}$, this is not in AOI form
  - But, $\overline{a \oplus b} = a \cdot b + \overline{a} \cdot \overline{b}$, this is in AOI form
  - Therefore, $a \oplus b = \overline{\overline{(a \oplus b)}} = \overline{a \cdot b + \overline{a} \cdot \overline{b}}$

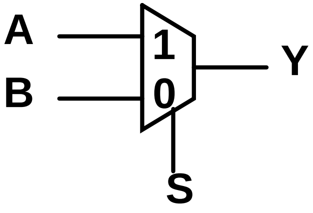


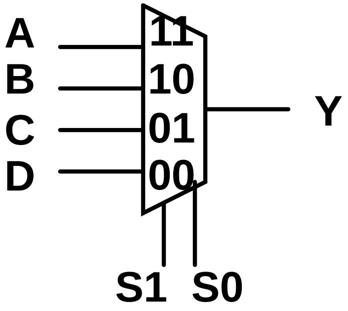
XOR Gate

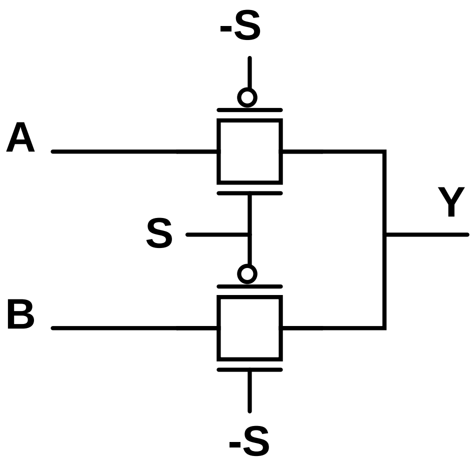


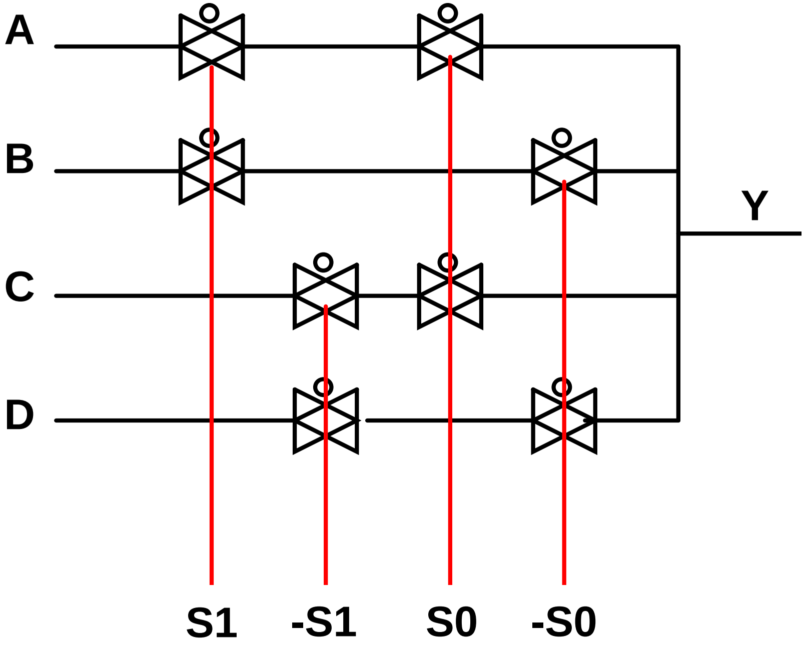
XNOR Gate

Multiplexer





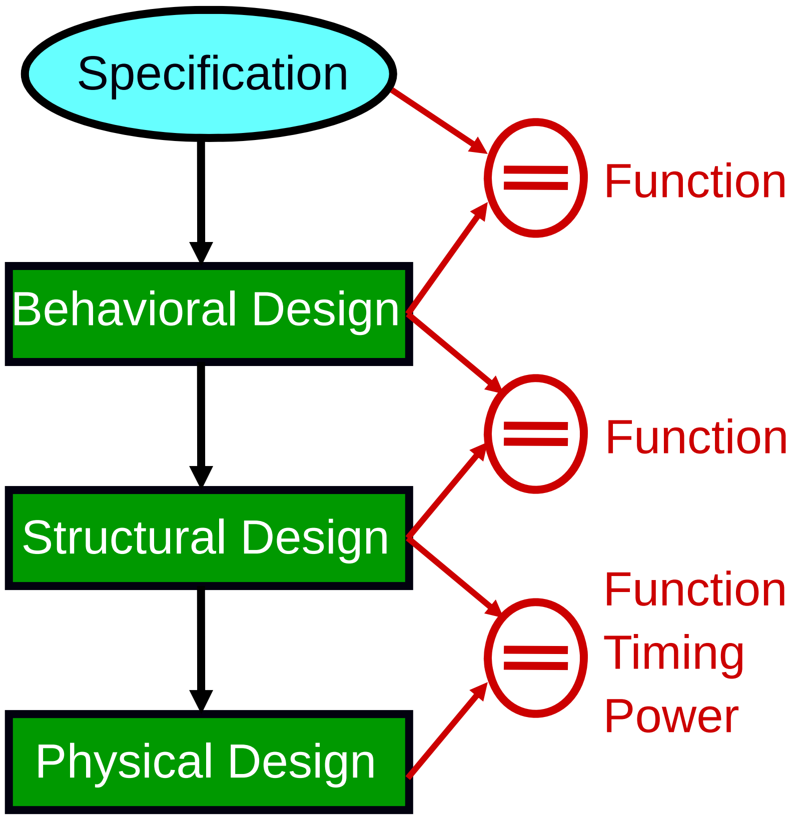




Static CMOS Summary

- In static circuits at every point in time (except when switching), the output is connected to either Vdd or Gnd through a low resistance path
  - Fan-in of *n* (or *n* inputs) requires 2*n* (*n* N-type and *n* P-type) devices
- Non-ratioed logic: gates operate independent of PMOS or NMOS sizes
- No path ever exists between Vdd and Gnd: low static power
- Fully-restored logic (NMOS passes “0” only and PMOS passes “1” only
- Gates must be inverting

Design Flow for a VLSI Chip



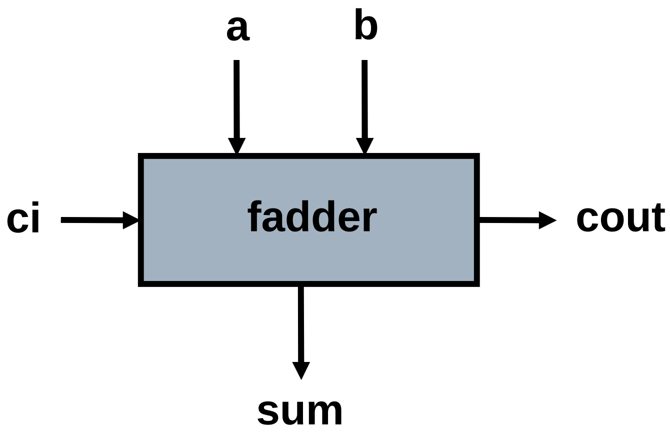
Circuit and System Representations

- □ Behavioral representation
  - ■ Functional, high level
  - ■ For documentation, simulation, verification
- □ Structural representation
  - ■ System level – CPU, RAM, I/O
  - ■ Functional level – ALU, Multiplier, Adder
  - ■ Gate level – AND, OR, XOR
  - ■ Circuit level – Transistors, R, L, C
  - ■ For design & simulation
- □ Physical representation
  - ■ For fabrication

Behavior Representation

- A one-bit full adder (Verilog)

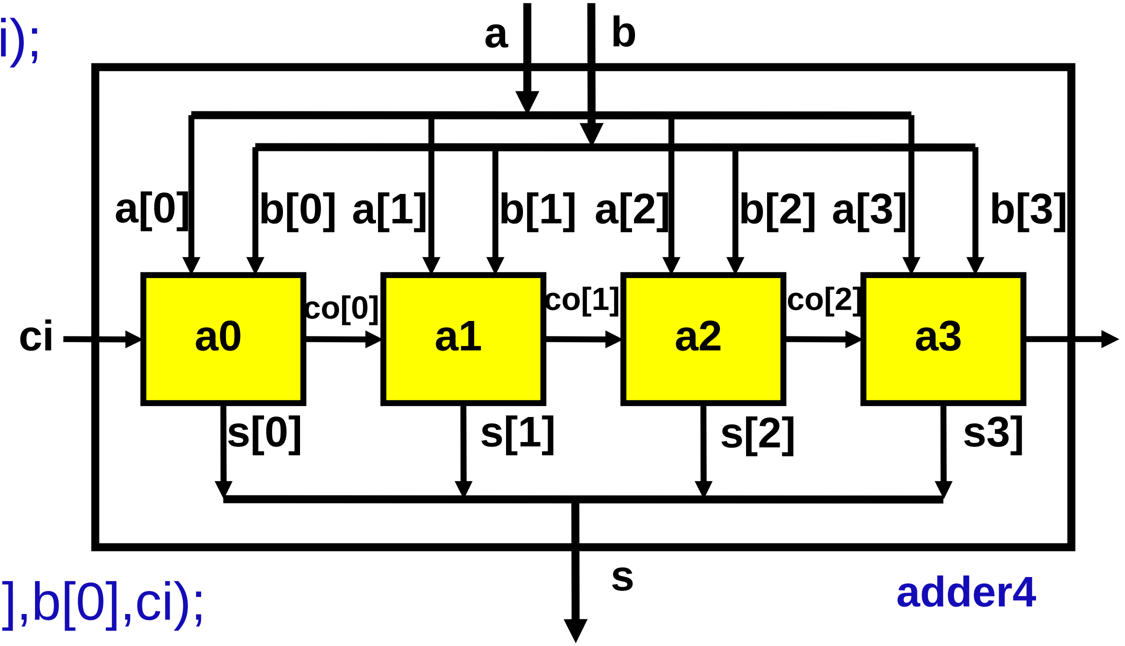
```
module fadder(sum,cout,a,b,ci);
output sum, cout;
input a, b, ci;
reg sum, cout;

always @(a or b or ci) begin
    sum = a^b^ci;
    cout = (a&b)|(b&ci)|(ci&a);
end
endmodule
```



Structure Representation

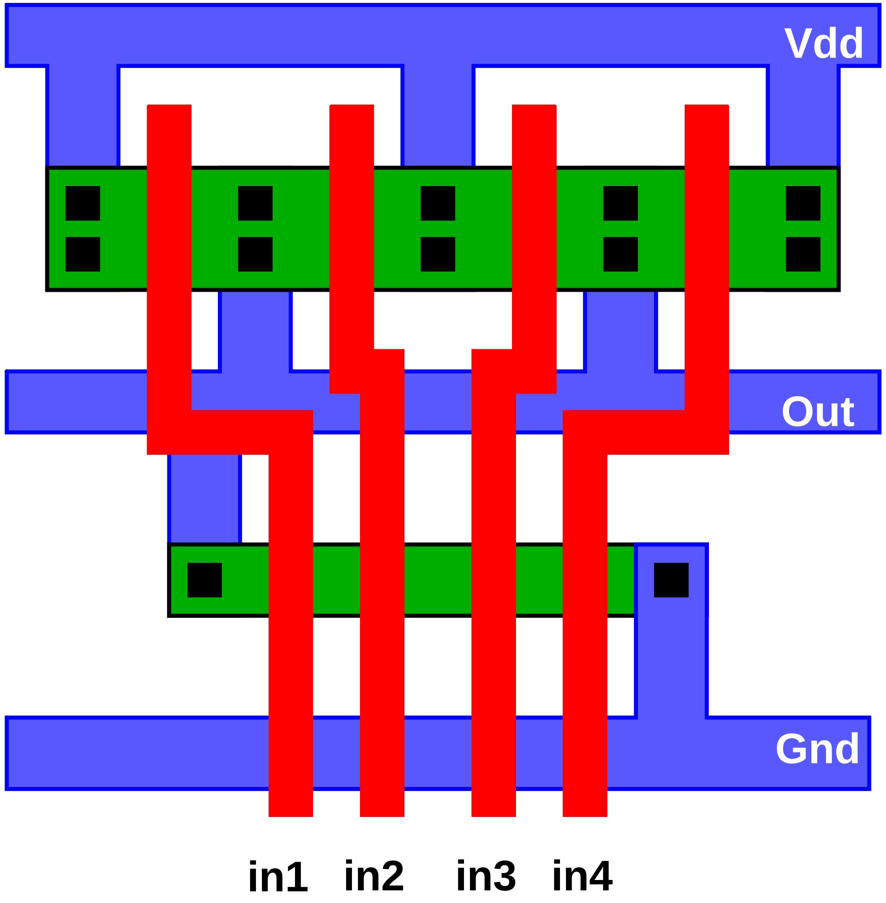
□ A four-bit full adder (Verilog)

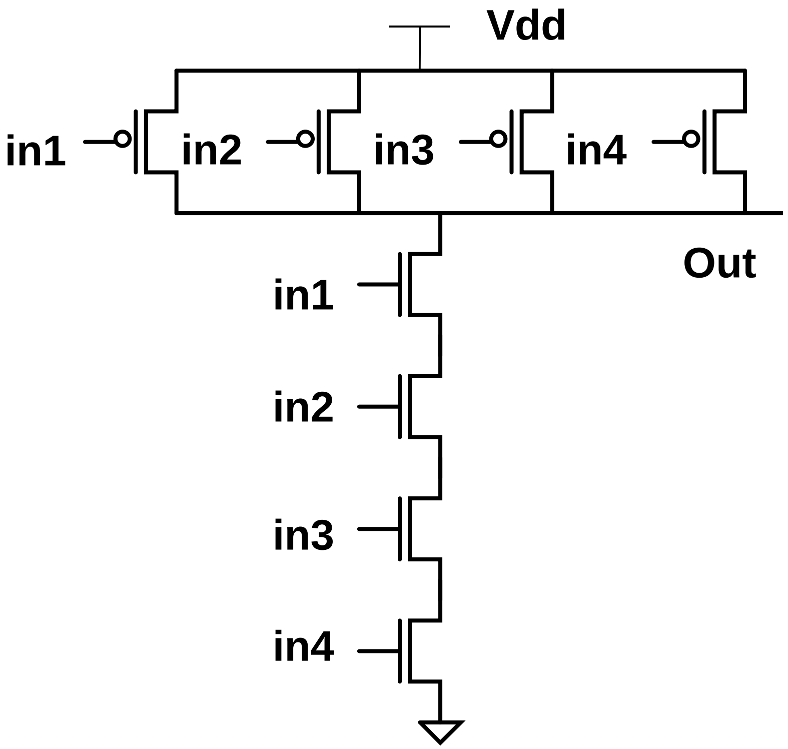
```
module adder4(s,c4,a,b,ci);
output[3:0] sum;
output c4;
input[3:0] a, b;
input  ci;
reg[3:0] s;
reg  c4;
wire[2:0] co;
    fadder a0(s[0],co[0],a[0],b[0],ci);
    fadder a1(s[1],co[1],a[1],b[1],co[0]);
    fadder a2(s[2],co[2],a[2],b[2],co[1]);
    fadder a3(s[3],c4,a[3],b[3],co[2]);
endmodule
```



Physical Representation

- □ Layout of a 4-bit NAND gate


Vdd
Out
Gnd
in1
in2
in3
in4


Vdd
in1
in2
in3
in4
Out
in1
in2
in3
in4