Introduction to CMOS VLSI Design

Circuits & Layout

Outline

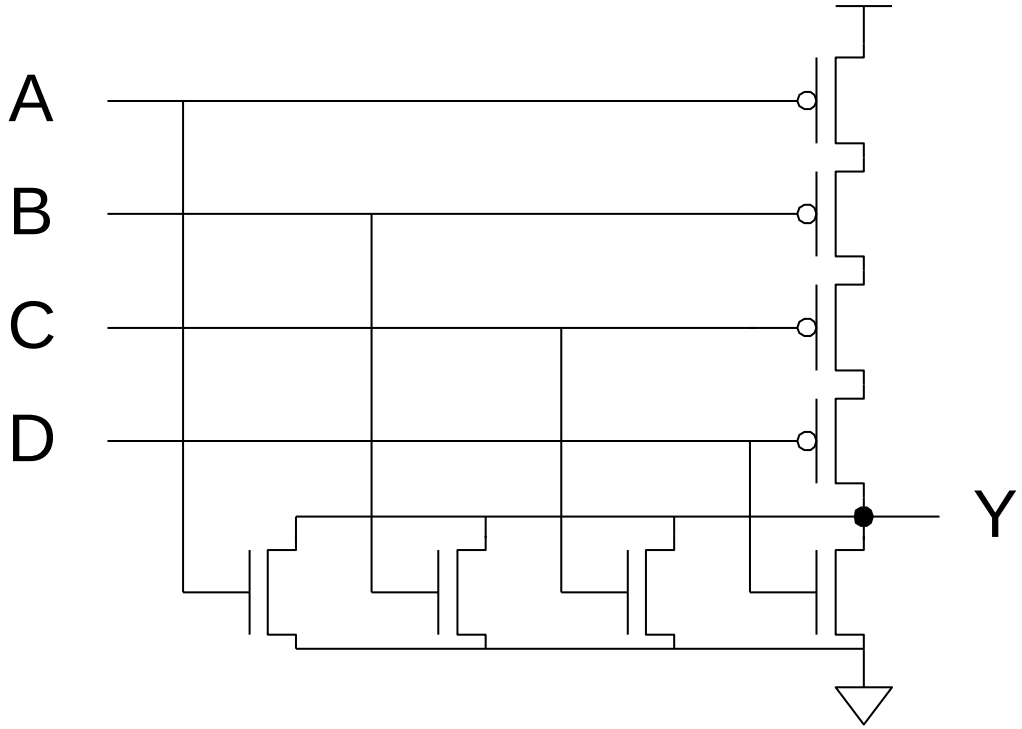
- CMOS Gate Design
- Pass Transistors
- CMOS Latches & Flip-Flops
- Standard Cell Layouts
- Stick Diagrams

CMOS Gate Design

- Activity:
  - Sketch a 4-input CMOS NAND gate

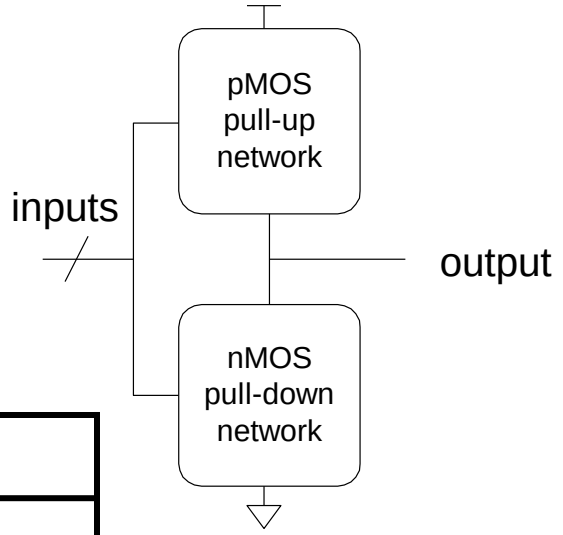
CMOS Gate Design

- Activity:
  - Sketch a 4-input CMOS NOR gate



Complementary CMOS

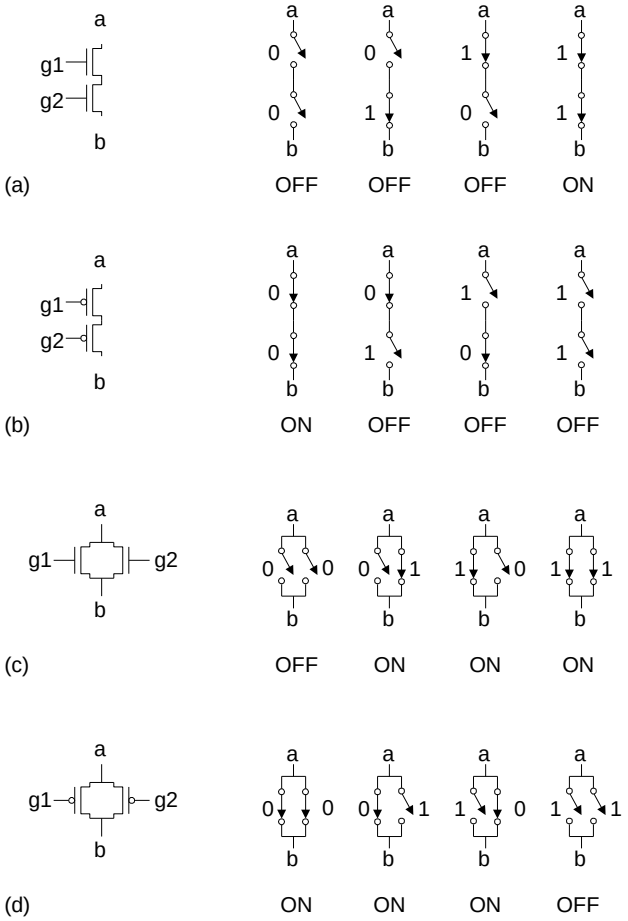
- Complementary CMOS logic gates
  - nMOS *pull-down network*
  - pMOS *pull-up network*
  - a.k.a. static CMOS



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

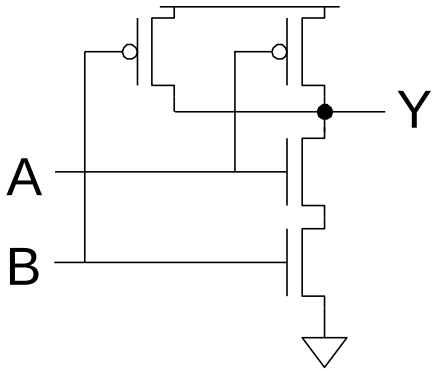
Series and Parallel

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



Conduction Complement

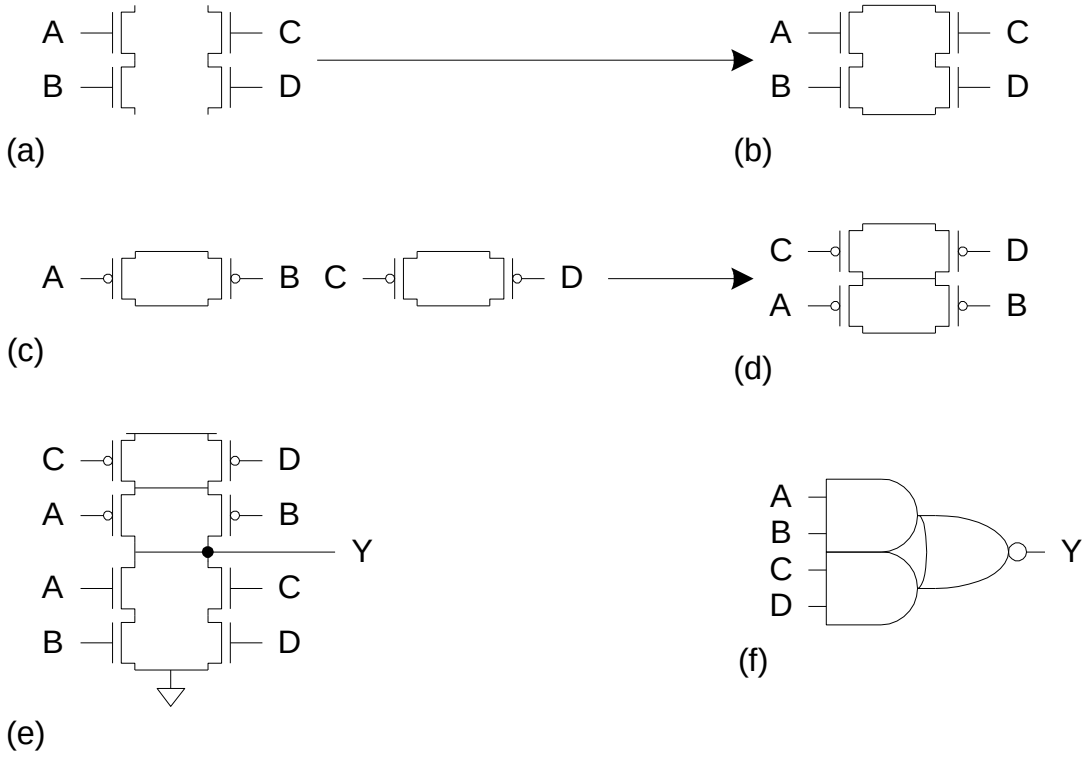
- Complementary CMOS gates always produce 0 or 1
- Ex: NAND gate
  - Series nMOS: Y=0 when both inputs are 1
  - Thus Y=1 when either input is 0
  - Requires parallel pMOS



- 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
- Ex: Y = (A.B + C.D)'

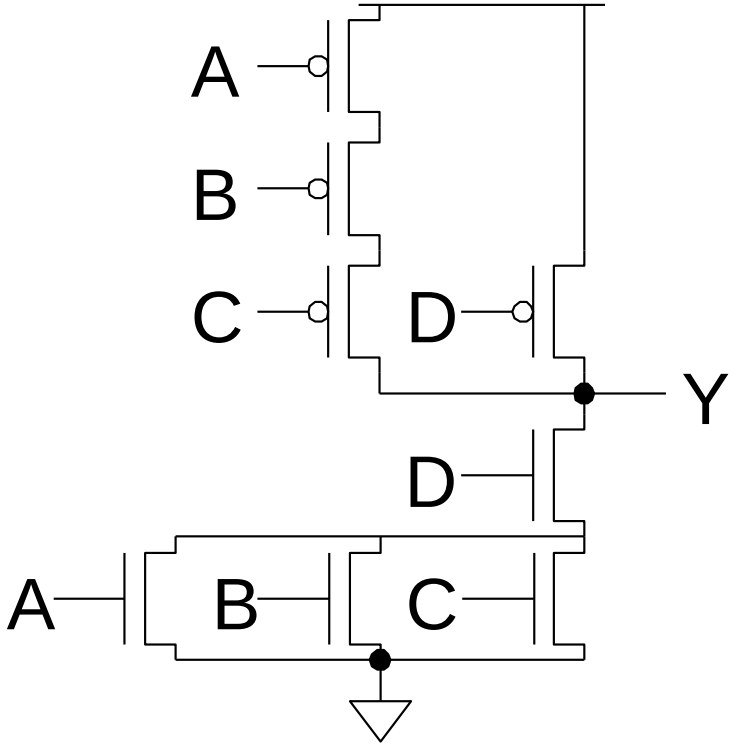


Example: O3AI

- Y = ((A+B+C).D)'

Example: O3AI

- Y = ((A+B+C).D)'

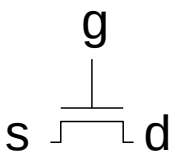


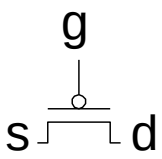
Signal Strength

- *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

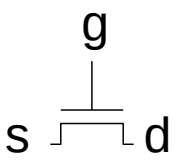
- Transistors can be used as switches

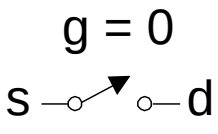


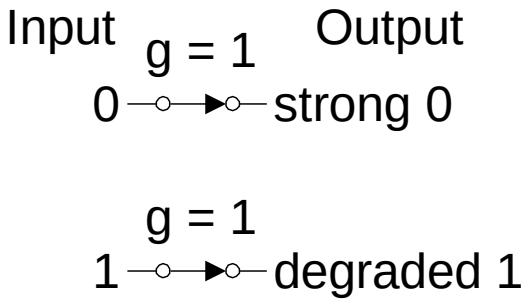


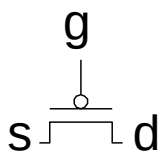
Pass Transistors

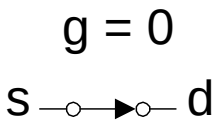
- Transistors can be used as switches

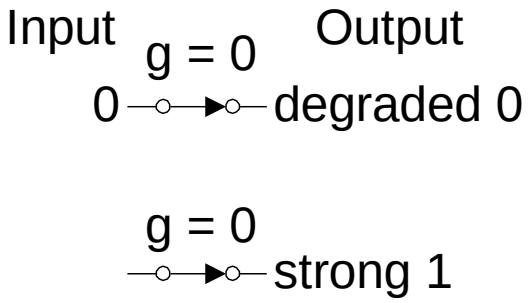










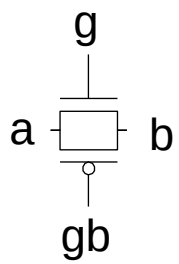


Transmission Gates

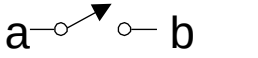
- Pass transistors produce degraded outputs
- *Transmission gates* pass both 0 and 1 well

Transmission Gates

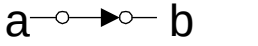
- Pass transistors produce degraded outputs
- *Transmission gates* pass both 0 and 1 well



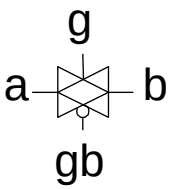
g = 0, gb = 1

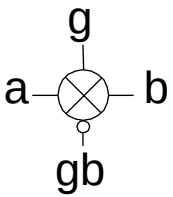


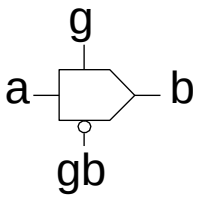
g = 1, gb = 0



Input	Output
g = 1, gb = 0 0	strong 0
g = 1, gb = 0 1	strong 1



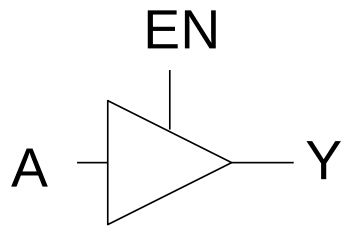


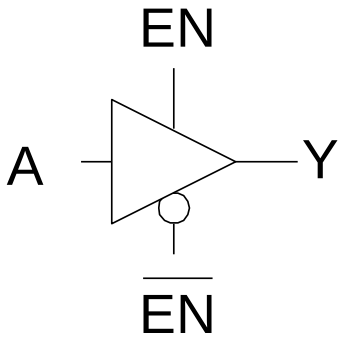


Tristates

- *Tristate buffer* produces Z when not enabled

EN	A	Y
0	0	
0	1	
1	0	
1	1	

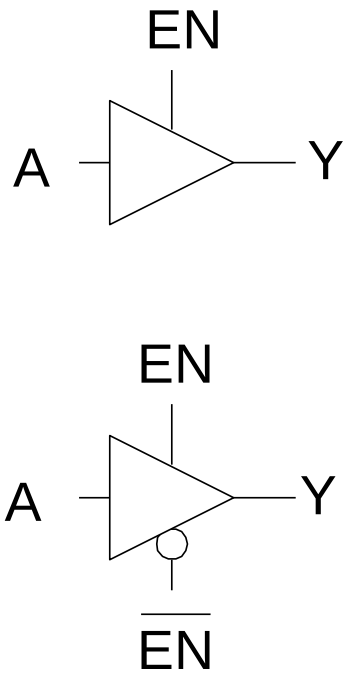




Tristates

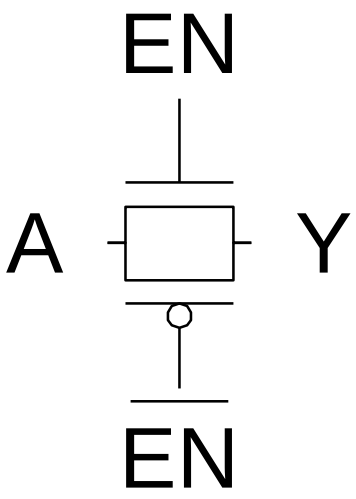
- *Tristate buffer* produces Z when not enabled

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



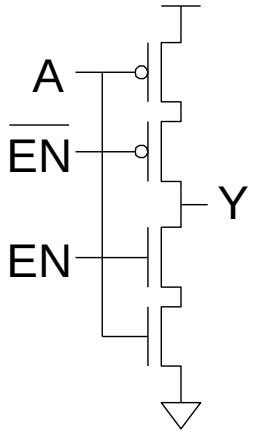
Nonrestoring Tristate

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



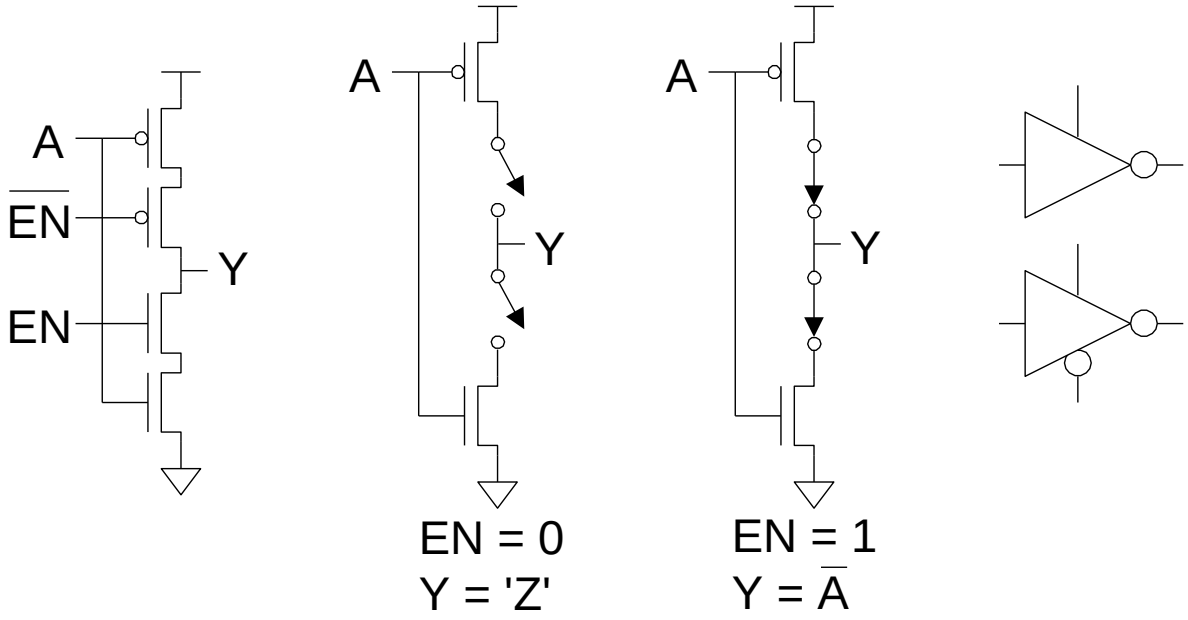
Tristate Inverter

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



Tristate Inverter

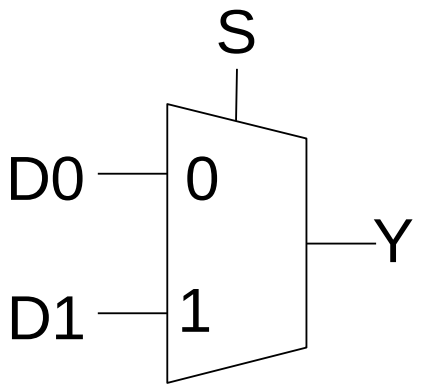
- Tristate inverter produces restored output
  - Violates conduction complement rule
  - Because we want a Z output



Multiplexers

- 2:1 *multiplexer* chooses between two inputs

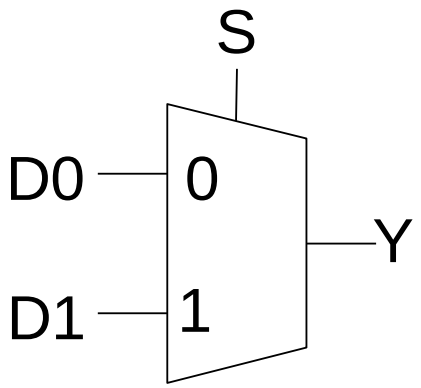
S	D1	D0	Y
0	X	0	
0	X	1	
1	0	X	
1	1	X	



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

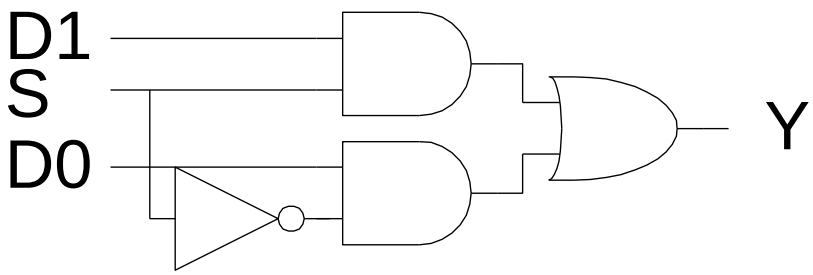


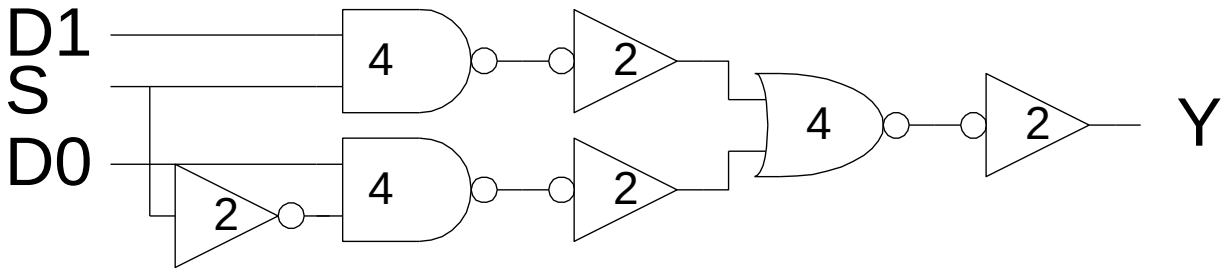
Gate-Level Mux Design

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

Gate-Level Mux Design

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



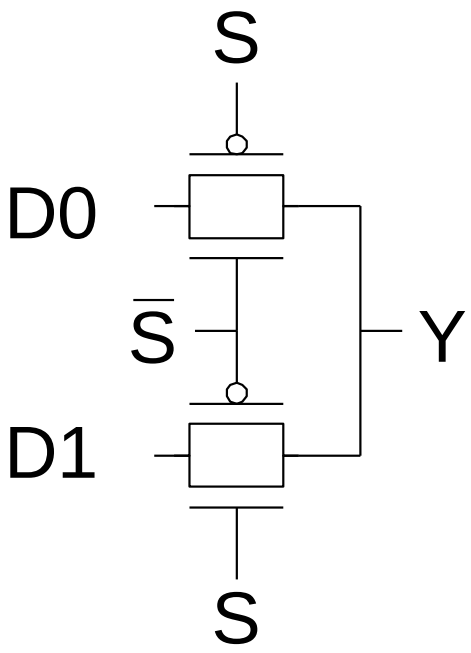


Transmission Gate Mux

- Nonrestoring mux uses two transmission gates

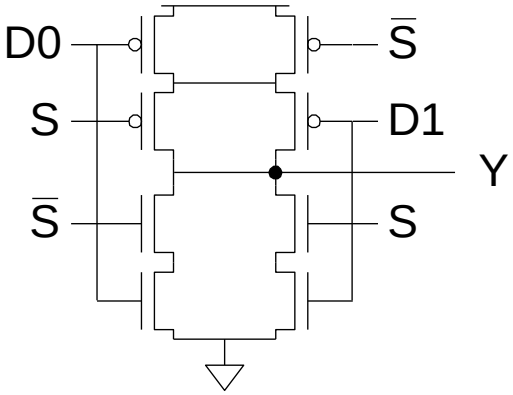
Transmission Gate Mux

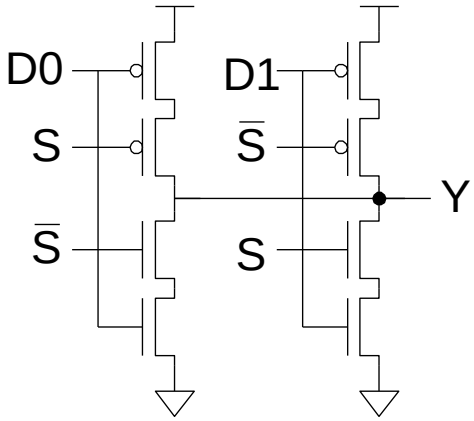
- Nonrestoring mux uses two transmission gates
  - Only 4 transistors

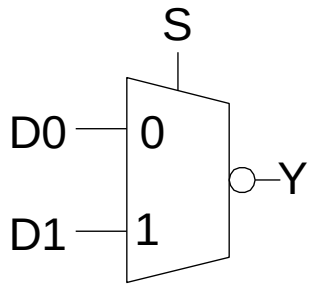


Inverting Mux

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





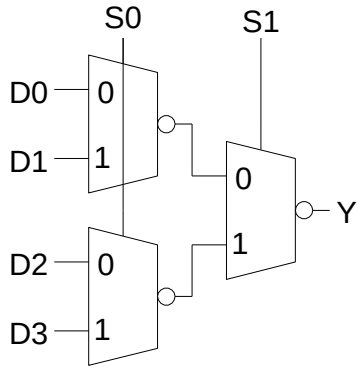


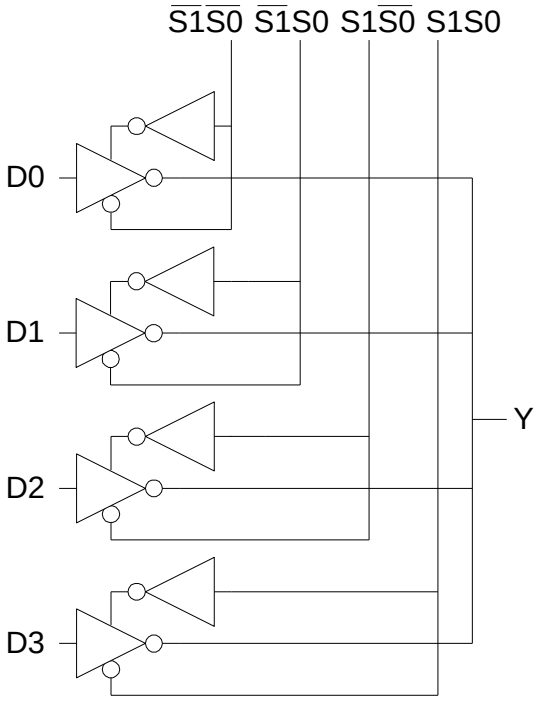
4:1 Multiplexer

- 4:1 mux chooses one of 4 inputs using two selects

4:1 Multiplexer

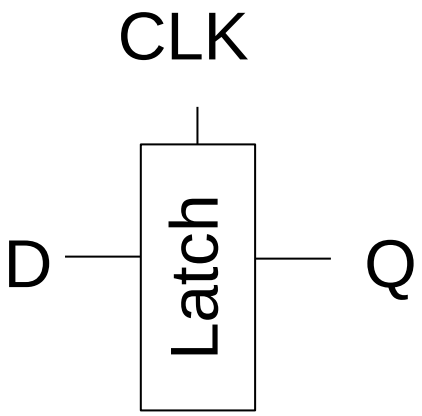
- 4:1 mux chooses one of 4 inputs using two selects
  - Two levels of 2:1 muxes
  - Or four tristates

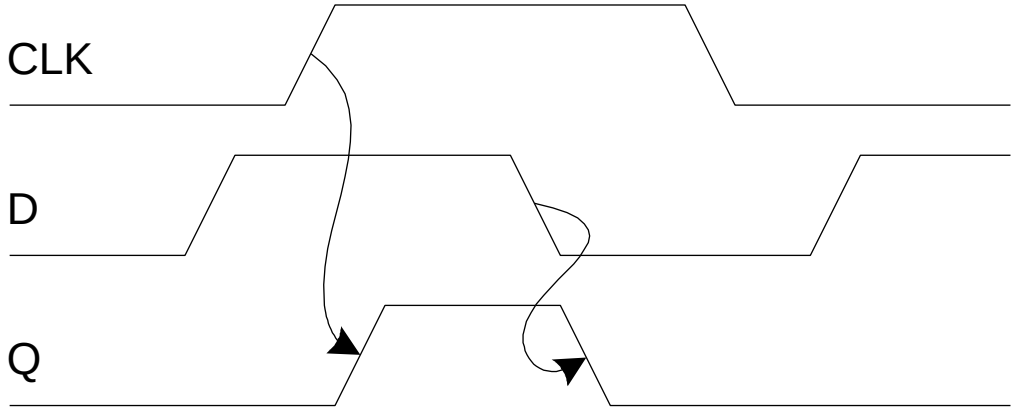




D Latch

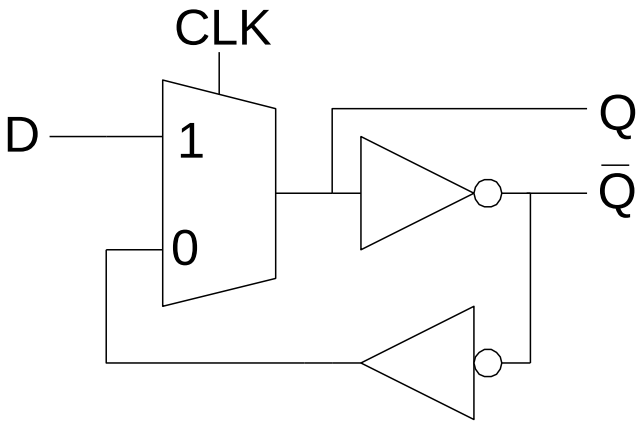
- When CLK = 1, latch is *transparent*
  - D flows through to Q like a buffer
- When CLK = 0, the latch is *opaque*
  - Q holds its old value independent of D
- a.k.a. *transparent latch* or *level-sensitive latch*

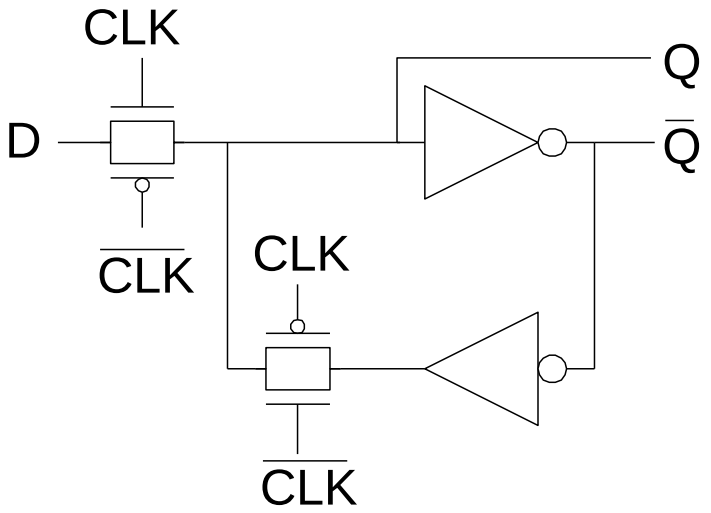




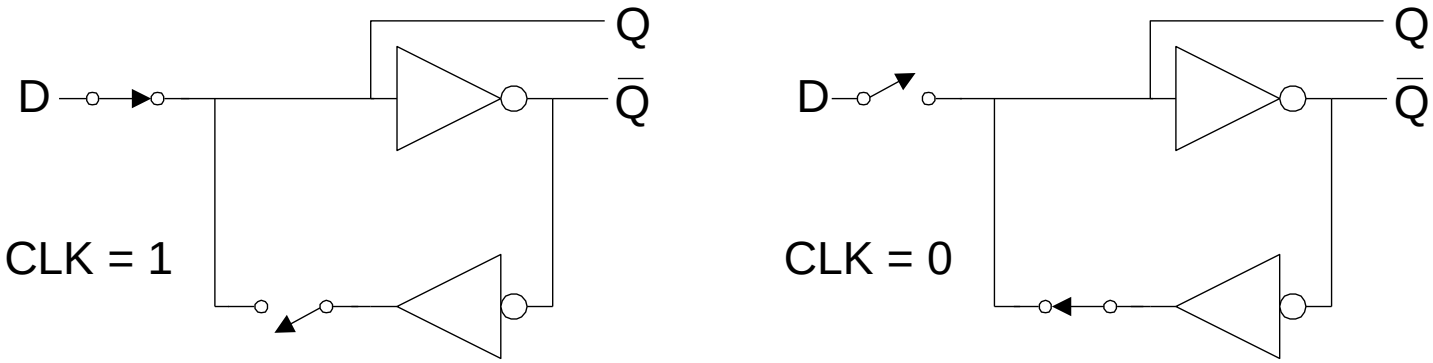
D Latch Design

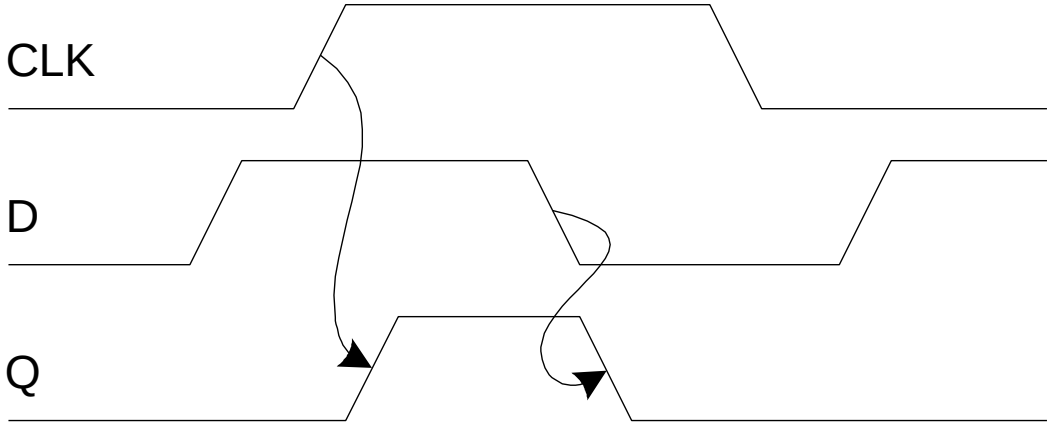
- Multiplexer chooses D or old Q





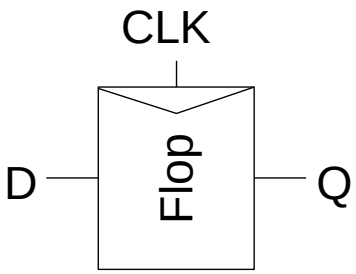
D Latch Operation

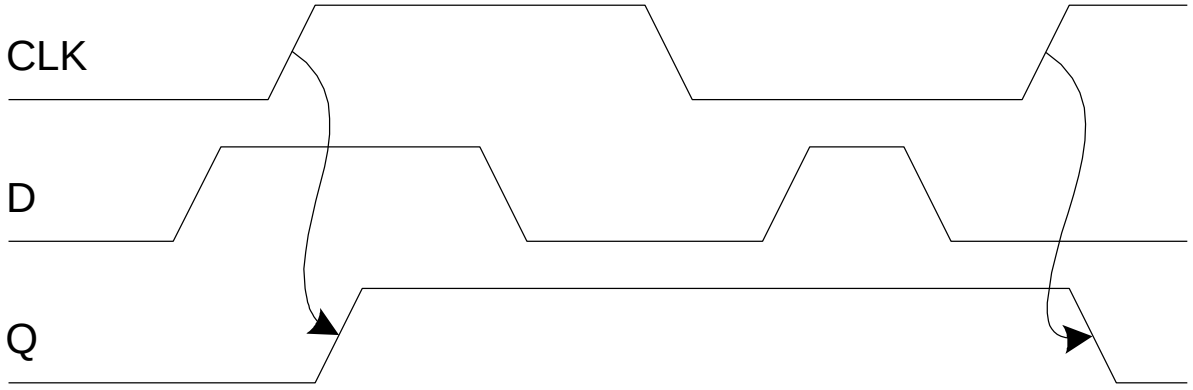




D Flip-flop

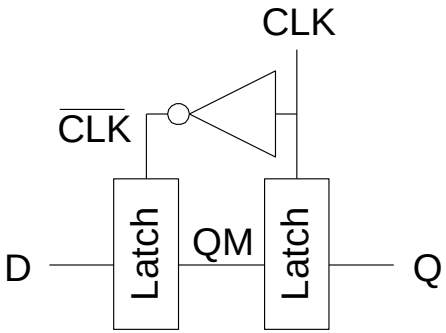
- When CLK rises, D is copied to Q
- At all other times, Q holds its value
- a.k.a. *positive edge-triggered flip-flop*, *master-slave flip-flop*

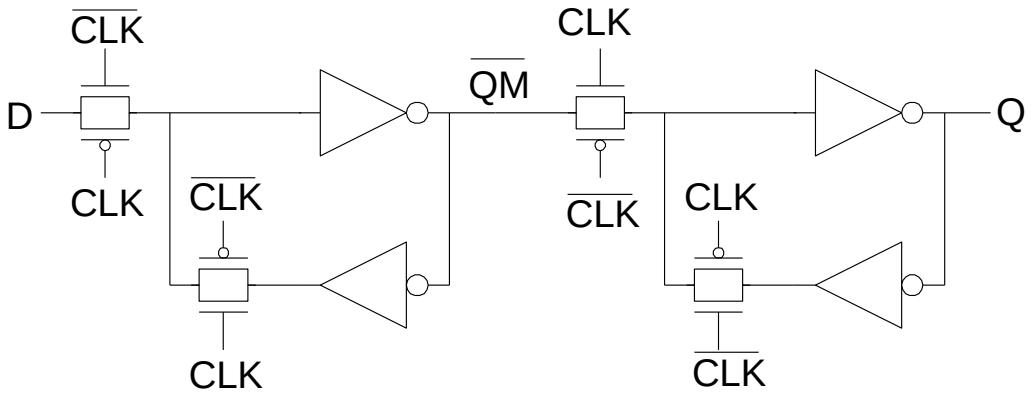




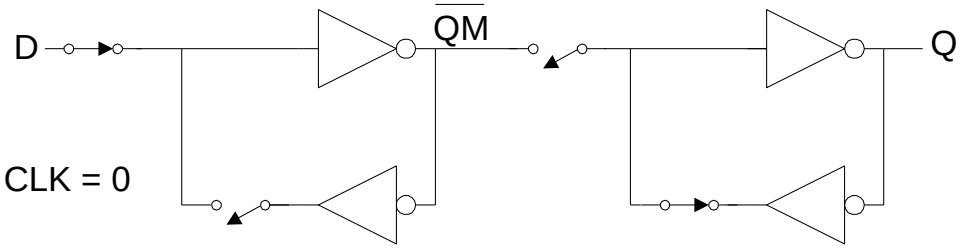
D Flip-flop Design

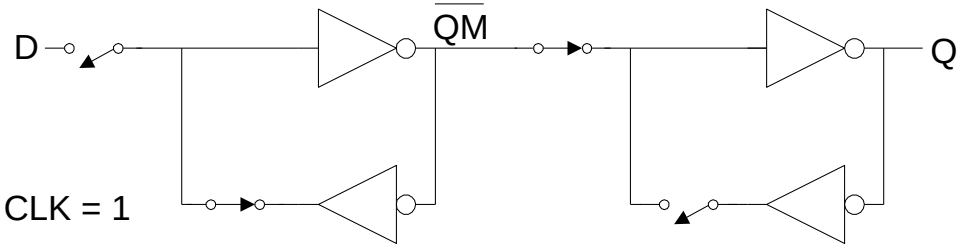
- Built from master and slave D latches

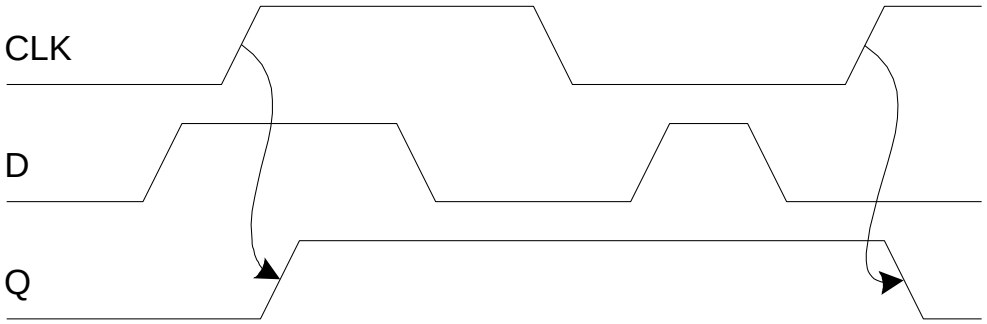




D Flip-flop Operation

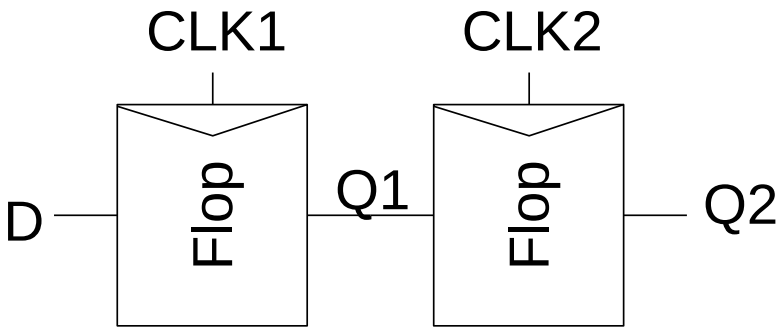


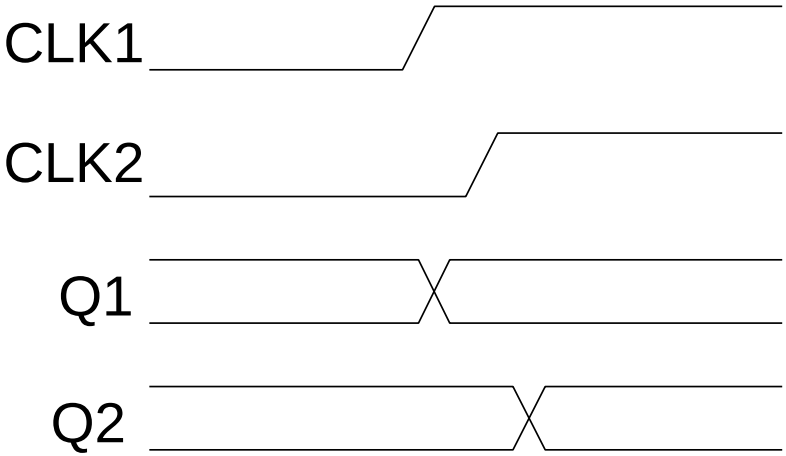




Race Condition

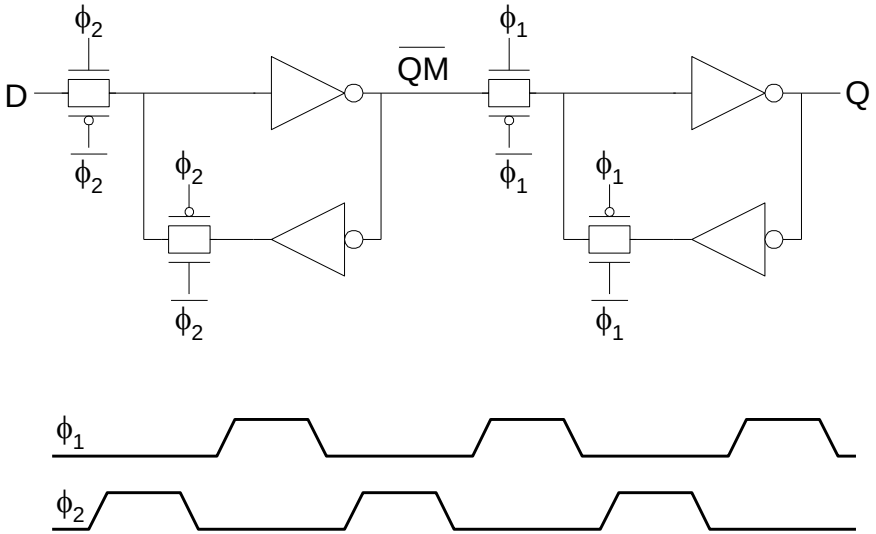
- Back-to-back flops can malfunction from clock skew
  - Second flip-flop fires late
  - Sees first flip-flop change and captures its result
  - Called *hold-time failure* or *race condition*





Nonoverlapping Clocks

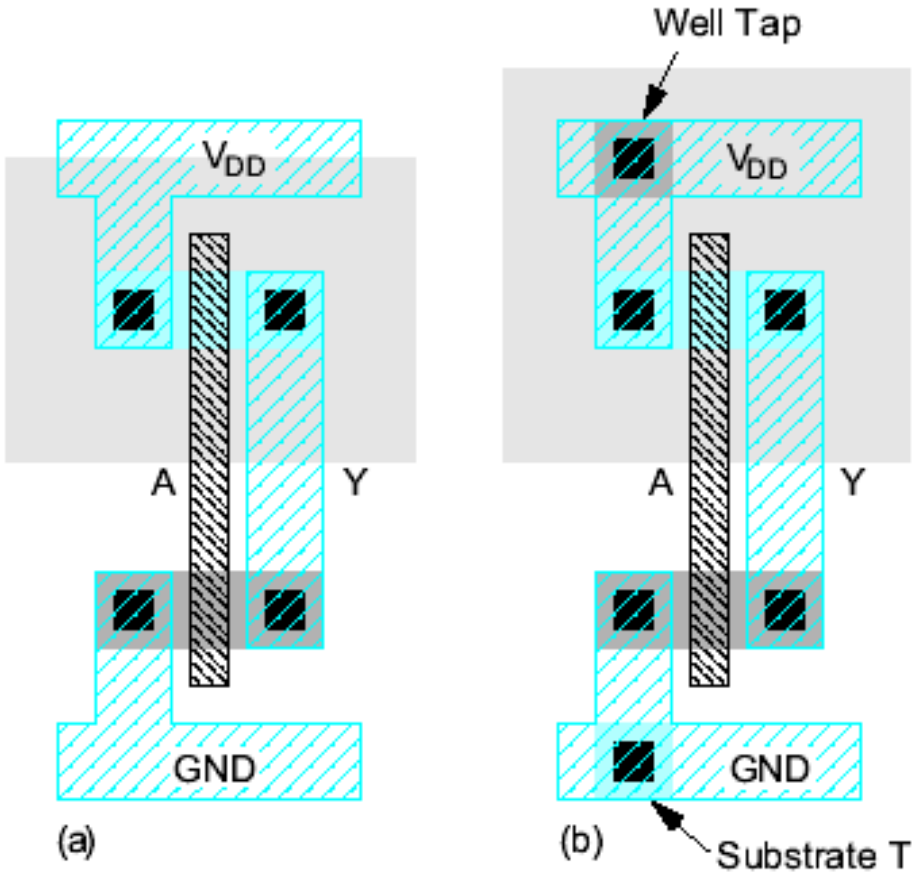
- Nonoverlapping clocks can prevent races
  - As long as nonoverlap exceeds clock skew
- We will use them in this class for safe design
  - Industry manages skew more carefully instead



Gate Layout

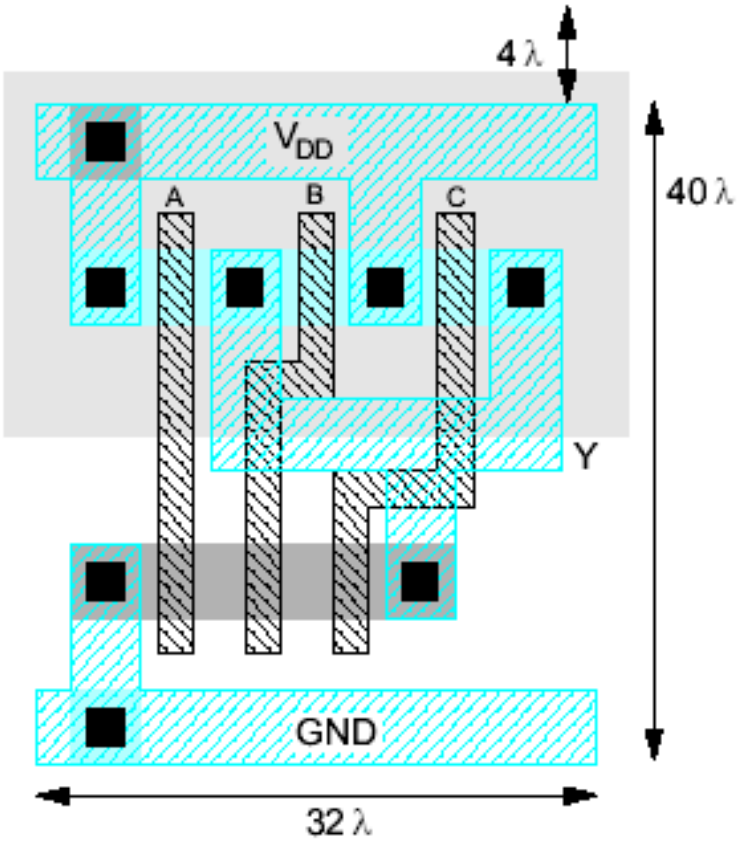
- Layout can be very time consuming
  - Design gates to fit together nicely
  - Build a library of standard cells
- Standard cell design methodology
  - V_{DD} and GND should abut (standard height)
  - Adjacent gates should satisfy design rules
  - nMOS at bottom and pMOS at top
  - All gates include well and substrate contacts

Example: Inverter



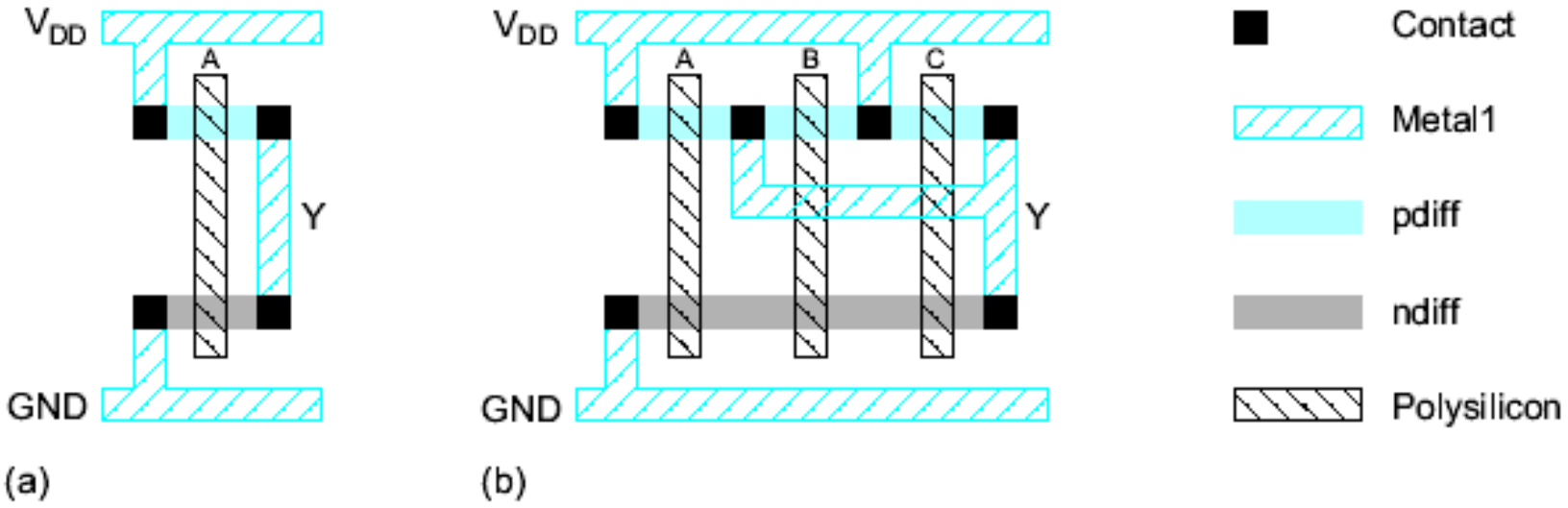
Example: NAND3

- Horizontal N-diffusion and p-diffusion strips
- Vertical polysilicon gates
- Metal1 V_{DD} rail at top
- Metal1 GND rail at bottom
- 32 λ by 40 λ



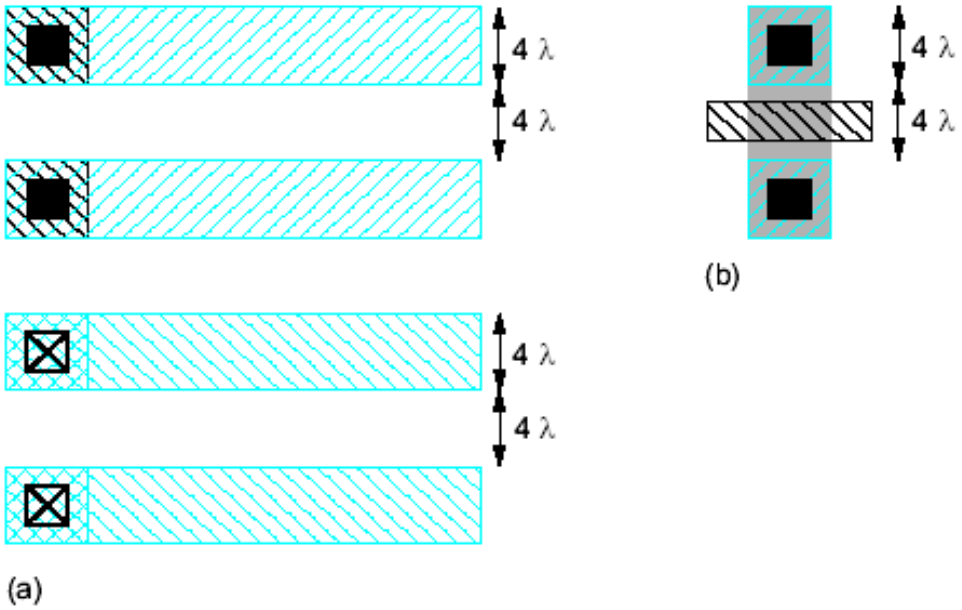
Stick Diagrams

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



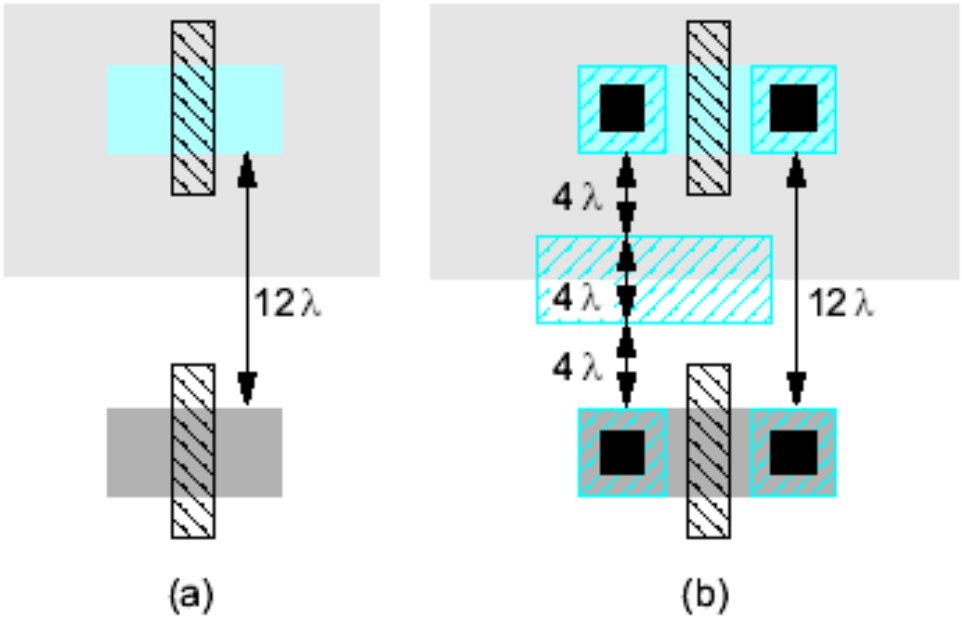
Wiring Tracks

- A *wiring track* is the space required for a wire
  - 4 λ width, 4 λ spacing from neighbor = 8 λ pitch
- Transistors also consume one wiring track



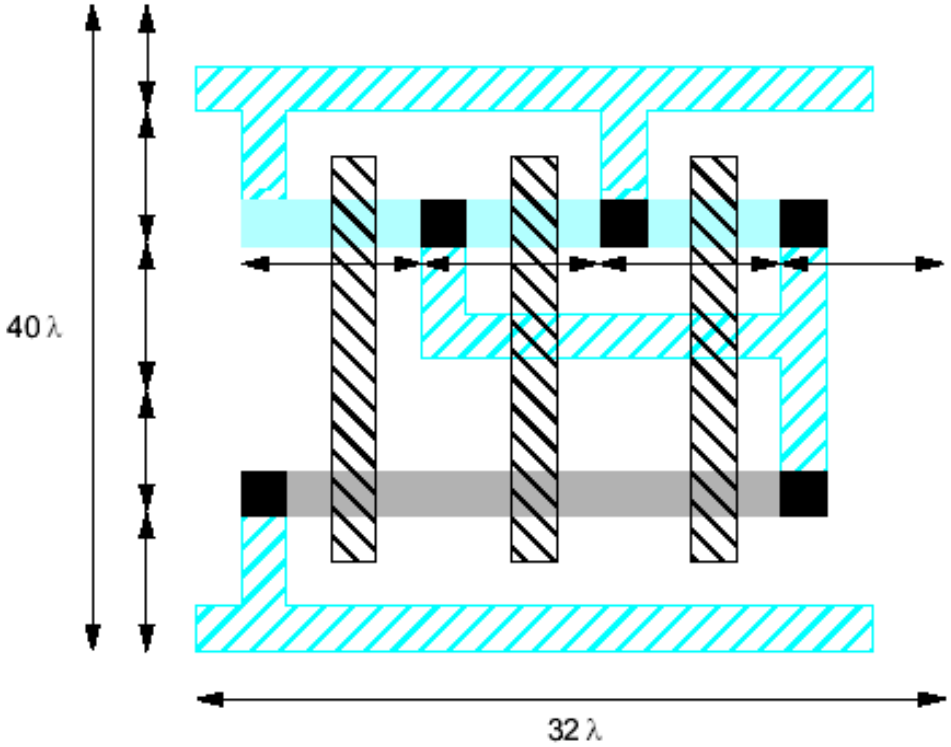
Well spacing

- Wells must surround transistors by 6 λ
  - Implies 12 λ between opposite transistor flavors
  - Leaves room for one wire track



Area Estimation

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

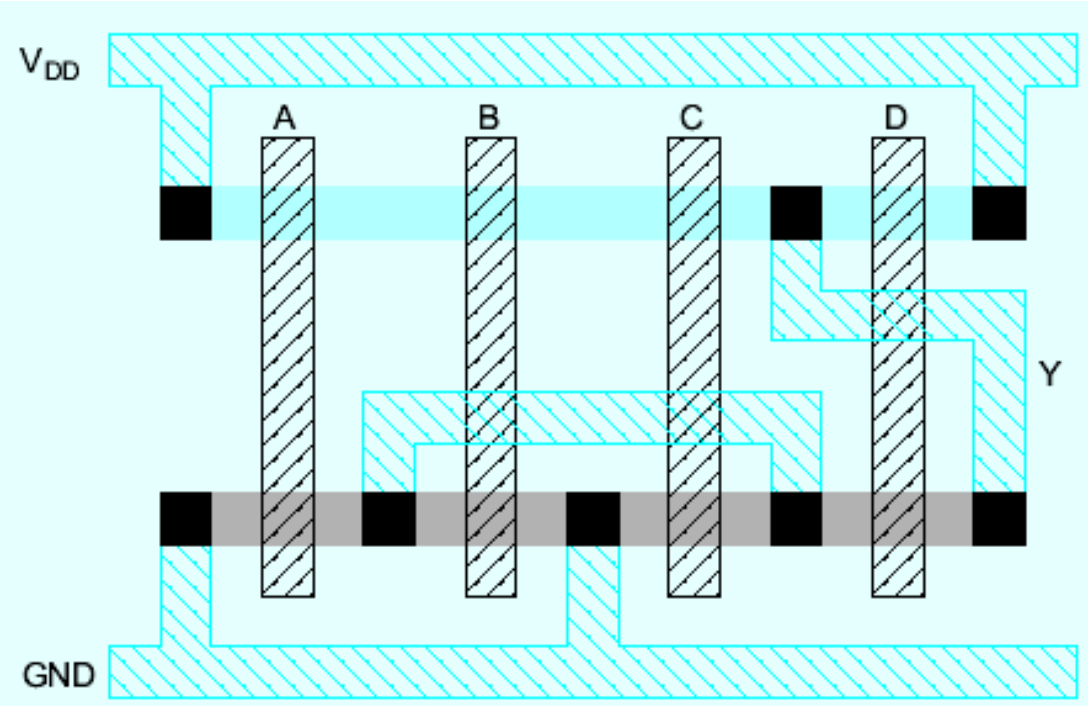


Example: O3AI

- Sketch a stick diagram for O3AI and estimate area
  - Y = ((A+B+C).D)'

Example: O3AI

- Sketch a stick diagram for O3AI and estimate area
  - Y = ((A+B+C).D)'



Example: O3AI

- Sketch a stick diagram for O3AI and estimate area
  - Y = ((A+B+C).D)'