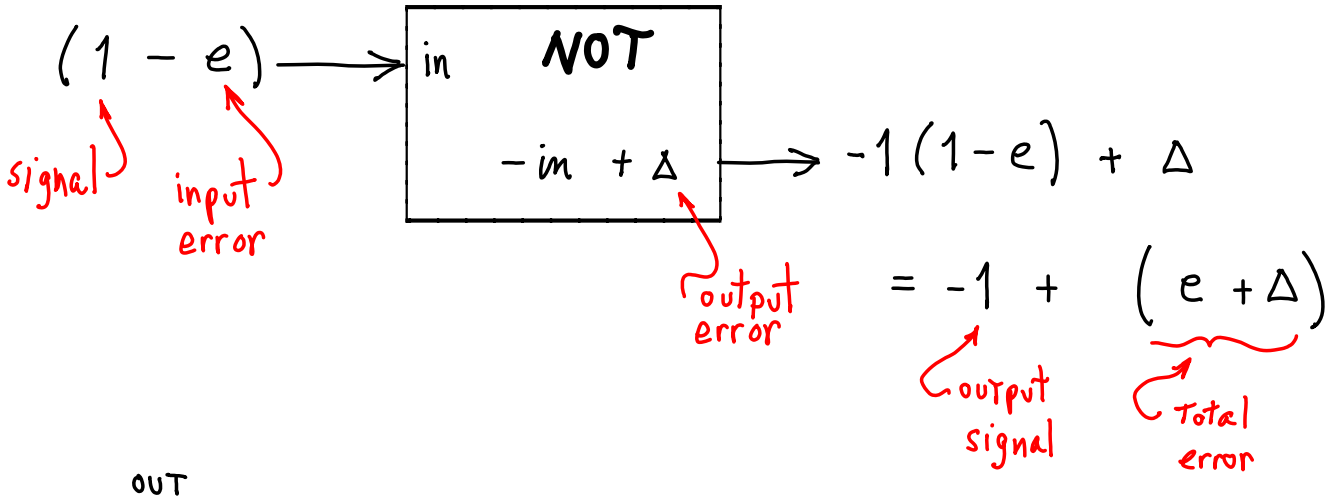
Lec-5a-devices

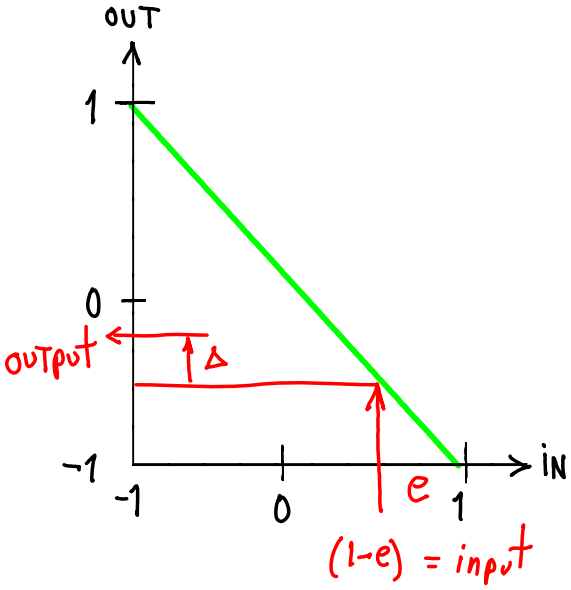
We want to build a Universal Turing Machine (rather, its equivalent, a Computer).
Such a machine consists of a Finite State Machine (FSM) and a Read/Write tape.
A FSM has two physical components, a state-element and a functional element.
The state-element defines what state the FSM is in currently.
The functional element determines what state is next and what symbol to output.
To build these two elements physically, all we need are basic switching devices.

All devices have errors.
We need Signal-Restoring, Non-Linear Logic.
Ohm's Law devices are LINEAR.

Linear NOT(x) device: NOT(x) = -1(x) + error

NOT(1) = -1 + error
NOT(-1) = 1 + error





Input error propagates to output.
Error grows.

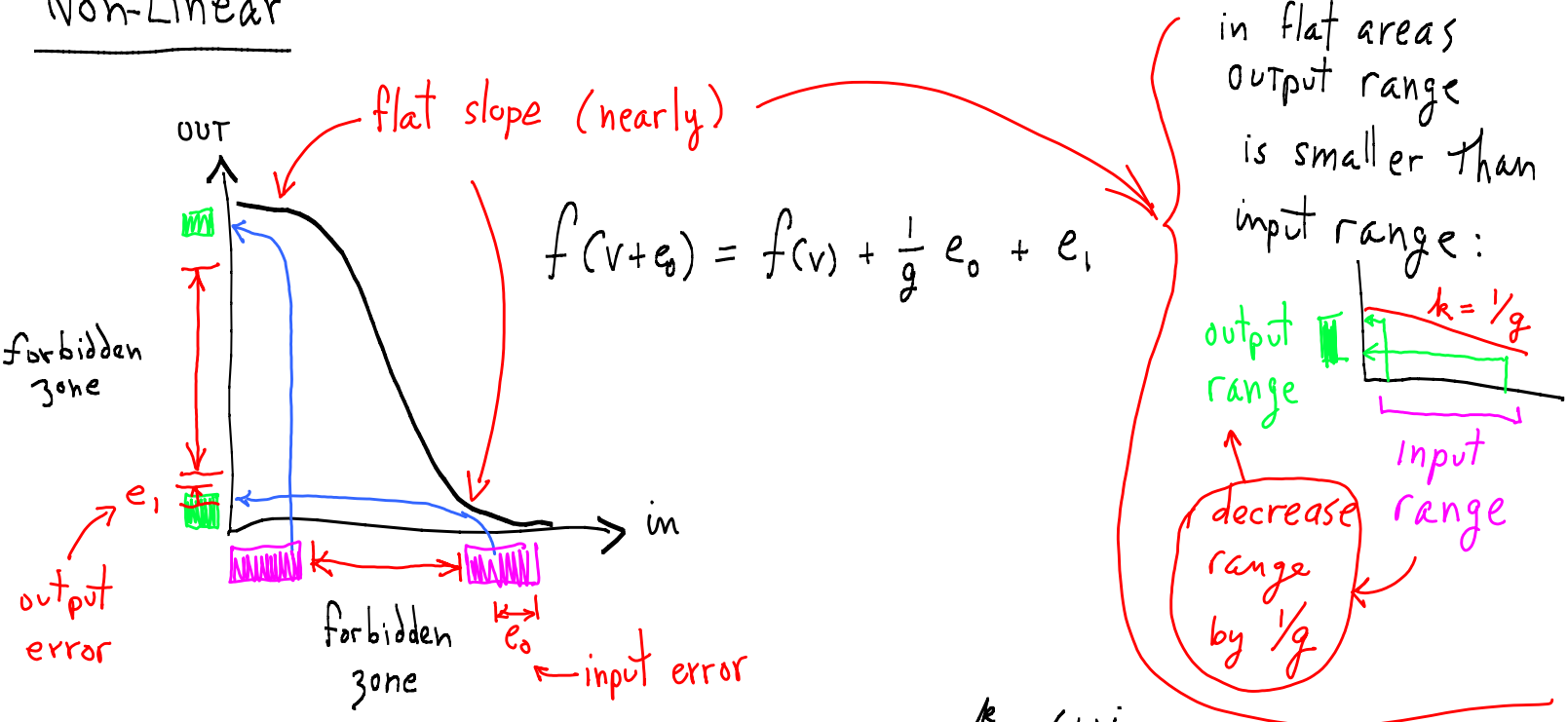
Output is input to another device,
error gets worse each step.

NOTE:
error is + or - randomly
===> random walk
===> after n devices, variance is sqrt(n)
===> error grows w/o bound

Take random step (either in the -1 or +1 direction).
How far from 0 can you expect to be after k steps? About k^1/2 away.
With probability 0 you will be at 0, and error gets unboundedly large.

We must Reduce error at each stage ==> exponentially decreasing effect in later stages.

Non-Linear



after k stages: $= 1 + \sum_{i=0}^{k} \left(\frac{1}{g}\right)^i e_i$

Converges!

If g is large enough (flat areas of curve are flat enough),

and

if output error size is not too big,

Then

output after k stages never hits FORBIDDEN ZONE.

So, if we plan to have a circuit with long device chains, we must have non-linear devices w/ suitable response curves.

Do we plan to have long chains? YES:

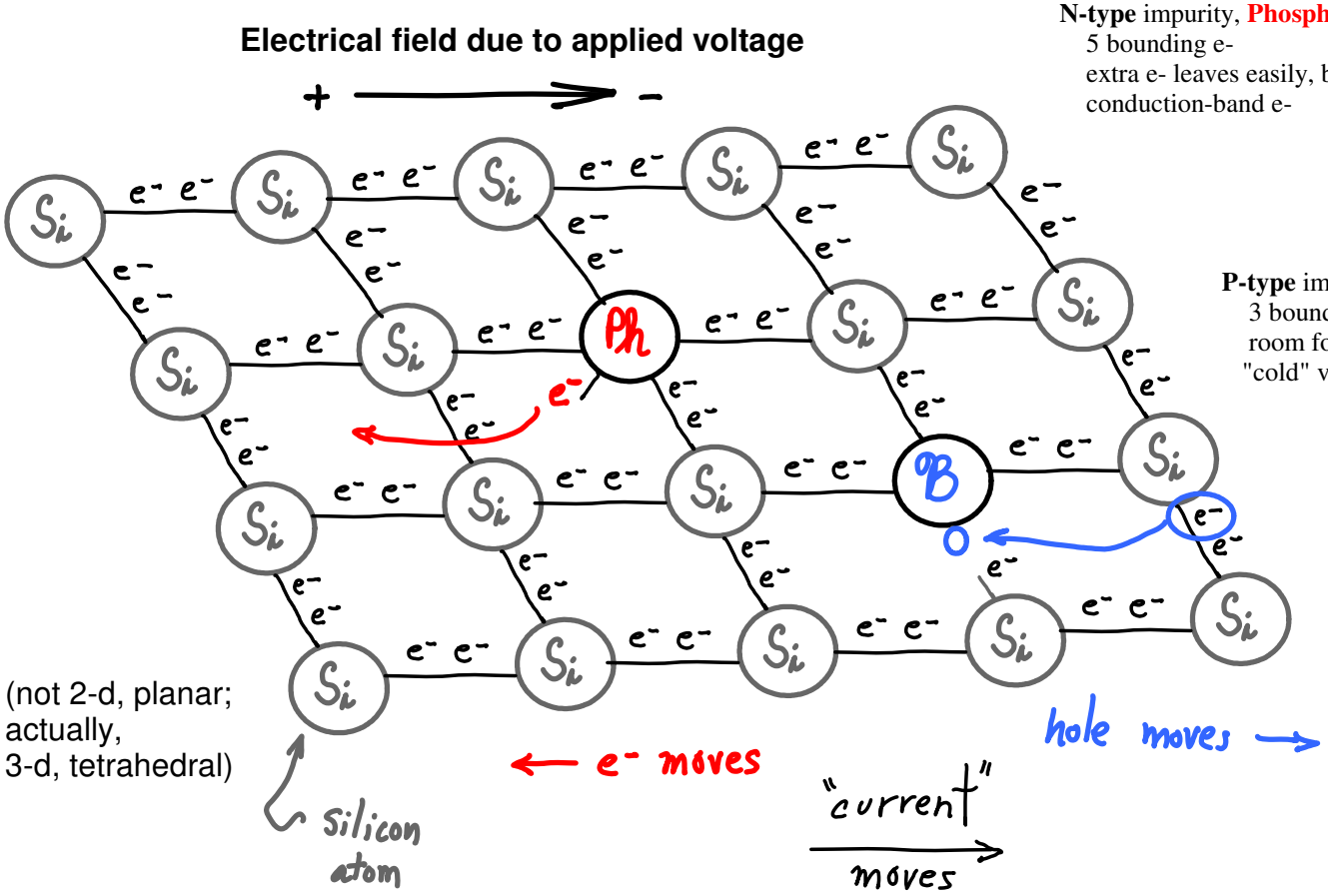
(1) feedback in system,
(2) chained data operations: D1 ==> D2 ==> D3 ==> D4 ...
(3) 1 Billion devices per cpu

Solid State devices – Semiconductors

Silicon crystals
Si has 4 bounding e-
bounds 4 Si neighbors
crystal is diamond-like

N-type impurity, **Phosphorous**:
5 bounding e-
extra e- leaves easily, becomes "hot" conduction-band e-

P-type impurity, **Boron**:
3 bounding e-
room for bounding e-
"cold" valence e- arrives,



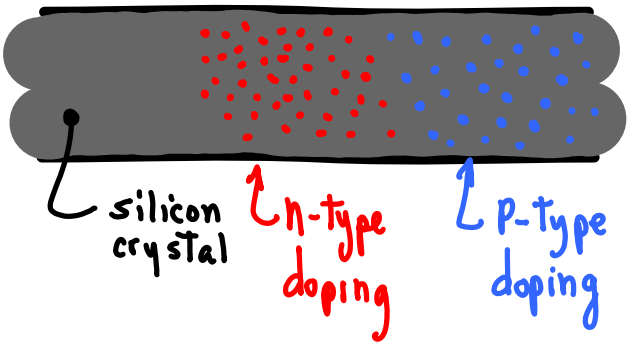
(not 2-d, planar;
actually,
3-d, tetrahedral)

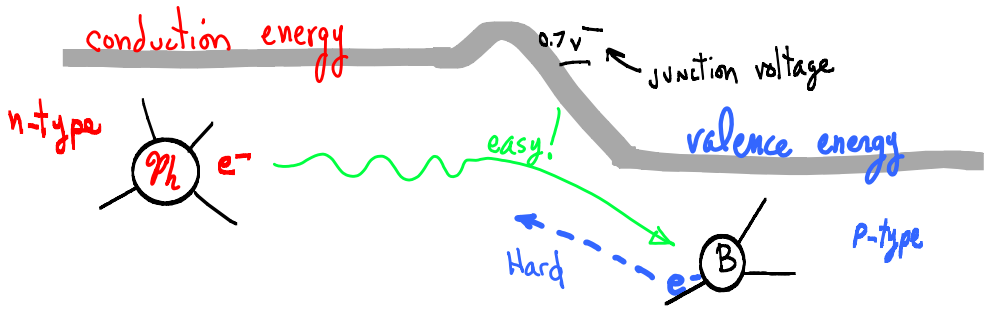
Holes and e- move in opposite directions,
"current" direction is same a hole movement.

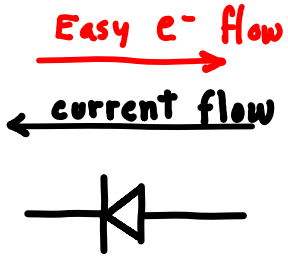
e- FLOW (opposite of "current" flow):

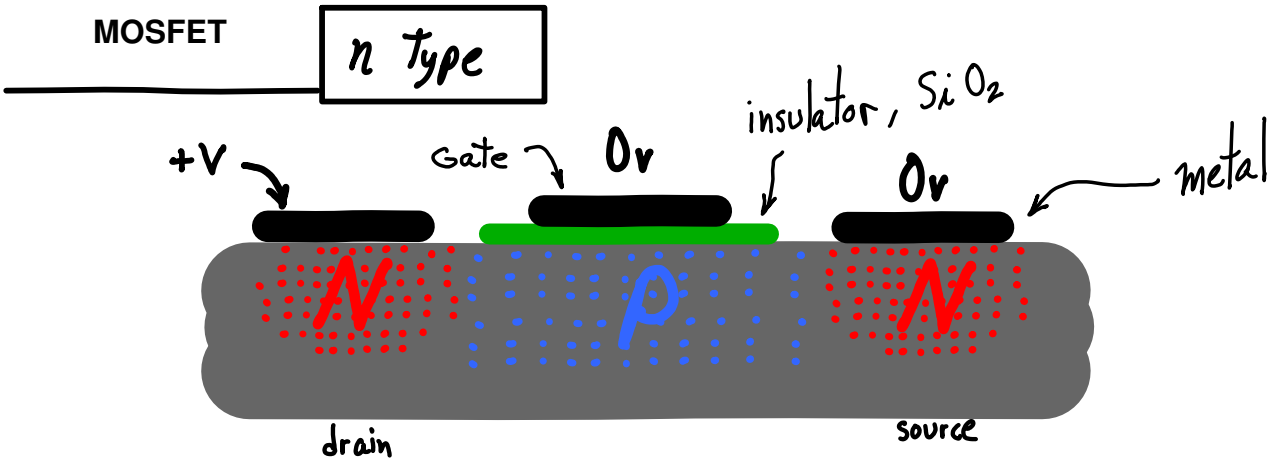
n-type ===> **p-type**, EASY
"hot" conduction energy e- are free to move.

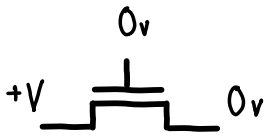
n-type <=== **p-type**, HARD
"cold" bonding energy e- need energy to become "hot" e-.









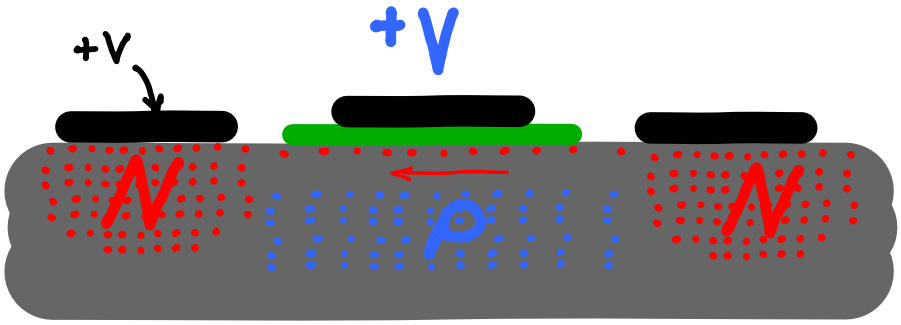


NOT-CONDUCTING
Vgate = 0
R-drain-source = BIG

Vgate = 0, holes populate channel.

Source N-well e- drop into valence band in channel.

+V at drain cannot pull valence-band e- from P-type to N-type.

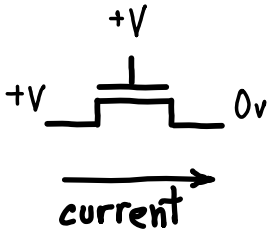


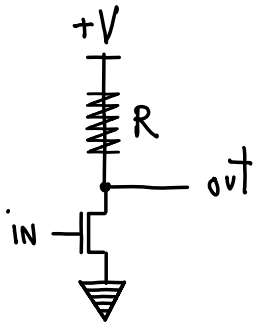
CONDUCTING
Vgate = +V
R-drain-to-source ~ 0

+V on gate drives holes away from P-type channel.

Conduction-band e- move from source N-type well.

+V on drain pulls conduction-band e- off.

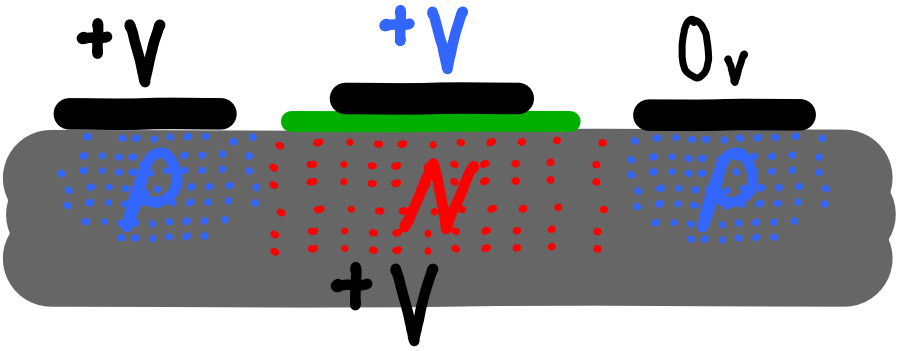


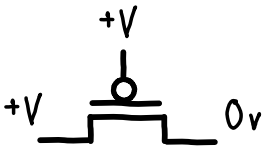


n-MOS inverter

MOSFET

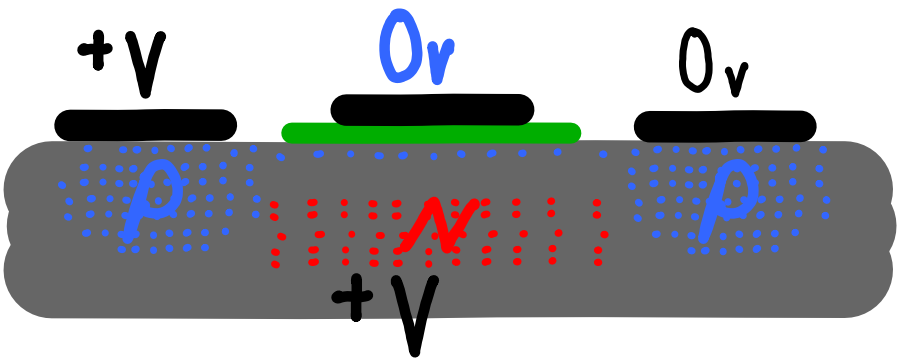
p type

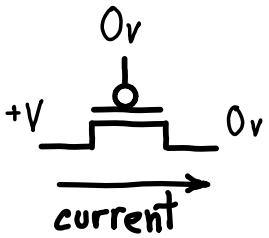




NOT-CONDUCTING
Vgate = +V
R-drain-source = BIG

Cannot pull valence-band e- from P-type to N-type.





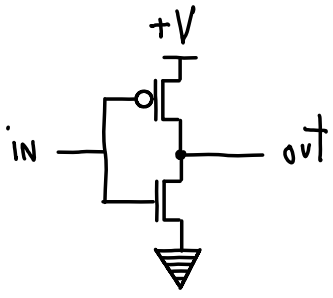
CONDUCTING
Vgate = 0
Resistance-drain-to-source ~ 0

+V on substrate drives e- away from N-type channel.

Holes move from N-channel to source P-well.

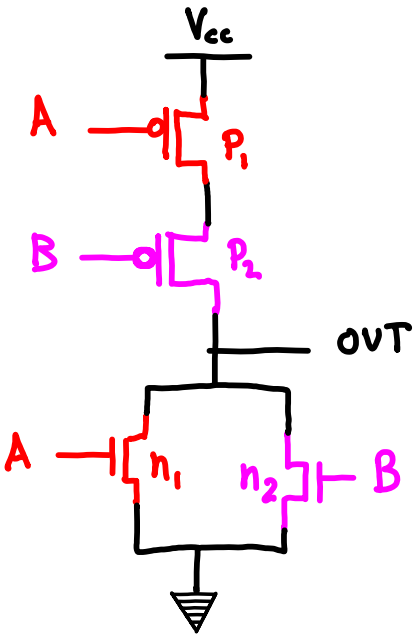
+V on drain pulls e- off.

CMOS inverter



in	out
0	+V
+V	0

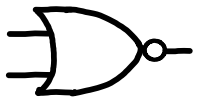
Basic Logic Gates



A	B	P_1	P_2	n_1	n_2	OUT
0	0	ϕ	ϕ	⊗	⊗	Vcc
0	1	ϕ	⊗	⊗	ϕ	0v
1	0	⊗	ϕ	ϕ	⊗	0v
1	1	⊗	⊗	ϕ	ϕ	0v

NOR

A	B	out
0	0	1
0	1	0
1	0	0
1	1	0

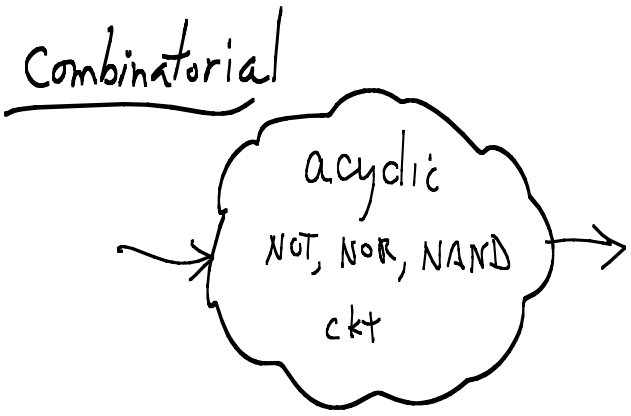


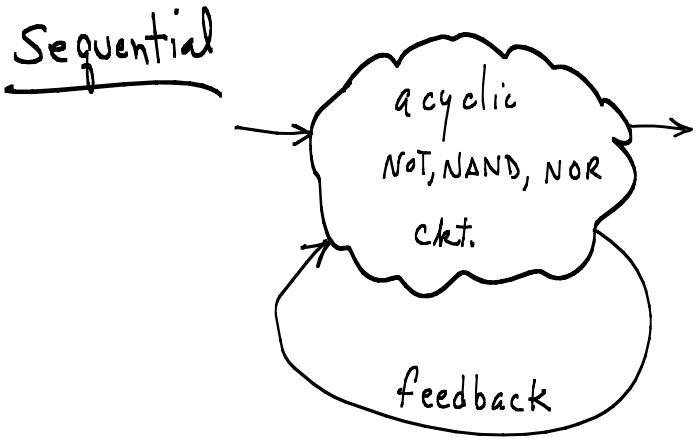
Logic CIRCUITS

Two kinds of logic circuits:
(1) w/ feedback, SEQUENTIAL: can hold STATE
(2) w/o feedback, COMBINATORIAL: realize FUNCTIONS

Basic logic gates: NOT, 2-input NOR, 2-input NAND.
That's all we need for both sequential and combinatorial circuits.

(NAND alone is sufficient, also NOR alone is sufficient.)





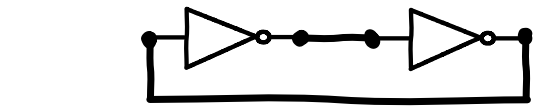
State = feedback

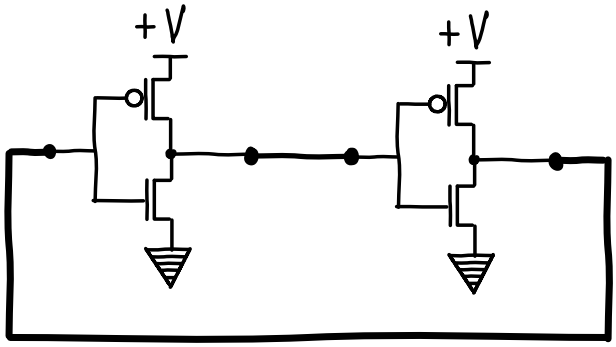
A circuit with feedback can "remember".
It is in a "state".

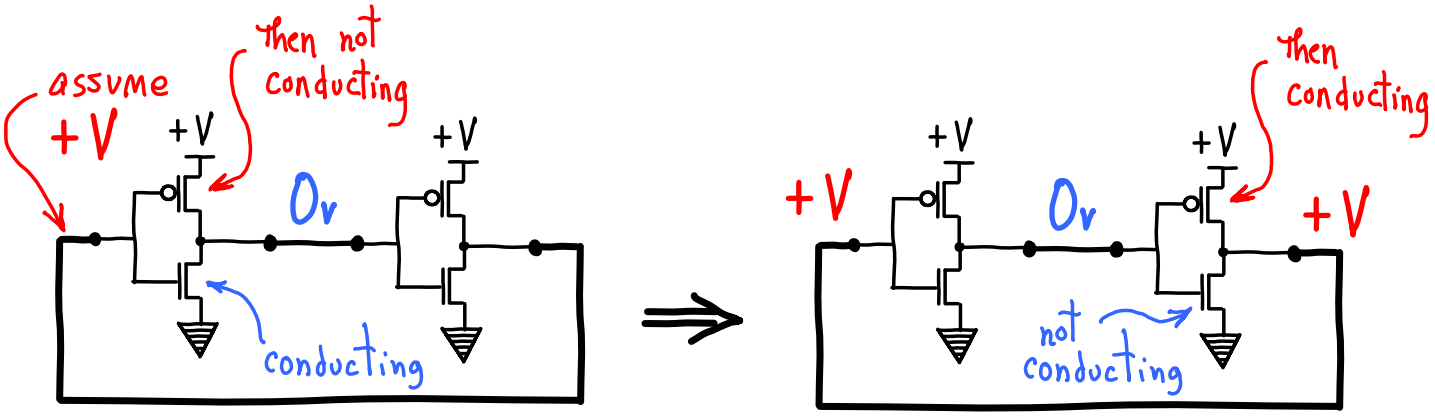
What state is it in?
That depends on its past state.

We cannot tell by looking.

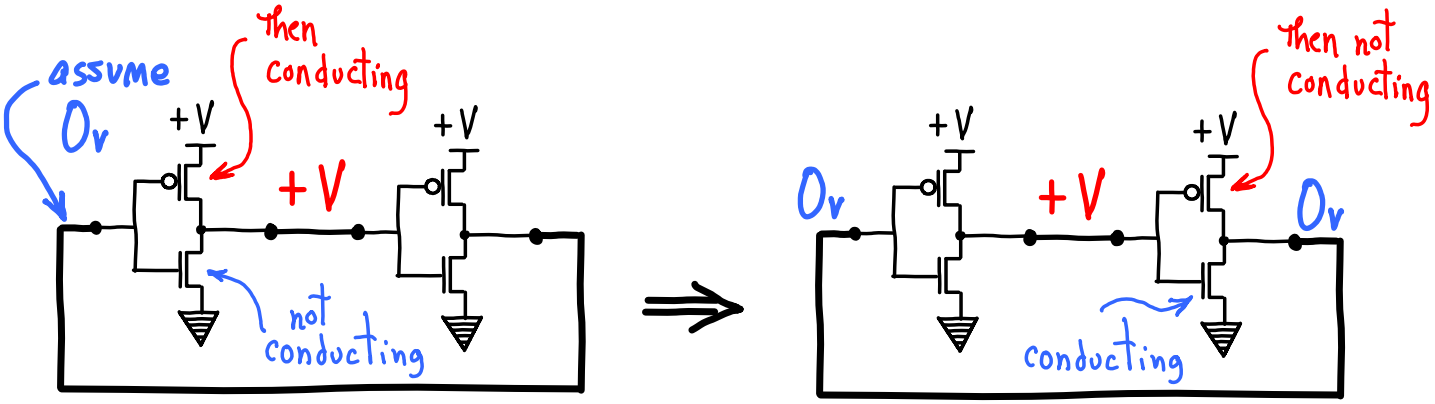
Let's analyze a circuit by assuming its state.
Then see if our assumption is consistent.







Consistent



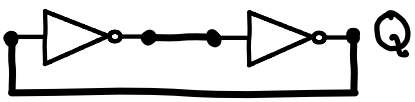
Consistent

Circuit has two stable states: aka, meta-stable, bi-stable.

What state when power is first turned on? Unknown, random.

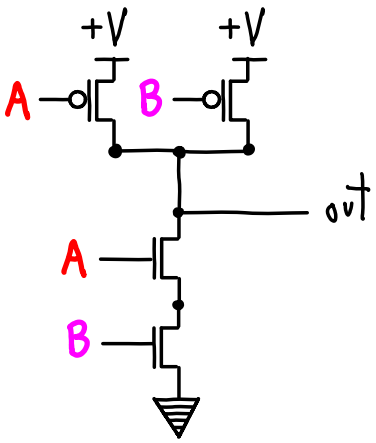
Can we set the state, using voltage inputs? NO. Is device useless!?!

Basic Sequential elements



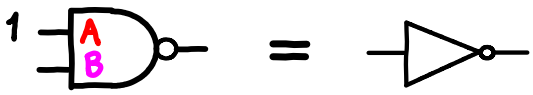
NOT-NOT loop stores data.
But also we want control.

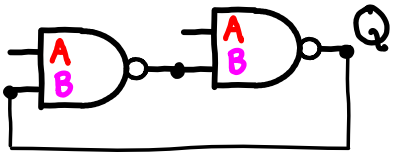
NAND



A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

when A = 1
Q = NOT(B)



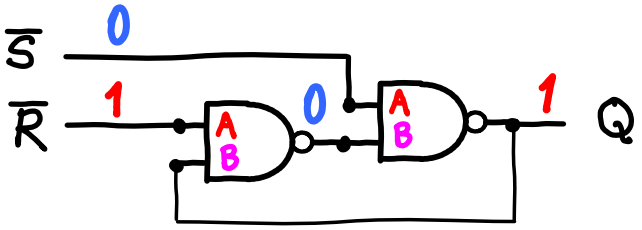


CIRCUITS W/ STATE

NOT-NOT circuit is stable in either of two states: **BISTABLE element**.

NAND-NAND circuit **with both A = 1**: **same as a NOT-NOT** circuit.

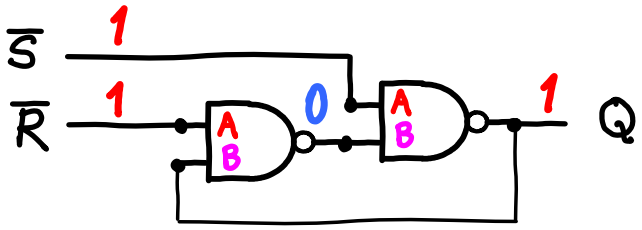
Is there something new here? YES: control.



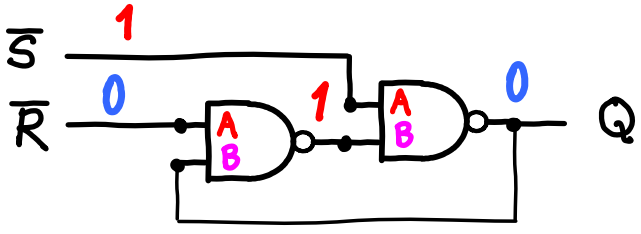
A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

A = 0: Q = 1

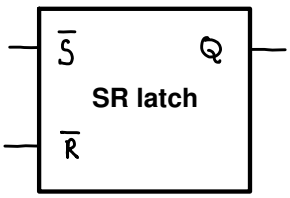
Pulsing a 0 sets Q = 1.



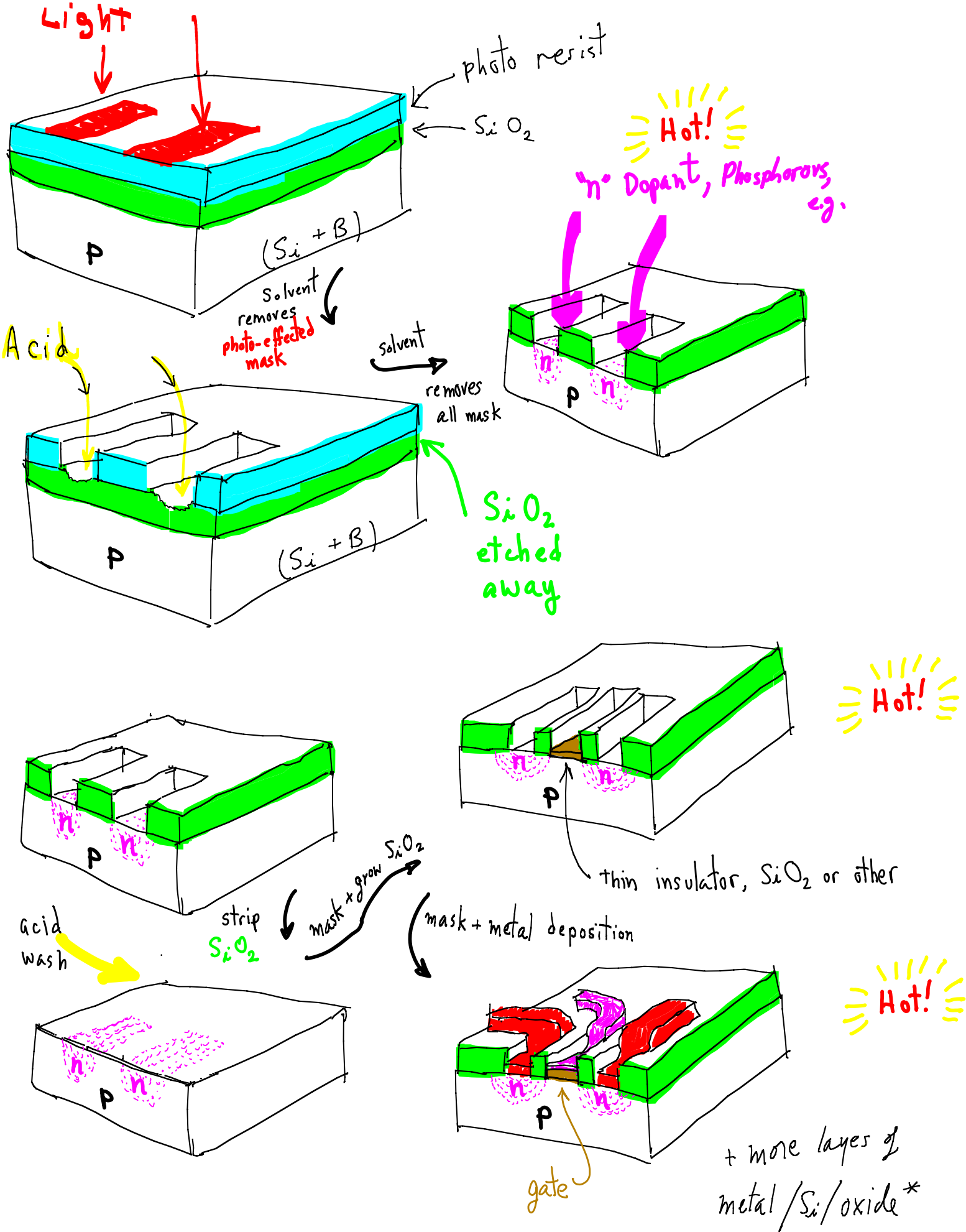
stable

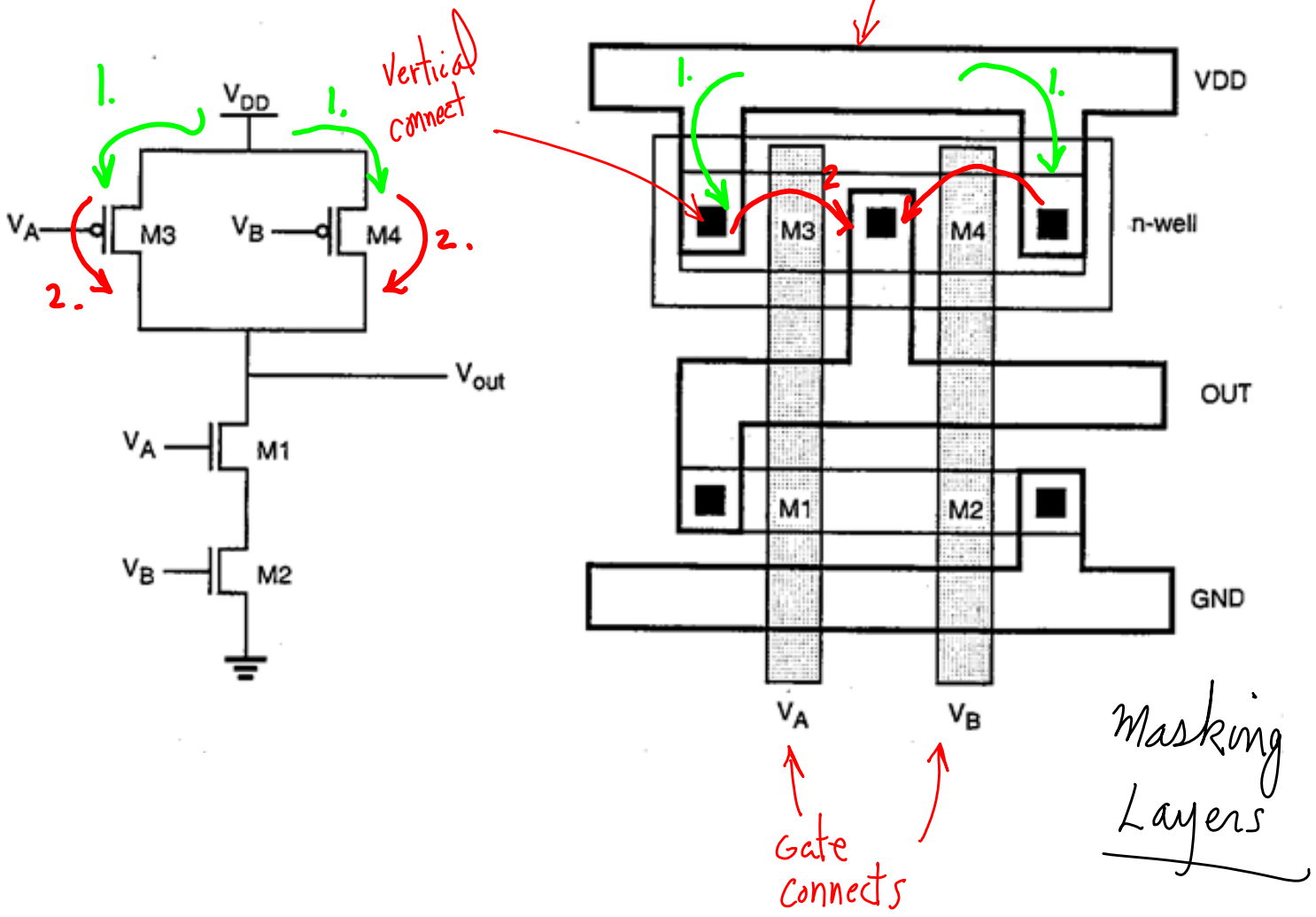


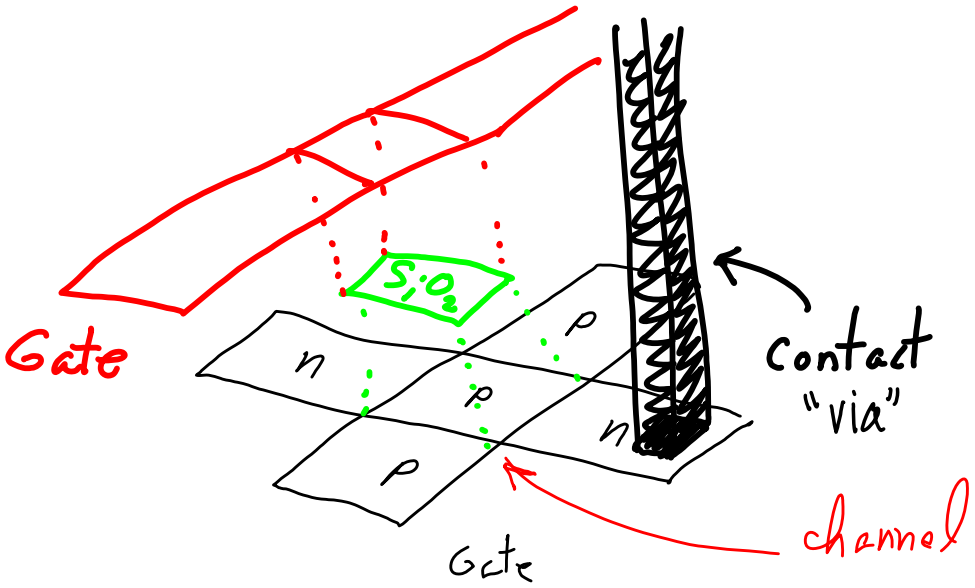
S̄	R̄	
1	1	stable
0	1	set Q = 1
1	0	reset Q = 0



Silicon circuit fabrication: build a transistor







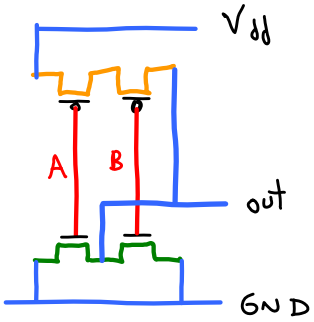
Chip is built up by layers.

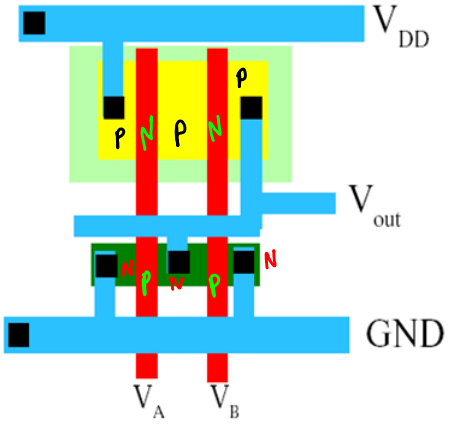
Each layer etched using a mask.

Bottom layer is n-type and p-type.

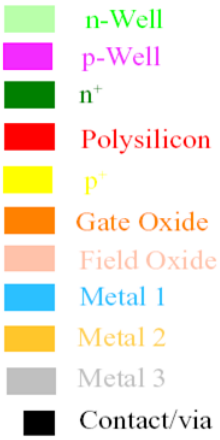
Next layer has gates to form transistors.

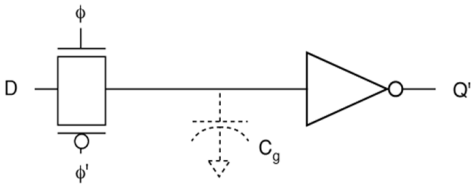
Upper layers are electrical connections.



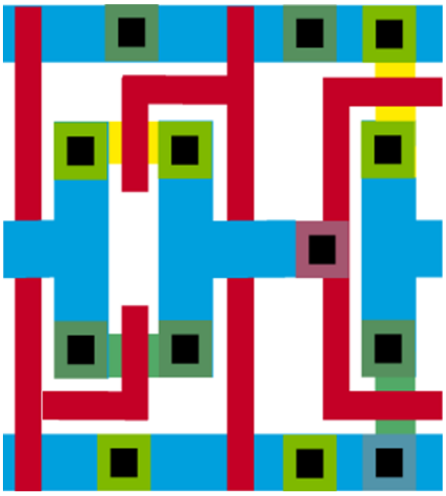


NOR Gate

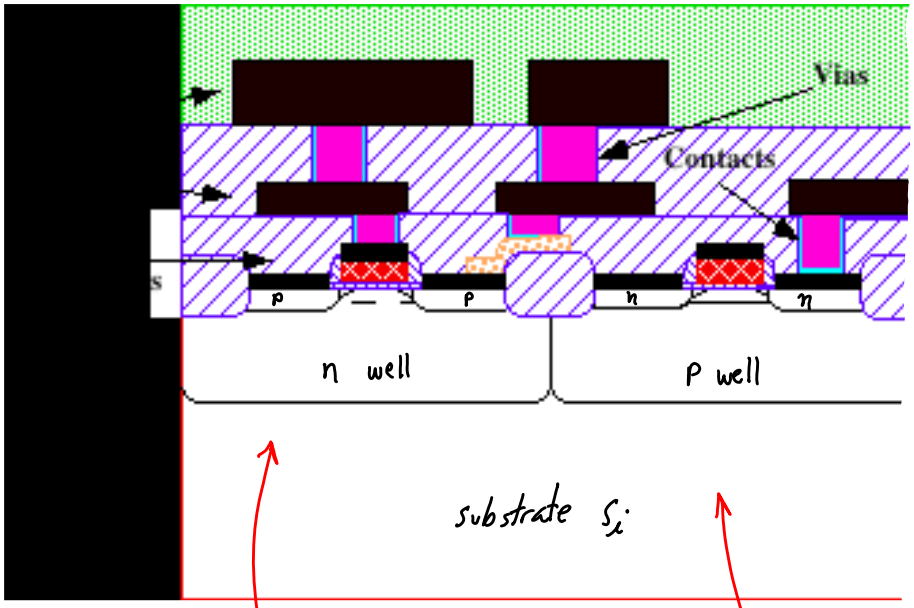




dynamic latch



Charge moves through n-type or p-type (both open at same time). Inverter gate set to 0 or 1. Transistors turned off, charge held on gate. Has to be recharged as leakage drains charge.



insulator

conductor, inter-connect

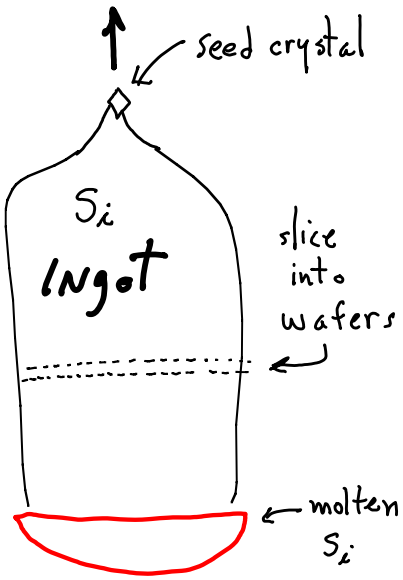
P-type transistor

n-type transistor

Lithography Line Sizes

Masks define lines,
lines define chip features

Smaller lines ===> More features



Silicon (Si) crystal formed:

-- seed crystal
-- molten Si bath
-- slow rotation, pulling

===> Si ingot

Si Wafer formed:

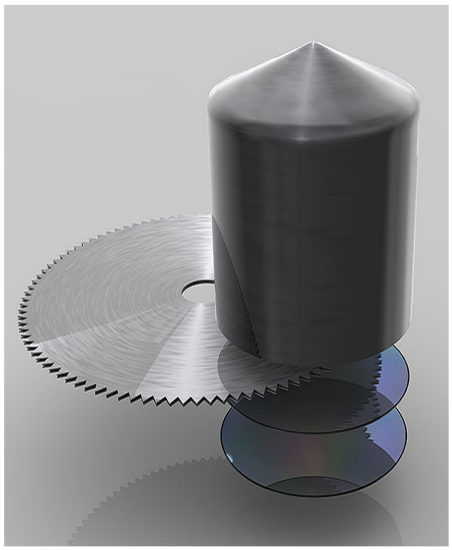
-- Cut ingot into thin slices

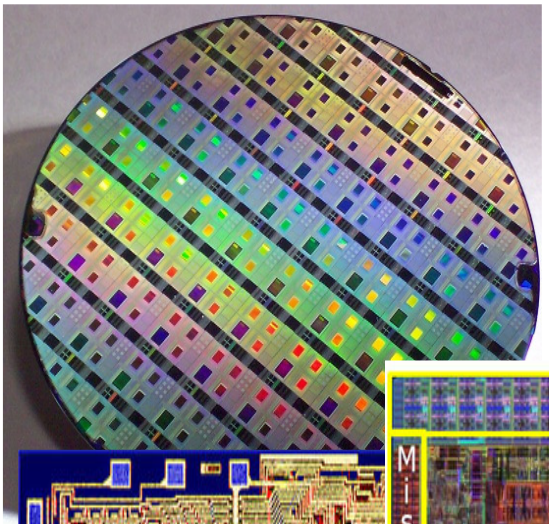
Circuits printed:

Masking/etching per layer.

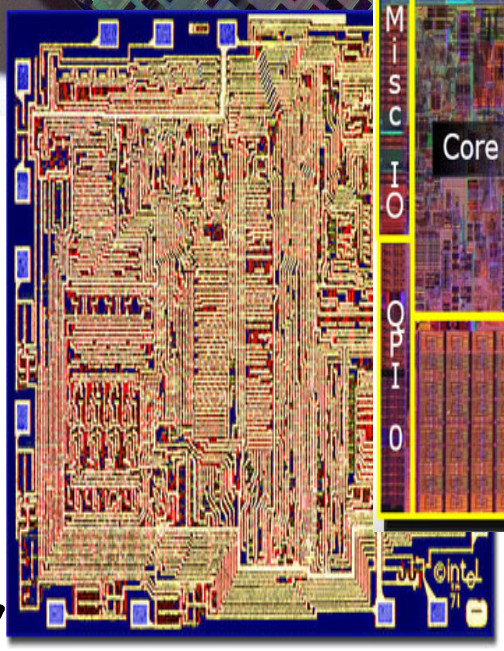
Die/chip removed:

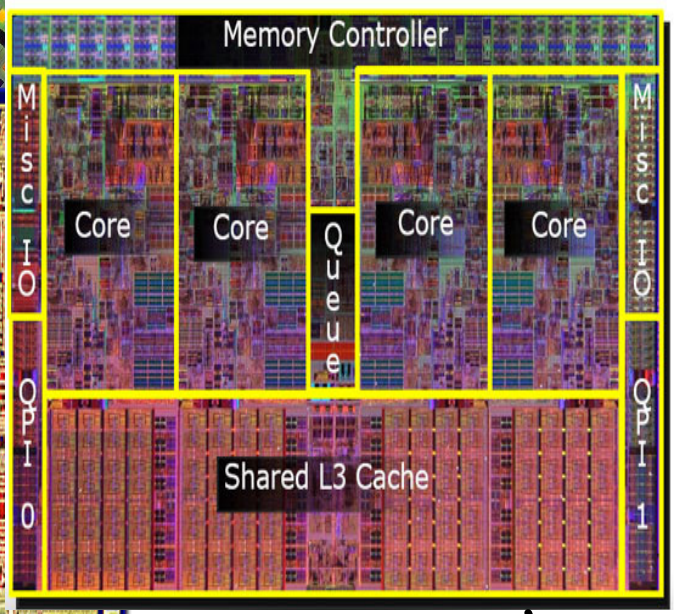
Each die is a complete chip circuit, cut from wafer.



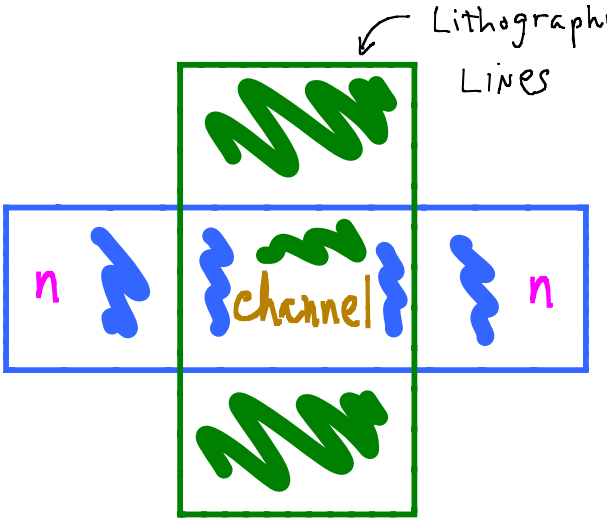


intel 4004





intel i7

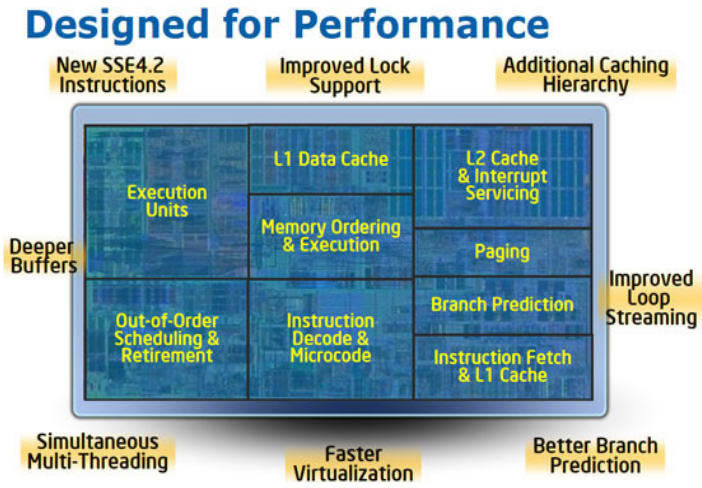


4004 10 μm 2.5k transistors

~ 3/4 MHz @ 8 cycles/instruction

i7, 10nm, ~1G transistors

~ 3 GHz @ 8 cycles/instruction



Better Process
more expensive equipment
+$

lines shrink

10nm

faster switching
→ faster clock

more transistors
→ more function
→ more usage
→ more sales
→ same price

smaller features → More defects, lower die yield, but smaller die → Moore's Law

2x #Tr / 2 yr

end of the road?

$(10\mu m \rightarrow 10\,nm)^2 : (X\ 10^3)^2 = X\ 10^6$