MOS TRANSISTORS AND CMOS LOGIC

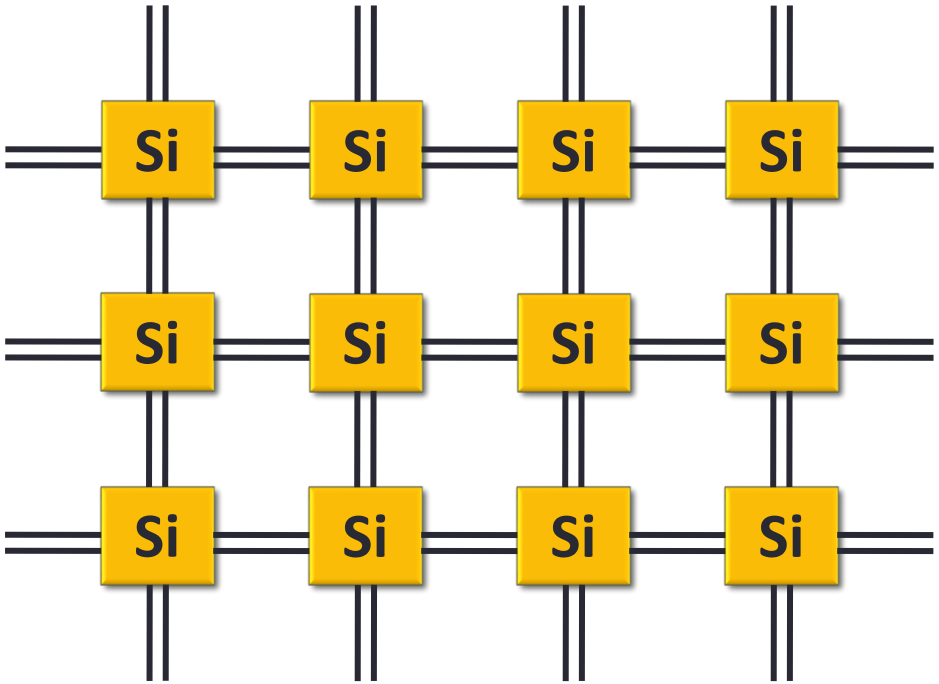
Dr Noor Mahammad Sk

High Performance Reconfigurable Computing Systems Engineering Group

Hprcse

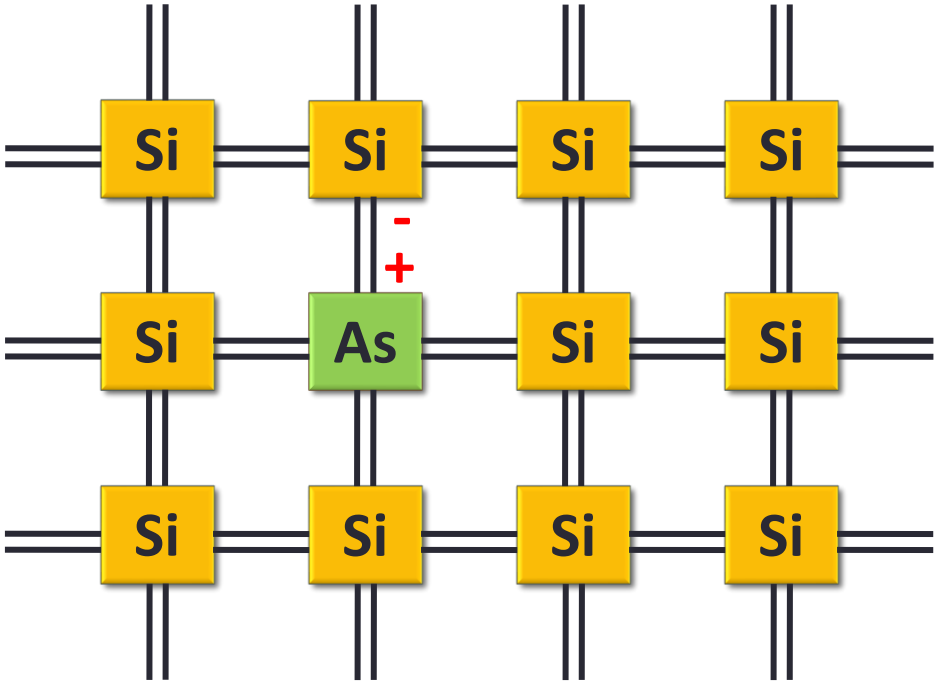
Silicon Lattice

- Transistors are built on a silicon substrate
- Silicon is a Group IV material
- Forms crystal lattice with bonds to four neighbors



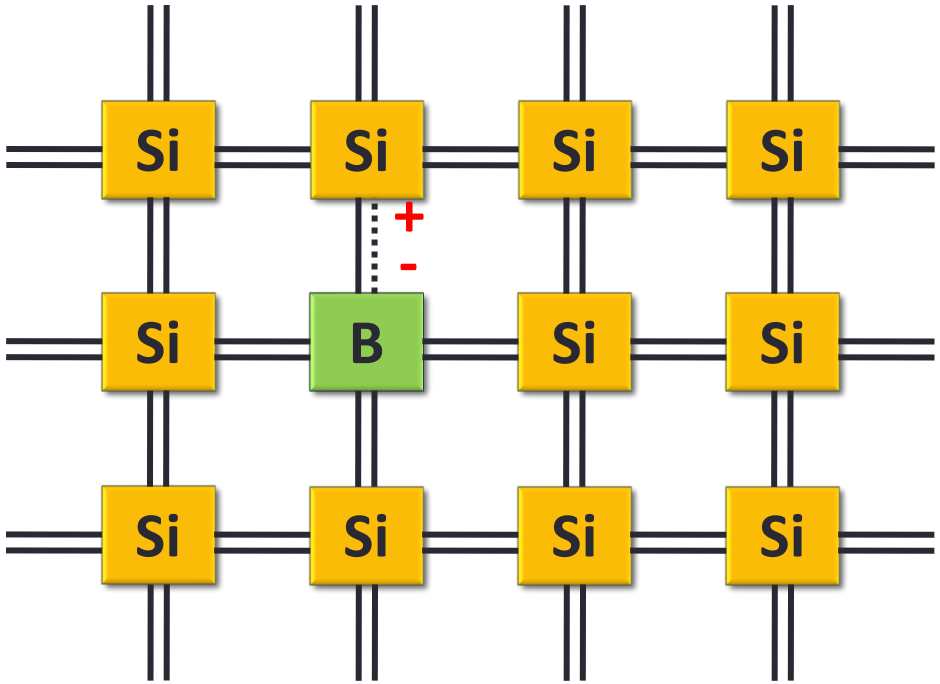
Dopants

- Silicon is a semiconductor
- Pure silicon has no free carriers and conducts poorly
- Adding dopants increases the conductivity
- Group V: extra electron (n-type)



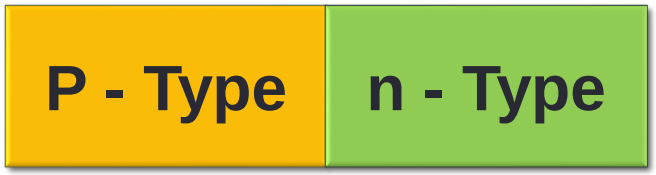
Dopants

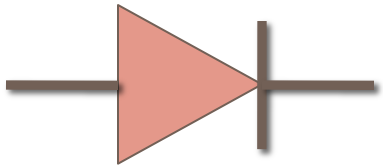
- Group III: missing electron, called hole (p-type)



p-n Junctions

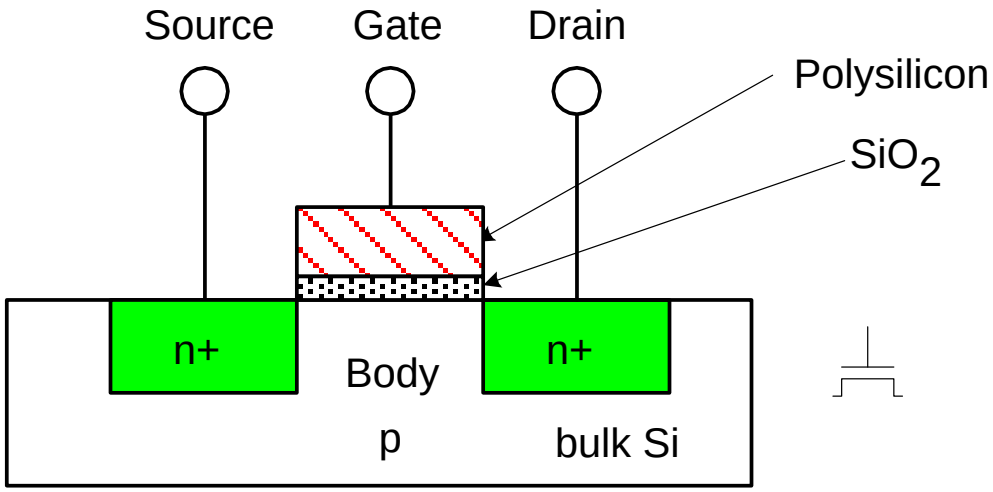
- A junction between p-type and n-type semiconductor forms a diode.
- Current flows only in one direction





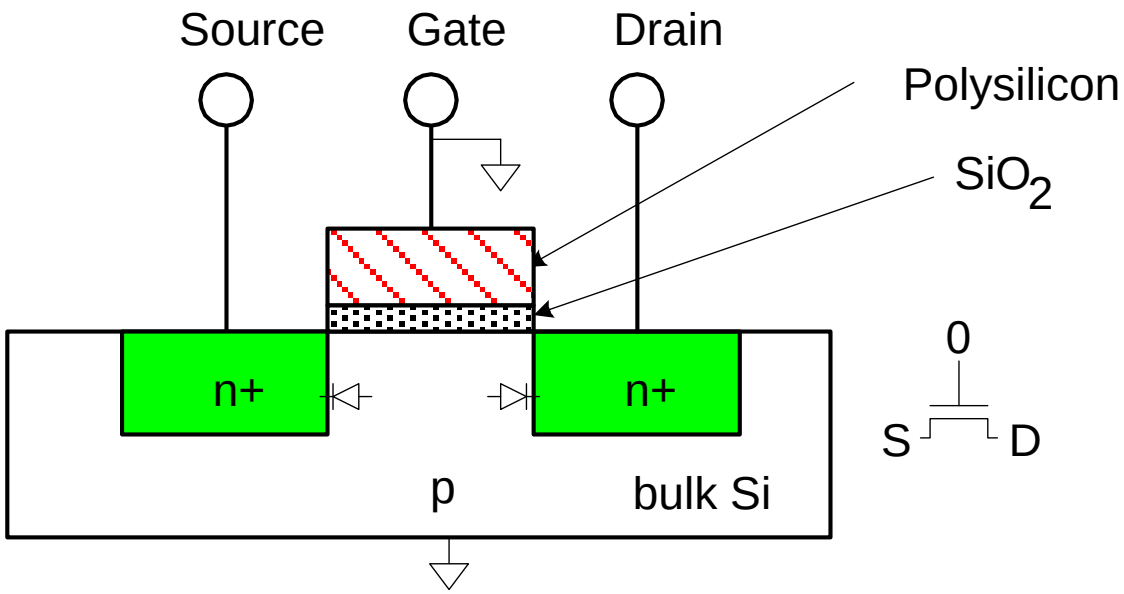
nMOS Transistor

- 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



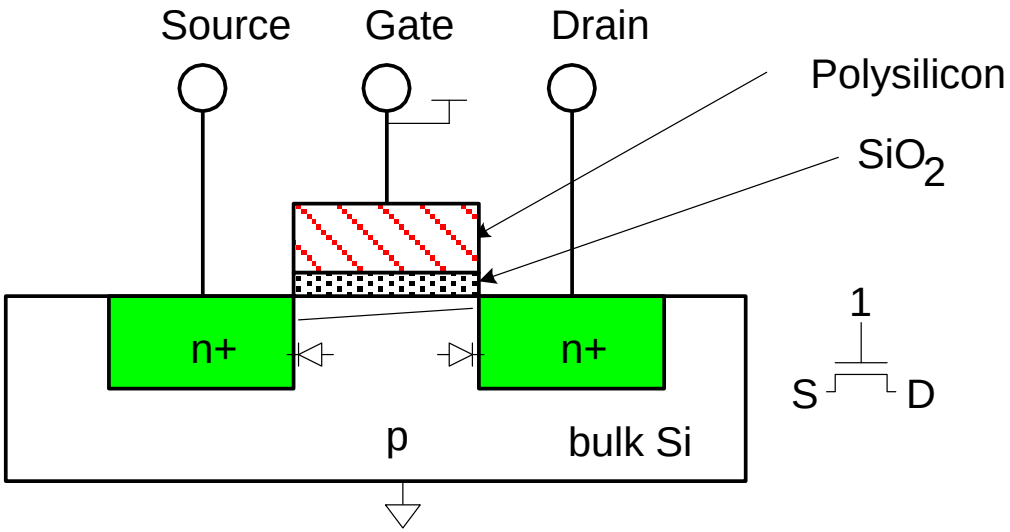
nMOS Operation

- Body is usually 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



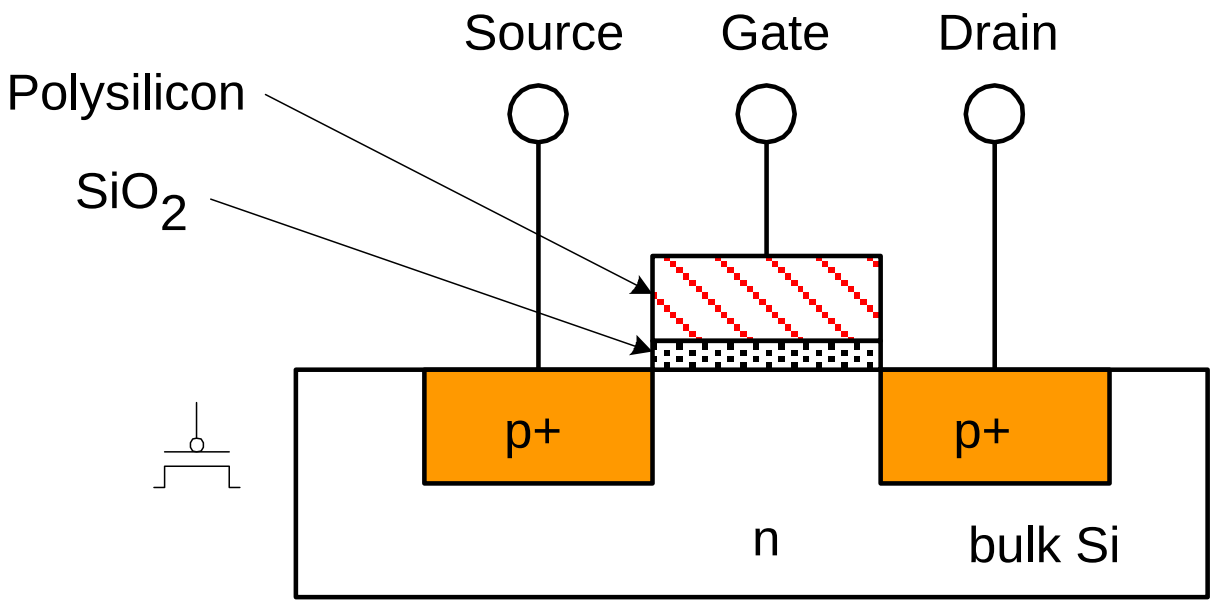
nMOS Operation Cont.

- 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



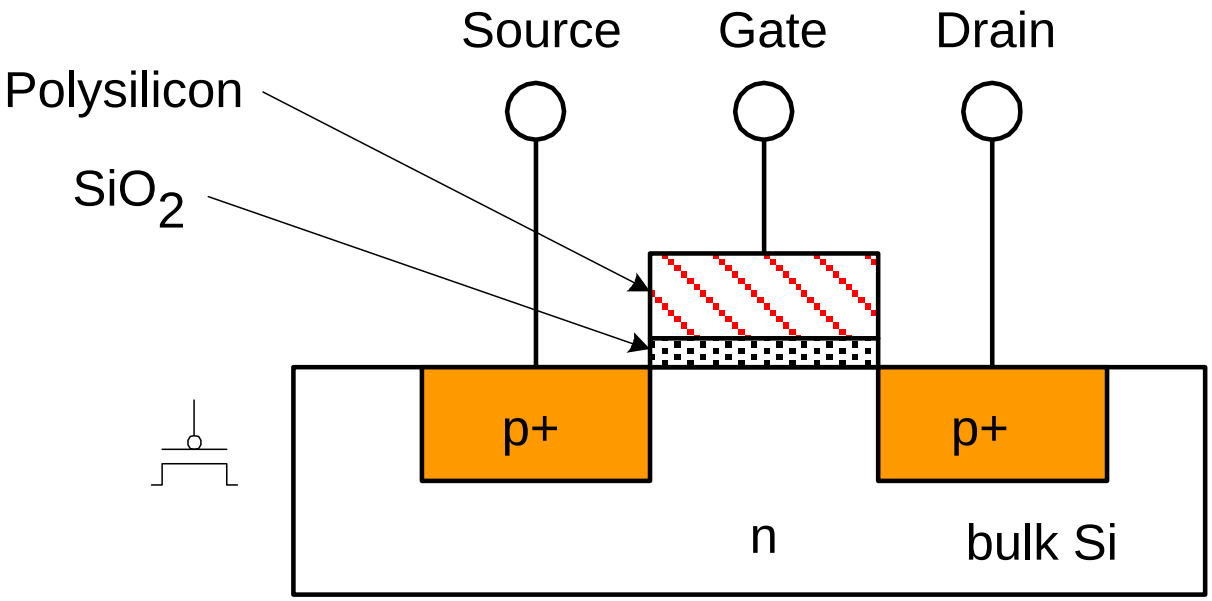
pMOS Transistor

- Similar, but doping and voltages reversed
- Body tied to high voltage (V_{DD})
- Gate low: transistor ON – the positive charges are attracted to the under side of the Si-SiO_2 interface,
- The sufficient low gate voltage inverts the channel and a conducting path of positive carriers is formed from source to drain



pMOS Transistor

- Similar, but doping and voltages reversed
- Body tied to high voltage (V_{DD})
- Gate high: transistor OFF – the source and drain junctions are reverse-biased and no current flow
- Symbol – Bubble on the gate indicates inverted behavior to nMOS

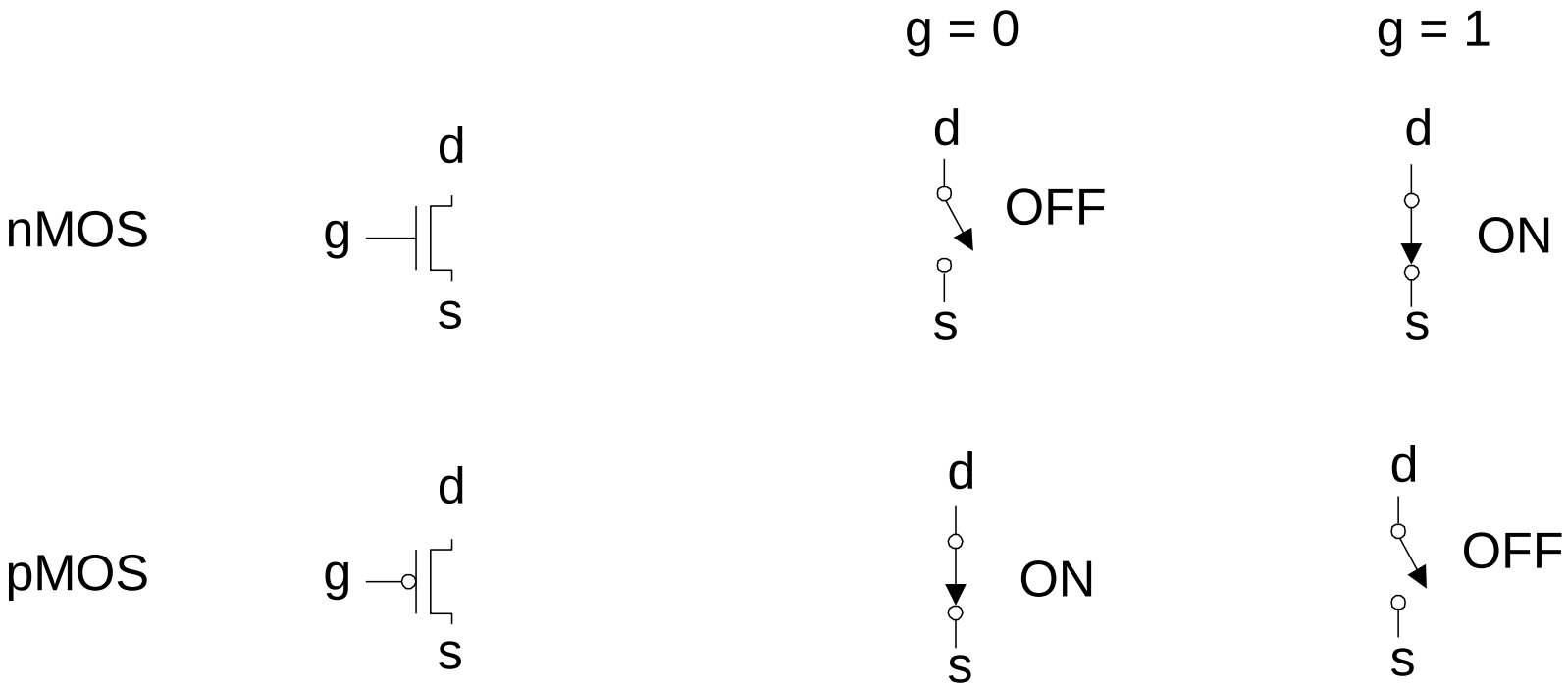


Power Supply Voltage

- GND = 0 V
- In 1980's, V_{DD} = 5V
- V_{DD} has decreased in modern processes
  - High V_{DD} would damage modern tiny transistors
  - Lower V_{DD} saves power
- V_{DD} = 3.3, 2.5, 1.8, 1.5, 1.2, 1.0, …

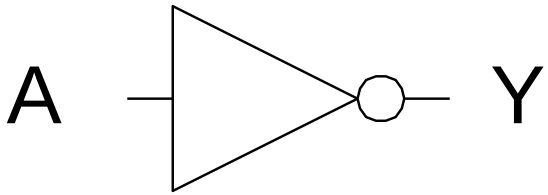
Transistors as Switches

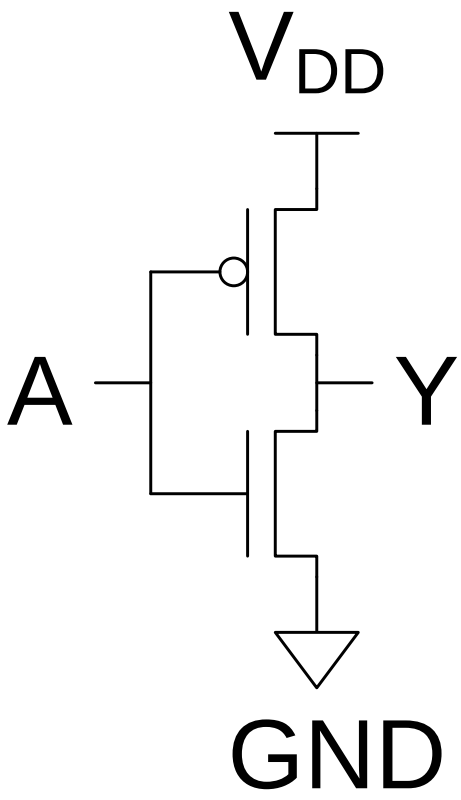
- We can view MOS transistors as electrically controlled switches
- Voltage at gate controls path from source to drain



CMOS Inverter

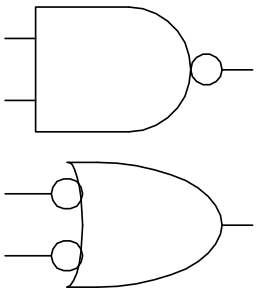
A	Y



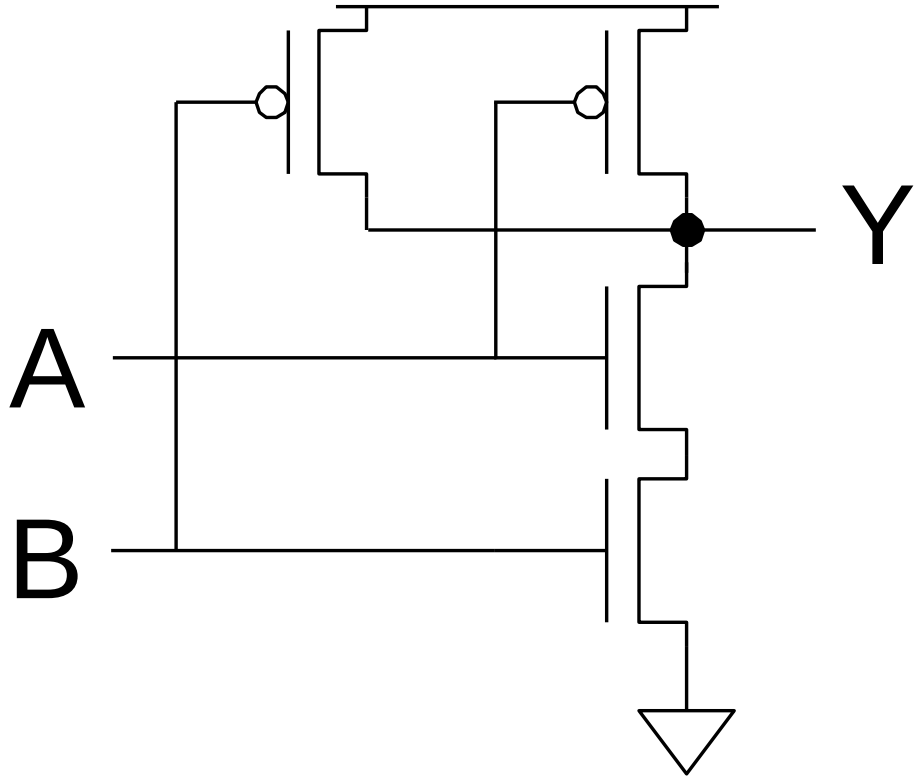


CMOS NAND Gate

A	B	Y
0	0	
0	1	
1	0	
1	1	

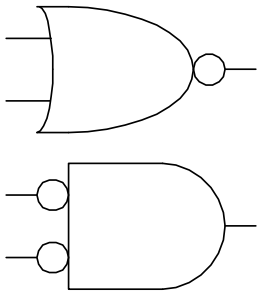


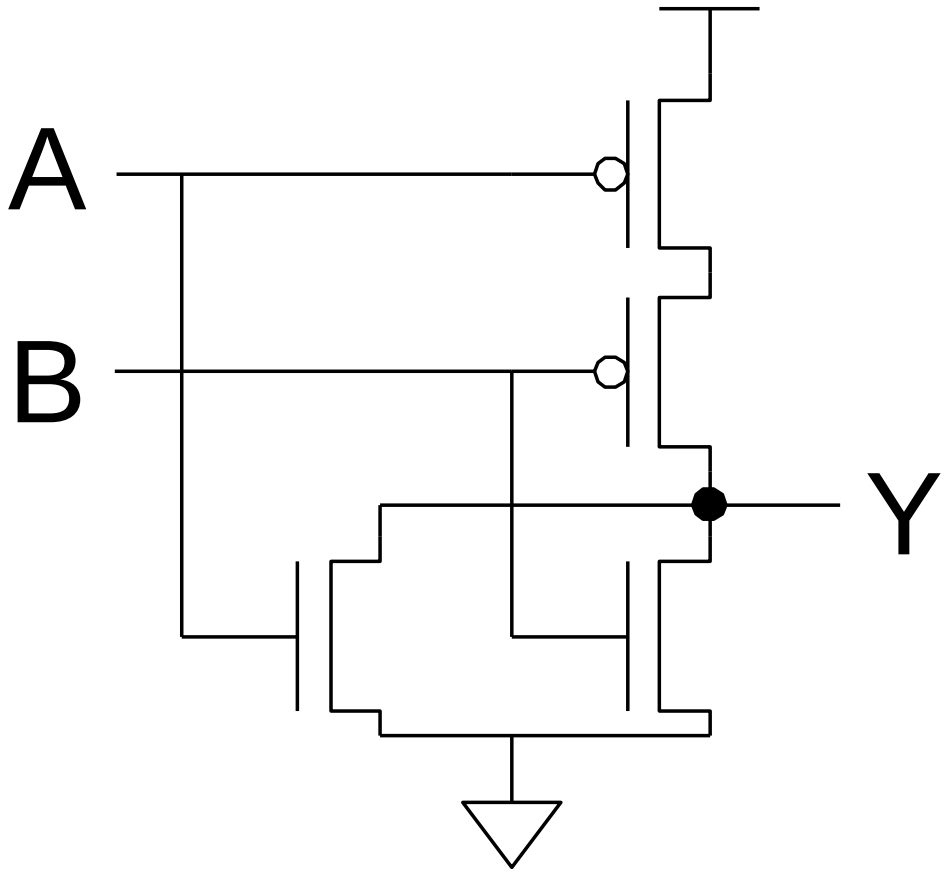
k-input NAND gates are constructed using k-series nMOS transistors and k parallel pMOS transistors



CMOS NOR Gate

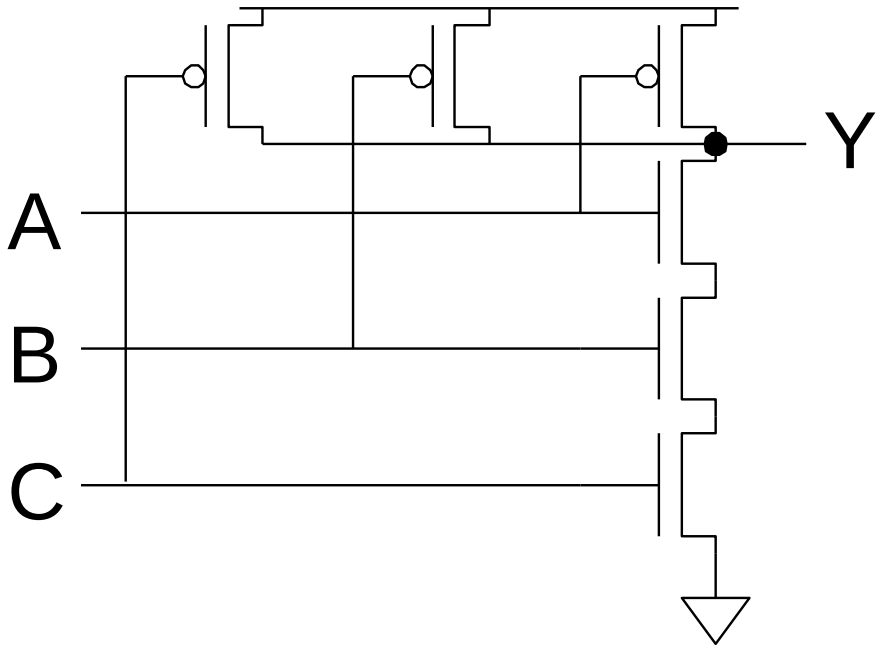
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0





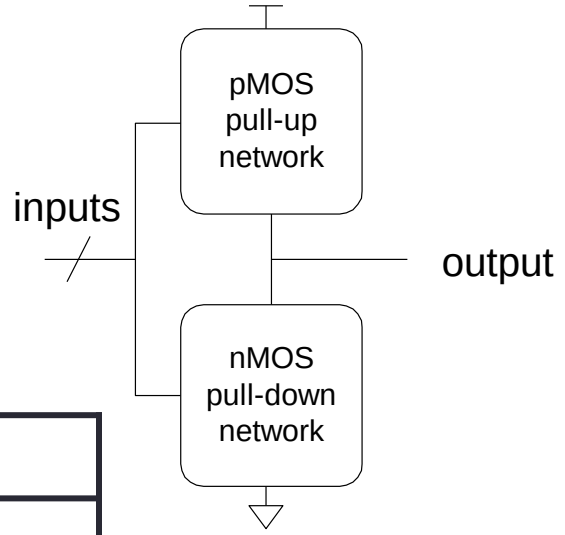
3-input NAND Gate

- Y pulls low if ALL inputs are 1
- Y pulls high if ANY input is 0



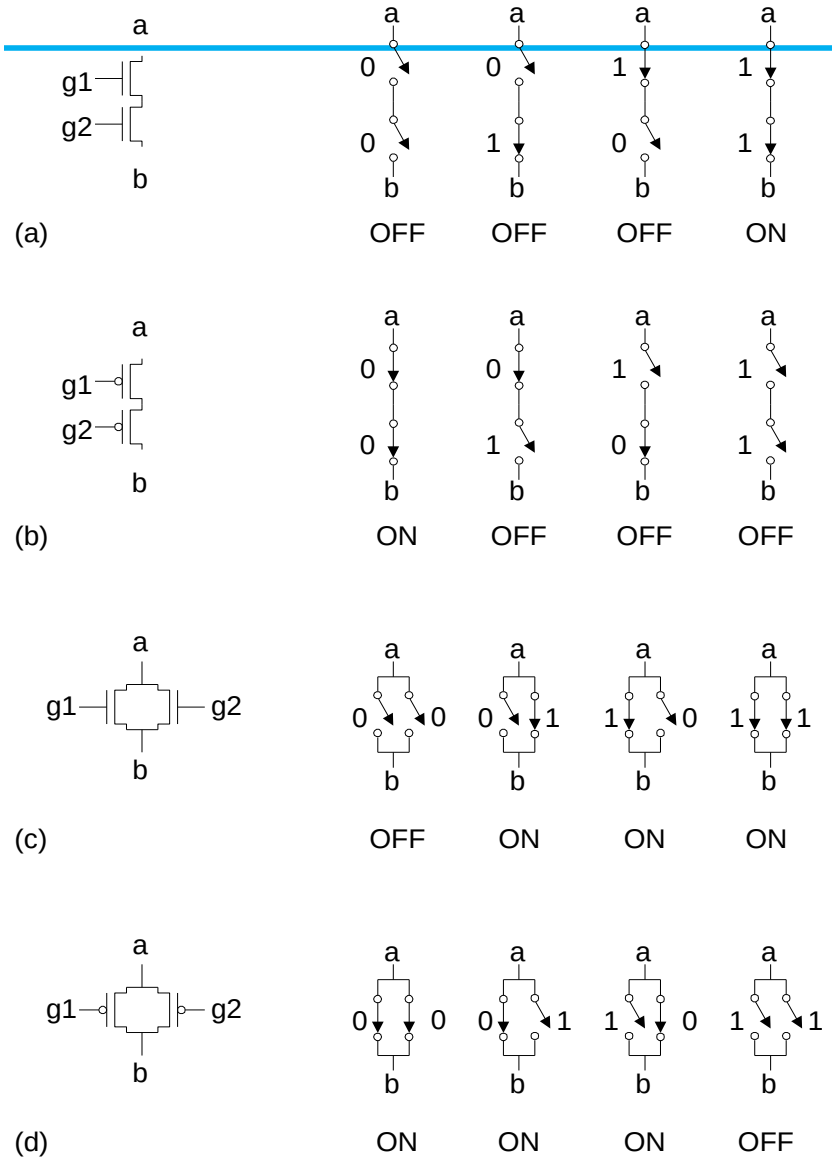
Complementary MOS (CMOS)

- Complementary CMOS logic gates
- nMOS *pull-down network*
- pMOS *pull-up network*
- a.k.a. static CMOS



	Pull-up OFF	Pull-up ON
Pull-down OFF	Z (float)	1
Pull-down ON	0	X (crowbar)

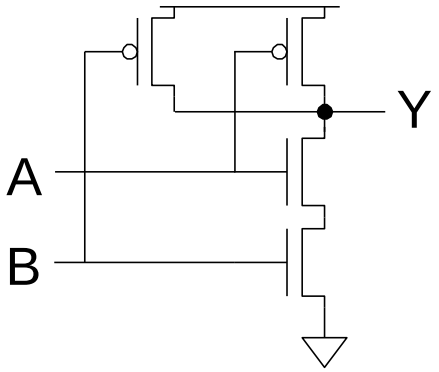
Series and Parallel



- nMOS: 1 = ON
- pMOS: 0 = ON
- *Series*: both must be ON
- *Parallel*: either can be ON

Conduction Complement

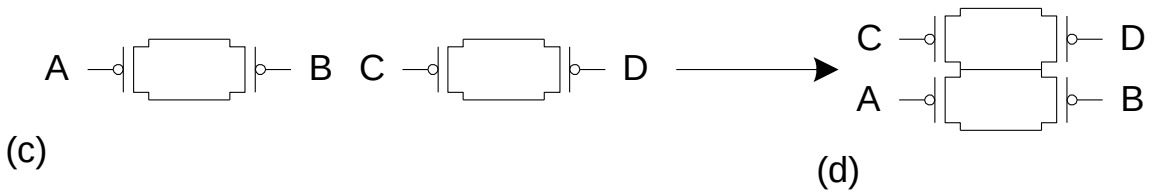
- Complementary CMOS gates always produce 0 or 1
- Ex: NAND gate
  - Series nMOS: Y=0 when both inputs are 1
  - Thus Y=1 when either input is 0
  - Requires parallel pMOS



- Rule of *Conduction Complements*
  - Pull-up network is complement of pull-down
  - Parallel -> series, series -> parallel

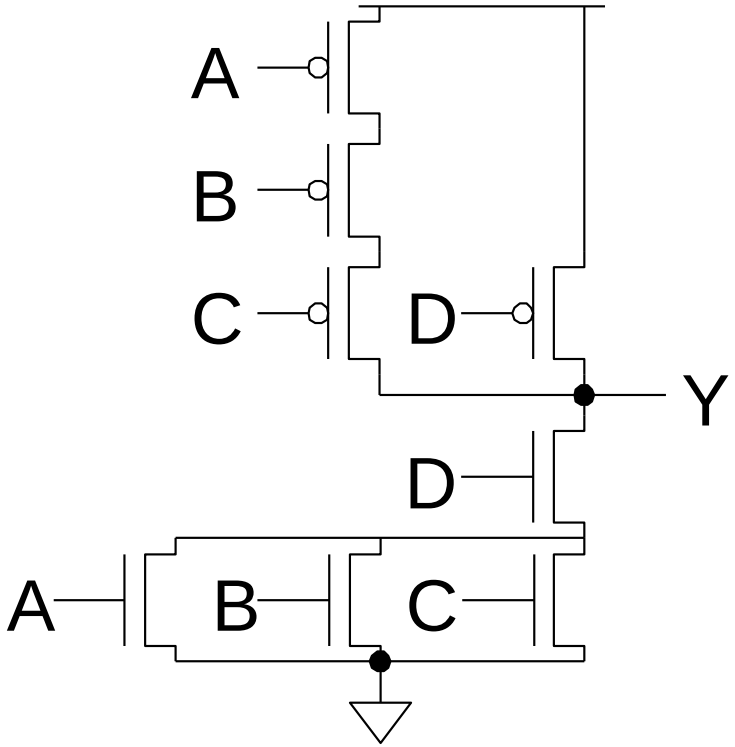
Compound Gates

- *Compound gates* can do any inverting function
- Example: $Y = \overline{A.B + C.D}$ (AND- AND-OR-INVERT, AOI22)



Example: O3AI

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

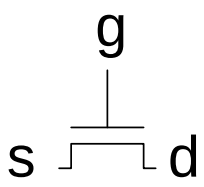


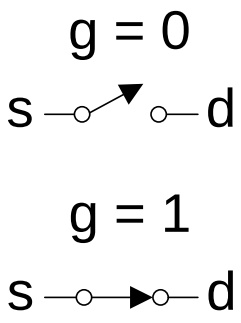
Signal Strength

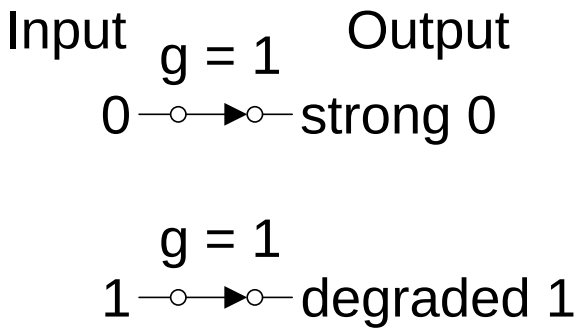
- *Strength* of signal
  - How close it approximates ideal voltage source
- V_{DD} and GND rails are strongest 1 and 0
- nMOS pass strong 0
  - But degraded or weak 1
- pMOS pass strong 1
  - But degraded or weak 0
- Thus nMOS are best for pull-down network

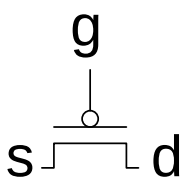
Pass Transistors

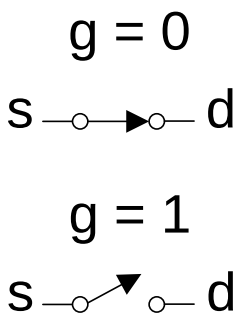
- Transistors can be used as switches

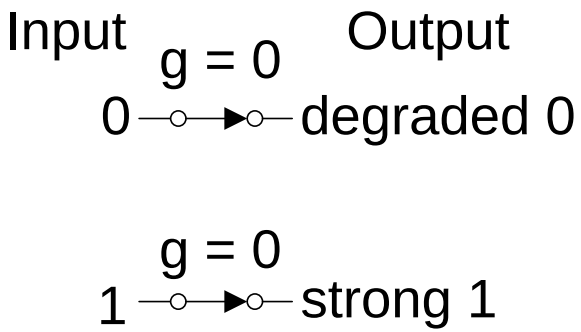






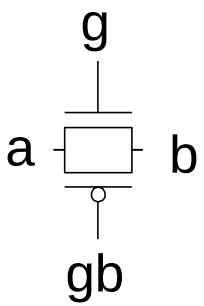


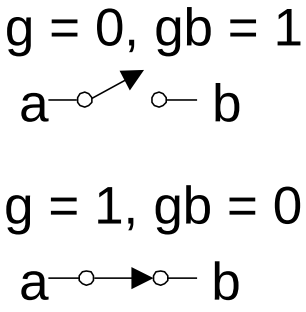


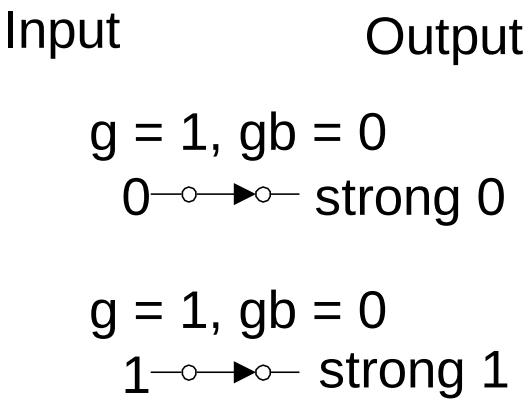


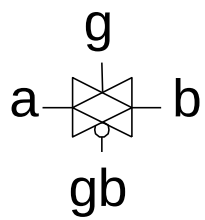
Transmission Gates

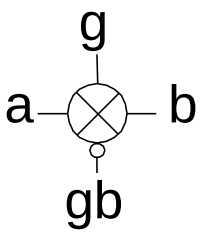
- Pass transistors produce degraded outputs
- *Transmission gates* pass both 0 and 1 well

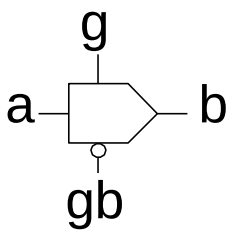






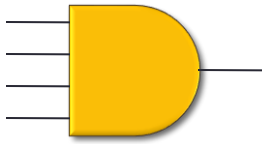


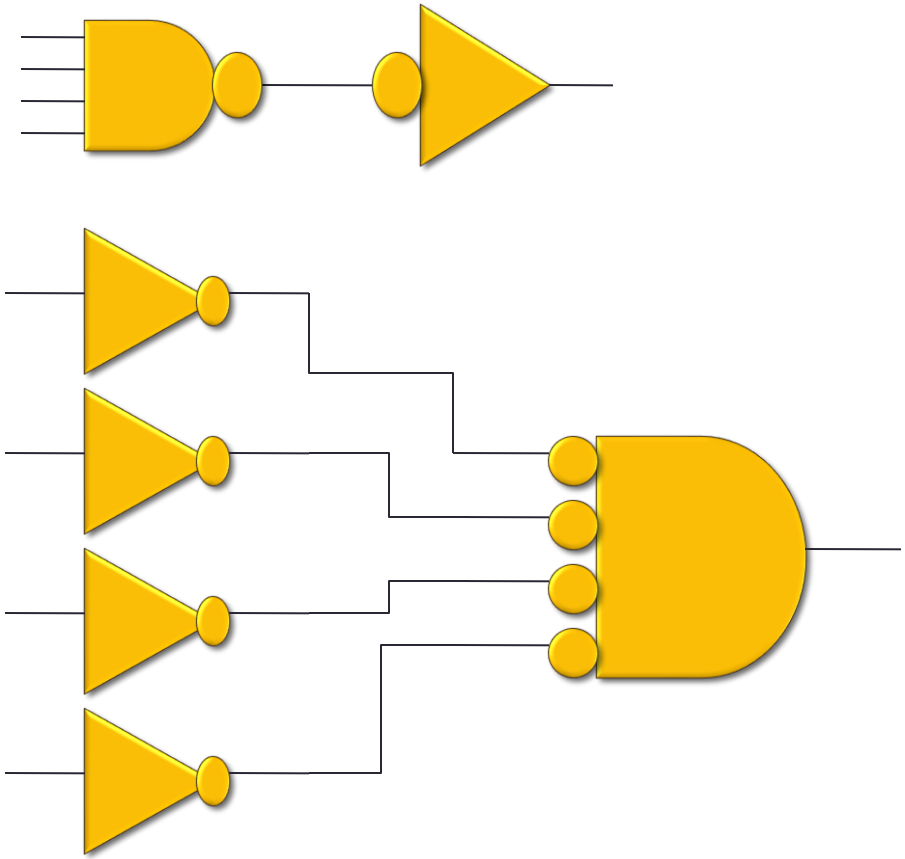




CMOS AND Gate Implementation

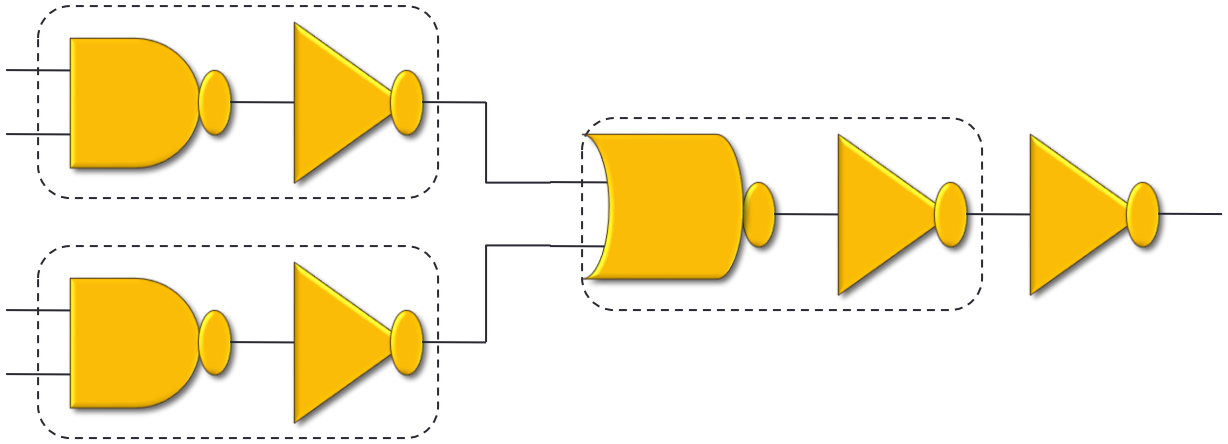
- A 4 – input AND gate built from two levels of inverting CMOS gates





CMOS AND Gate Implementation

- 4 –input AND gate could be built with two AND gates, an OR gate, and an inverter

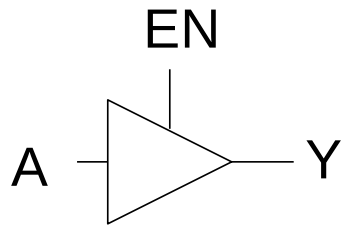


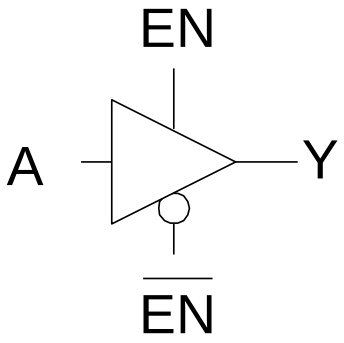
It consumes 20 transistors

Tristates

- *Tristate buffer* produces Z when not enabled

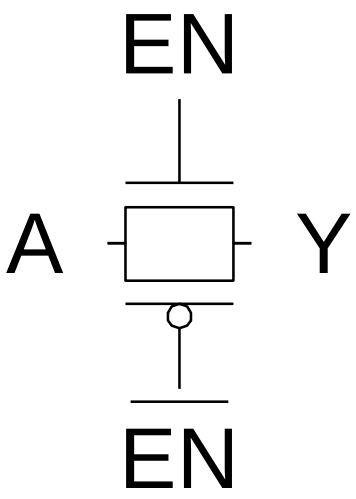
EN	A	Y
0	0	Z
0	1	Z
1	0	0
1	1	1





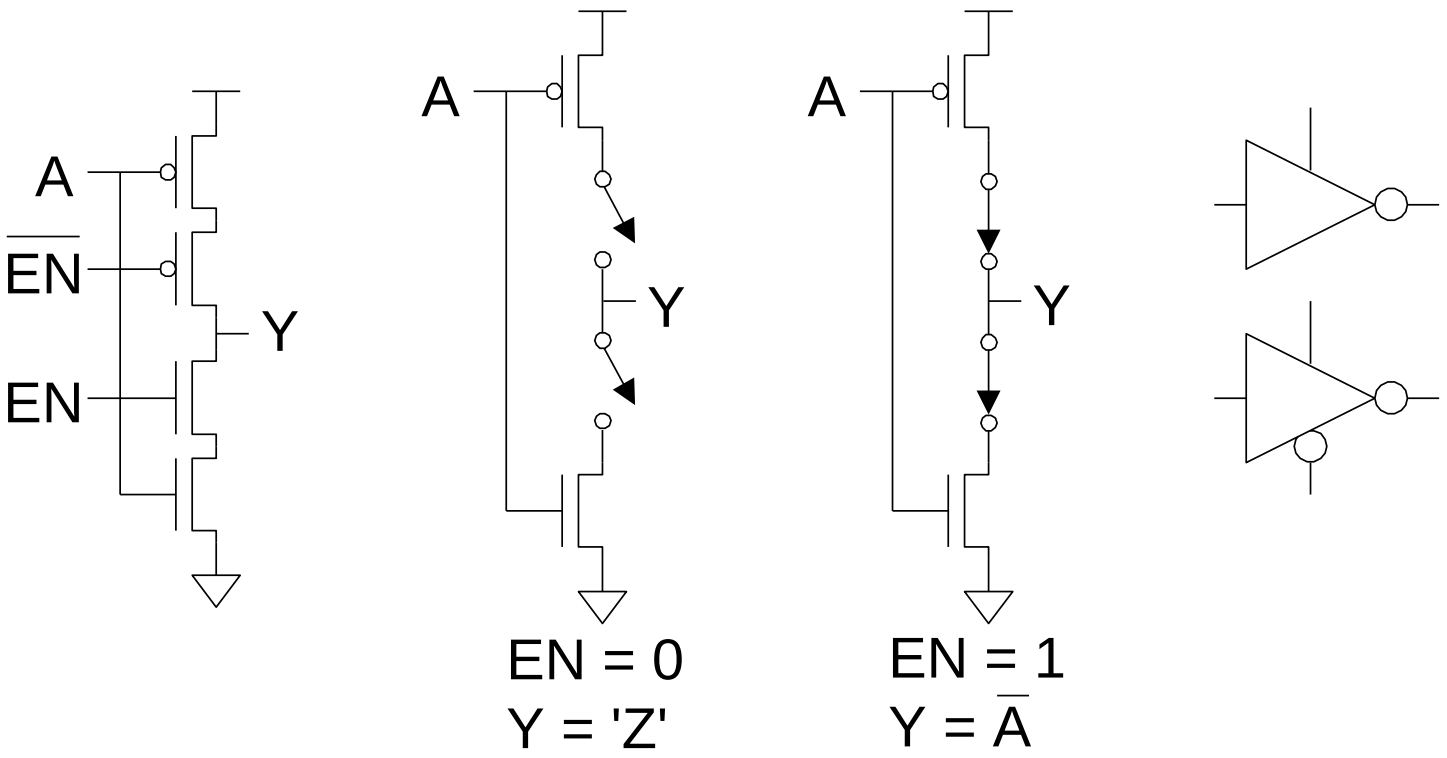
Nonrestoring Tristate

- Transmission gate acts as tristate buffer
  - Only two transistors
  - But *nonrestoring*
    - Noise on A is passed on to Y



Tristate Inverter

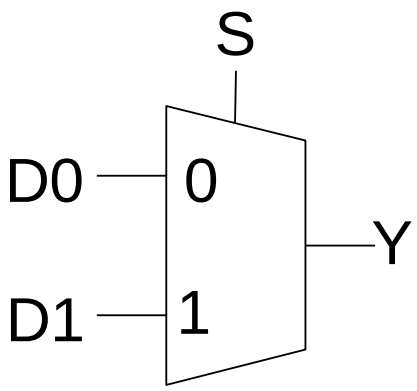
- Tristate inverter produces restored output
  - Violates conduction complement rule
  - Because we want a Z output



Multiplexers

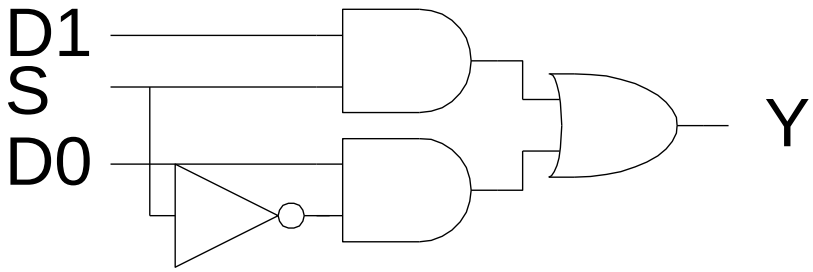
- 2:1 multiplexer chooses between two inputs

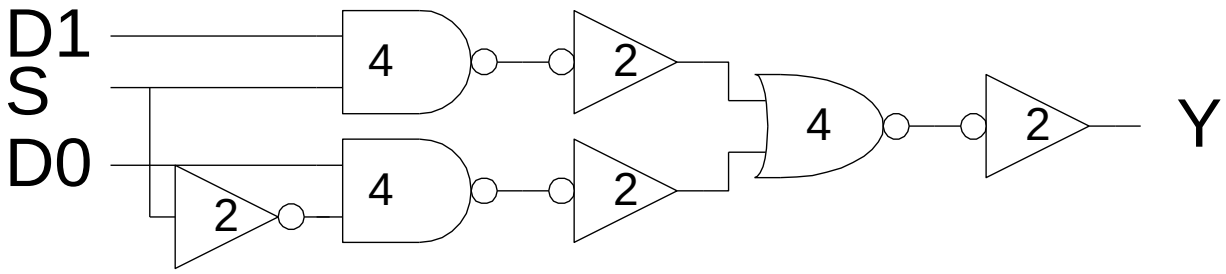
S	D1	D0	Y
0	X	0	0
0	X	1	1
1	0	X	0
1	1	X	1



Gate-Level Mux Design

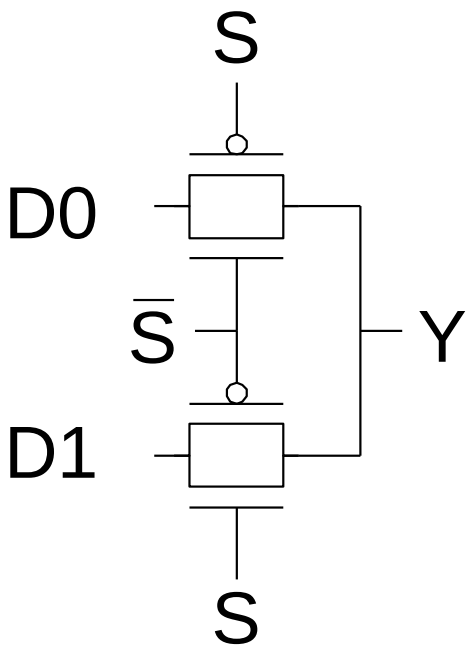
- $Y = SD_1 + \overline{S}D_0$ (too many transistors)
- How many transistors are needed? 20





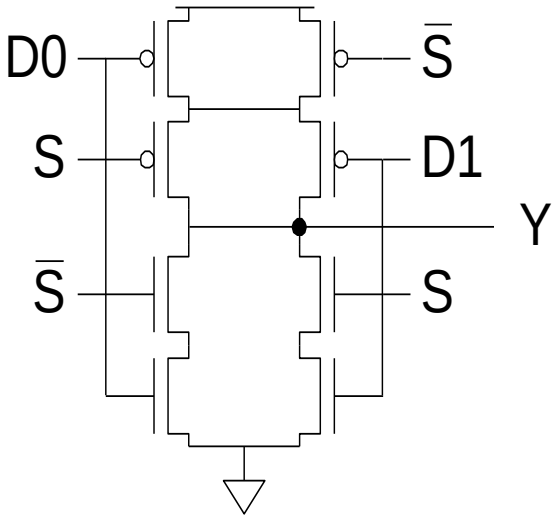
Transmission Gate Mux

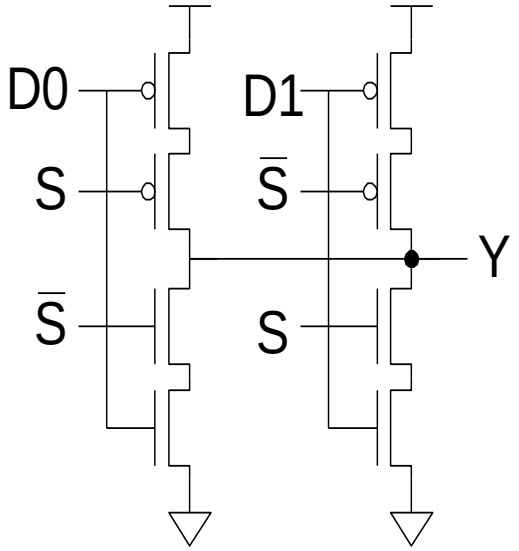
- Nonrestoring mux uses two transmission gates
  - Only 4 transistors

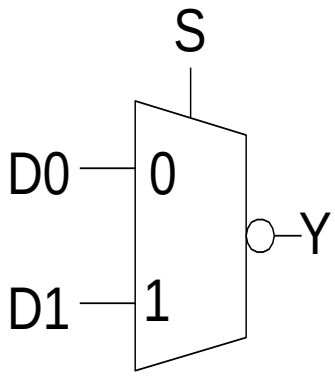


Inverting Mux

- Inverting multiplexer
  - Use compound AOI22
  - Or pair of tristate inverters
  - Essentially the same thing
- Noninverting multiplexer adds an inverter

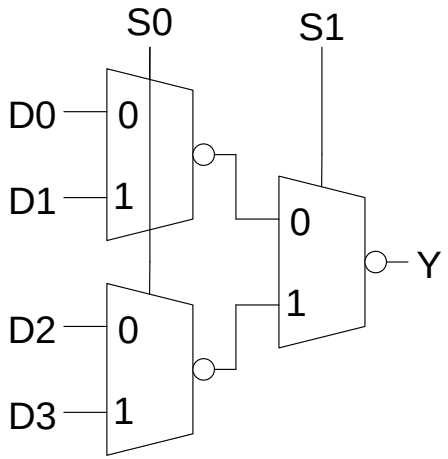


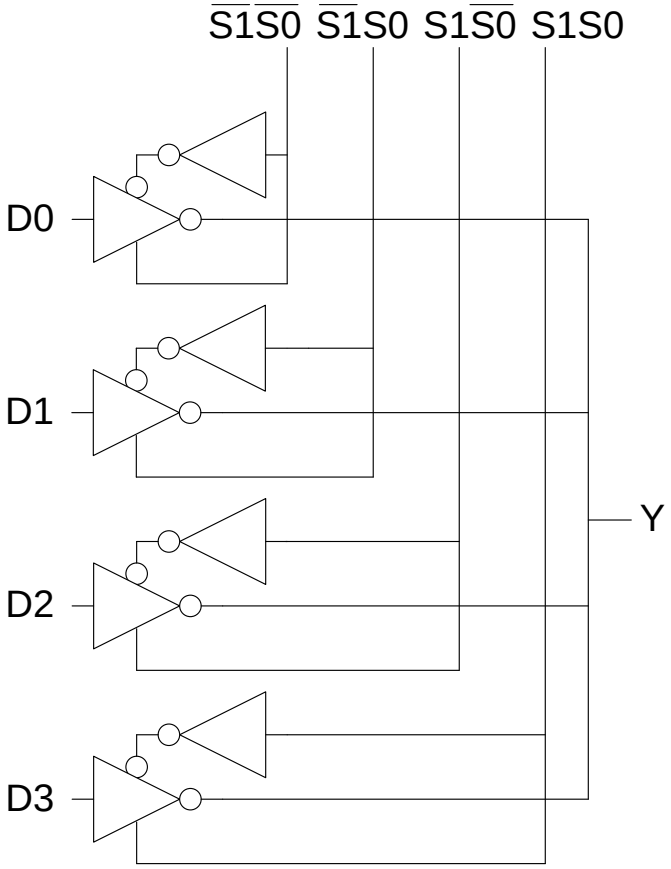




4:1 Multiplexer

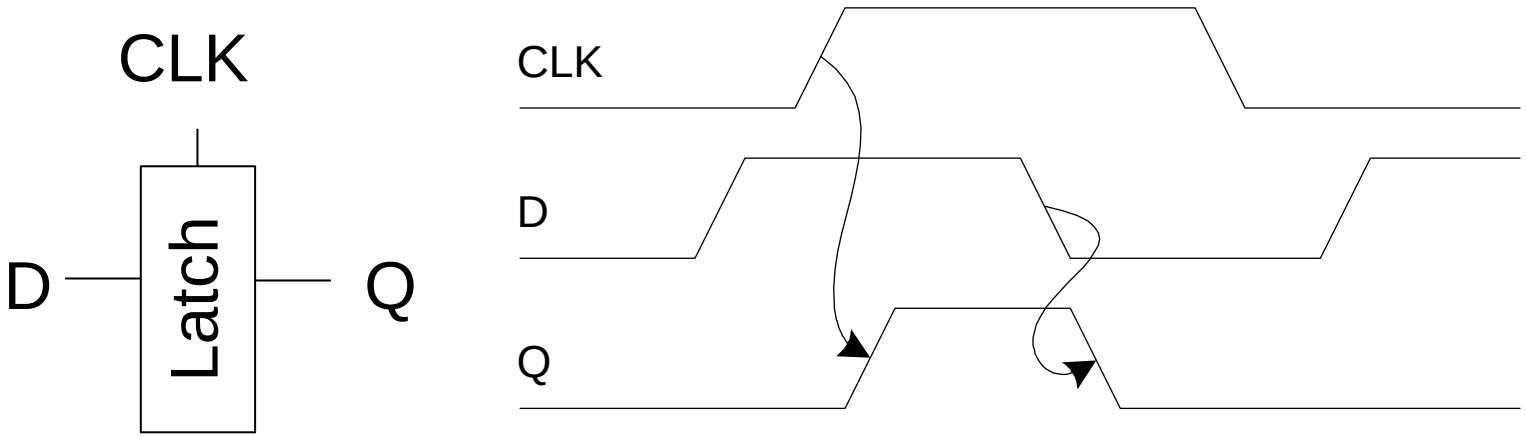
- 4:1 mux chooses one of 4 inputs using two selects
  - Two levels of 2:1 muxes
  - Or four tristates





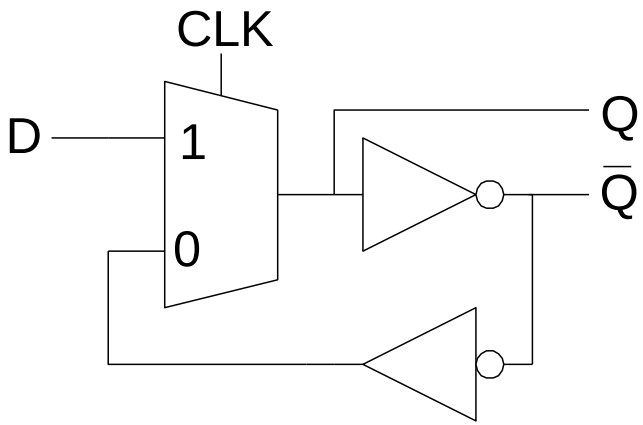
D Latch

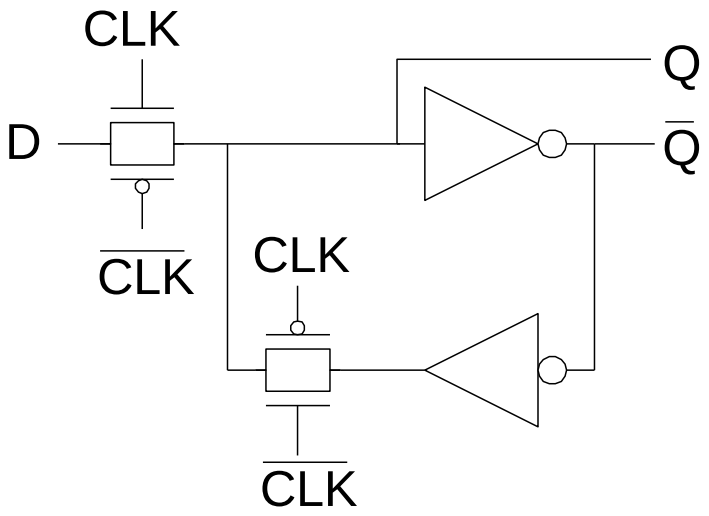
- When CLK = 1, latch is *transparent*
  - D flows through to Q like a buffer
- When CLK = 0, the latch is *opaque*
  - Q holds its old value independent of D
- a.k.a. *transparent latch* or *level-sensitive latch*



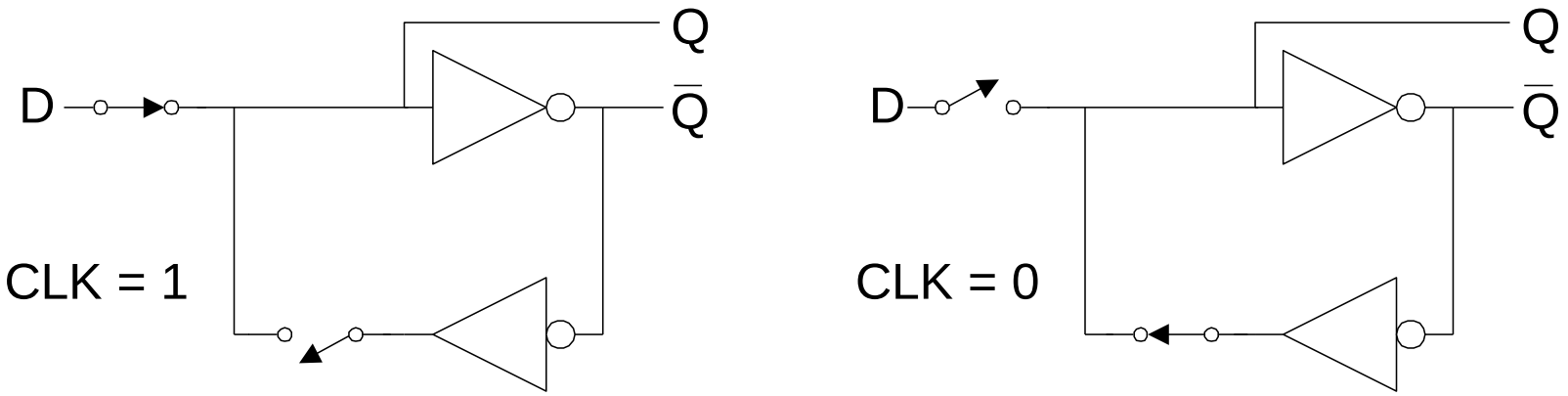
D Latch Design

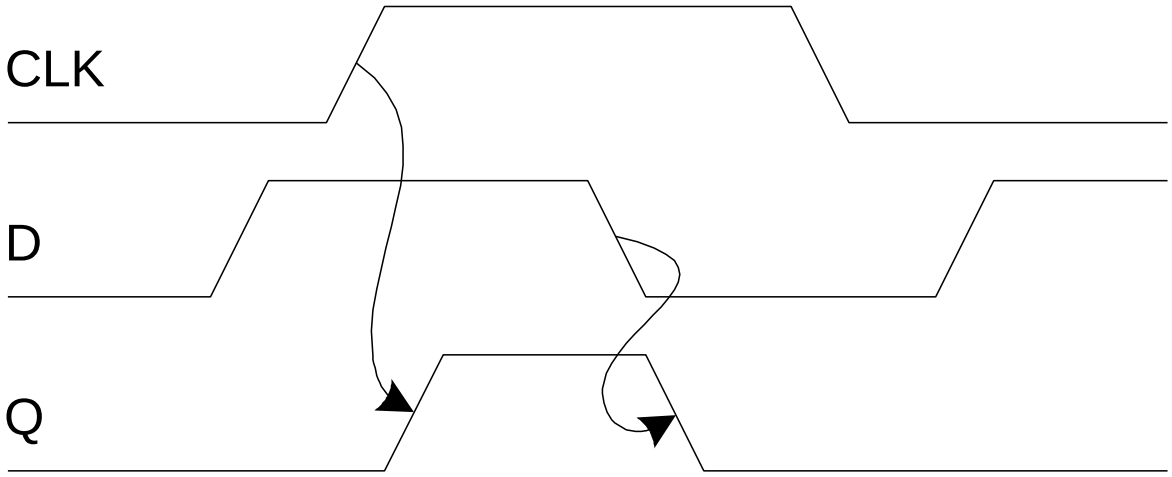
- Multiplexer chooses D or old Q





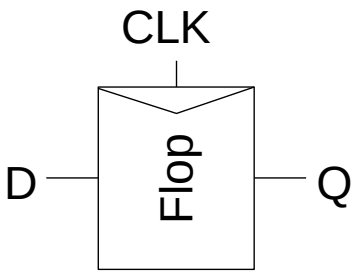
D Latch Operation

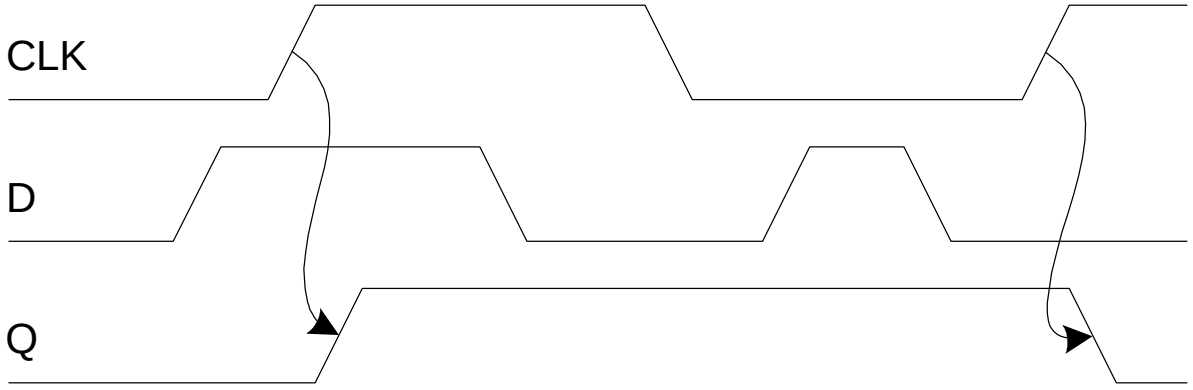




D Flip-flop

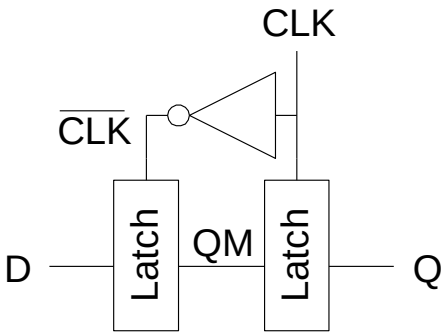
- When CLK rises, D is copied to Q
- At all other times, Q holds its value
- a.k.a. *positive edge-triggered flip-flop*, *master-slave flip-flop*

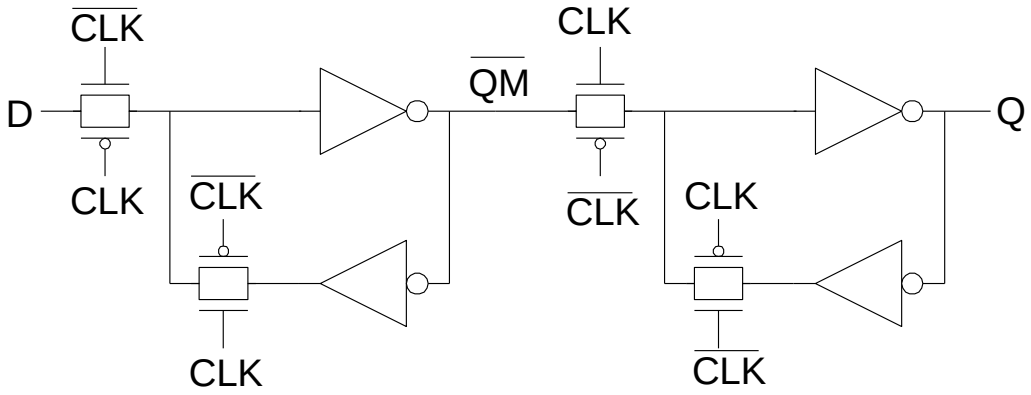




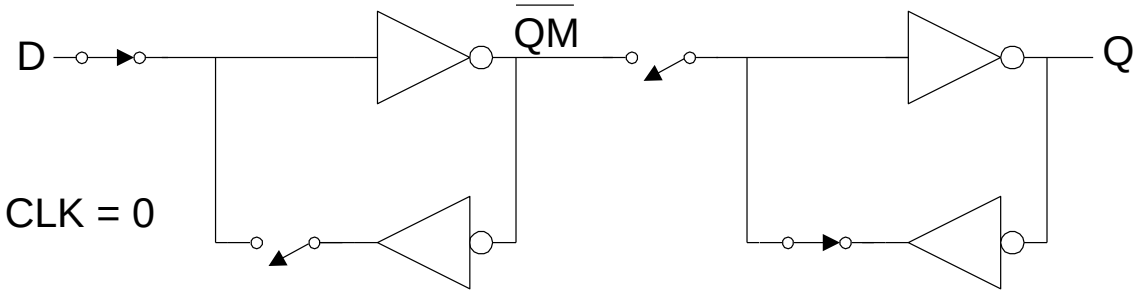
D Flip-flop Design

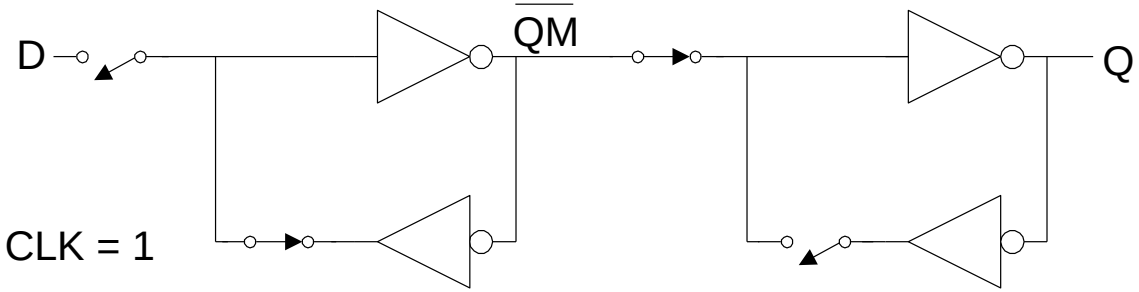
- Built from master and slave D latches

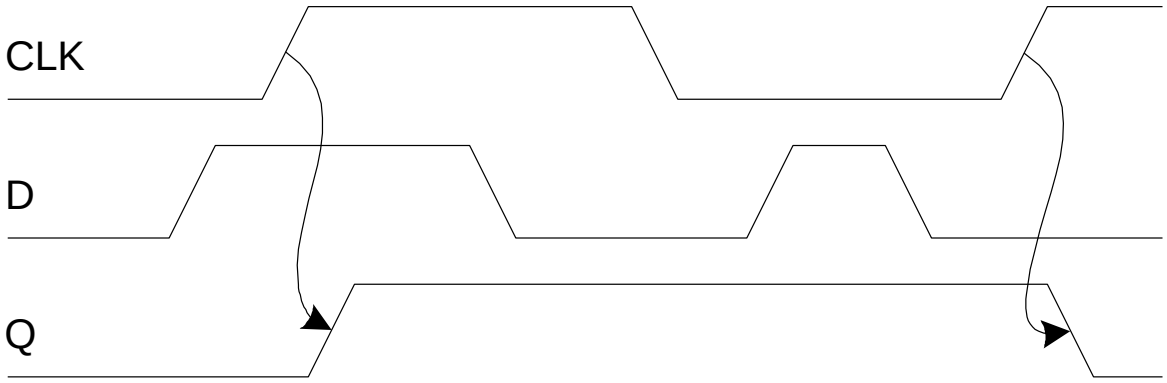




D Flip-flop Operation

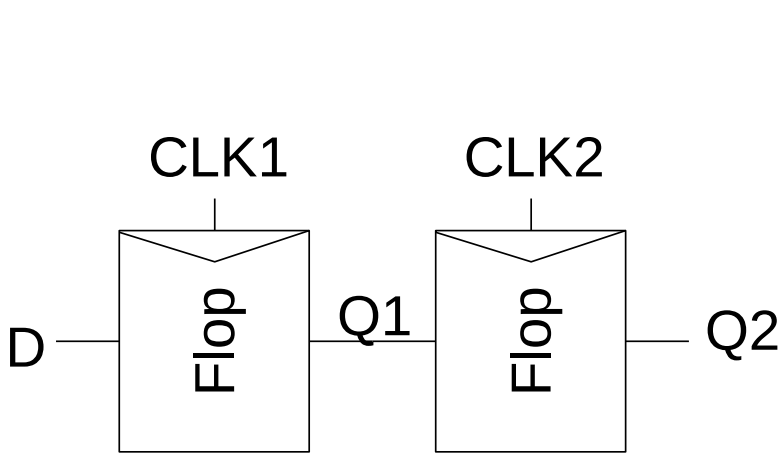


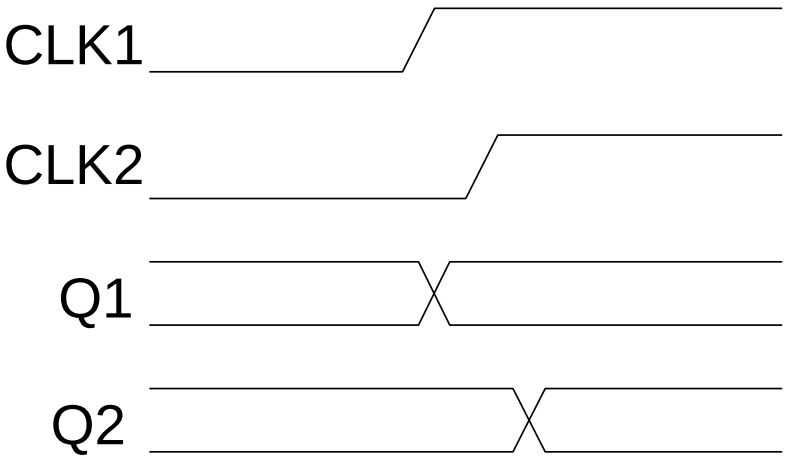




Race Condition

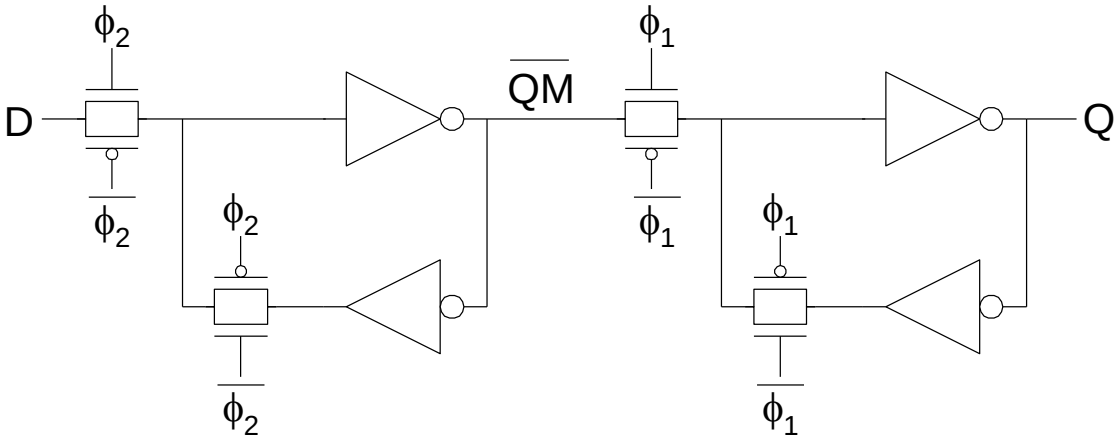
- Back-to-back flops can malfunction from clock skew
  - Second flip-flop fires late
  - Sees first flip-flop change and captures its result
  - Called *hold-time failure* or *race condition*

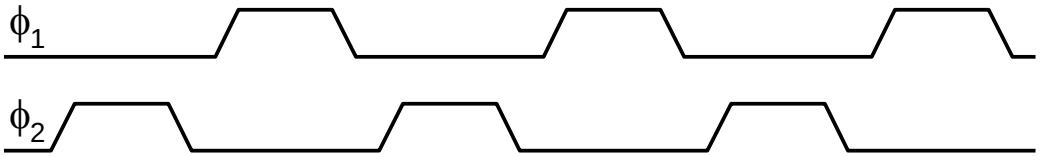




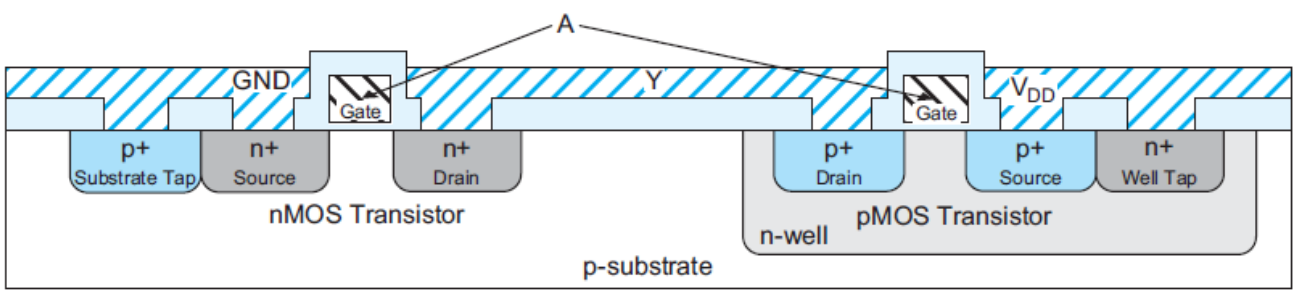
Nonoverlapping Clocks

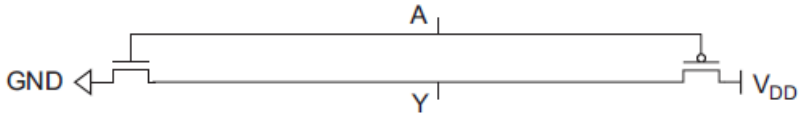
- Nonoverlapping clocks can prevent races
  - As long as nonoverlap exceeds clock skew
- We will use them in this class for safe design
  - Industry manages skew more carefully instead



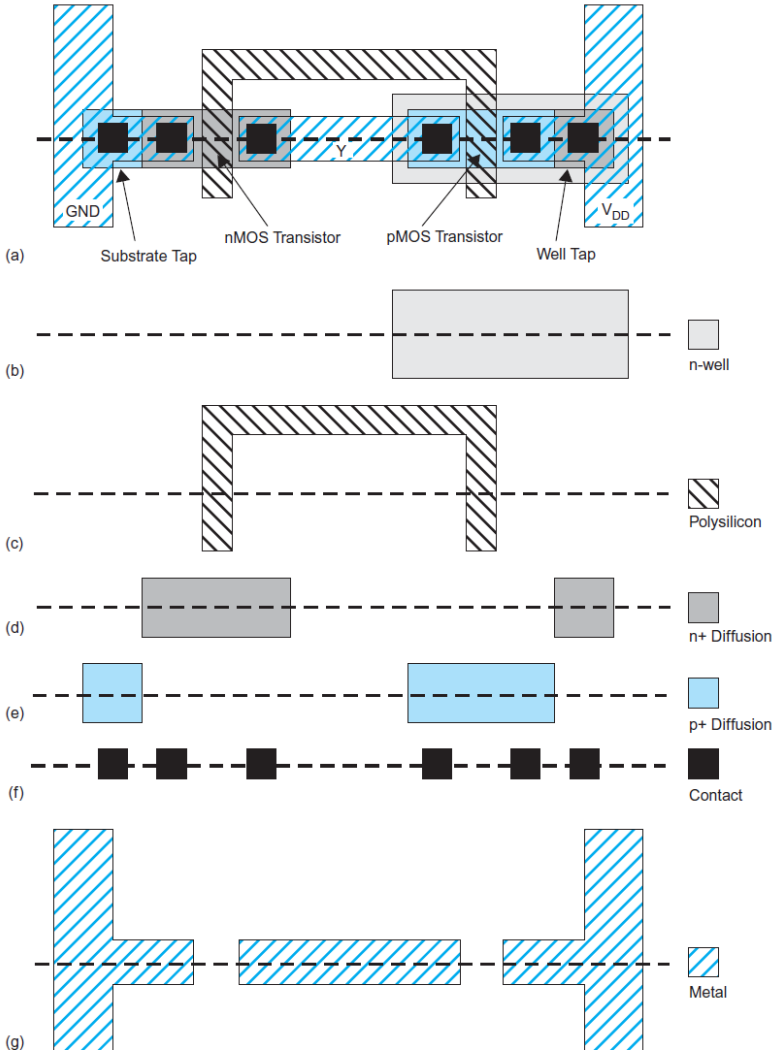


Inverter cross-section





Inverter mask set



Layout Design Rules

- Layout design rules describe how small features can be and how closely they can be reliably packed in a particular manufacturing process.
- Industrial design rules are usually specified in microns.
- This makes migrating from one process to a more advanced process or a different foundry's process difficult because not all rules scale in the same way.
- Universities sometimes simplify design by using scalable design rules that are conservative enough to apply to many manufacturing processes

Layout Design Rules

- Mead and Conway popularized scalable design rules based on a single parameter, λ, that characterizes the resolution of the process.
- λ is generally half of the minimum drawn transistor channel length.
- This length is the distance between the source and drain of a transistor and is set by the minimum width of a polysilicon wire.
- For example, a 180 nm process has a minimum polysilicon width (and hence transistor length) of 0.18 um and uses design rules with λ = 0.09 um

Feature Size

- Lambda-based rules are necessarily conservative because they round up dimensions to an integer multiple of λ.
- However, they make scaling layout trivial;
- The same layout can be moved to a new process simply by specifying a new value of λ.
- Design rules in terms of λ.
- The potential density advantage of micron rules is sacrificed for simplicity and easy scalability of lambda rules.
- Designers often describe a process by its *feature size*.
- Feature size refers to minimum transistor length, so λ is half the feature size.

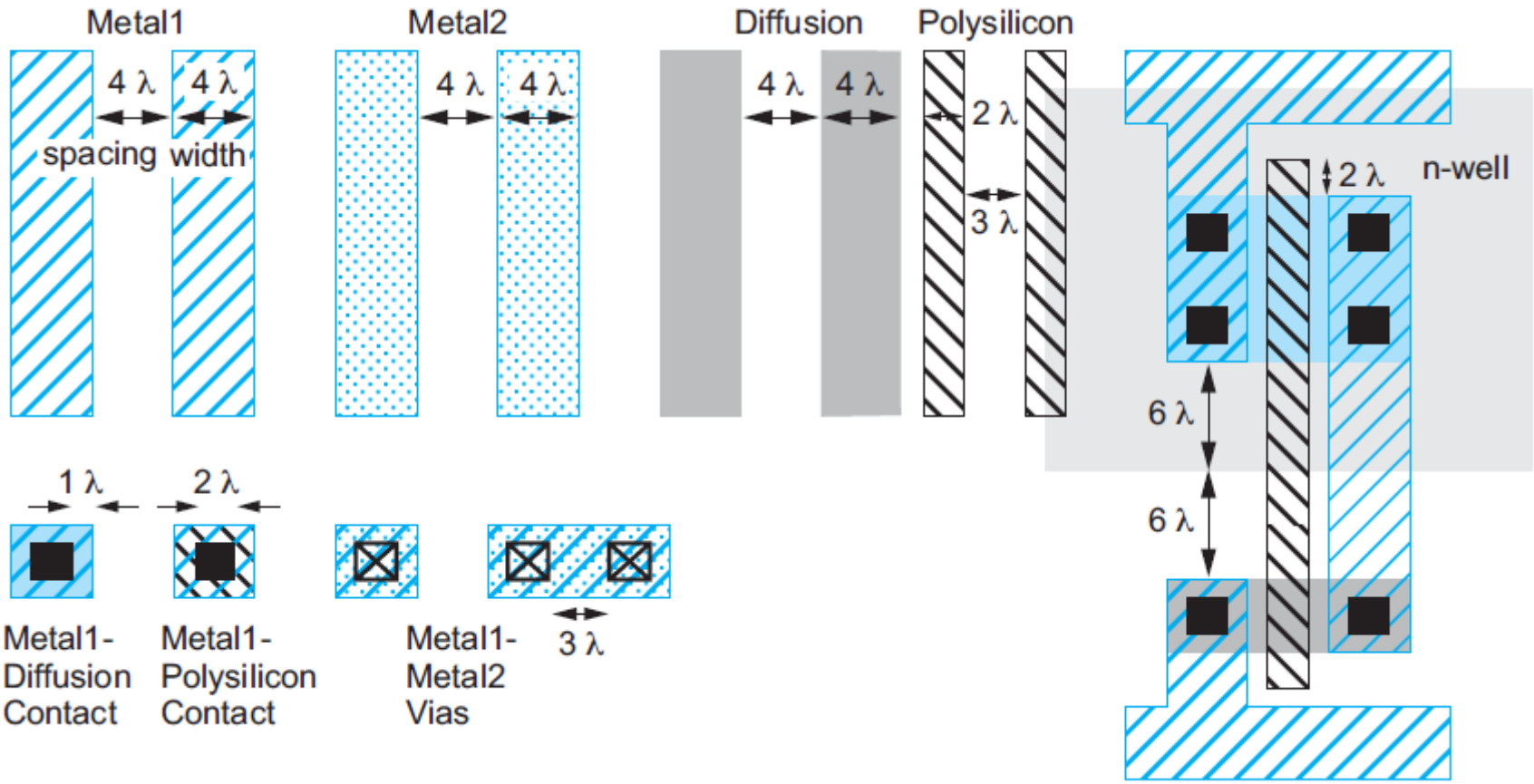
MOSIS Design Rules

- The MOSIS service is a low-cost prototyping service that collects designs from academic, commercial, and government customers and aggregates them onto one mask set to share overhead costs and generate production volumes sufficient to interest fabrication companies.
- MOSIS has developed a set of scalable lambda-based design rules that covers a wide range of manufacturing processes.
- The rules describe
- The minimum width to avoid breaks in a line,
- Minimum spacing to avoid shorts between lines, and
- Minimum overlap to ensure that two layers completely overlap.

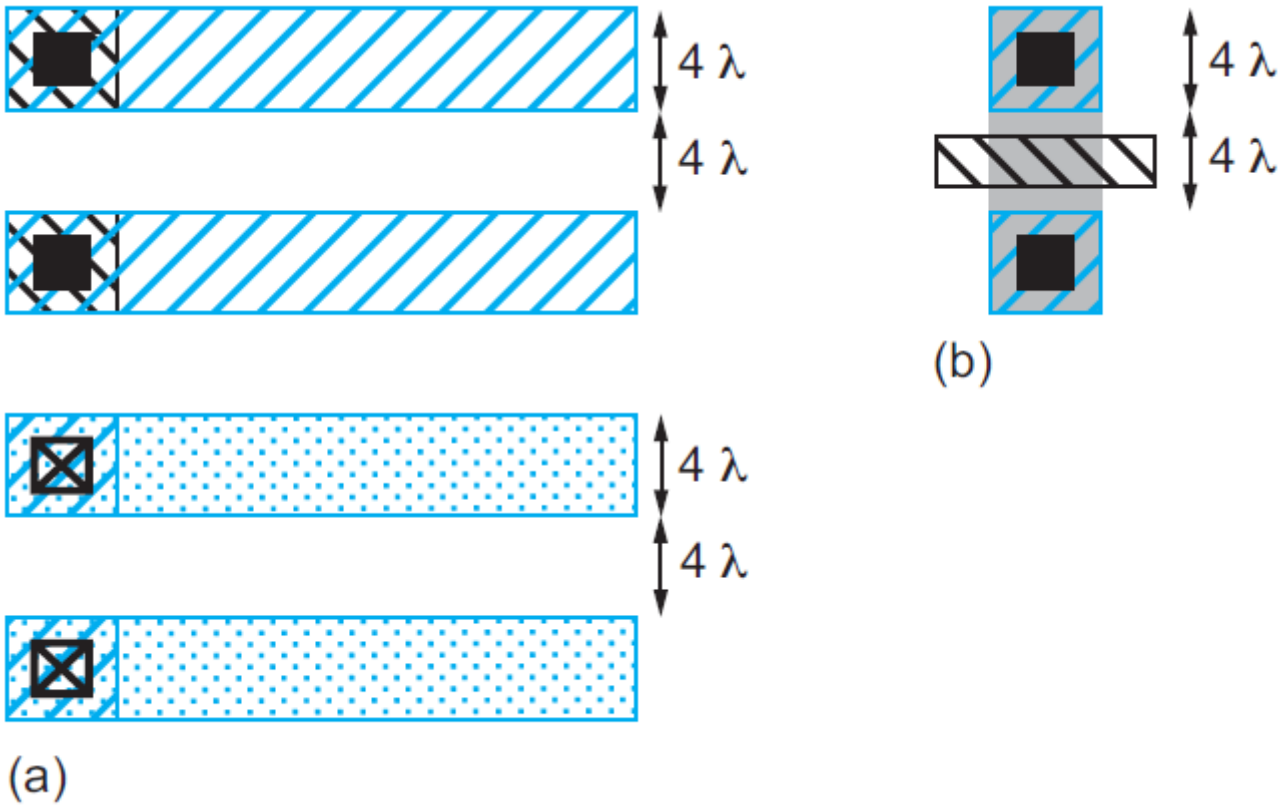
Layout Design Rules

- Metal and diffusion have minimum width and spacing of 4λ.
- Contacts are 2λ × 2λ and must be surrounded by 1λ on the layers above and below.
- Polysilicon uses a width of 2λ.
- Polysilicon overlaps diffusion by 2λ where a transistor is desired and has a spacing of 1λ away where no transistor is desired.
- Polysilicon and contacts have a spacing of 3λ from other polysilicon or contacts.
- N-well surrounds pMOS transistors by 6λ and avoids nMOS transistors by 6λ.

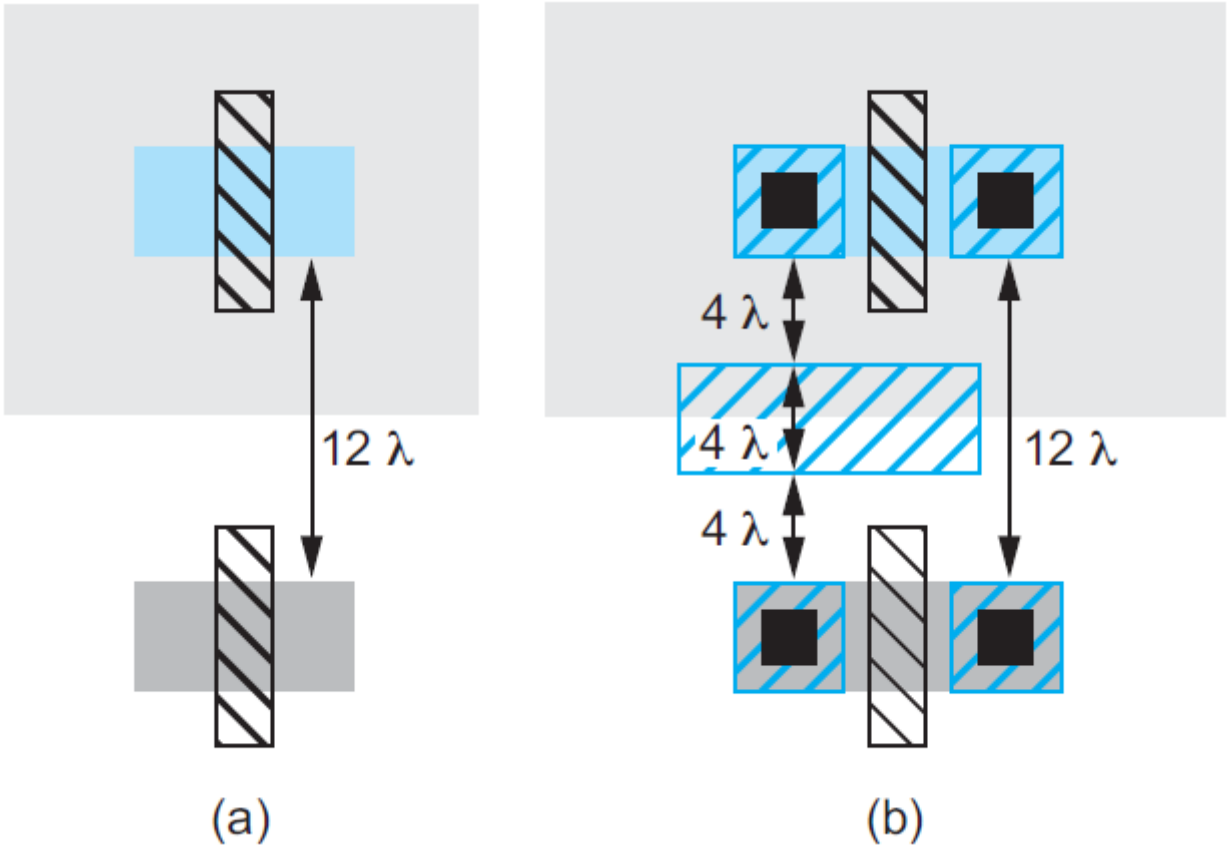
Simplified λ-based design rules



Pitch of routing tracks



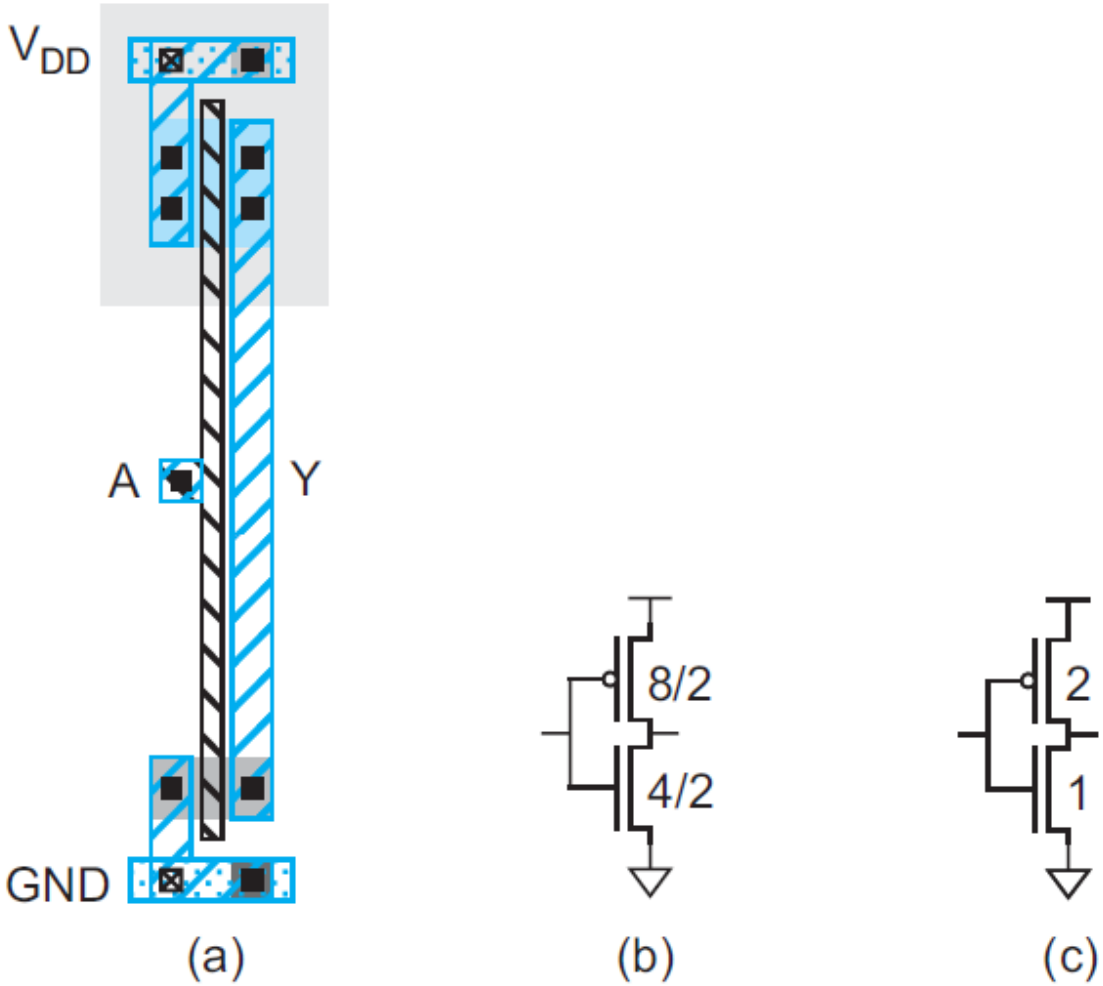
Spacing between nMOS and pMOS transistors



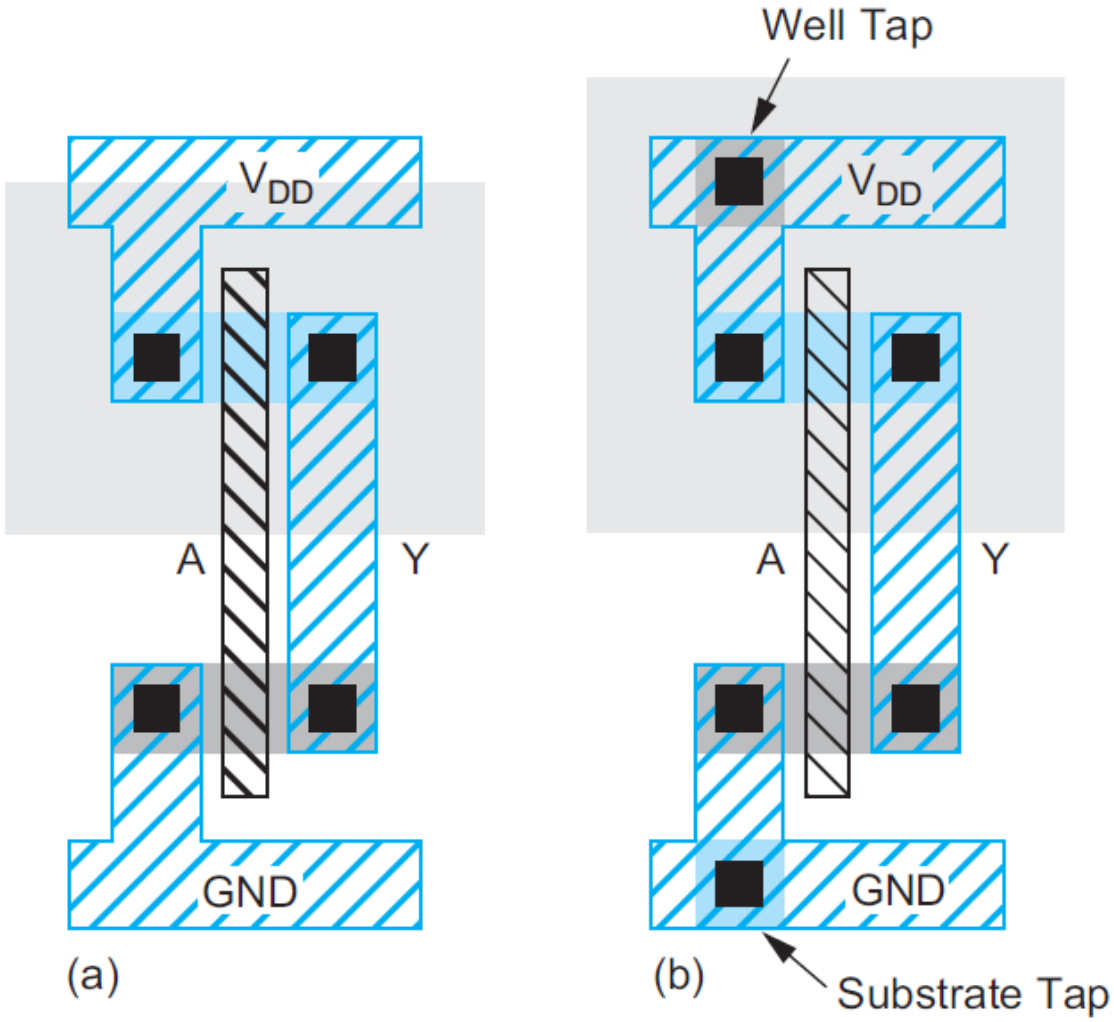
Transistor Size

- Transistor dimensions are often specified by their Width/Length (W/L) ratio
- the nMOS transistor - polysilicon crosses n-diffusion has a W/L of 4/2.
- Such a minimum-width contacted transistor is often called a unit transistor.
- In a 0.6 um process, this corresponds to an actual width of 1.2 um and a length of 0.6 um.
- pMOS transistors are often wider than nMOS transistors because holes move more slowly than electrons so the transistor has to be wider to deliver the same current.
- A unit inverter layout with a unit nMOS transistor and a double-sized pMOS transistor.
- In digital systems, transistors are typically chosen to have the minimum possible length because short-channel transistors are faster, smaller, and consume less power.

Inverter with dimensions labeled



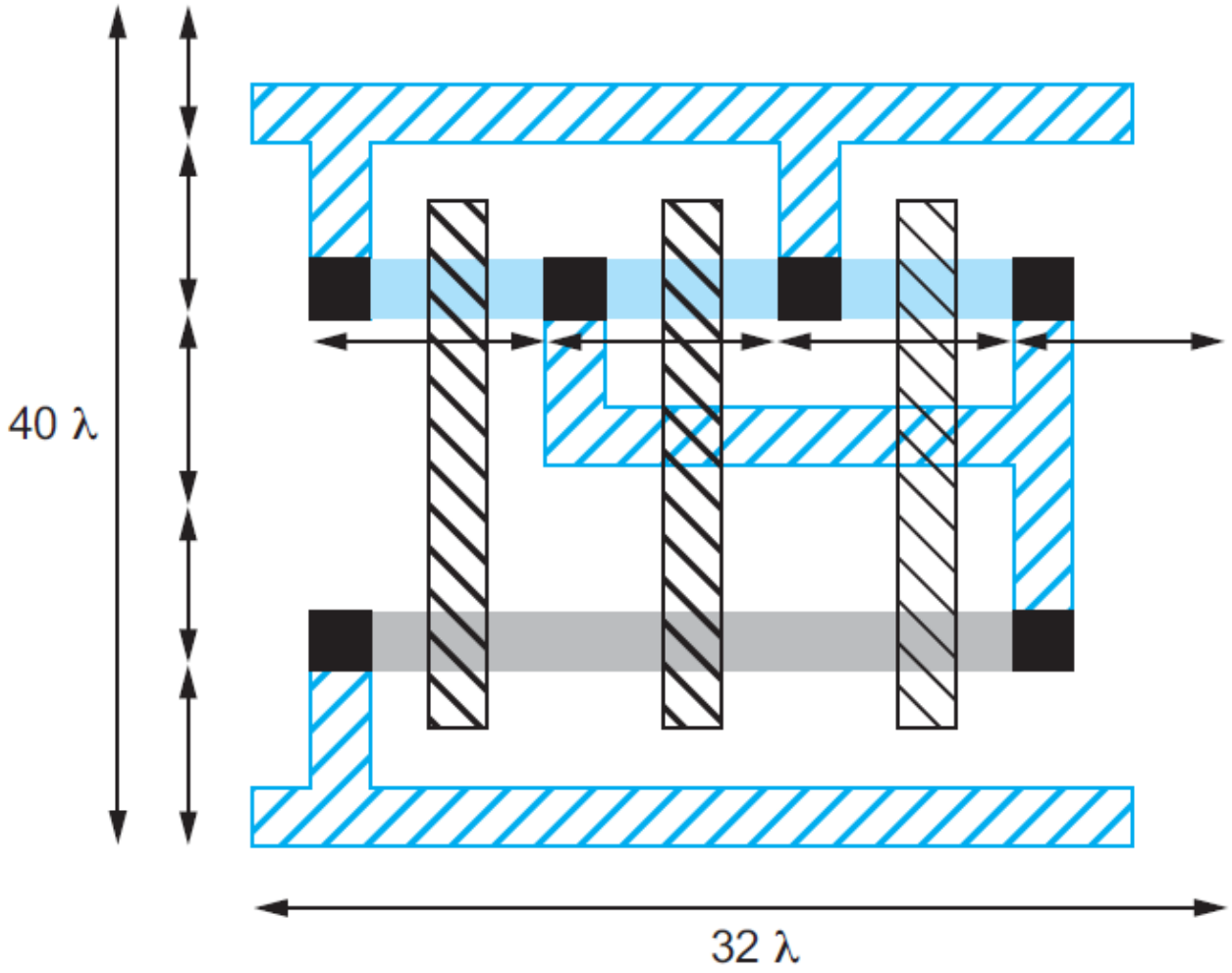
Gate Layouts



3-input NAND gate

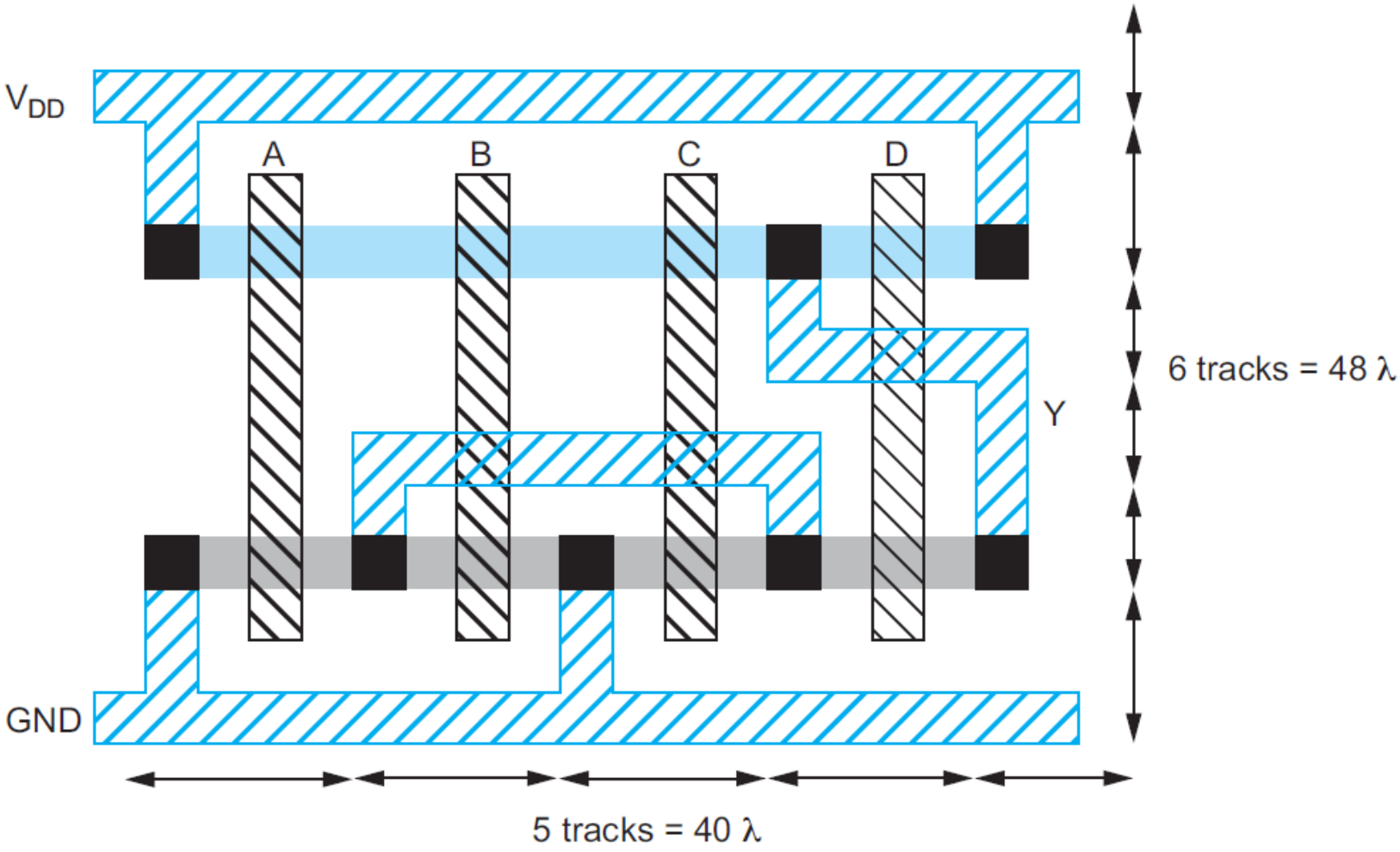
- Notice how the nMOS transistors are connected in series while the pMOS transistors are connected in parallel.
- Power and ground extend 2λ on each side so if two gates were abutted the contents would be separated by 4λ, satisfying design rules.
- The height of the cell is 36λ, or 40λ if the 4λ space between the cell and another wire above it is counted.
- There are four vertical wire tracks, multiplied by 8λ per track to give a cell width of 32λ.
- There are five horizontal tracks, giving a cell height of 40λ.

3-input NAND gate area estimation



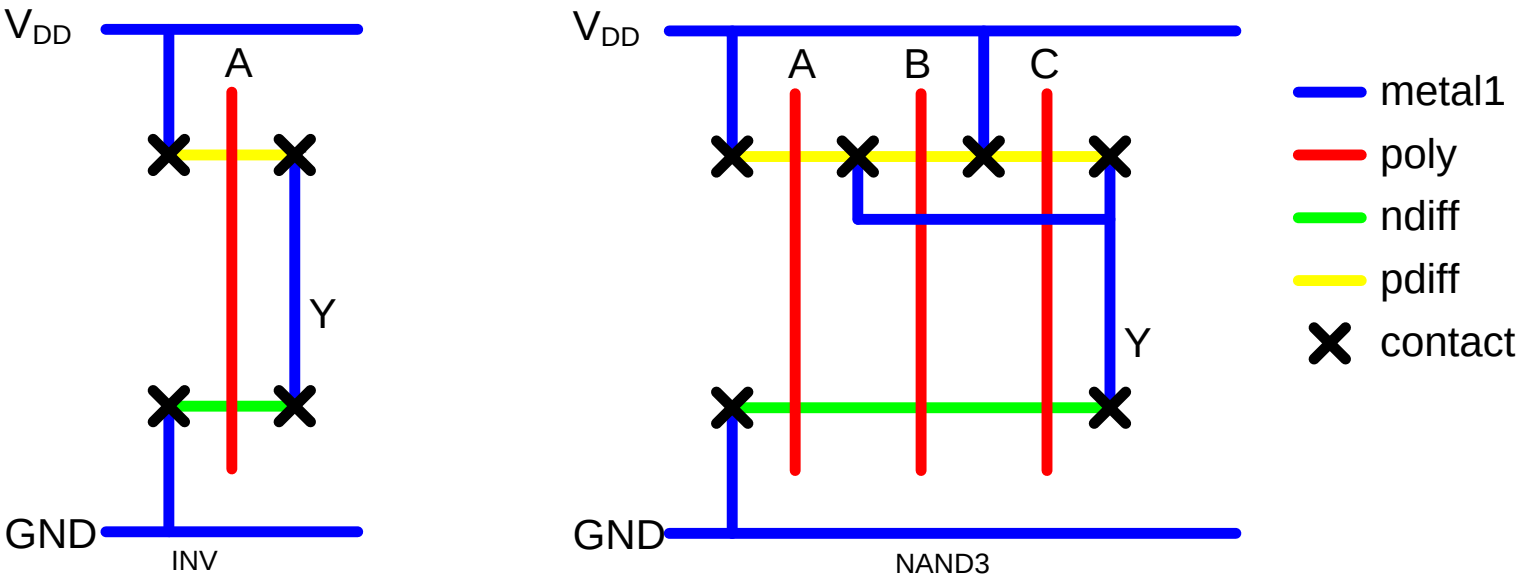
CMOS compound gate for function

$Y = (A + B + C) \cdot D$



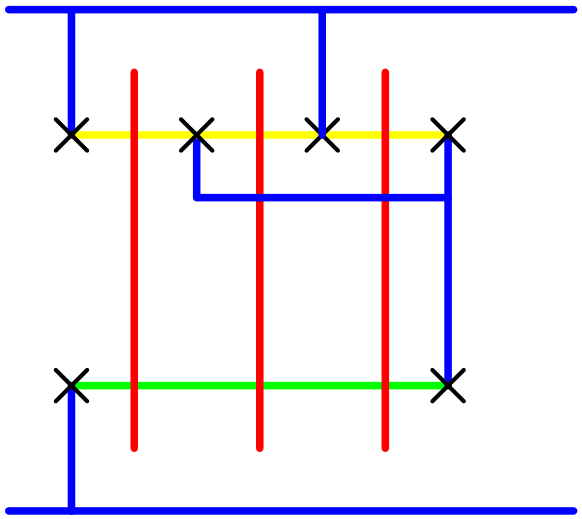
Stick Diagrams

- *Stick diagrams* help plan layout quickly
  - Need not be to scale
  - Draw with color pencils or dry-erase markers

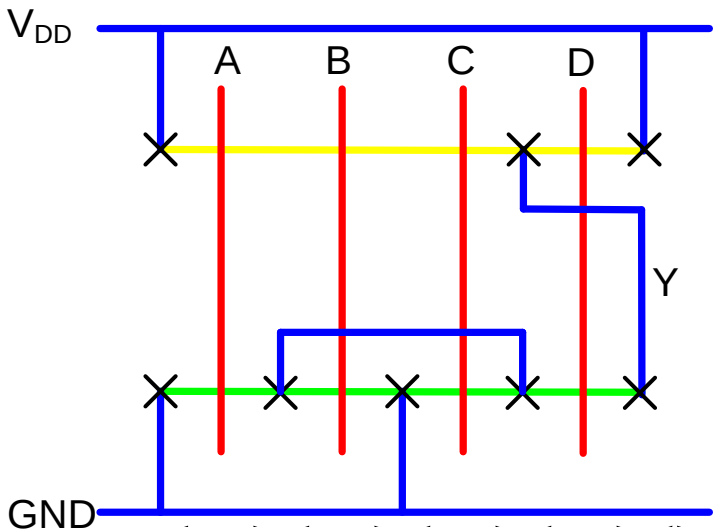


Area Estimation

- Estimate area by counting wiring tracks
  - Multiply by 8 to express in λ



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



Design Partitioning

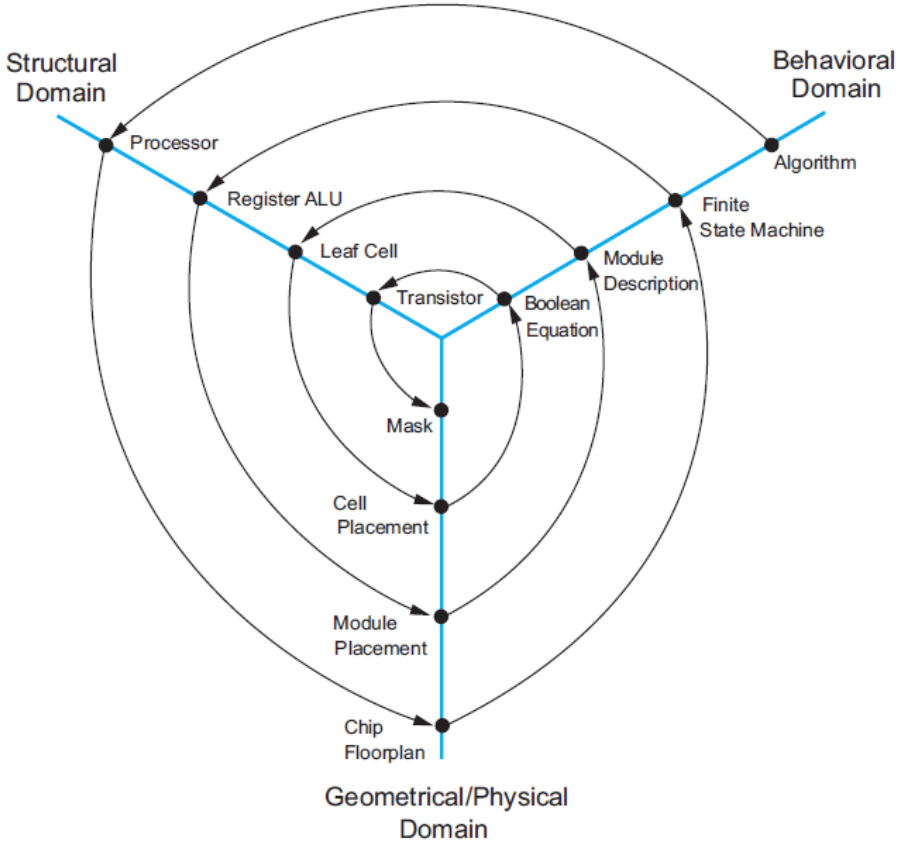
- The greatest challenge in modern VLSI design is not in designing the individual transistors but rather in managing system complexity.
- Modern *System-On-Chip* (SOC) designs combine memories, processors, high-speed I/O interfaces, and dedicated application-specific logic on a single chip.
- They use hundreds of millions or billions of transistors and cost tens of millions of dollars (or more) to design.
- The implementation must be divided among large teams of engineers and each engineer must be highly productive.

Design Partitioning

- If the implementation is too rigidly partitioned, each block can be optimized without regard to its neighbors, leading to poor system results.
- Conversely, if every task is interdependent with every other task, design will progress too slowly.
- Design managers face the challenge of choosing a suitable trade-off between these extremes.
- There is no substitute for practical experience in making these choices, and talented engineers who have experience with multiple designs are very important to the success of a large project.
- Design proceeds through multiple levels of abstraction, hiding details until they become necessary.
- The practice of *structured design*, which is also used in large software projects, uses the principles of hierarchy, regularity, modularity, and locality to manage the complexity.

- Digital VLSI design is often partitioned into five levels of abstractions: *architecture* design, *microarchitecture* design, *logic* design, *circuit* design, and *physical* design.
- Architecture describes the functions of the system.

Behavioral, Structural, and Physical Domains - Y-chart



Reference

- CMOS VLSI Design – A Circuits and Systems Perspective by **Neil H. E. Weste,** Pearson Publications