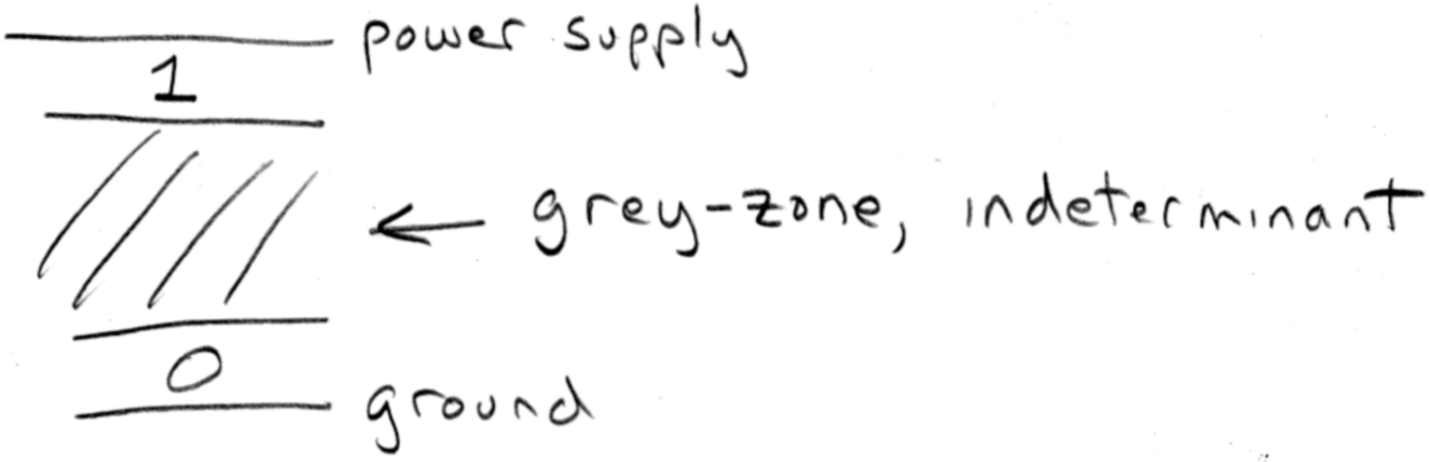
Digital Circuits

Only 2 voltages allowed

near power supply = "1"

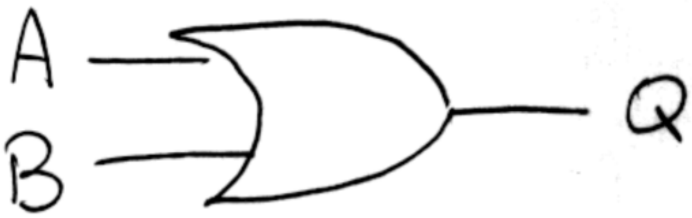
near ground = "0"



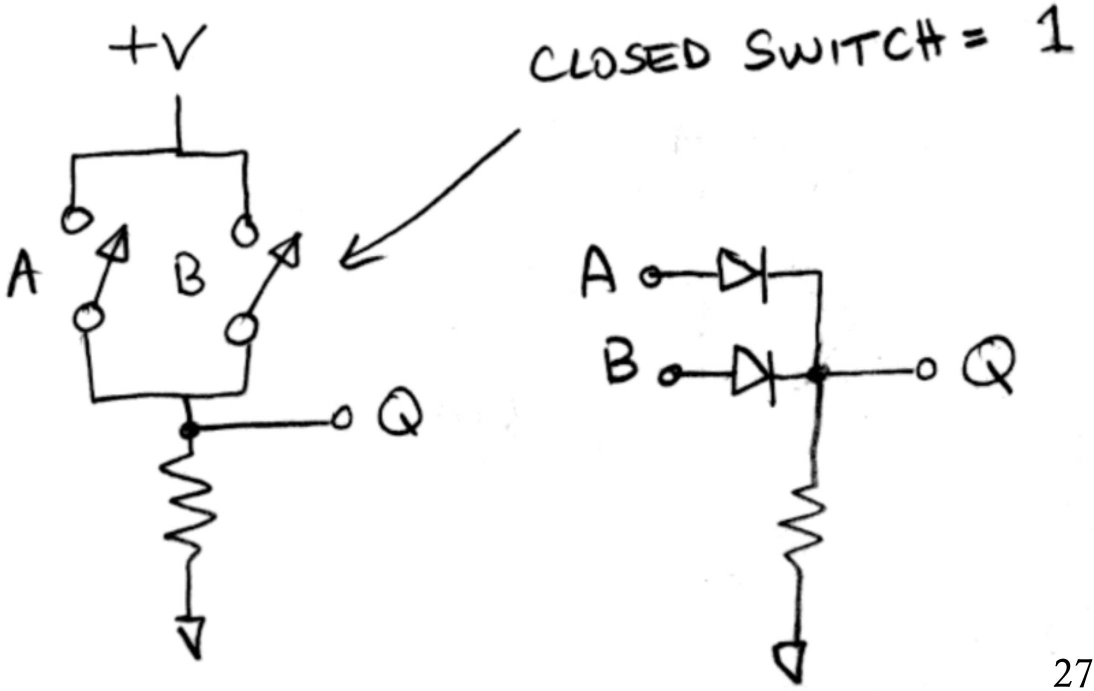
"slam" the voltage into the ground or powersupply, Don't worry about reaching an equilibrium. Fast!!

Logic Gates OR

$$\overbrace{Q = A + B}^{\text{BOOLEAN ALGEBRA}}$$

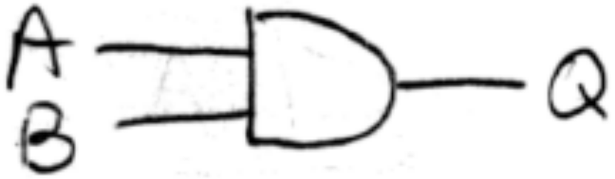


A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

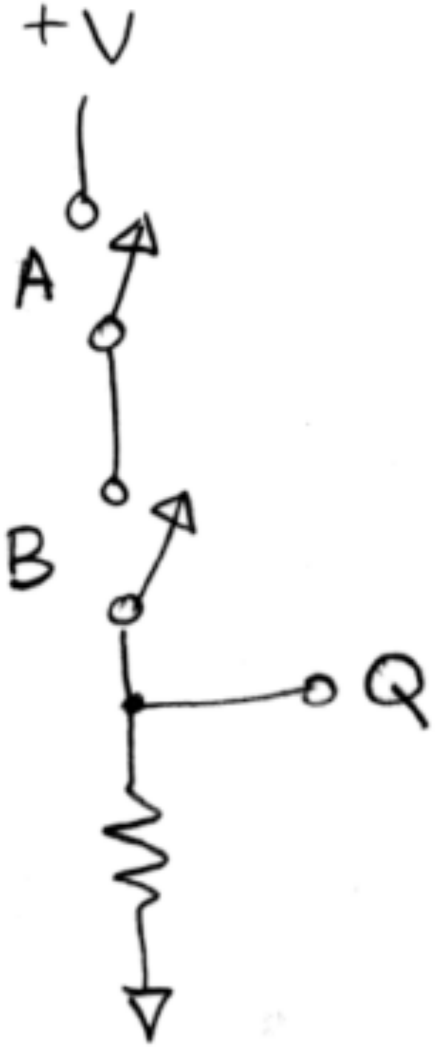

CLOSED SWITCH = 1

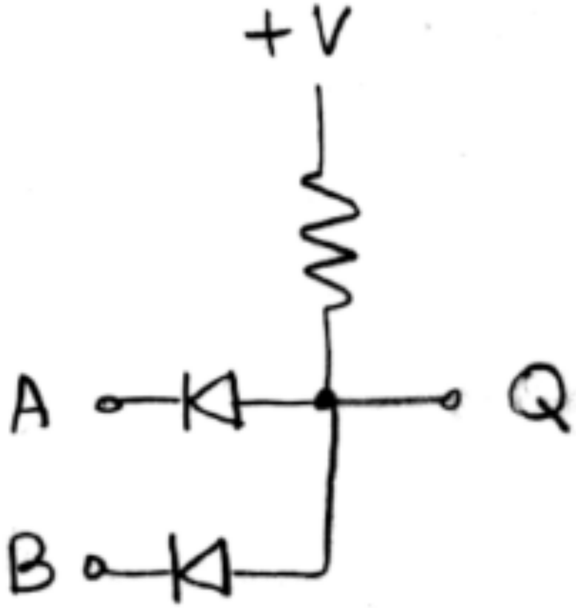
AND

$$Q = AB$$



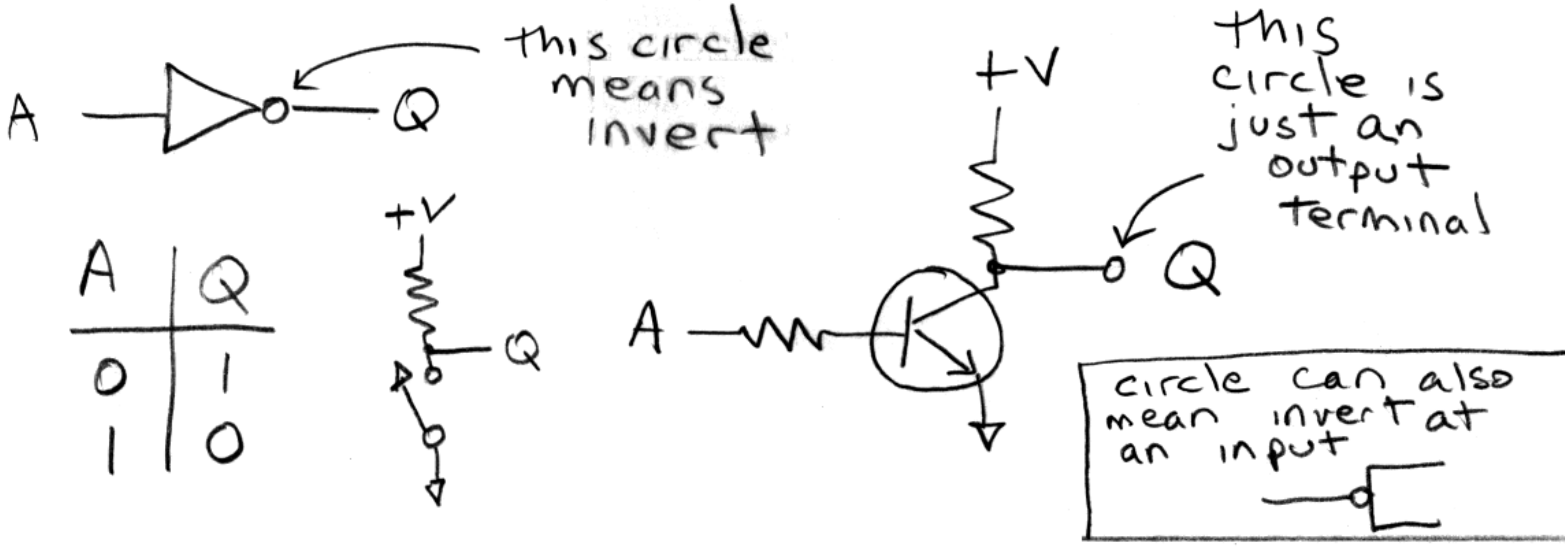
A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1





NOT (INVERT)

$$Q = \overline{A}$$



this circle means invert

this circle is just an output terminal

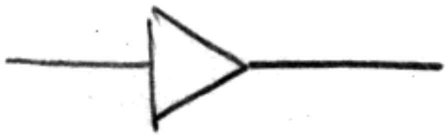
circle can also mean invert at an input

A	Q
0	1
1	0

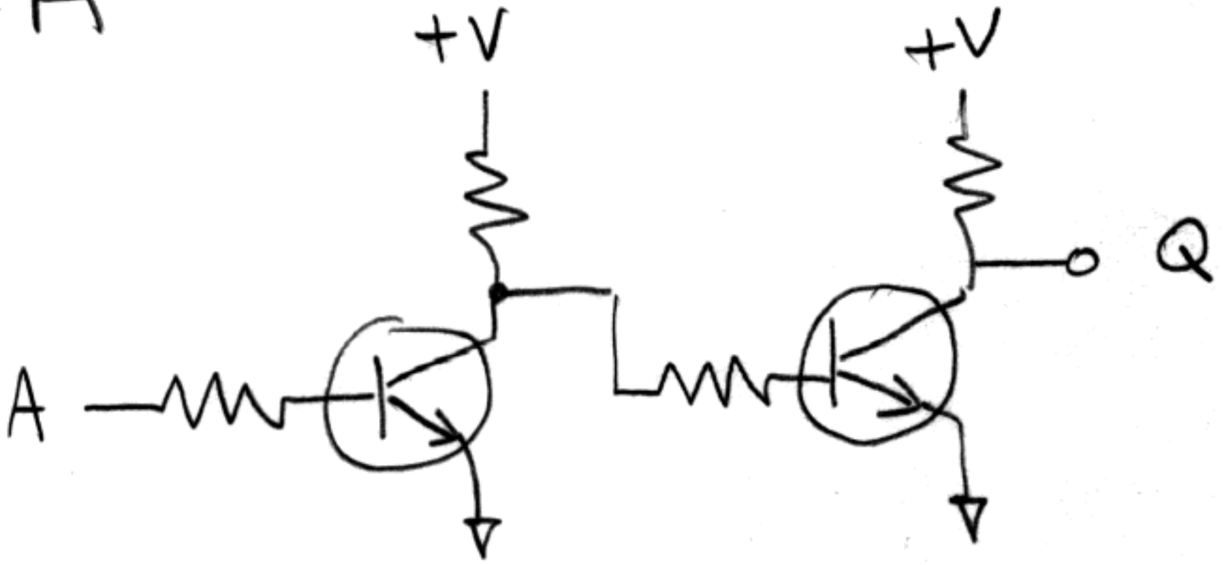
- Using AND, OR, and NOT you can make any logic circuit, up to the most complex computer

BUFFER

$Q = A$



A	Q
0	0
1	1

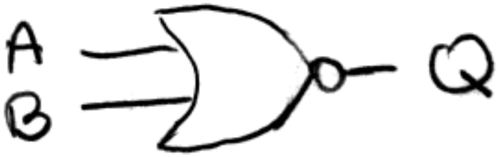


current gain,
"fan out", many inputs
from one output

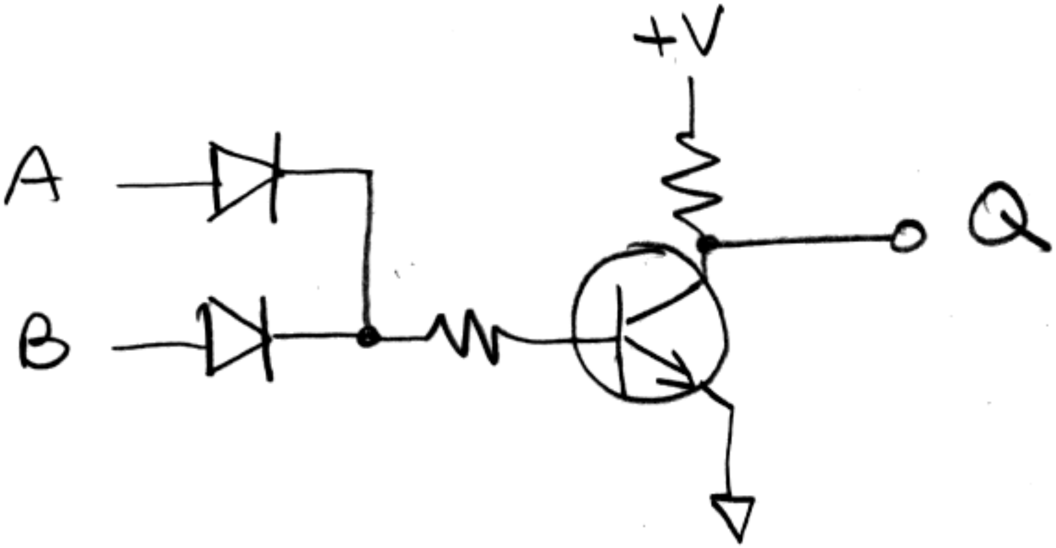
NOR

$$Q = \overline{(A+B)} = \bar{A}\,\bar{B} \leftarrow$$

both A and B must be 0 for Q to be 1.



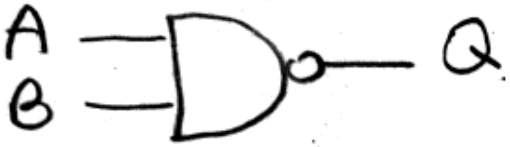
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0



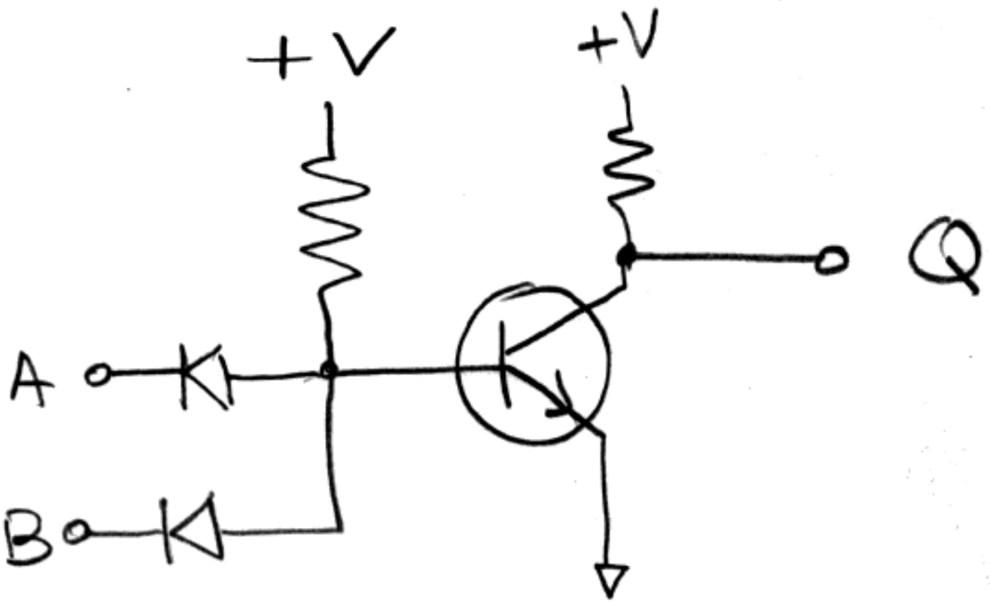
NAND

$$Q = \overline{(AB)} = \overline{A} + \overline{B}$$

if either A or B is 0, Q is 1.



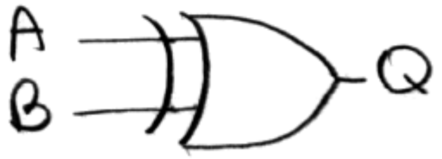
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0



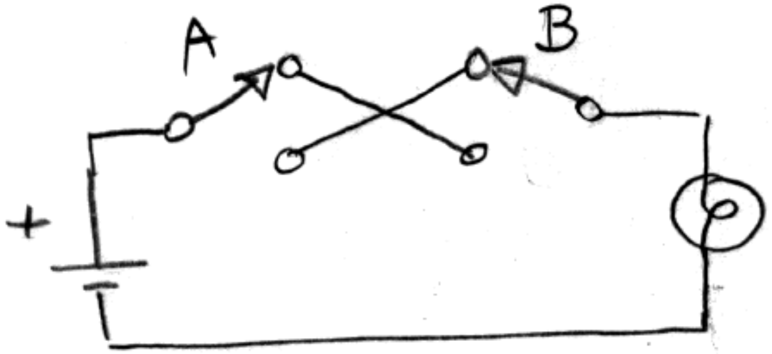
EXCLUSIVE OR (XOR)

$Q = A \oplus B$

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0



"one but not both"

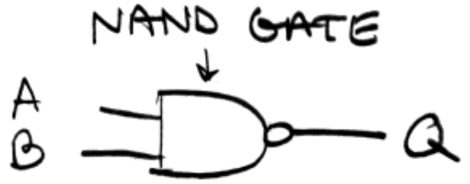


used in house wiring

- One input controls whether to invert the other.
- Using just XOR you can make any logic circuit.

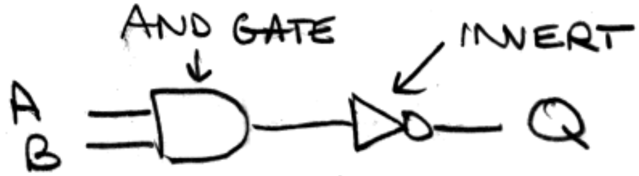
a little more about Boolean math

RECALL NAND



A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

SAME AS



$Q = \overline{(AB)}$

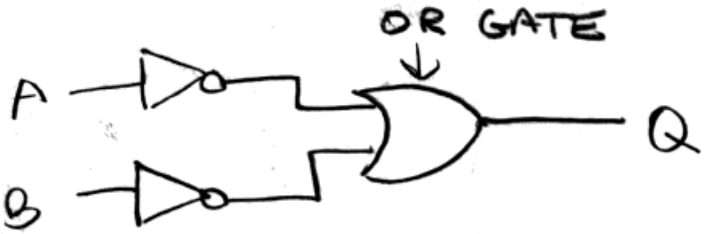
↑

equivalent expressions

↓

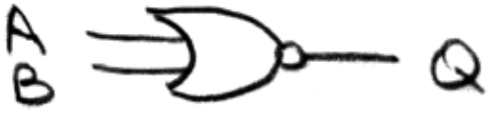
$Q = \overline{A} + \overline{B}$

SAME AS



So $\overline{(AB)} = \overline{A} + \overline{B}$

similarly for NOR



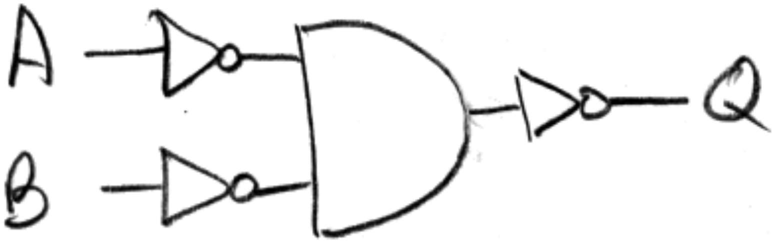
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

$$\overline{(A+B)} = \bar{A}\,\bar{B}$$

more examples

OR

$$Q = A + B = \overline{(\overline{A}\,\overline{B})}$$

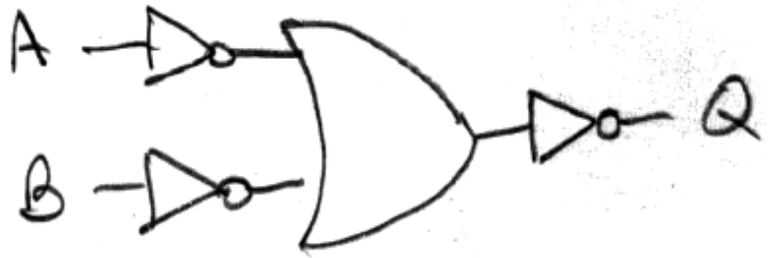


if both A and B are 0 Q is 0

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

AND

$$Q = AB = \overline{(\bar{A} + \bar{B})}$$



A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

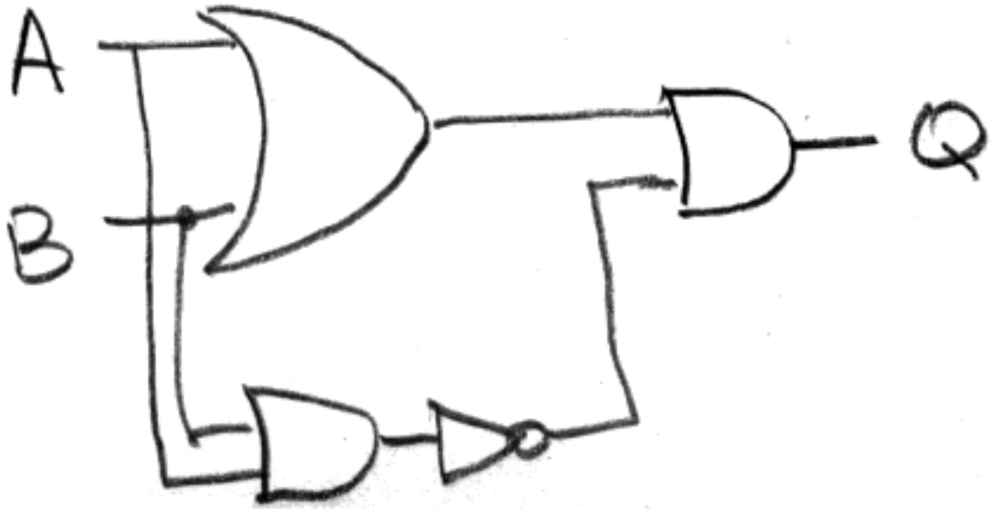
if either A or B
is 0

Q is 0

XOR

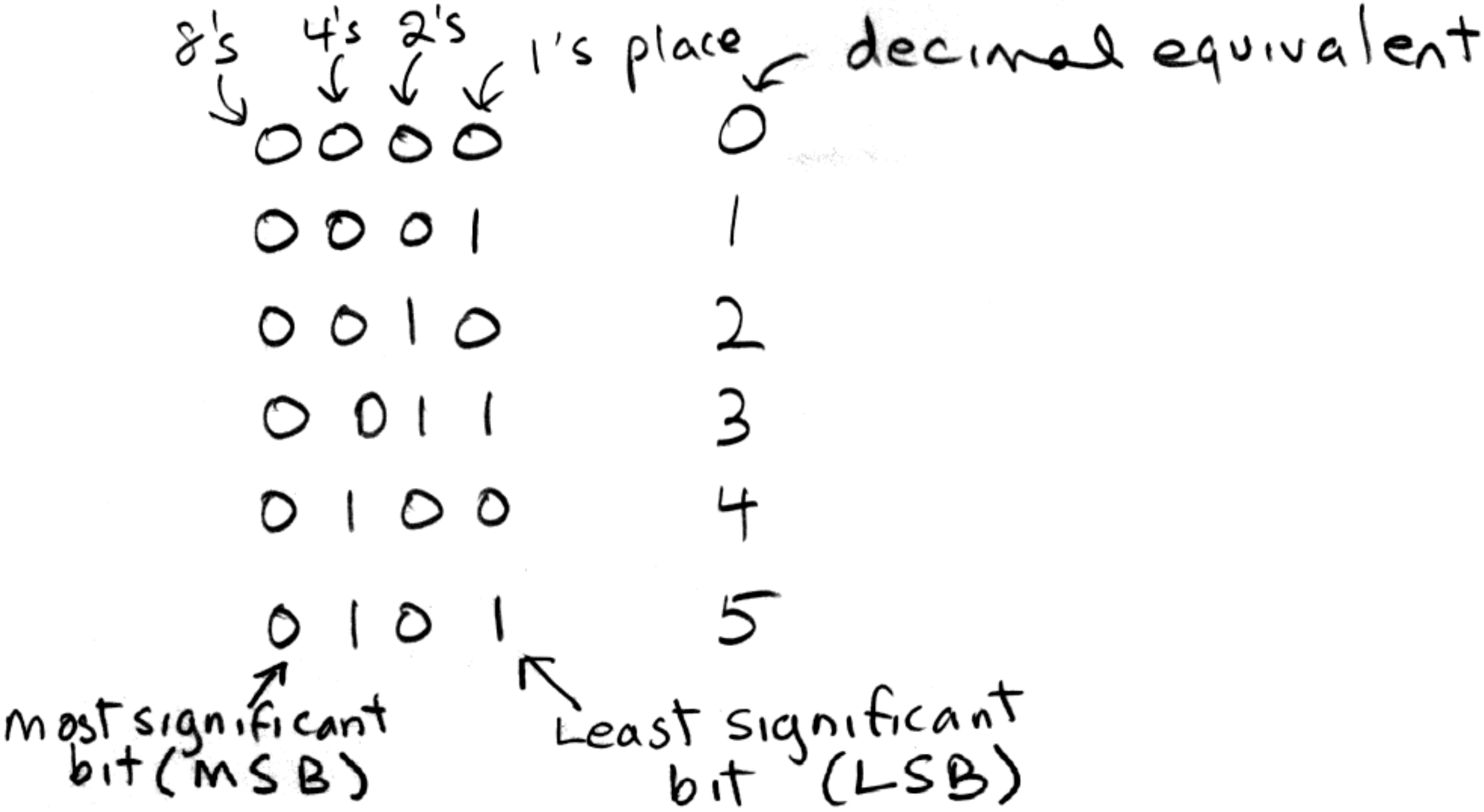
$$Q = A \oplus B = (A+B)\overline{(AB)}$$

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0



XOR is like OR, with the added stipulation that if A and B are both 1, then Q is 0.

Binary – Base 2



8's	4's	2's	1's place	decimal equivalent
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5

Just like base 10, with 1's, 10's, 100's place

Base 2, 10, 16

0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F

Numbers in computers most efficiently handled in powers of 2, e.g. hexadecimal (“hex”) = base 16.

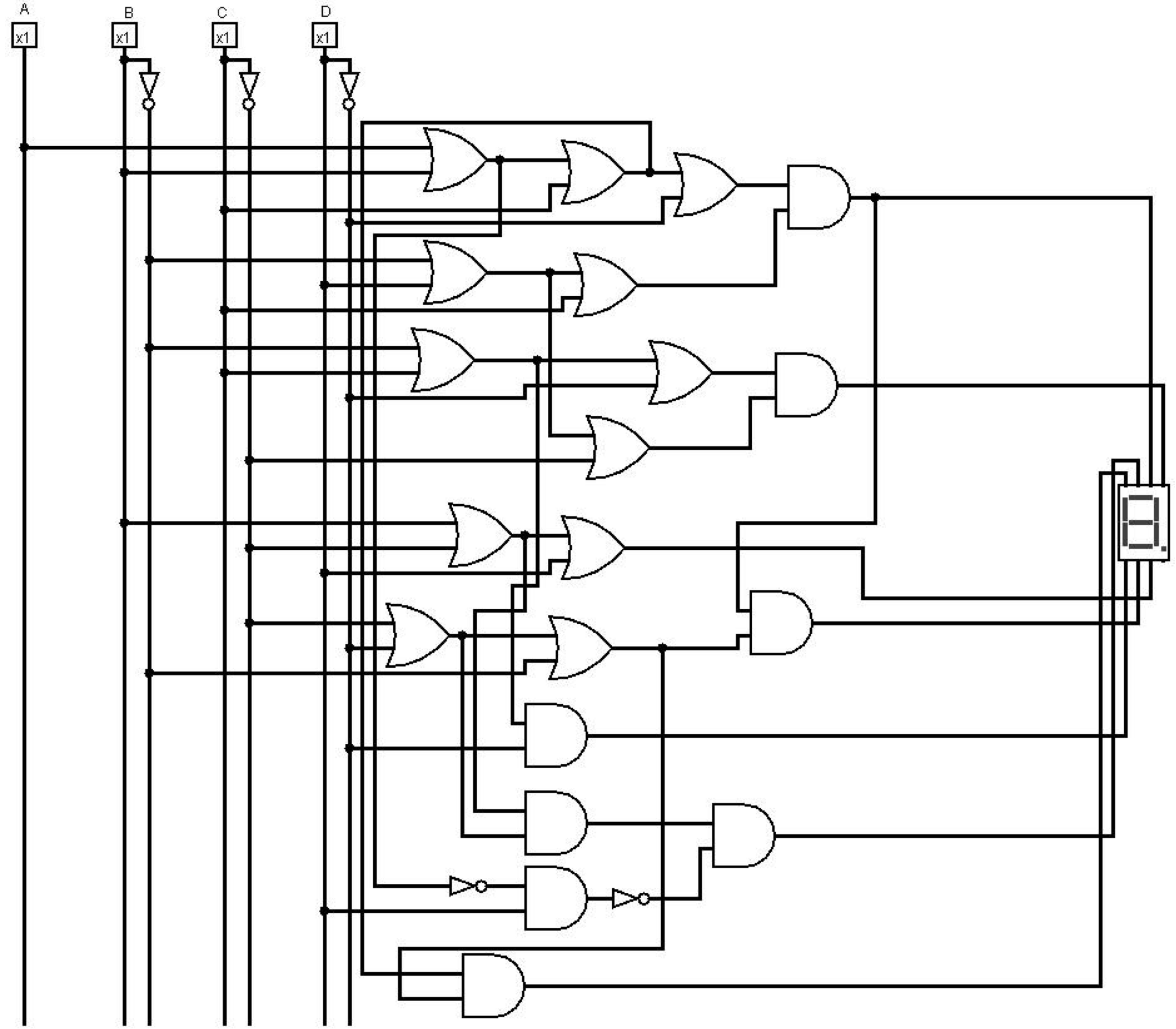
8 bits is a “byte”
4 bits is a “nibble” (one hex digit)

One byte is two hex digits, 0-255

$$42_{10} = 2A_{16} = \underbrace{0010}_{2_{16}}\underbrace{1010}_{A_{16}}{}_{2}$$

For human interfaces, and in calculators, we use “binary coded decimal” (BCD), where 4 bits only represents 0-9.

BCD to 7-Segment Display



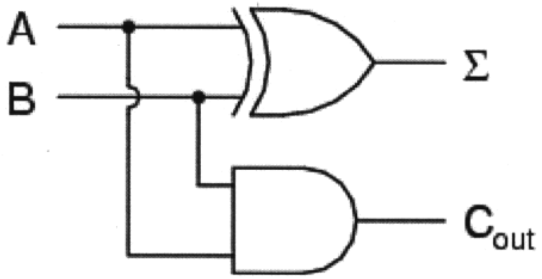
The *decoding* scheme for each segment being *on* or *off* for each BCD number is stored by this pattern of gates.

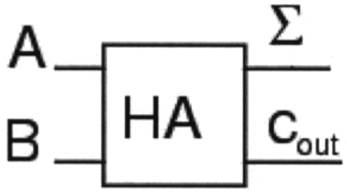
It represents a kind of *memory* (read only).

Binary Addition

$$\begin{array}{rr} \overset{1}{0}\,\overset{1}{1}\,1\,1_2 & 7_{10} \\ +0\,1\,1\,0_2 & +6_{10} \\ \hline 1\,1\,0\,1_2 & 13_{10} \end{array}$$

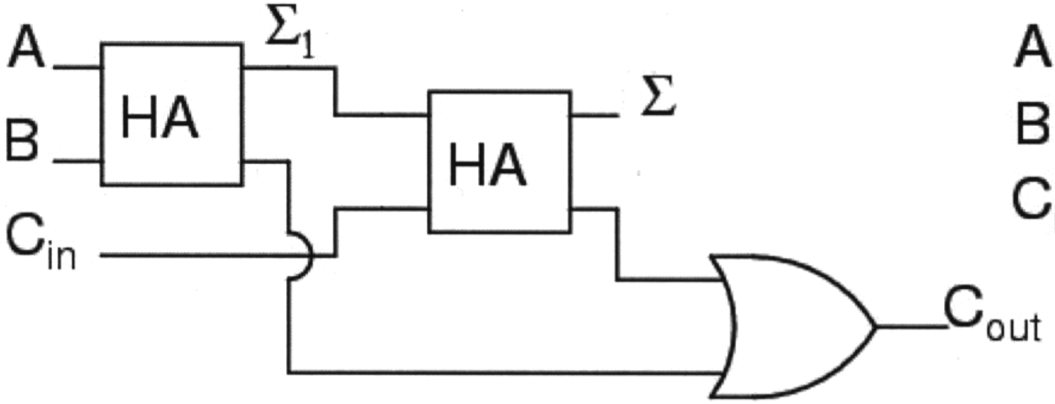
Half Adder

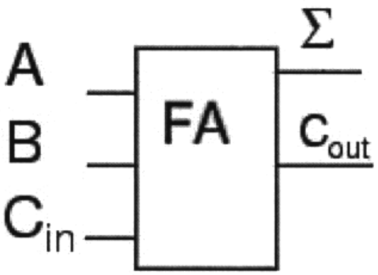


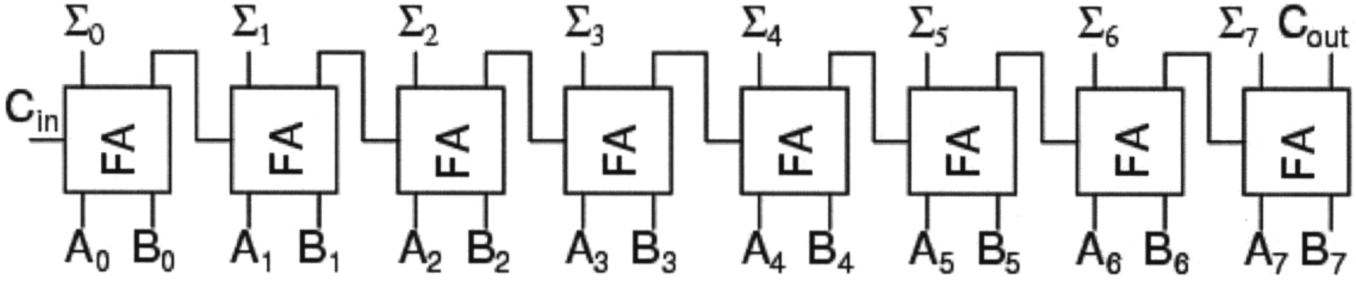


A	B	Σ	C_{out}
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Full Adder







Logical modules are built up hierarchically to reach levels of greater complexity than can be understood as collections of gates.

www.allaboutcircuits.com

Negative Binary Numbers

1's complement

3	0011
2	0010
1	0001
0	0000
-0	1111
-1	1110
-2	1101

2's complement

3	0011
2	0010
1	0001
0	0000
-1	1111
-2	1110
-3	1101

ignore this carry (modulo 16)

$$\begin{array}{rr} 1\;\,1\;1 & \\ 0111_2 & 7_{10} \\ -1110_2 & -2_{10} \\ \hline 0101 & 5_{10} \end{array}$$

subtraction is same logic as addition with 2's complement

Since only 4 bits are supported in this example, the numbers "wrap around" to 0 after counting past the largest number $1111_2=15_{10}$, dropping the 16 bit and leaving the remainder. The "modulo 16" operation finds the remainder after division by 16.

Other Binary Representations

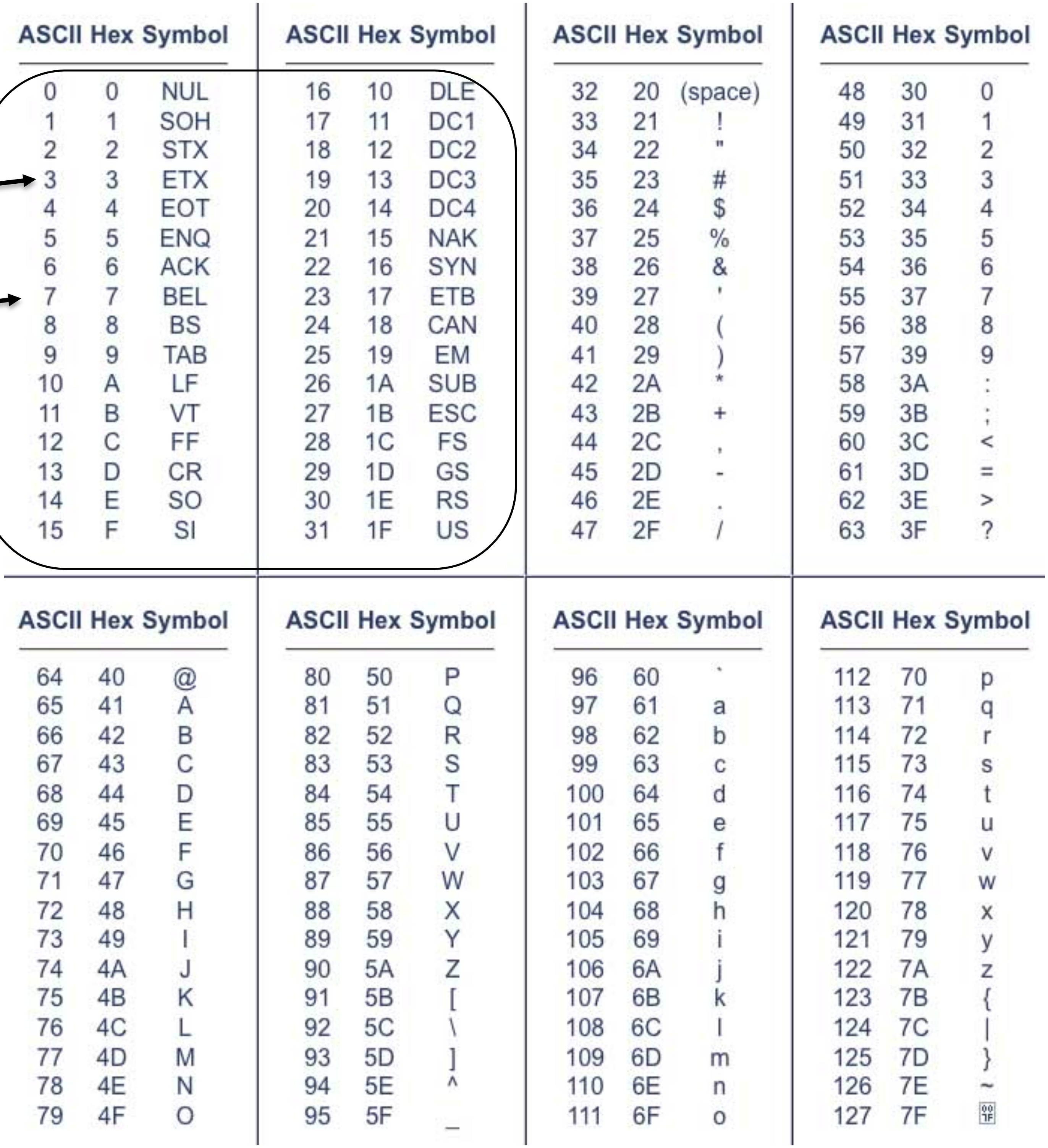
- ASCII
  - American Standard Code for Information Interchange
  - the human/computer Rosetta stone.
  - 7 bits (2^7=128)
  - upper and lower case, digits, control chars, punctuation, space, tab, line-feed, etc.
- Unicode (consortium decides)
  - extends ASCII to 2 bytes,
  - multiple languages, even emojis
  - first 128 still ASCII

- MIDI
  - Musical Instrument Digital Interface
  - 2-3 bytes per instruction
  - which note, how loud, what instrument, pedal, etc.

control characters

ctr-c = “end of text”

ctr-g = “bell”

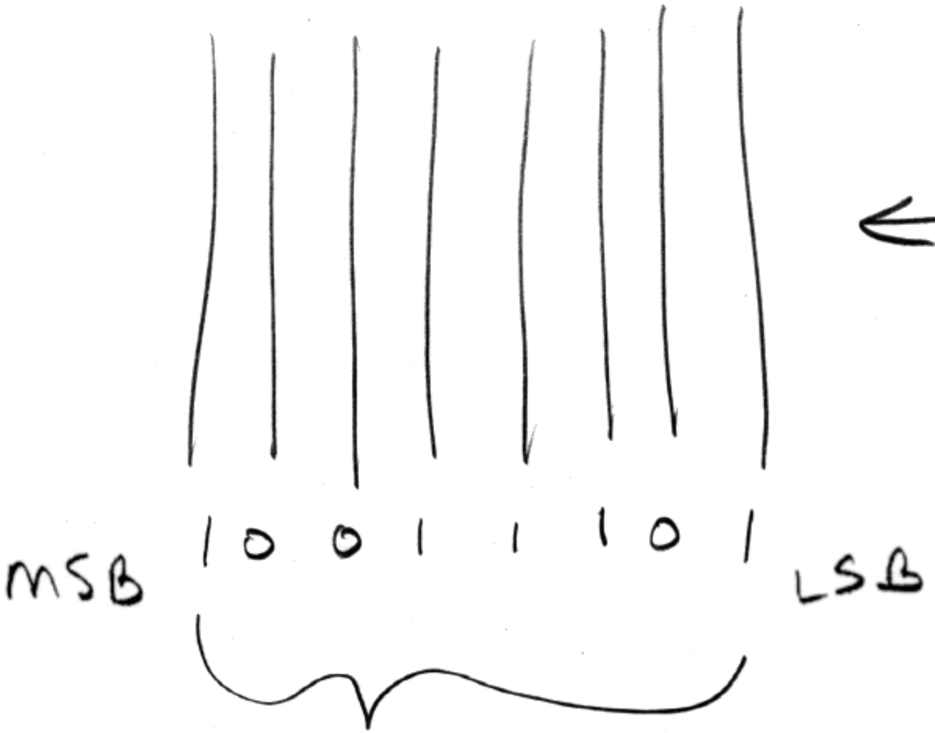


ASCII	Hex	Symbol	ASCII	Hex	Symbol	ASCII	Hex	Symbol	ASCII	Hex	Symbol
0	0	NUL	16	10	DLE	32	20	(space)	48	30	0
1	1	SOH	17	11	DC1	33	21	!	49	31	1
2	2	STX	18	12	DC2	34	22	"	50	32	2
3	3	ETX	19	13	DC3	35	23	#	51	33	3
4	4	EOT	20	14	DC4	36	24	$	52	34	4
5	5	ENQ	21	15	NAK	37	25	%	53	35	5
6	6	ACK	22	16	SYN	38	26	&	54	36	6
7	7	BEL	23	17	ETB	39	27	'	55	37	7
8	8	BS	24	18	CAN	40	28	(	56	38	8
9	9	TAB	25	19	EM	41	29	)	57	39	9
10	A	LF	26	1A	SUB	42	2A	*	58	3A	:
11	B	VT	27	1B	ESC	43	2B	+	59	3B	;
12	C	FF	28	1C	FS	44	2C	,	60	3C	<
13	D	CR	29	1D	GS	45	2D	-	61	3D	=
14	E	SO	30	1E	RS	46	2E	.	62	3E	>
15	F	SI	31	1F	US	47	2F	/	63	3F	?

ASCII	Hex	Symbol	ASCII	Hex	Symbol	ASCII	Hex	Symbol	ASCII	Hex	Symbol
64	40	@	80	50	P	96	60	`	112	70	p
65	41	A	81	51	Q	97	61	a	113	71	q
66	42	B	82	52	R	98	62	b	114	72	r
67	43	C	83	53	S	99	63	c	115	73	s
68	44	D	84	54	T	100	64	d	116	74	t
69	45	E	85	55	U	101	65	e	117	75	u
70	46	F	86	56	V	102	66	f	118	76	v
71	47	G	87	57	W	103	67	g	119	77	w
72	48	H	88	58	X	104	68	h	120	78	x
73	49	I	89	59	Y	105	69	i	121	79	y
74	4A	J	90	5A	Z	106	6A	j	122	7A	z
75	4B	K	91	5B	[	107	6B	k	123	7B	{
76	4C	L	92	5C	\	108	6C	l	124	7C	|
77	4D	M	93	5D	]	109	6D	m	125	7D	}
78	4E	N	94	5E	^	110	6E	n	126	7E	~
79	4F	O	95	5F	_	111	6F	o	127	7F	007F

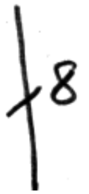
Binary Numbers in Hardware - Parallel

① in parallel
(usually within a machine)



← called a "bus", or along a ribbon cable

sometimes drawn as a single line

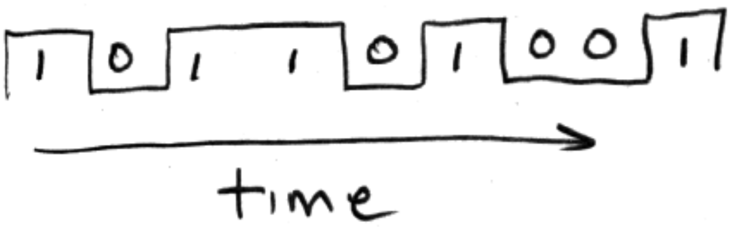


← number of bits in bus

Serial Data

② in series

(usually as a signal along a single wire or pair of wires, or wireless)



e.g. USB (universal serial bus)

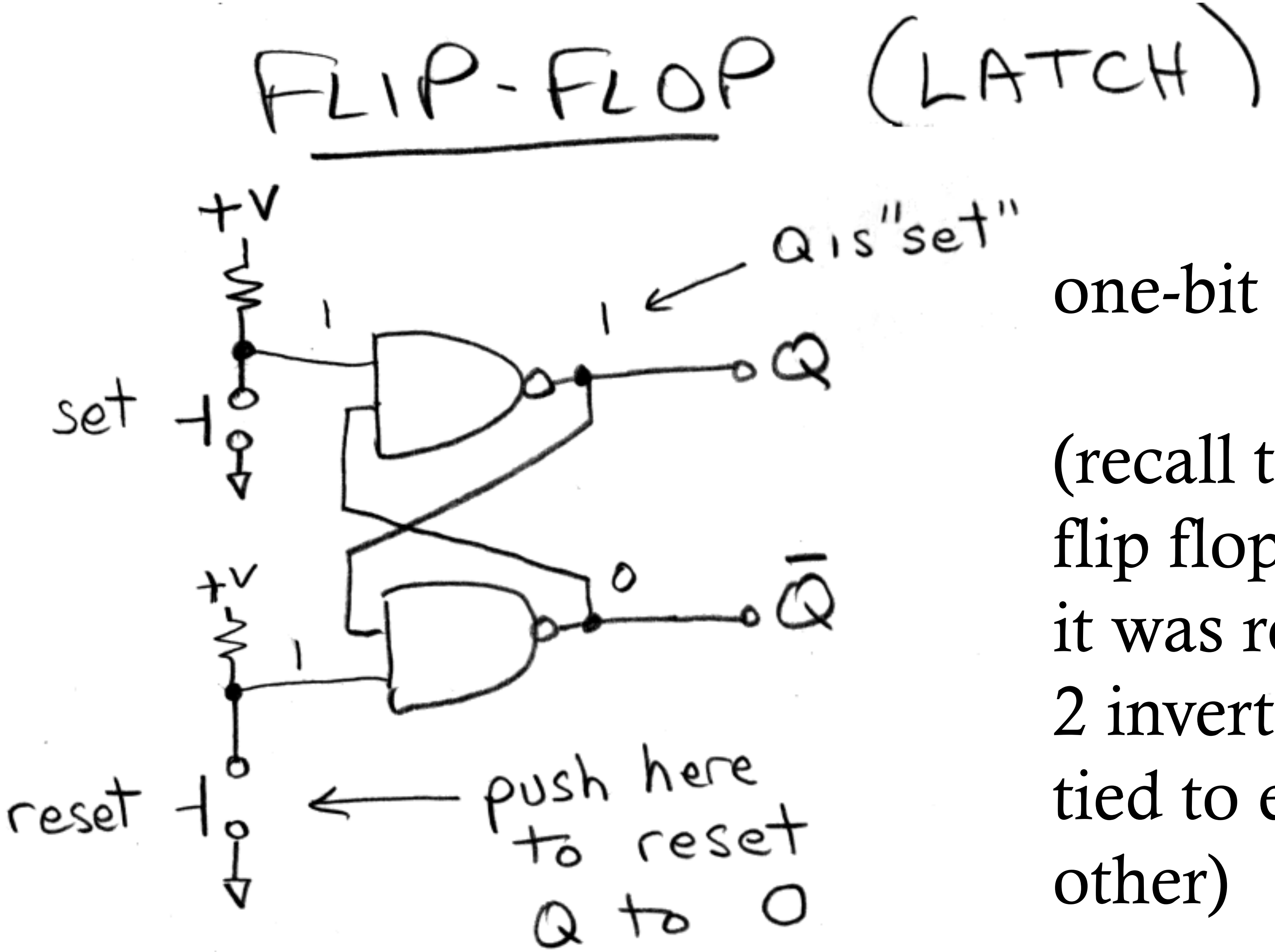
ethernet

Serial vs. Parallel Data

- Parallel
  - Fast transmission rate
  - Used inside computers for short distances
  - Parallel Busses (32 and 64 bit wide are standard)
  - Expensive in terms of numbers of wires connectors

- Serial
  - Slower transmission rate
  - Used for long distance transmission
  - Few or even single wires (USB, ethernet, …)
  - Good for fiber optics cables and wireless

Bistable Flip-Flop with gates

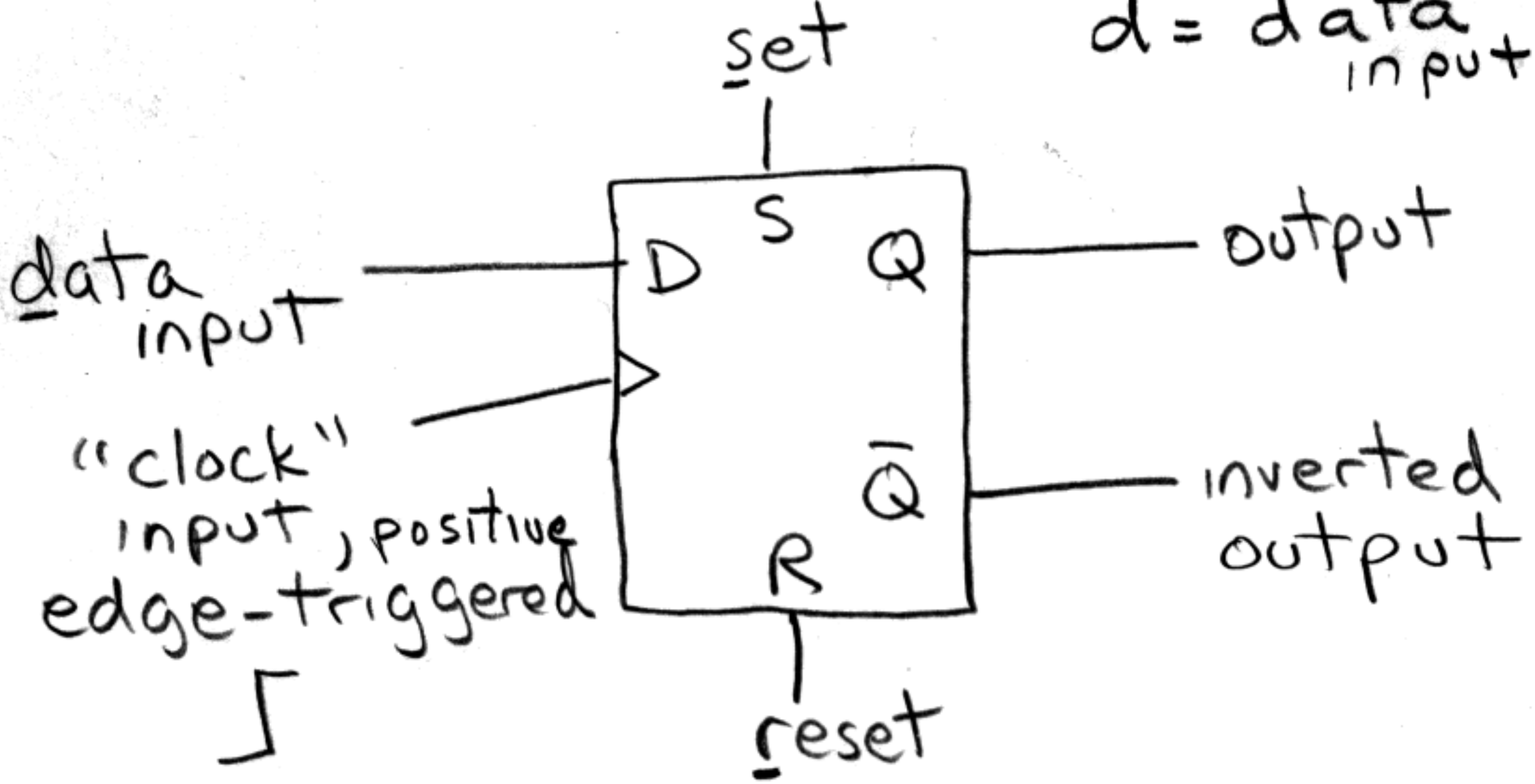


one-bit *memory*

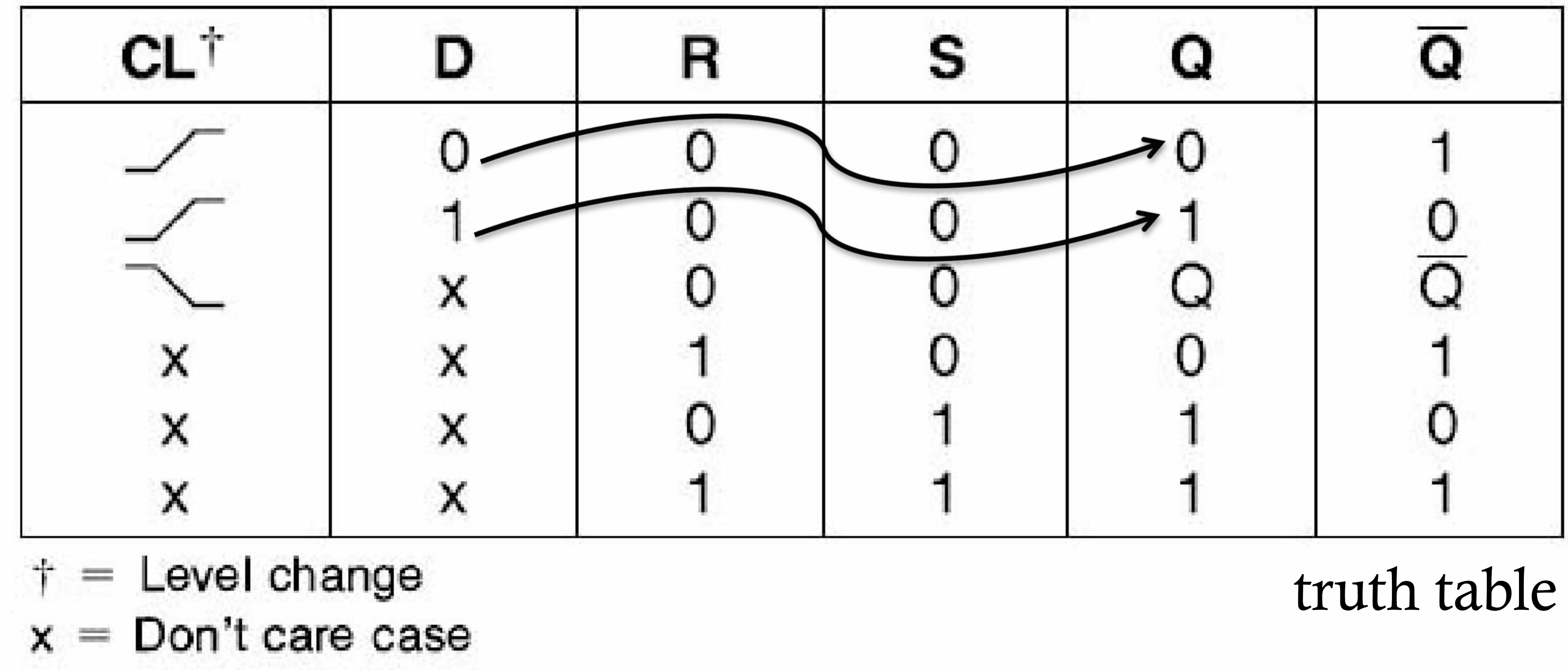
(recall transistor flip flop…
it was really
2 inverters
tied to each
other)

"D" FLIP-FLOP

s = set
r = reset
d = data input



when the clock goes from 0 → 1
whatever is at D is "latched"
at Q. S and R override

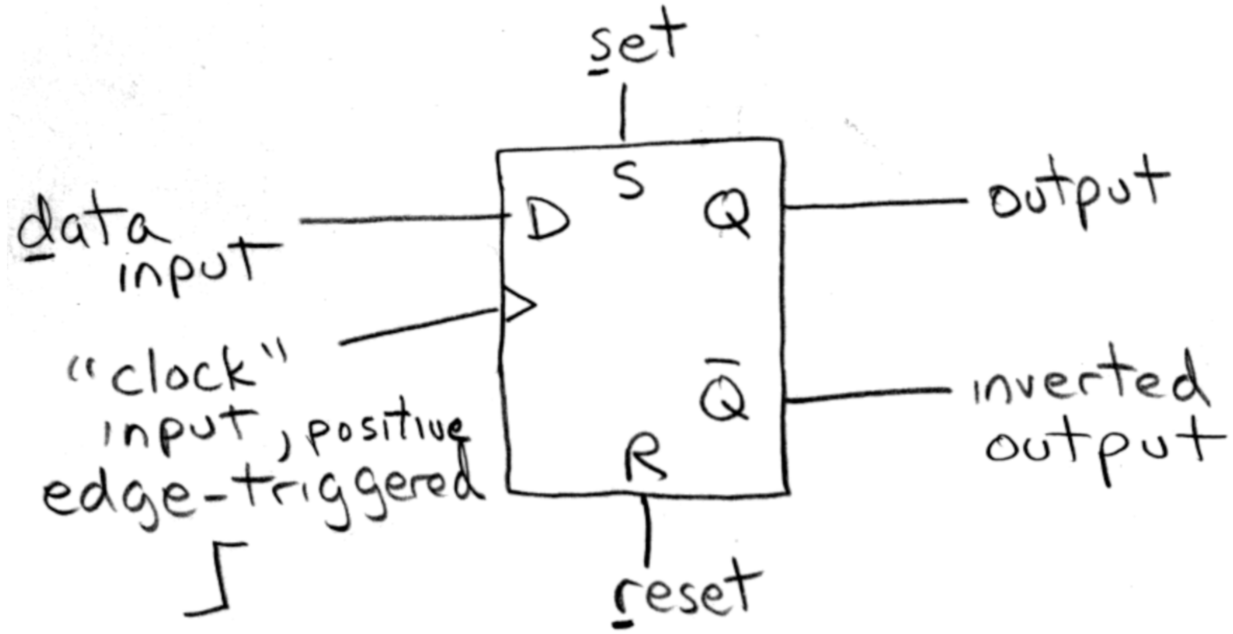


CL†	D	R	S	Q	$\overline{Q}$
↑	0	0	0	0	1
↑	1	0	0	1	0
↓	x	0	0	Q	$\overline{Q}$
x	x	1	0	0	1
x	x	0	1	1	0
x	x	1	1	1	1

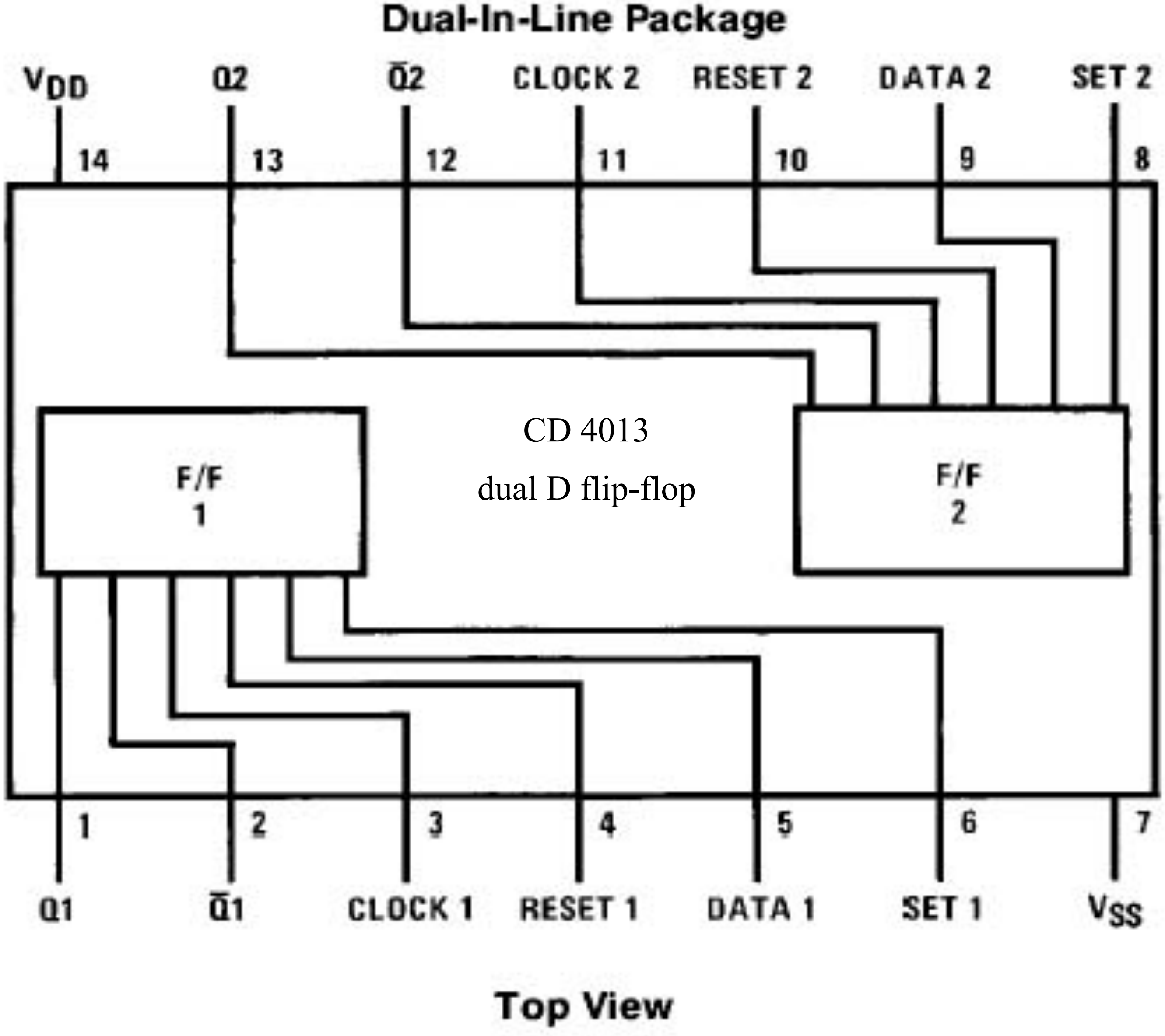
† = Level change

x = Don't care case

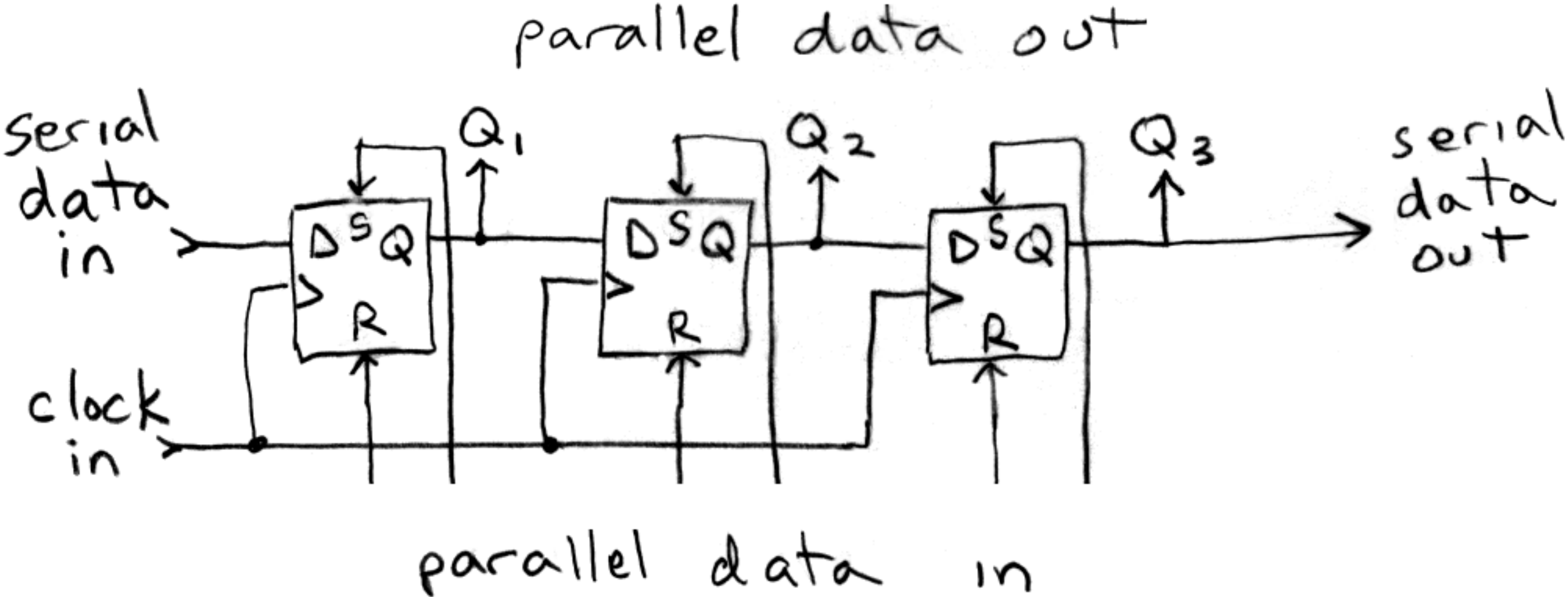
truth table



CD4013 - two D flip-flops on a DIP chip



Shift Register

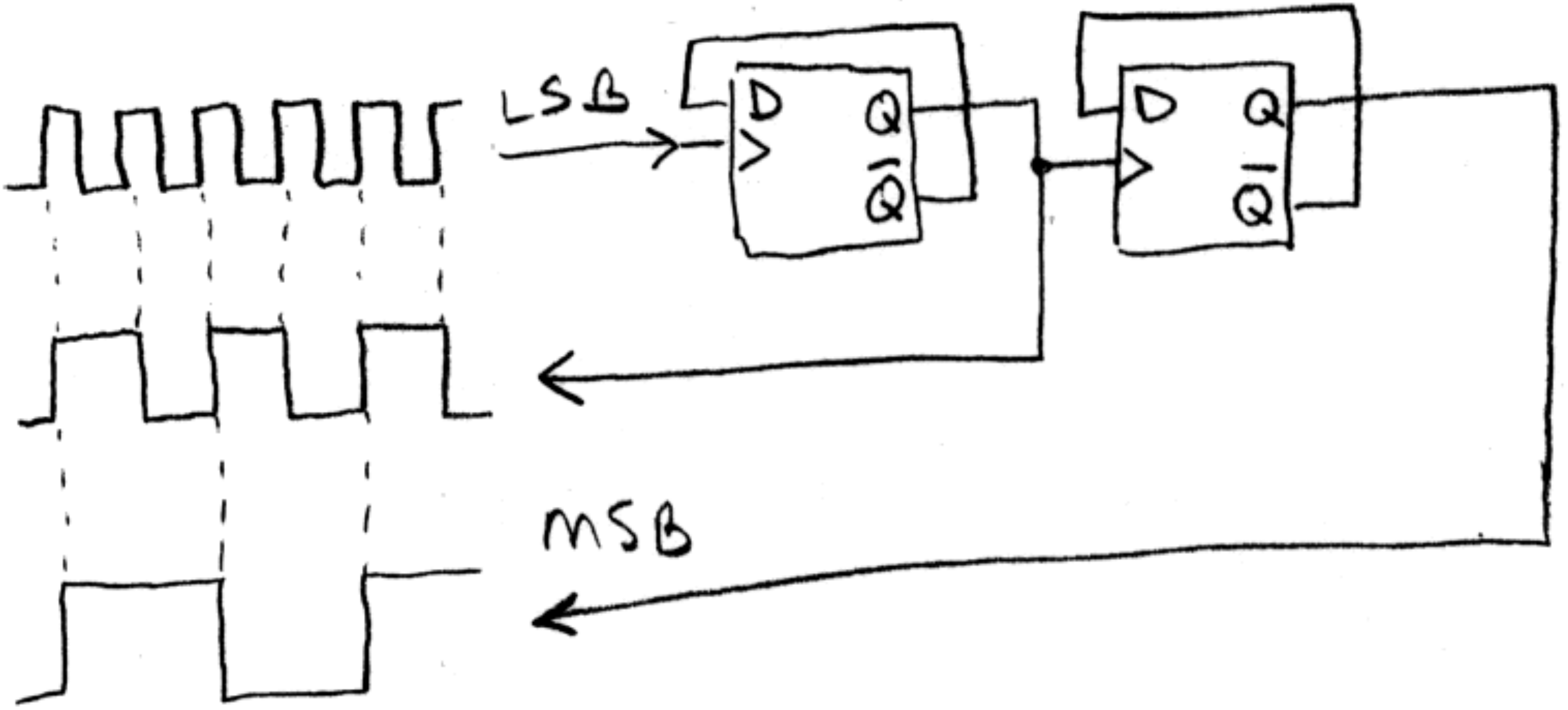


Serial-to-Parallel

or

Parallel-to-Serial

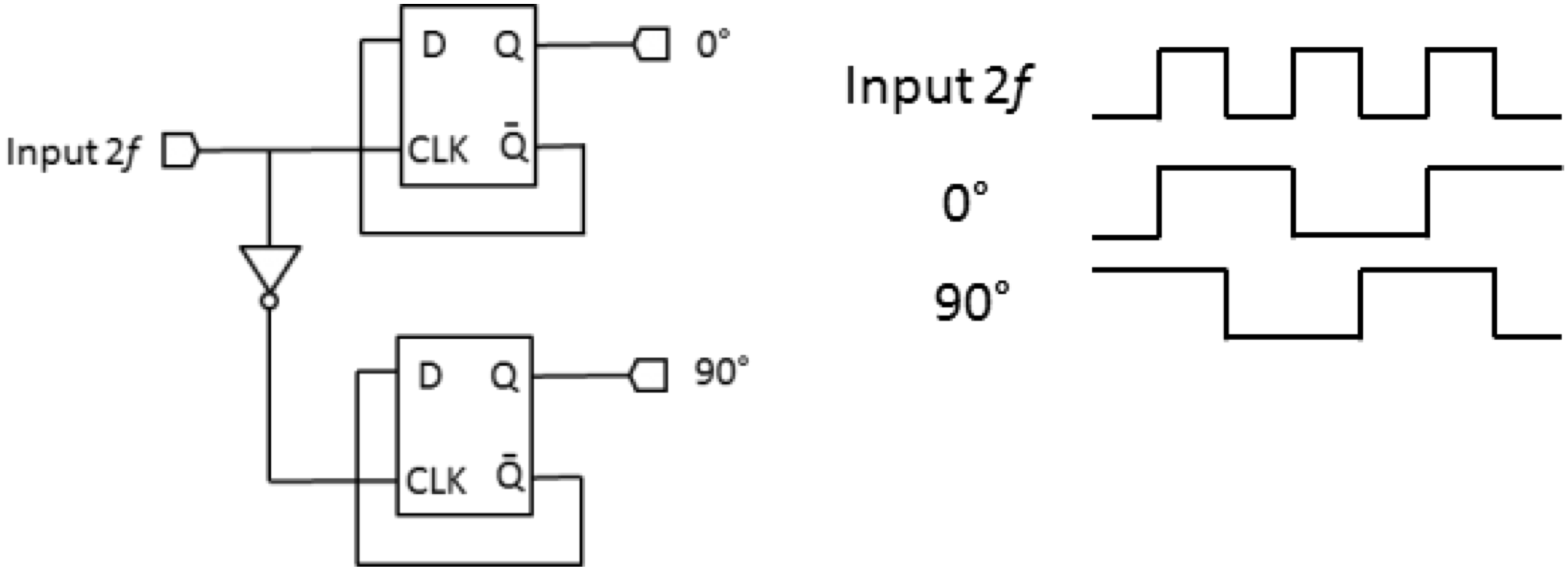
Divide-by-2 Ripple Counter



Propagation delay (for big numbers) leads to ambiguous states for short periods of time during *ripple* of clocks down the line.

Digital Phase Splitter

- Produces 2 square waves with 90° phase shift between them.
- Used in signal processing as a digital version of cos and sin.

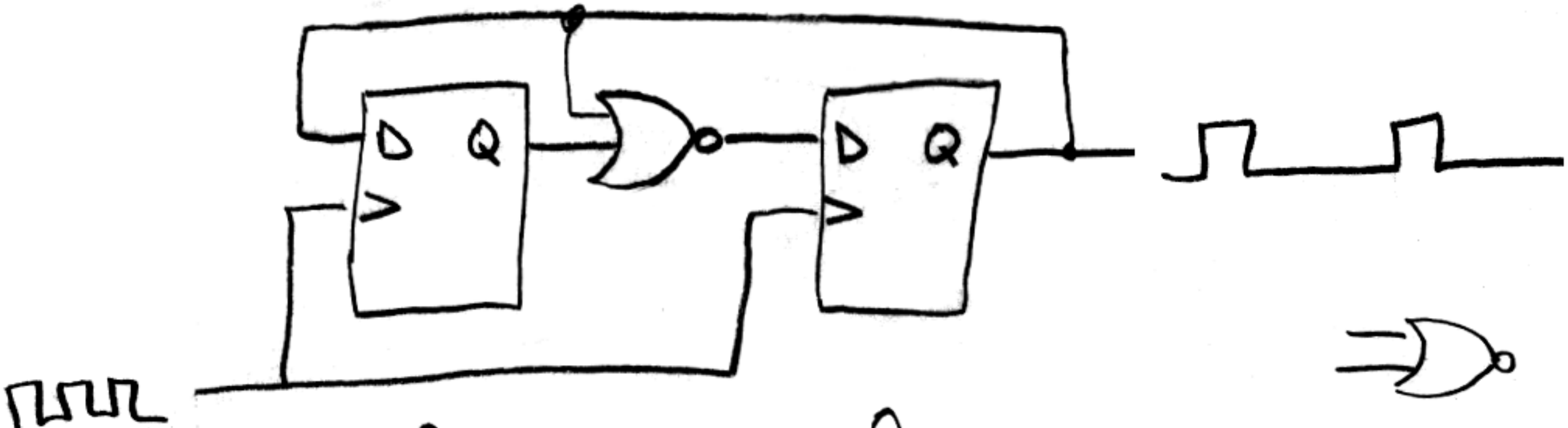


https://www.markimicrowave.com/blog/top-7-ways-to-create-a-quadrature-90-phase-shift/

State Machine

- Synchronous changes governed by global clock
- Next state determined by present state
- Avoids problems with propagation delays
- Basis of the computer

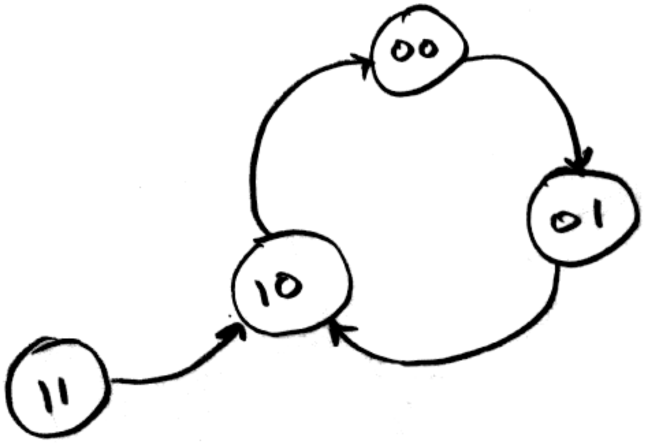
Divide by 3 counter



0	0
0	1
1	0
0	0

NOR

A B	Q
00	1
01	0
10	0
11	0

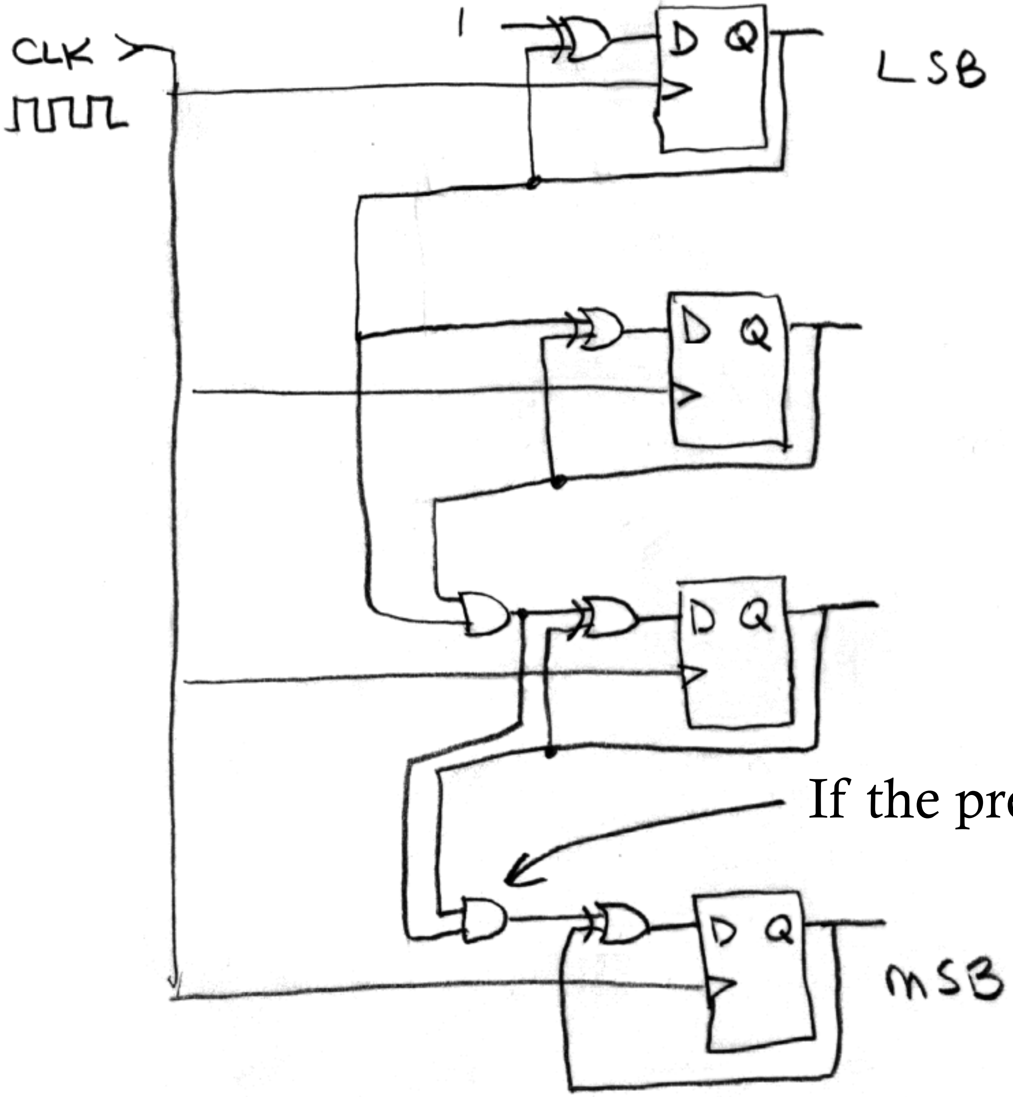


State machine repeats in at most 2^n cycles for n bits.

Horowitz and Hill, p .514

Avoiding Ripple - Synchronous Counter

Carry if all lower significant bits are 1
then you change state (recall XOR)

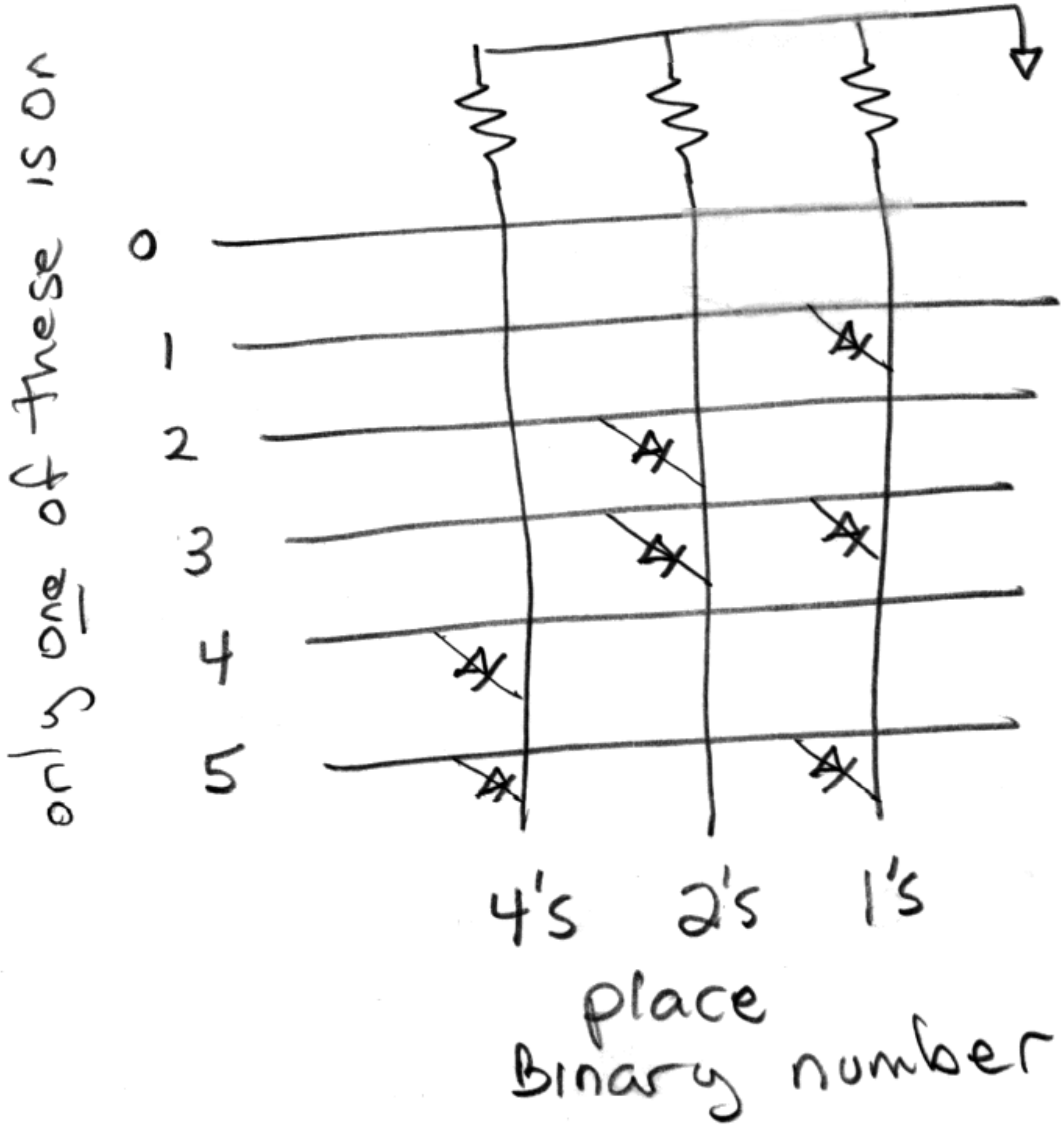


If the previous digit and all the digits before it are 1, then you change state.

0000
0001
0010
0011
0100
0101
0110
0111
1000

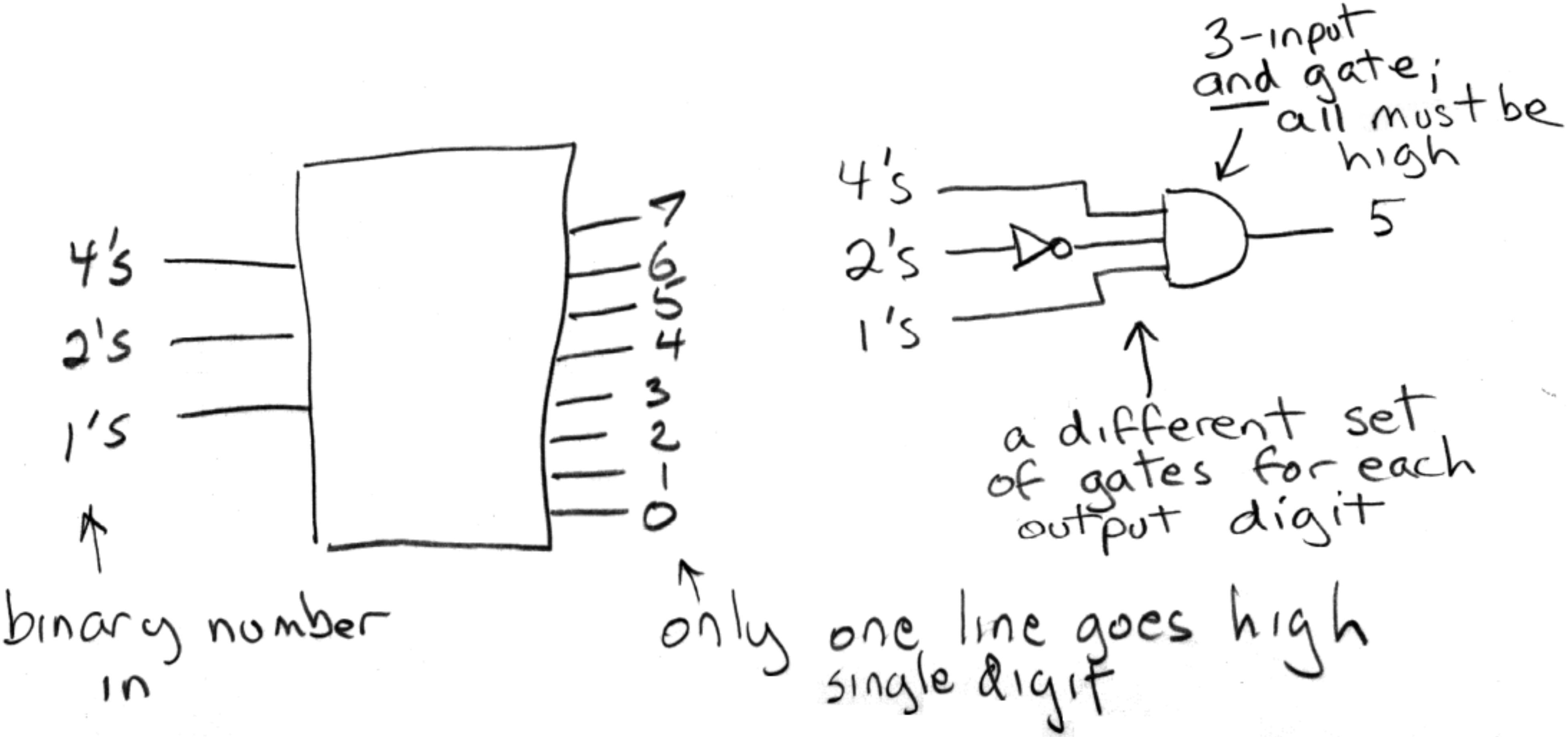
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Binary Encoder



This is actually a "Read Only Memory" or ROM, addressed by separate lines for 0-5 and returning those values as binary numbers.

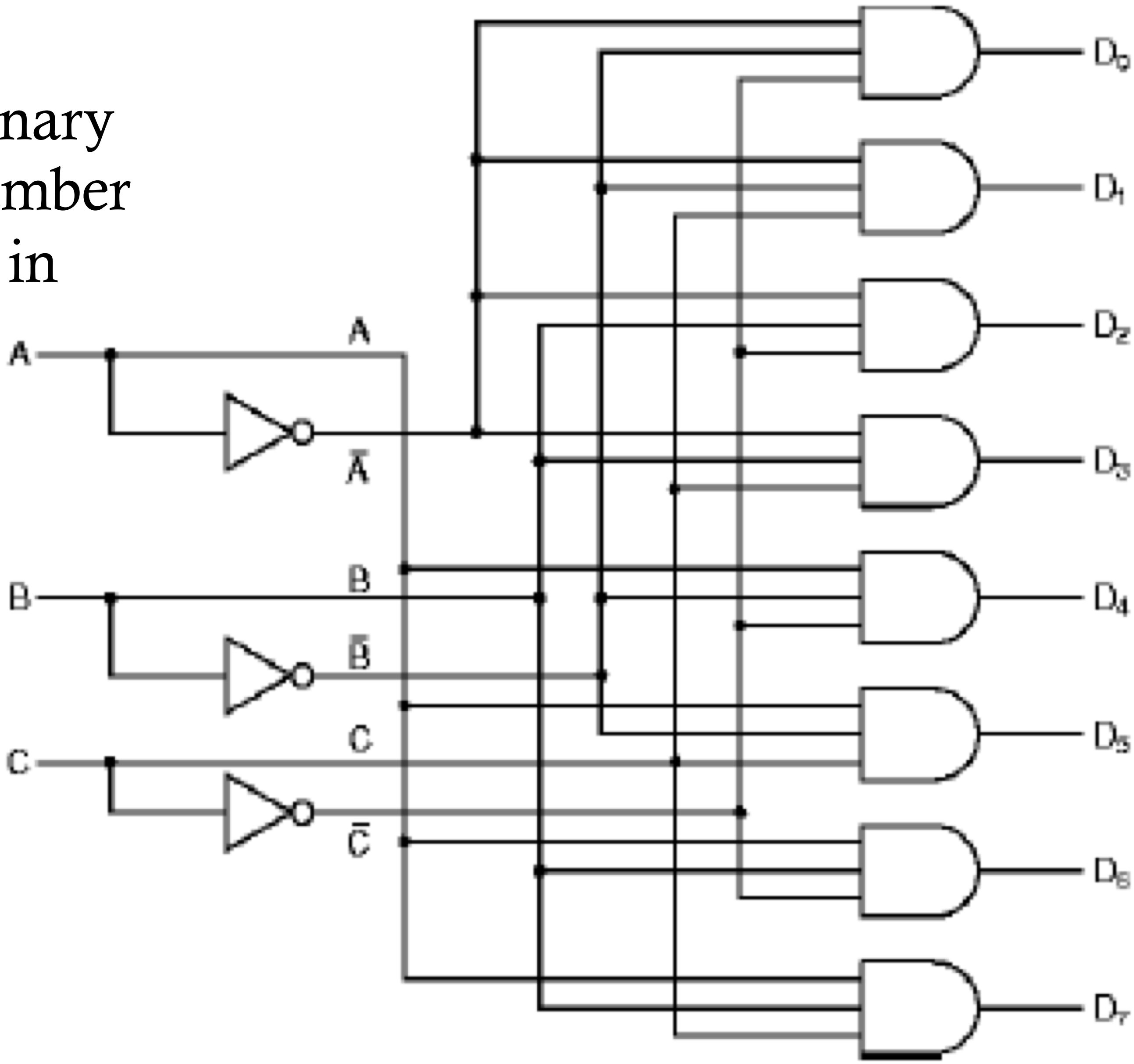
Binary Decoder


4's
2's
1's
binary number in
7
6
5
4
3
2
1
0
only one line goes high
single digit
4's
2's
1's
5
3-input and gate; all must be high
a different set of gates for each output digit

Performs the opposite function from the binary *encoder*: activates a single line corresponding to a particular binary number at the input.

Binary Decoder

binary
number
in

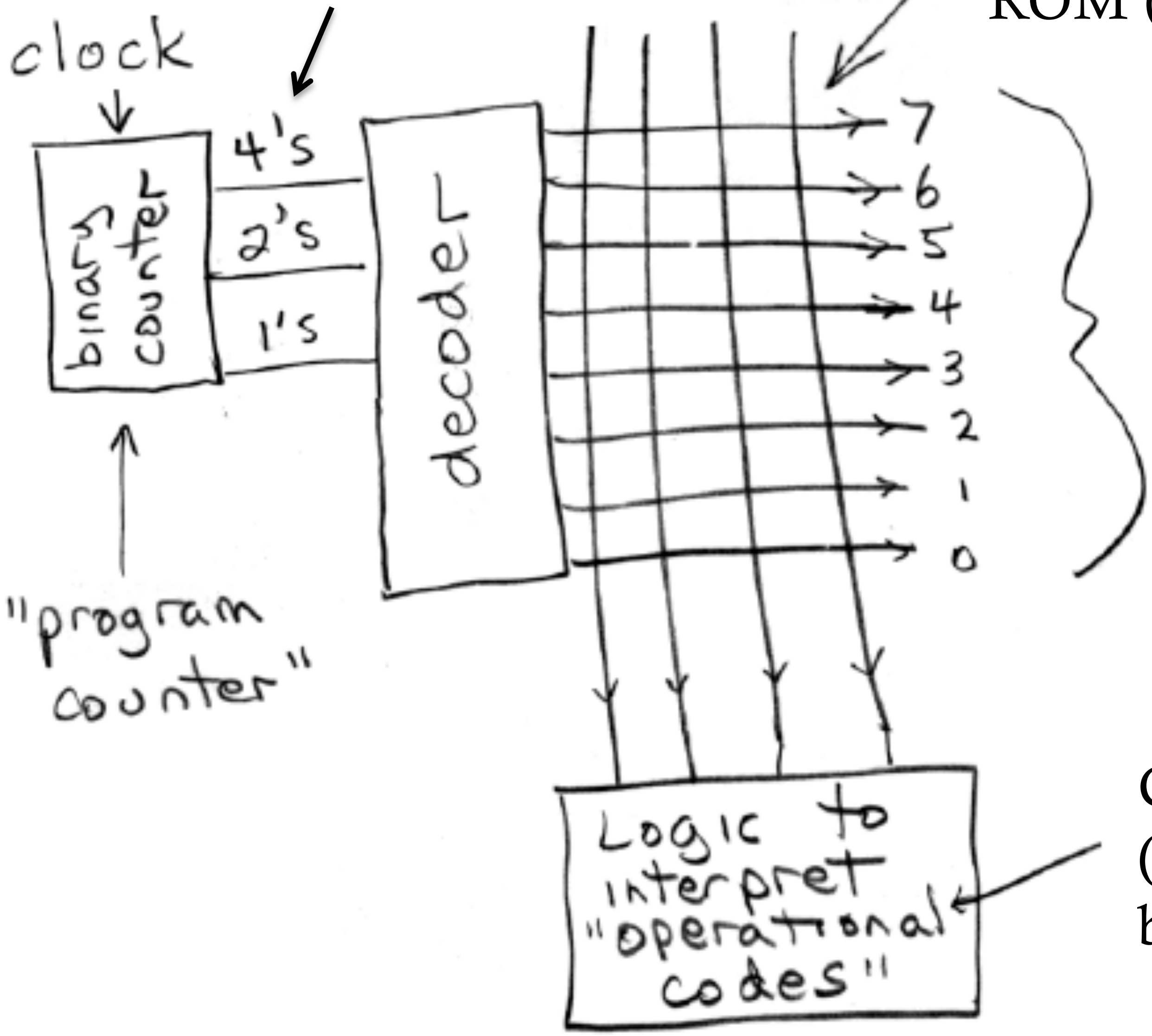


n data inputs
2^n data outputs

3-to-8 decoder

only one
output high
at a given time

Simple Computer



address bus

data bus

put diodes here for program: ROM (Read Only Memory)

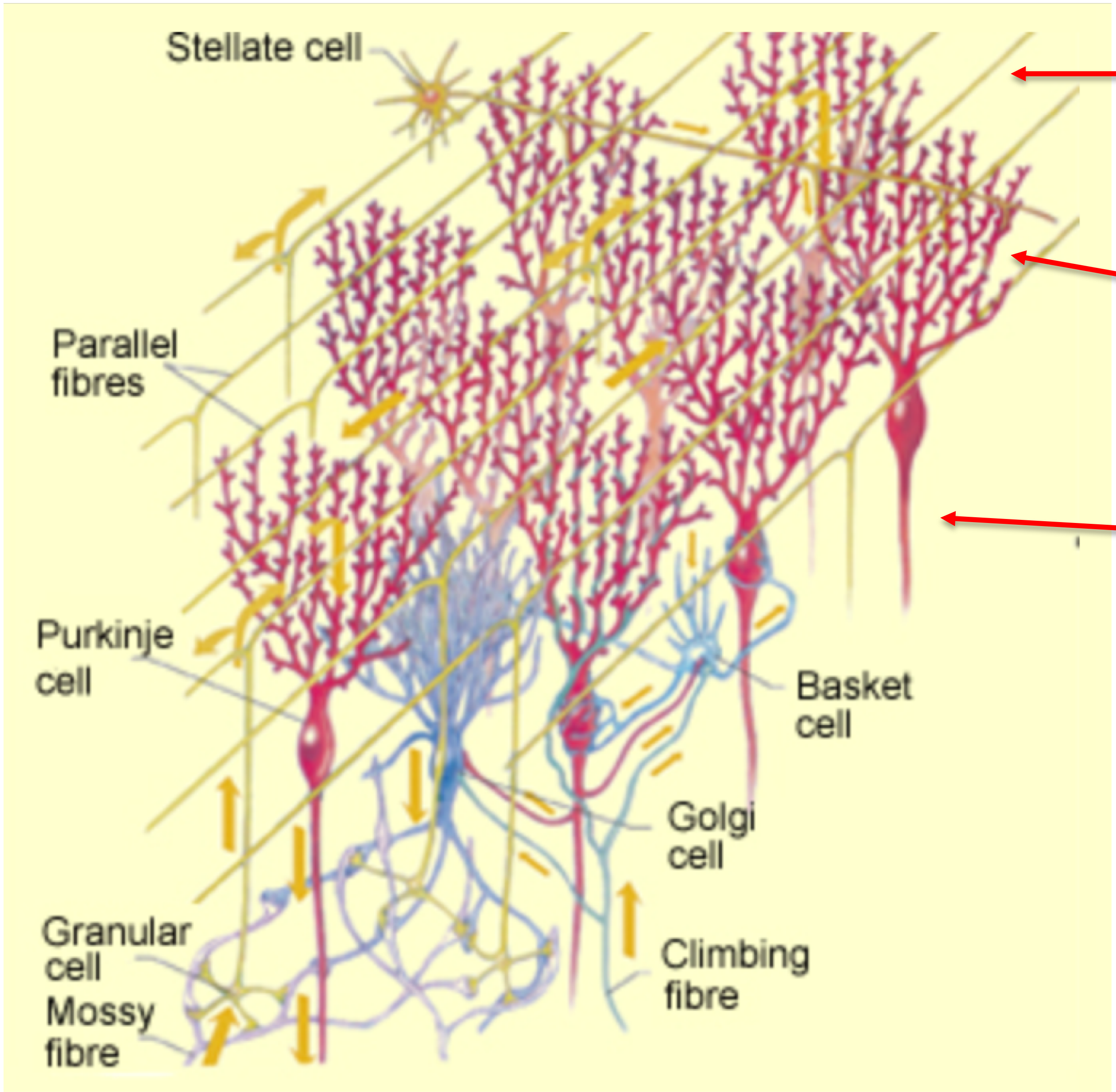
only one of these lines high at a given time (sequential)

Central Processing Unit (CPU) performs actions based on "Op Codes"

Central Processing Unit (CPU)

- CPU recognizes binary numbers and does specific tasks.
- For example, the number 5 could be decoded to activate a circuit to perform binary addition between two numbers.
- These machine-specific “operational codes” (op codes) form a “machine code” language for a given hardware platform, specific to a manufacturer (e.g. Intel).
- The numbers to be added and the resulting sum could be stored in specialized locations in the CPU known as “registers,” or they could be at addresses in memory specified in the program just after the op code.

Memory Addressing in the Cerebellum



Parallel fibers
= address bus.

Purkinje cell dendrites
= decoders.
THESE CAN LEARN!
(by synapses changing)

Purkinje cell axons
= data bus.

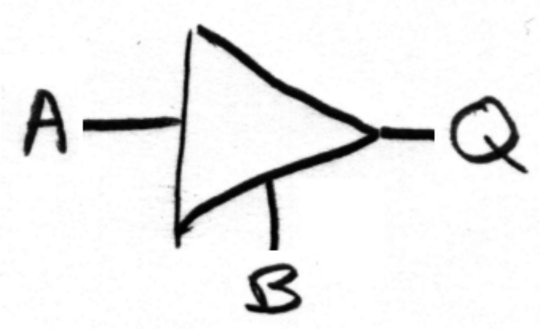
Each Purkinje cells is addressed by >100,000 parallel fibers in the cerebellum.

- http://thebrain.mcgill.ca

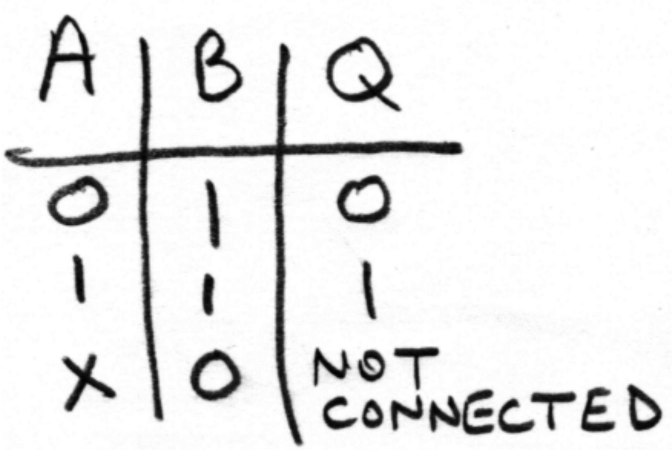
Now let's replace diodes in memory with gates, so we can change what is in the memory and learn. To read and write on the *same* data bus, we need…

Tri-State: a new kind of digital output

- Three output states instead of two:
  1, 0, and "off" (high impedance)
- Permits 2 outputs to share same wire

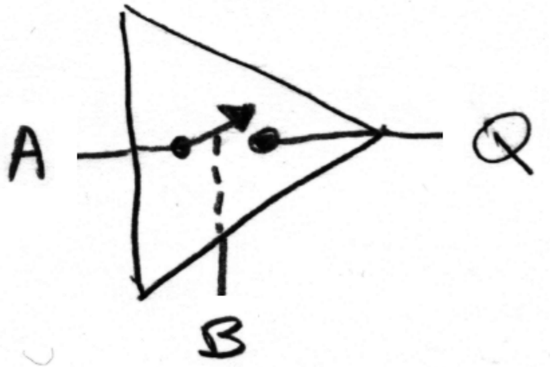


control input can disconnect output entirely



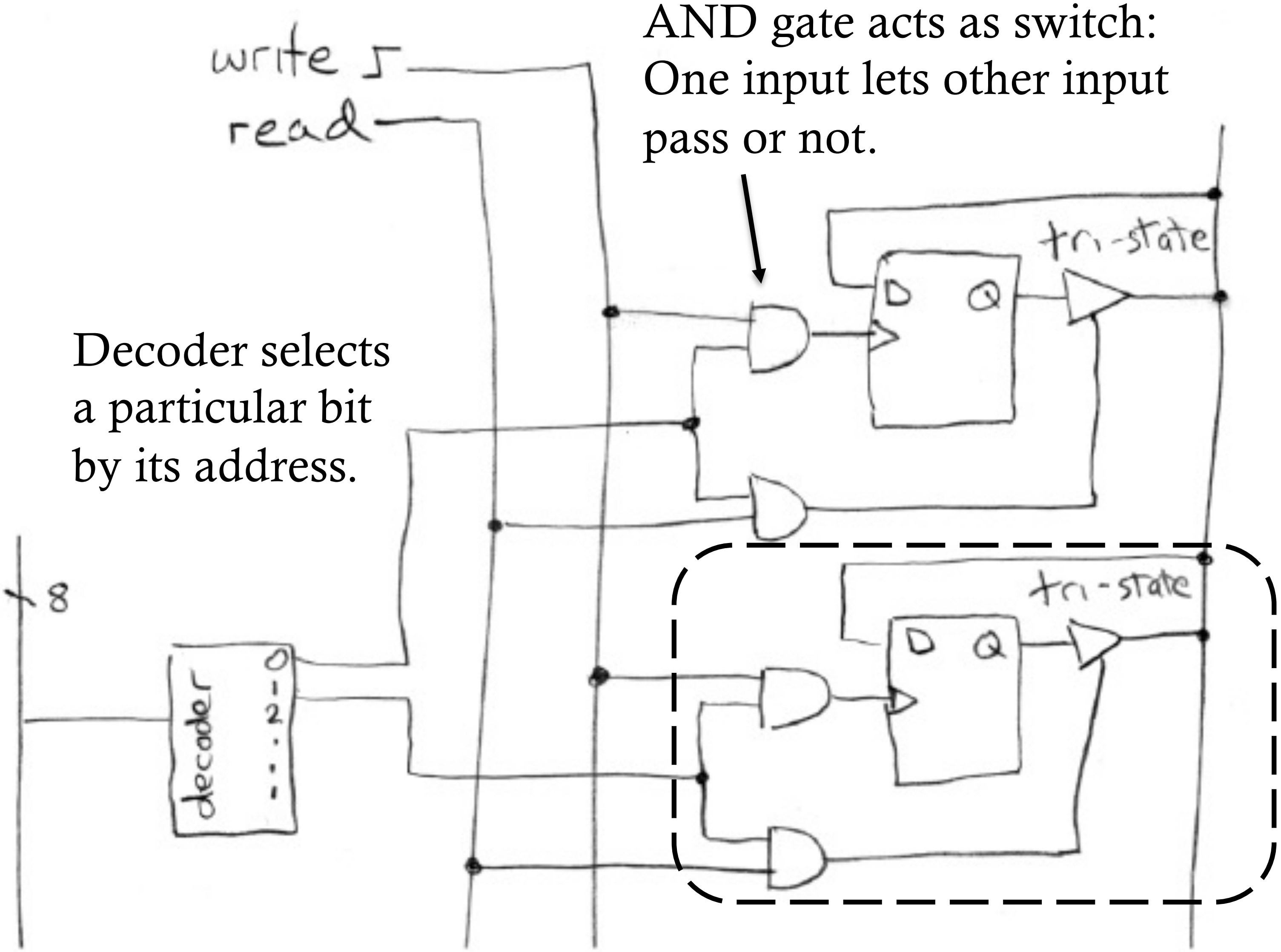
A	B	Q
0	1	0
1	1	1
X	0	NOT CONNECTED

X = EITHER 0 OR 1



Random Access Memory (RAM) can learn!

can now read and write on the same data bus.



AND gate acts as switch: One input lets other input pass or not.

Decoder selects a particular bit by its address.

1 bit of RAM stored in flip-flop

address bus

control bus

single data bus read and write

- Originally RAM was only used to store data.
- Big step: putting the program itself into RAM (Von Neuman 1949) replaces programming with diodes.

- Also introduction of special Op Codes:
  - "Go To" sets program counter to any location.
  - "If" tests something and controls program counter based on the result.
  - "Go Sub" jumps to a new section of code (subroutine)
  - "Return" jumps back.
  - "Stack" (first-in-last-out memory) stores return address, so subroutines can call other subroutines.
  - Leads to inevitable "stack overflow."

Power-on condition

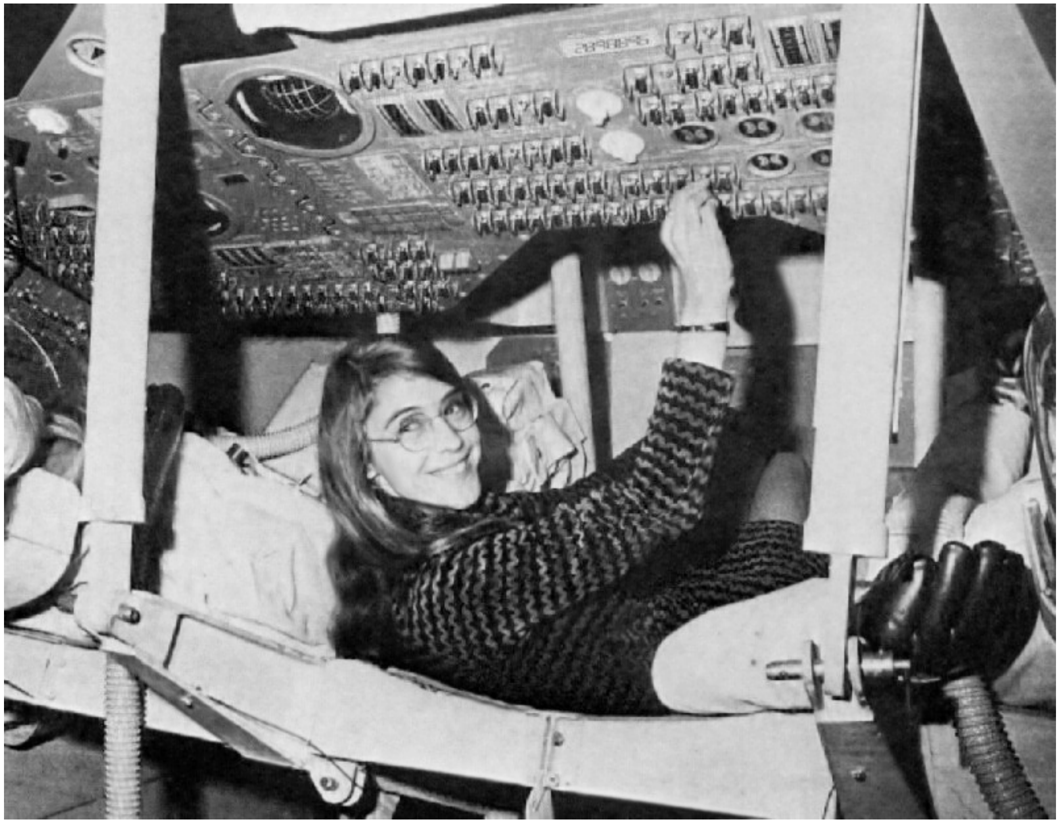
- State machines (computers) need an unambiguous power-on condition, an initial state.
- A crash is generally an unintended state, from which a power reset may be required.
- Some part a computer memory must be non-volatile,.
  - ROM (read only memory – write at the factory)
  - PROM (programmable ROM - write once at home)
  - EPROM (erasable PROM – erase with UV light and rewrite)
  - EEPROM (electronically EPROM – rewrite electronically)
- Basic Input/Output System (BIOS)
  - stored in EEPROM, bootstraps the *operating system*
  - thus “flashing” (rewriting) the BIOS” is risky

Other Additions to Basic Computer

- CPU becomes more complex
  - Internal “registers” (one-word memories) to hold numbers for immediate operations
  - Input-Output (I/O) lines
  - hardware clocks and counters
- Hardware Interrupt
  - ability to send the Program Counter to special location in response to hardware event
  - old value stored on a stack, as in Go Sub
  - leads to less reliable systems because of stack

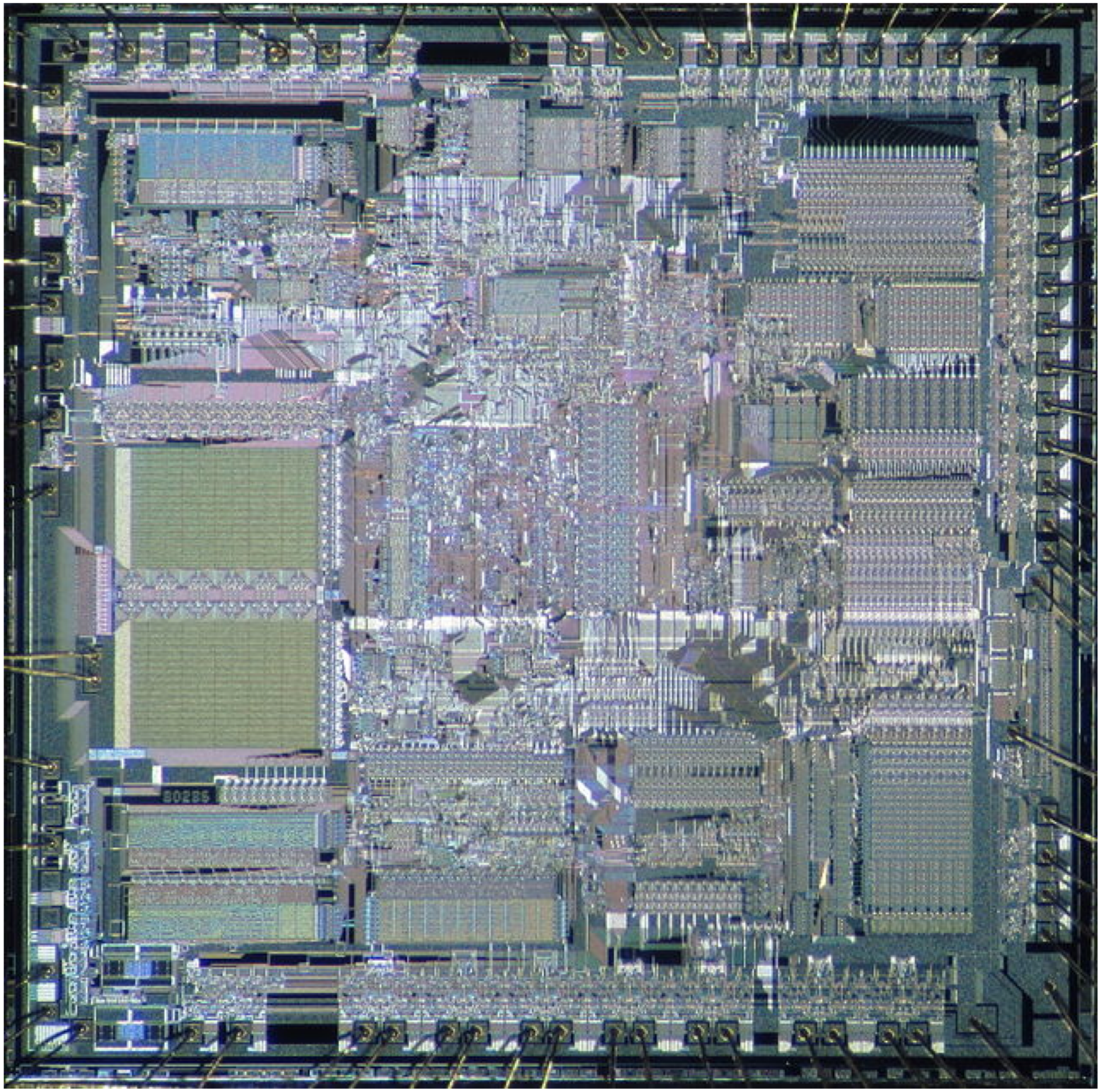
Interrupt Priority -

- Margaret Hamilton: lead Apollo flight software designer.
- Apollo 11 guidance computer became overloaded with interrupts three minutes before first moon landing.
- Her software prioritized interrupts, avoiding abort.



https://en.wikipedia.org/wiki/Margaret_Hamilton_(scientist)

Intel 80286 CPU – IBM PC 1982



134,000 transistors

2.6 million instructions per second

Other Processing Units and Architectures

- MPU (Mathematics Processing Unit)
  - Higher functions (multiply, exponent, trig, log)

- GPU (Graphical Processing Unit)
  - Array processing (many computations in parallel)
  - Display buffer (store image while computing next)
  - Geometric transformations (rotate, translate, scale)
  - Texture mapping (interpolate image onto new grid)
  - Occlusion computation (Z-buffer)
  - GPU now programmed for other purposes using a special array processing language, e.g., CUDA.

Higher-Level Languages

(All are built on Machine Code)

- ASSEMBLER
  - English (ASCII) version of machine code, largely one instruction for each op code.
  - Specific to the particular CPU.
  - Fastest code, if well written.
- COMPILER
  - Abstract language converted by a “compiler” into machine code before running .
  - e.g., Fortran, C, C++.
  - <u>Not</u> machine specific. Different compilers for each type of CPU. Big breakthrough!
  - Generally fast code.

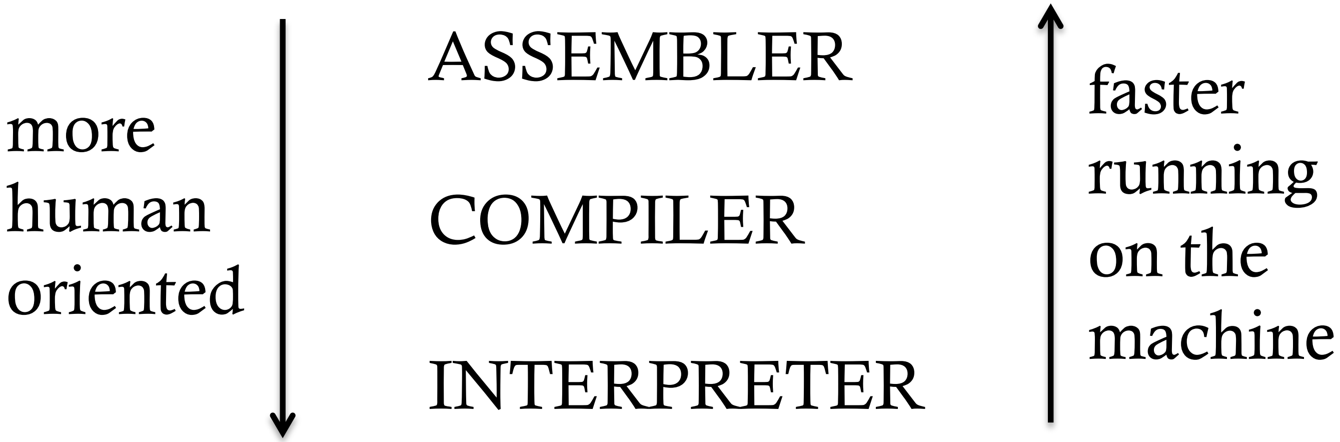
Grace Hopper (Amazing Grace)

- Wanted to program in a human language rather than numbers.
- Conceptualized machine-independent languages
- First compiler 1952,
- Coined the phase “debugging” when a moth was found in her computer
- Rear Admiral, US Navy

http://fivethirtyeight.com/features/the-queen-of-code/

Higher-Level Languages

- INTERPRETER
  - – Program is “Interpreted” at run-time.
  - – e.g. JAVA (first compiled to machine-independent Java Virtual Machine (JVM) byte-codes
  - – Good for web applets (JAVA) where interpreter runs on the local client.
  - – Permits interaction in real time (MATLAB, Python).
  - – Slower than compiled code.



more human oriented

ASSEMBLER

COMPILER

INTERPRETER

faster running on the machine

Object Oriented Programming

- C++, Java, etc.
- Each Class of Object contains public functions and private variables.
- Expedites larger programs by modularizing:
  - Proliferation of global variables avoided.
  - Access and manipulation of variables protected.
  - Inheritance of shared features in classes avoids redundancy in code.
- Application Programming Interface (API)
  - Set of libraries (classes) presenting an interface to underlying systems for the programmer.

- Graphical User Interface (GUI)
  - Replaced ASCII standard input/output in C/C++
  - Microsoft Foundation Classes (MFC)
  - Cocoa for Apple.
  - GUI built into Java
- Embedded systems
  - Stand-alone microprocessors
  - Running in appliances, cars, thermostats, etc.
  - Increasingly includes wireless communications ("Internet of Things")

- Web-based languages
  - – HTML
  - – Java applets
- Server Side Applications
  - – Dynamically generates webpages
  - – Structured Query Language (SQL)
  - – Active Server Pages (ASP)
- “Apps”
  - – iPhones and iPads
  - – Android
- Cloud-based computing
  - – Is this the future?

Scott Guthrie
Executive VP Microsoft
“father” of ASP
Head of Cloud and AI

Graphical Programming

- Program not inputted as ASCII, but rather by dragging and connecting icons.
- e.g., LabView.
- “Easier” for non-programmers.
- Lacks full power (granularity) and exactness of representation of real programming language.

Integrated Development Environment (IDE)

- IDE’s replace the original UNIX Method of a “make” file, listing where all the resources were stored.
- An IDE is an application providing comprehensive facilities for software development
- e.g., Visual Studio (Microsoft), Xcode (Apple) and Eclipse (open source, cross-platform)
- Each IDE stores its environment differently:
  - A recent advance is the open software *CMake*, which can translate a project from one IDE to another.

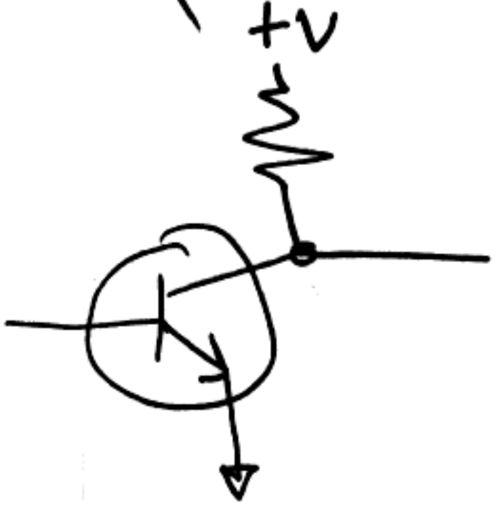
Practical Hardware Considerations

Density of circuity limited by *heat.*

- First solid state computers (1960's)

Resistor Transistor Logic

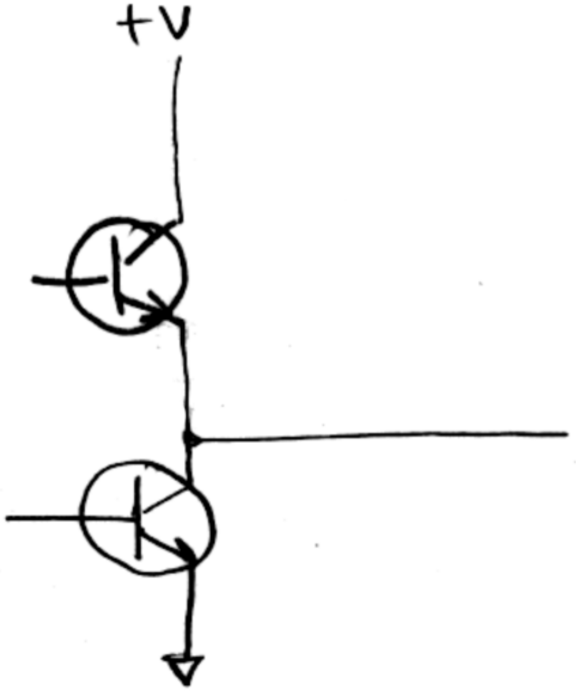
(RTL)



very wasteful
of power.
Discrete transistors

- First integrated circuits (1970's)
- Small-scale integration

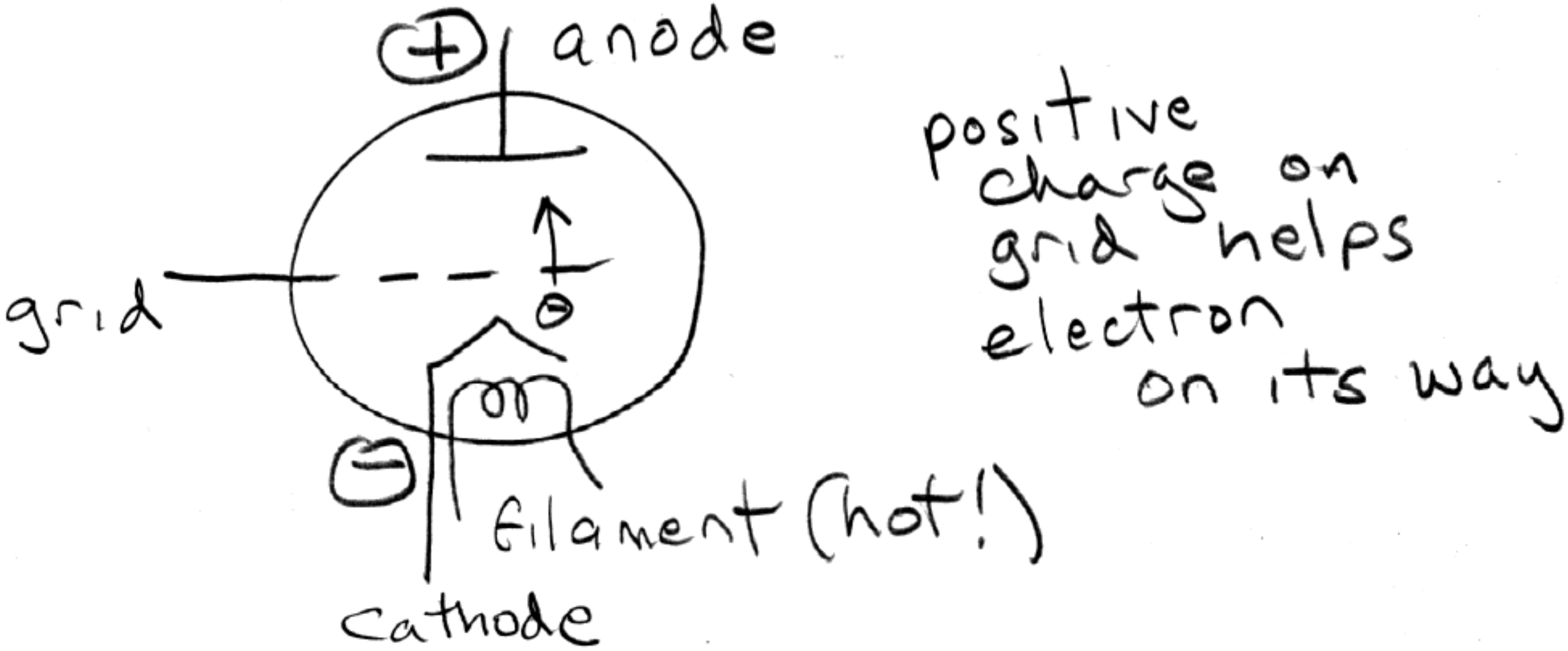
Transistor Transistor Logic (TTL)



More conservative of power

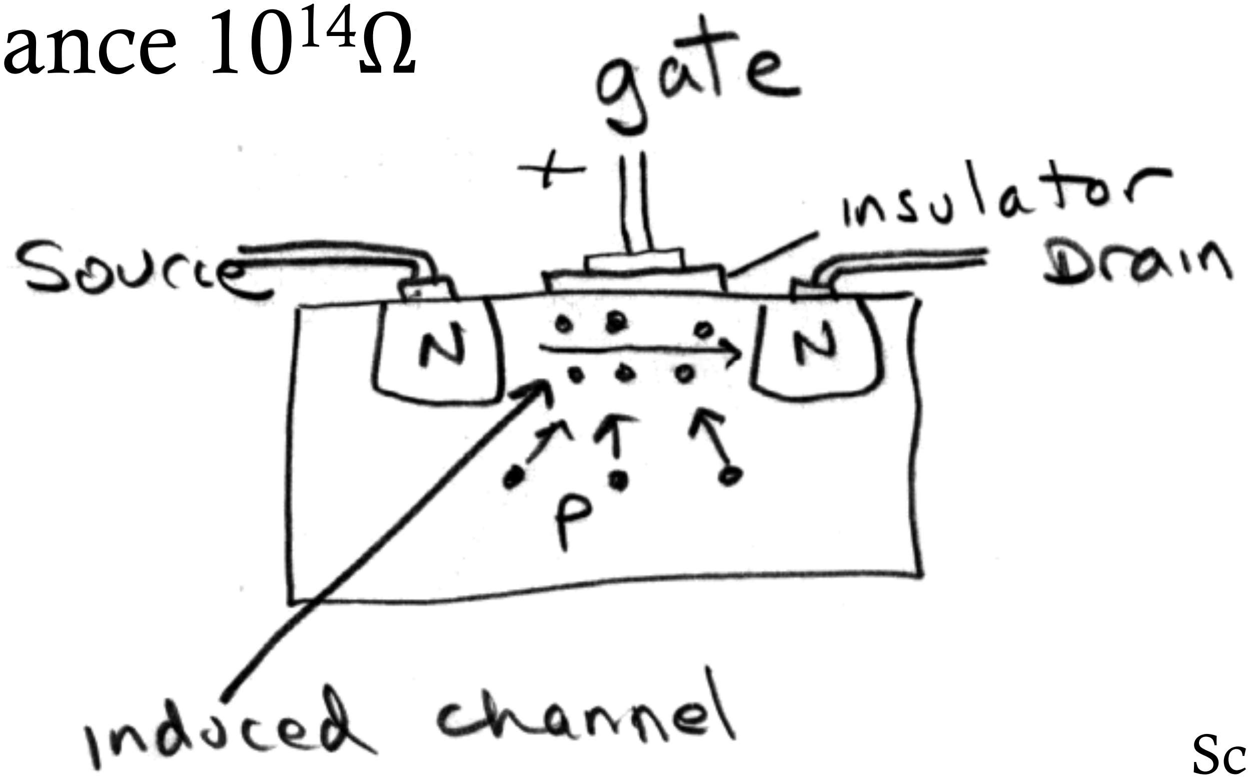
made into integrated circuits

- Bipolar transistors controlled by input current.
- Large-scale integrated circuits had to wait for transistors that needed less current.
- Vacuum tubes use input voltage (rather than current) to control output. Inspired the FET.



Field Effect Transistor (FET)

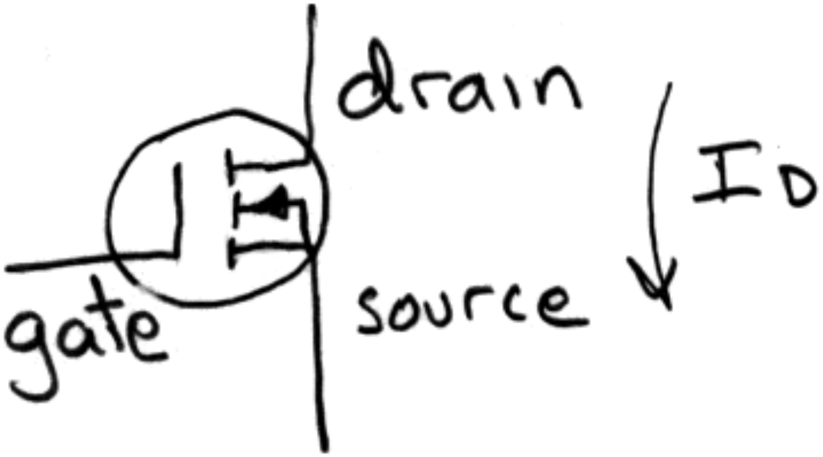
- Metal Oxide Semiconductor FET (MOSFET)
- Voltage at gate controls electron current from source to drain (field effect).
- Insulator (metal oxide) at gate *very* high impedance $10^{14}\Omega$



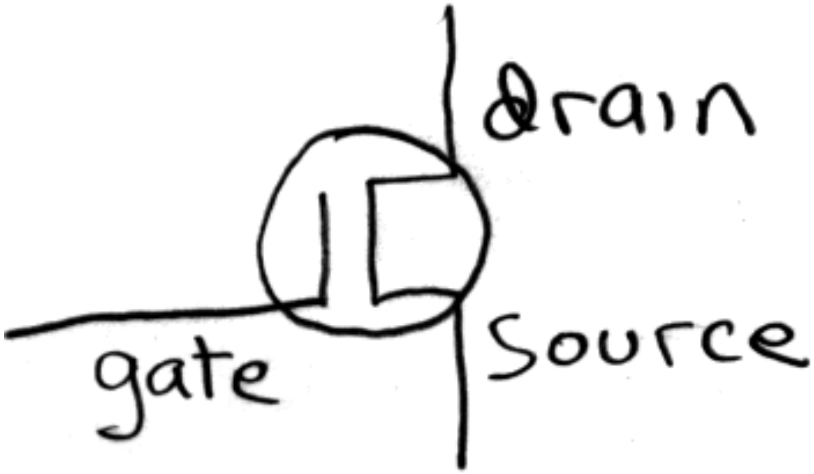
Scherz 4.3.4

MOSFET

- Enhancement vs. depletion mode
- *n* channel vs. *p* channel



Full symbol

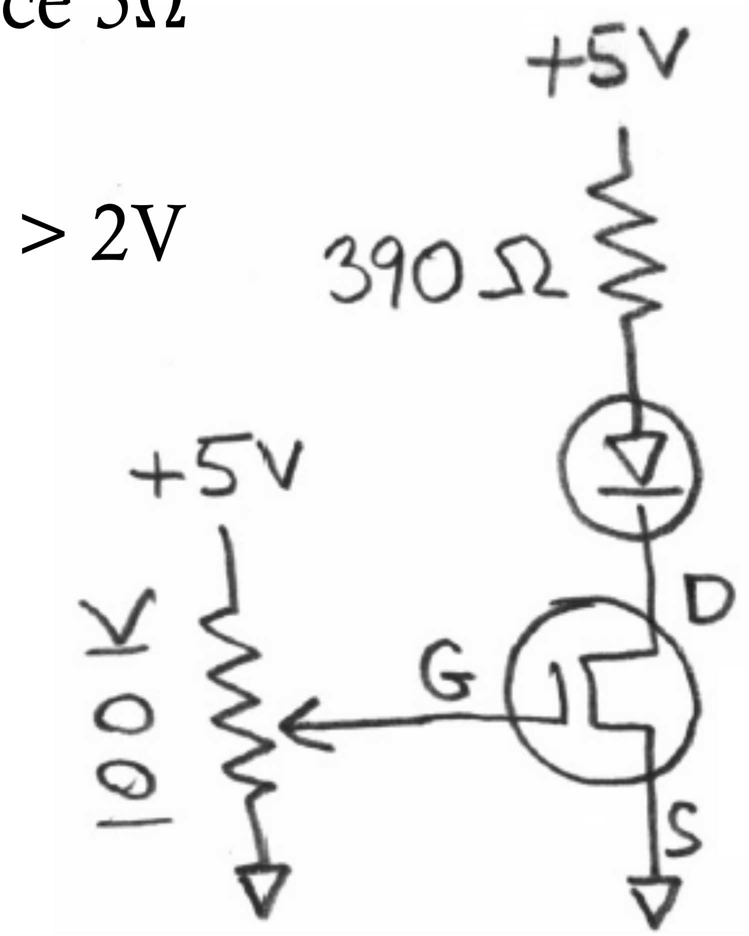


abreviated symbol

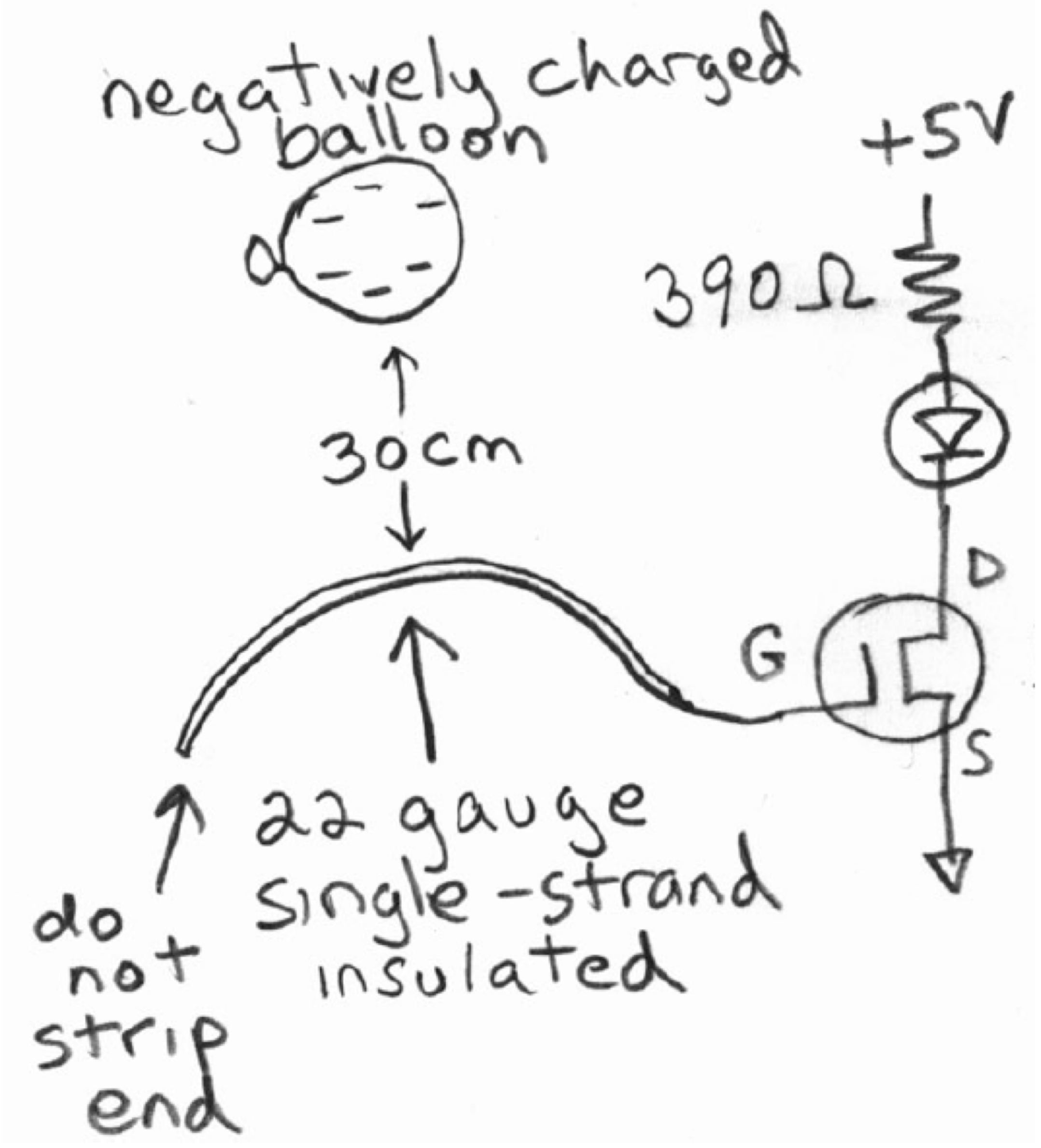
- We will use the BS170 MOSFET

  (not this year)

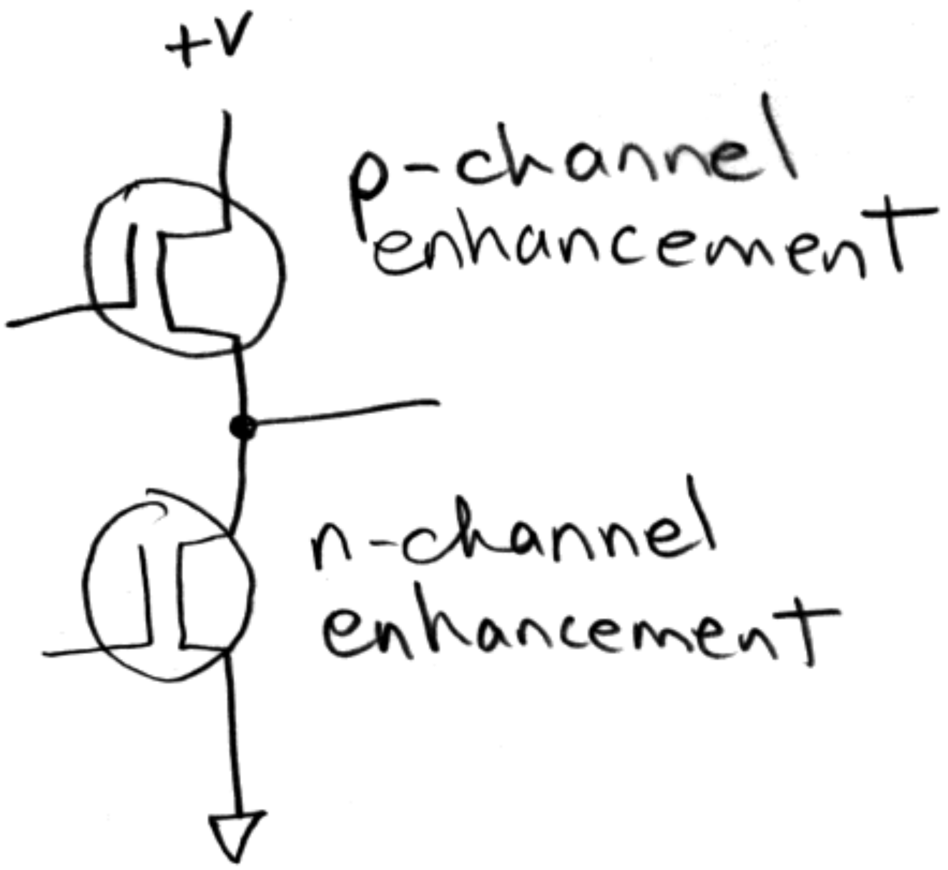
  - *N* channel enhancement mode
  - Drain-source on-resistance 5Ω
  - Max I_D 500mA
  - LED turns on when $V_{GS} > 2V$
  - Gate resistance 10^{14} Ω
  - Gate capacitance 60 pF



- Such high input impedance at gate that static electricity can be sensed
- Balloon pushed charge to the gate end of the wire (breaking the rule that the voltage is the same everywhere on a piece of wire!
- This makes some FETs susceptible to damage from static electricity when touched.



Complementary MOS (CMOS)



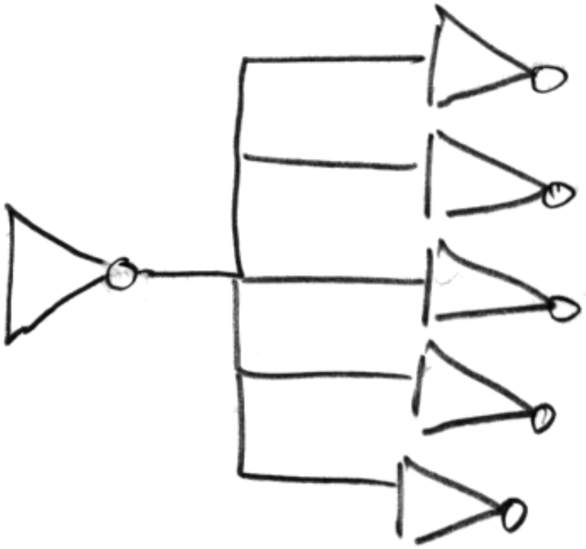
much lower power
hence can pack more circuitry
into a chip for same heat.

Moore’s Law

- # of transistors that can be placed inexpensively on an integrated circuit doubles approximately every 2 years
- Also applies to processing speed, memory capacity, amount of data in the world.
  - 90 percent of the data that now exists in the world has been created in just the last two years.

Gordon Moore
Co-Founder Intel

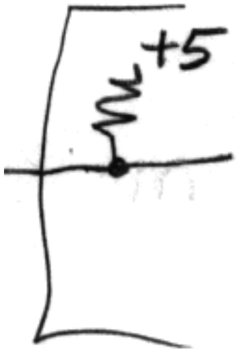
Fan-out



- Digital logic circuits are not perfect:
  $$Z_{\text{out}} \neq 0, Z_{\text{in}} \neq \infty$$
- They are rated as to their fan-out, i.e., how many inputs one output can drive.

Unused inputs and outputs

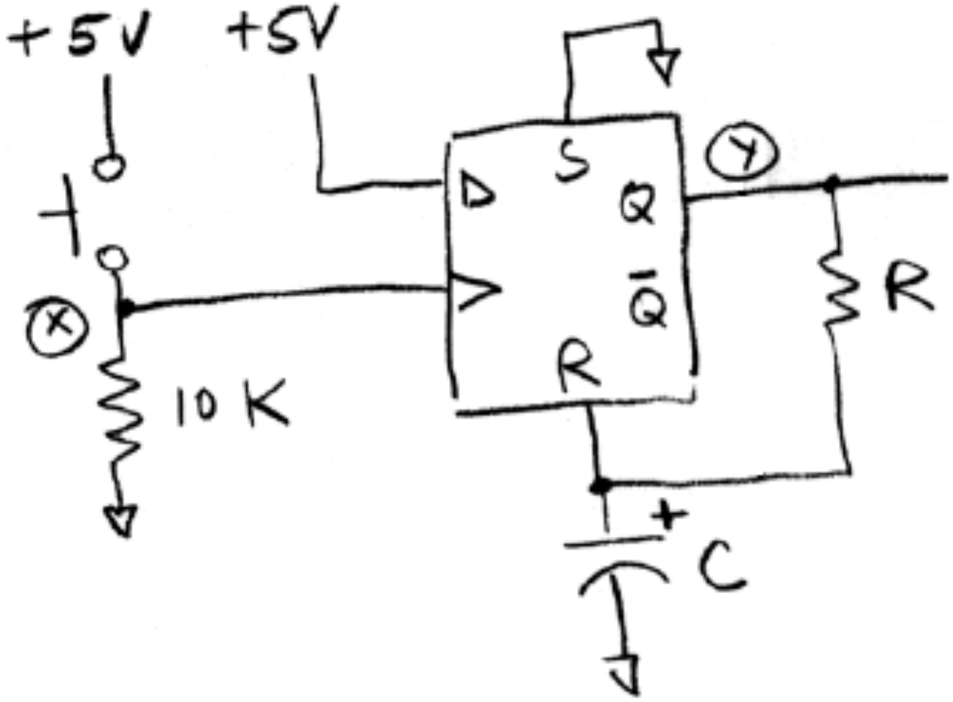
- Digital Inputs
  - Generally have pull-up resistors (float high)
  - Still should tie them high or low, since they are high-impedance and susceptible to noise



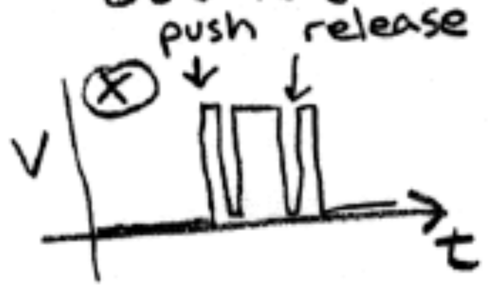
- Digital Outputs
  - Can be left open without risk

Avoiding Contact Bounce

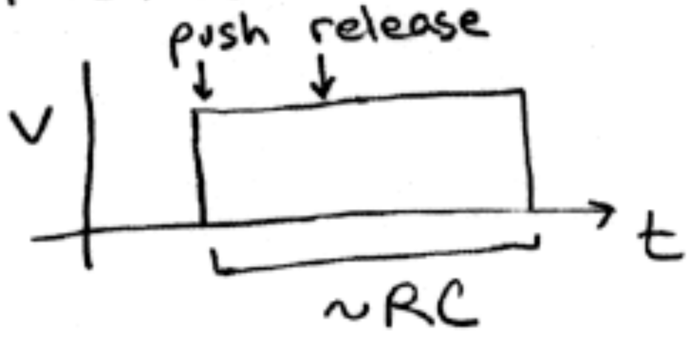
monostable multivibrator – "one shot"



the switch will show contact bounce

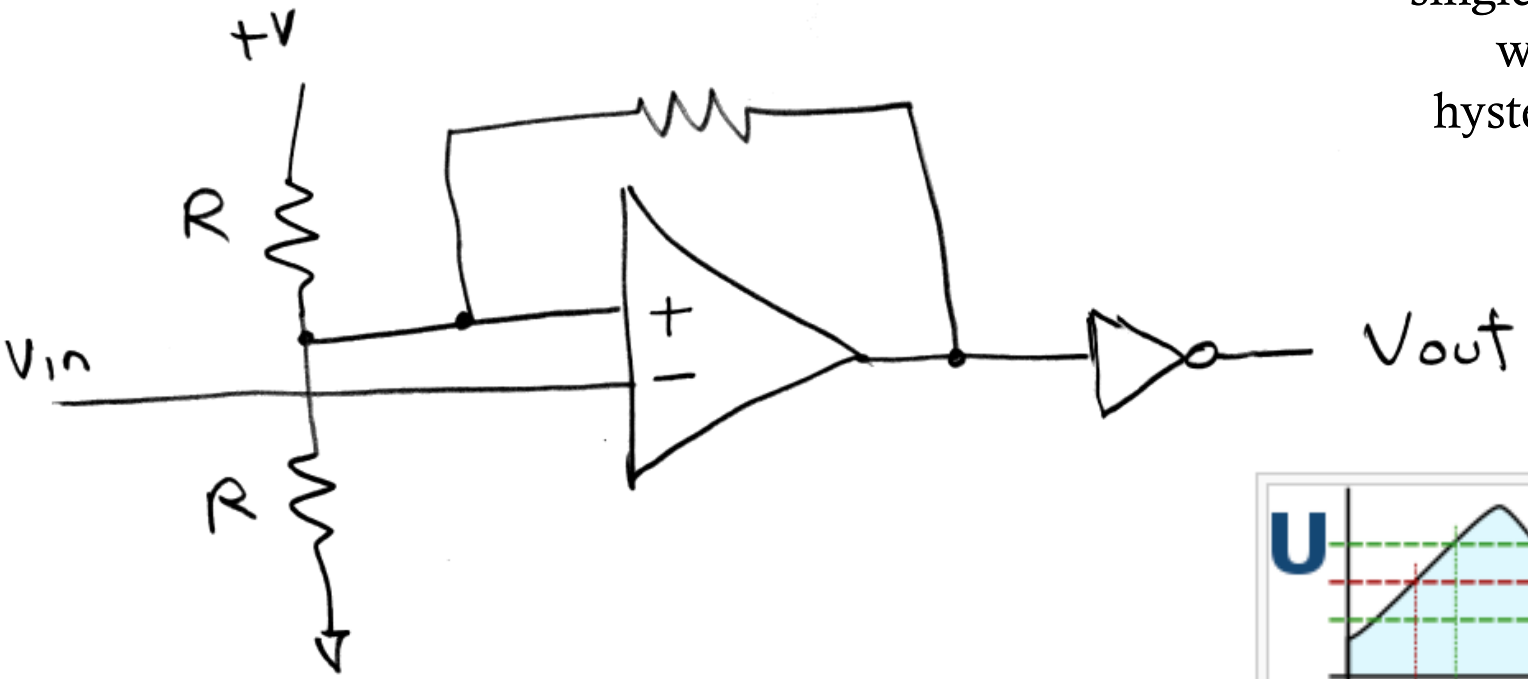


The output, Q, will not show this bounce



Digital circuits and computers are so fast that a mechanical switch will be seen to open and close many times with a single push or release.

Schmitt Trigger

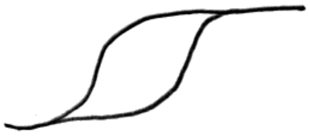


single threshold without hysteresis (A)

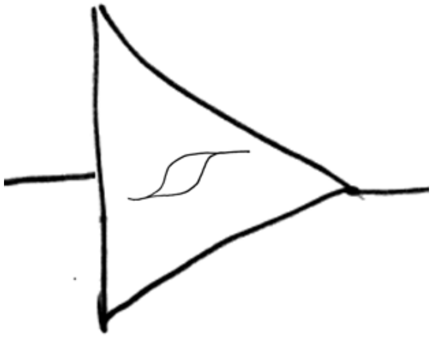
2 thresholds with hysteresis (B)

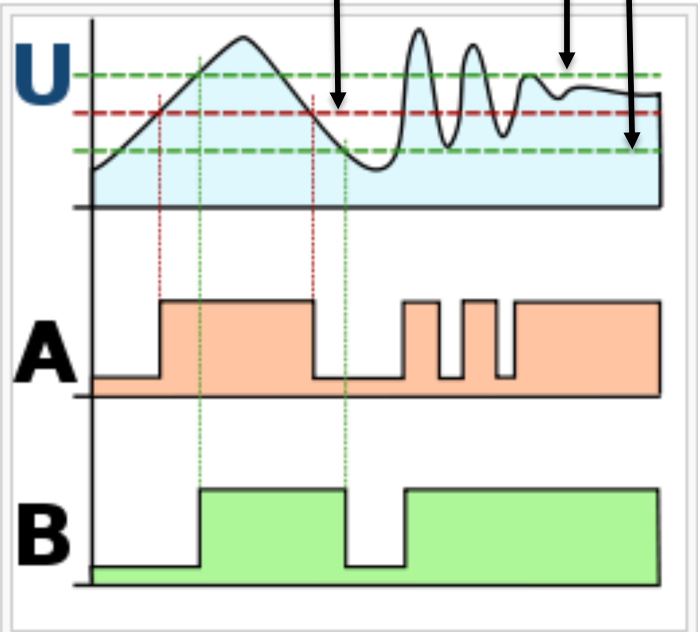
Hysteresis prevents chatter during indeterminate state between 0 and 1

this is the symbol



for The hysteresis

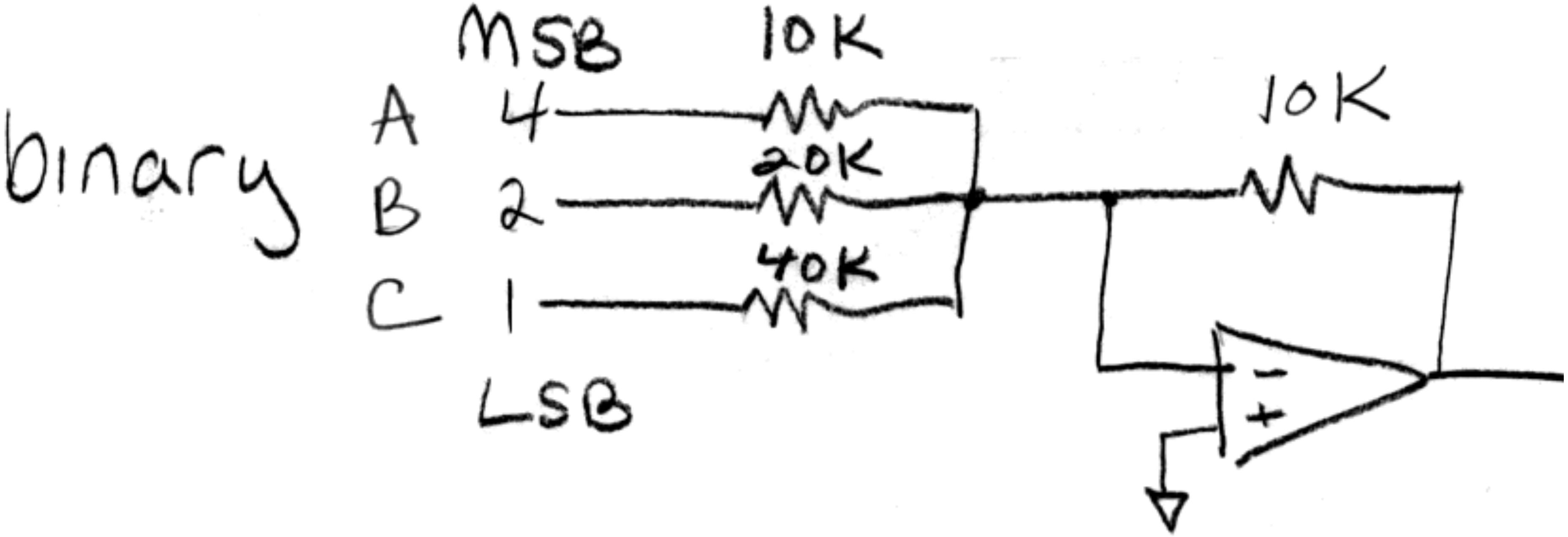




The effect of using a Schmitt trigger (B) instead of a comparator (A).

Wikipedia - Schmitt_trigger

Digital to Analog (D/A) Converter

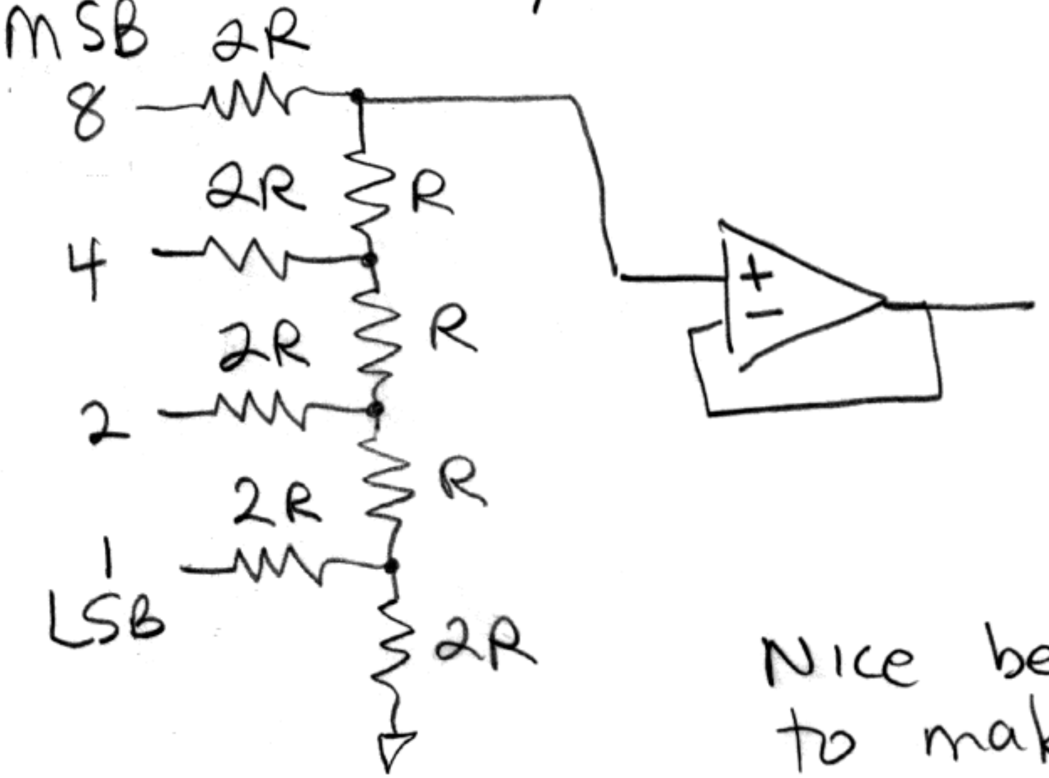


$$V_{out} = -\left(A + \frac{1}{2}B + \frac{1}{4}C\right)$$

- Numbers into voltages, recall inverting adder…
- Resistances go up as power of 2 with bit number, hard to keep them all with same absolute accuracy.

Actual D/A Converter

"R|2R LADDER"

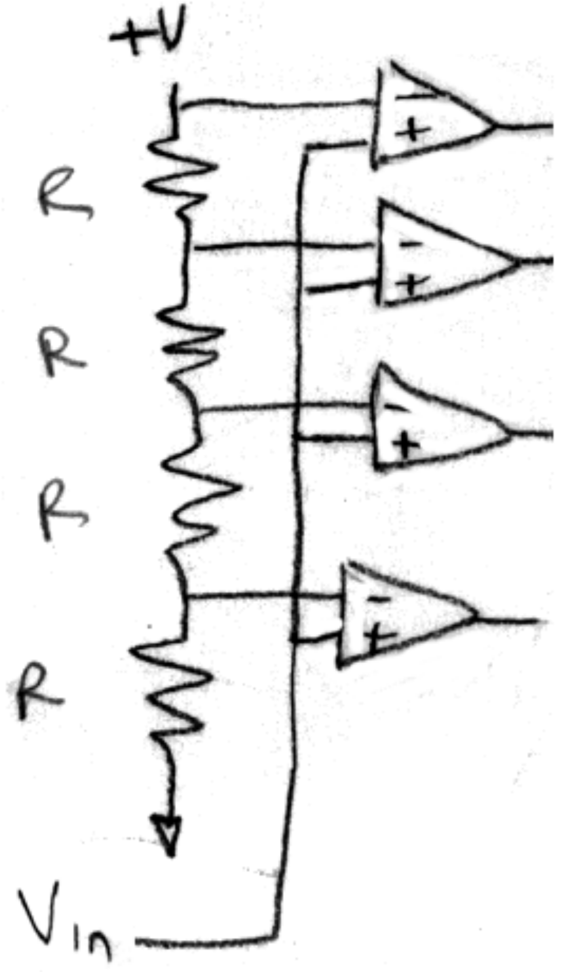


think superposition and see what happens with all other bits set to 0.

Nice because easier to make matching resistors.

- Easier to manufacture, only 2 resistor values.
- Typical D/A converters from 8-24 bit.

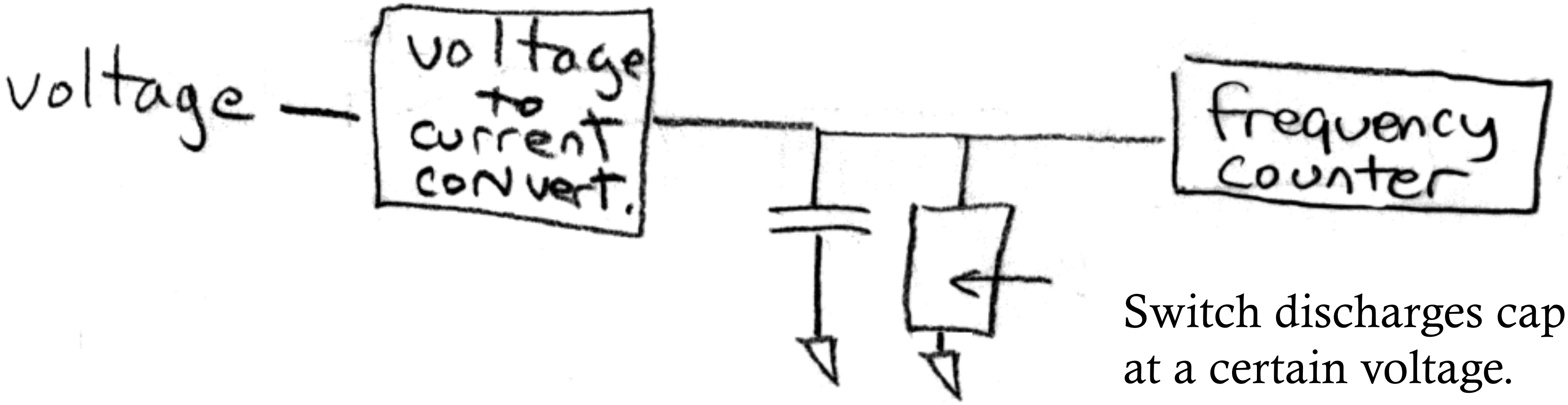
“Flash” A/D converter



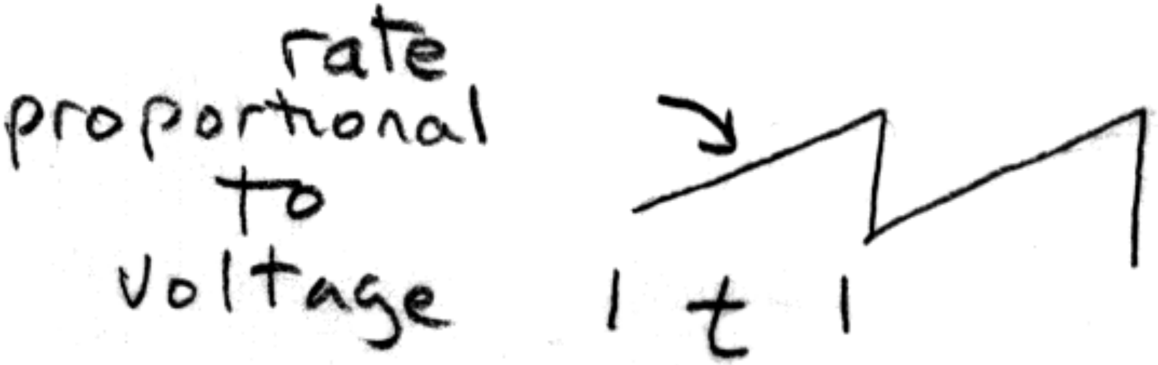
The comparators turn on sequentially from the bottom up, Additional logic required to encode the output as
a binary number.

- Generally fewer bits, lower precision
- Very Fast (video, radio, microwave)

Slow but accurate A/D Converter



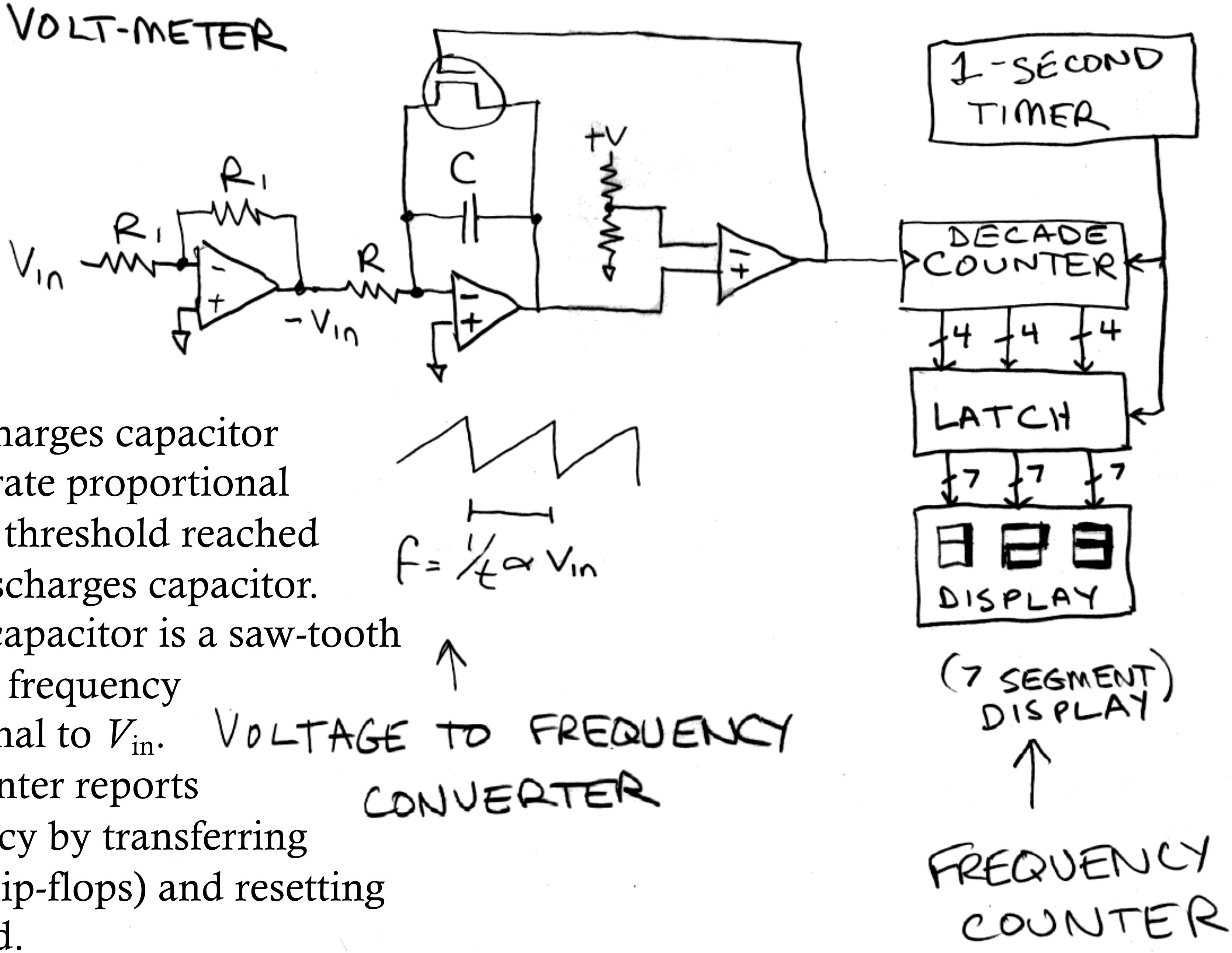
Switch discharges cap at a certain voltage.



$V \propto \frac{1}{t}$

- Linear, high precision, very slow, good for digital volt meters.
- Many ways to make a “voltage to frequency converter” (VFC) also called “voltage controlled oscillator” (VCO).

Your Voltmeter



Integrator charges capacitor at constant rate proportional to V_{in} , until threshold reached and FET discharges capacitor. Voltage on capacitor is a saw-tooth wave whose frequency is proportional to V_{in}. Decade counter reports that frequency by transferring it to latch (flip-flops) and resetting every second.

Dynamic Range

- Determines accuracy (nearness to truth) and precision (ability to discriminate 2 values)
- In analog circuits, the largest possible signal over the "noise floor" generally expressed in dB.
- In digital circuits, the largest number represented by *n* bits over the smallest is $2^{n}-1$
- To quickly estimate dynamic range, learn that 0-10 corresponds to

  1, 2, 4, 8, 16, 32, 64, 128, 256, 512, ~1000
- For example, 24 bits ≈ $16 \times 10^{6} = 36$ dB
- Digital audio is typically 16 bits ≈ $64 \times 10^{3} \approx 24$ dB

Digital Noise

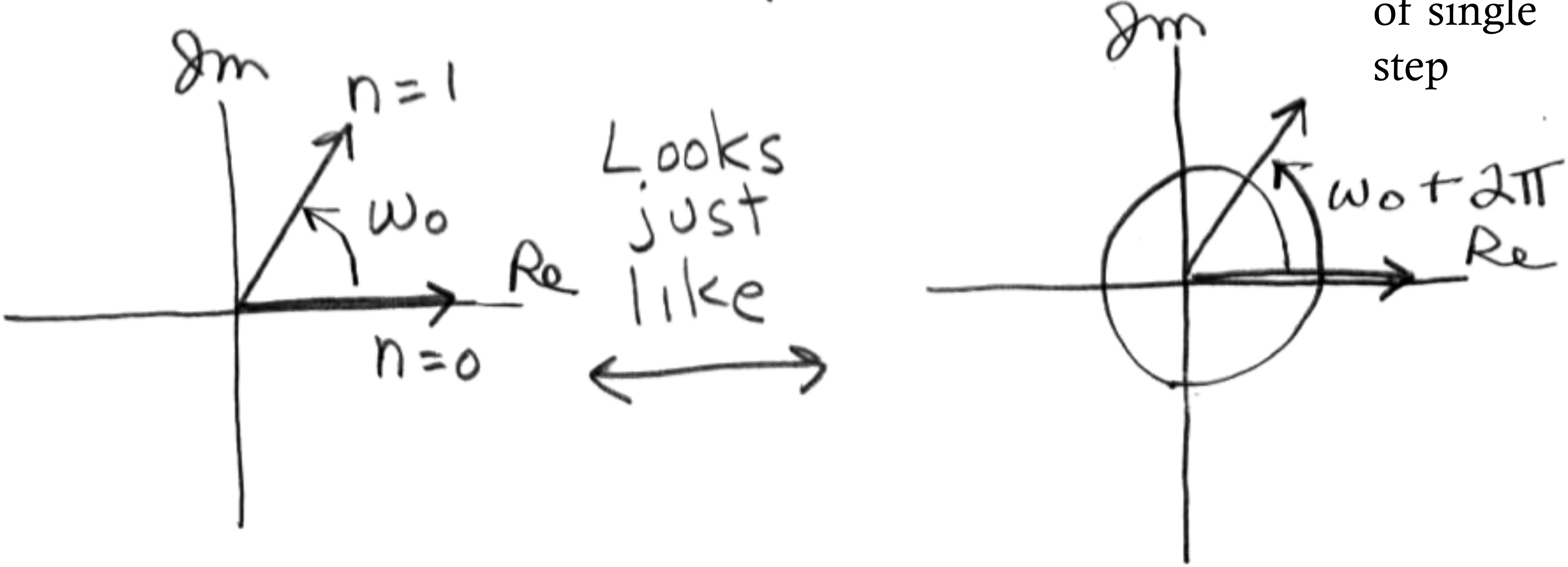
- Noise *does* exist in digital systems, due to a certain error rate in the 1's and 0's.
- Important, especially in transmission and storage of data – a certain percentage of bits are lost.
- Noise can be greatly reduced by using "redundancy" to detect and correct errors.
- e.g., add extra "parity bit", 1 when the number of 1 bits is even, and 0 when odd.
- Internal circuitry of modern computer itself has practically zero errors.

Nyquist Sampling Theorem

for discrete phasor $e^{j\omega_0 n}$

integer number of steps ↓ n

phase of single step → ω_0

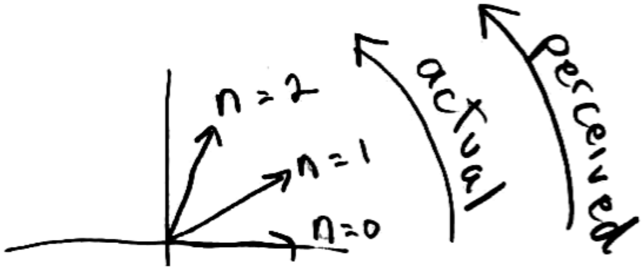


Between one sample and the next, the phasor may have taken any number (and direction) of extra complete revolutions.

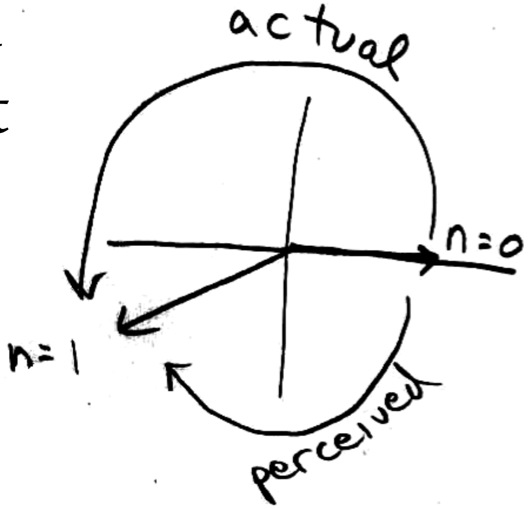
Any frequency above half the sampling frequency will be aliased (appear as a non-existent lower frequency)

Half the sampling frequency is called the “Nyquist frequency”

Analog signals must be filtered to remove frequencies above the Nyquist frequency *before* sampling into a digital form.

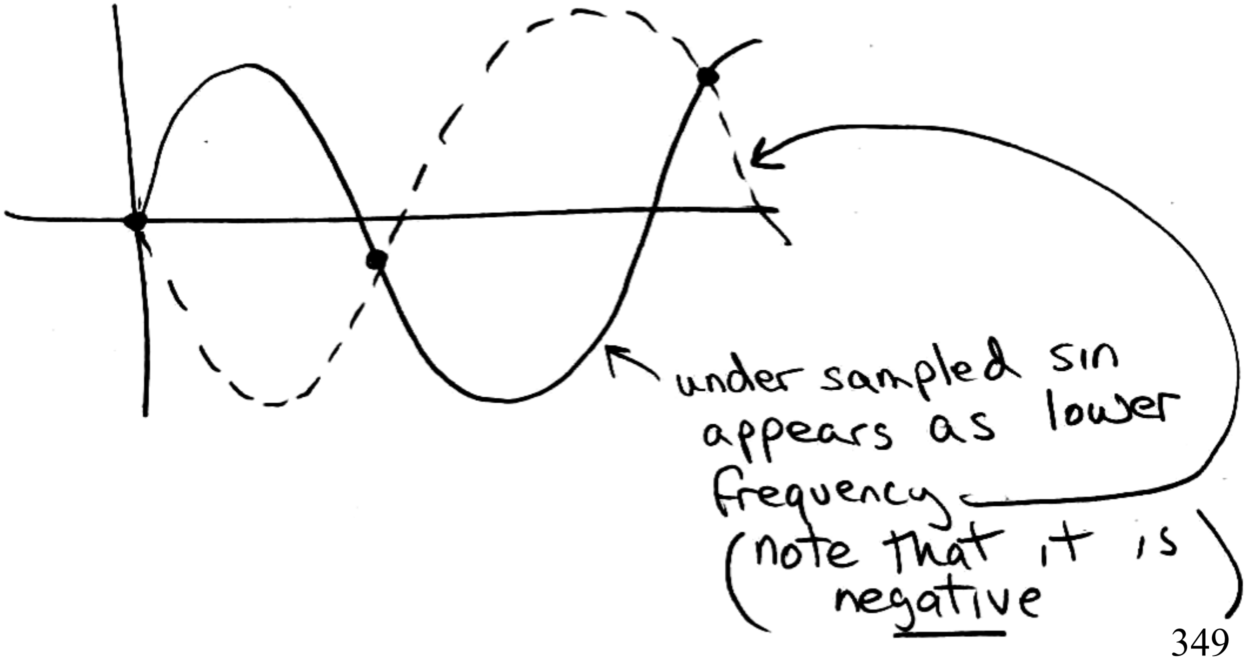


$>2 \frac{\text{samples}}{\text{cycle}}$



$<2 \frac{\text{samples}}{\text{cycle}}$

ALIASING



under sampled sin appears as lower frequency (note that it is <u>negative</u>)

Digital Signal Processors

- Special hardware for rapid signal processing
  - Pipeline architecture for data, *not* a computer (less flexible).
  - Switchable in terms of connections of pipelines and which functions are activated.
  - Real-time processing in Video, Audio, Beam-Forming (Radar, Ultrasound), Data Compression, etc.

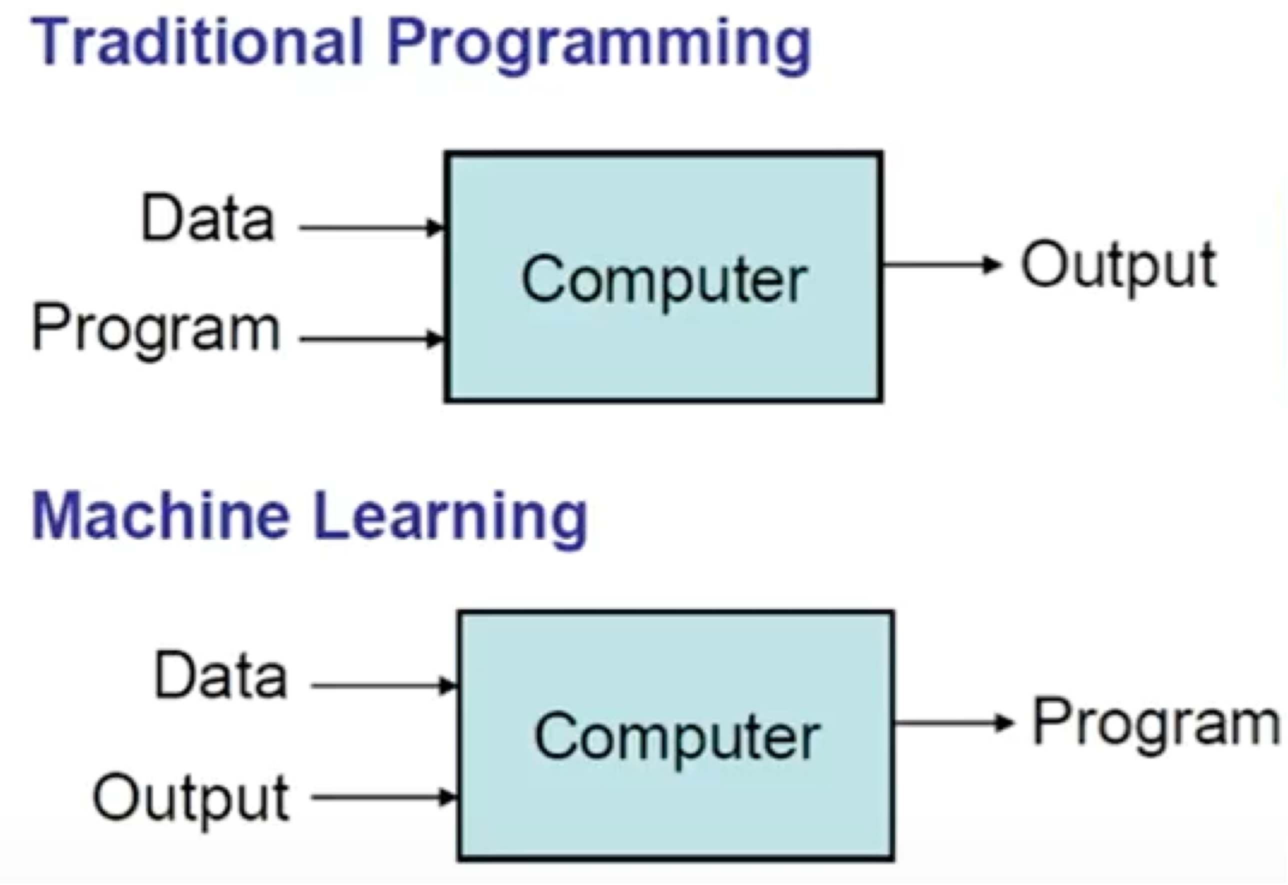
Field Programmable Gate Array (FPGA)

- Designing a custom IC is very expensive
  - only makes sense if large quantities expected, or for research with large funding.
- FPGA is a programmable IC full of gates
  - like a memory, but remembers circuits, not numbers
  - user specifies complex set of interconnected gates
  - burned into a special template IC
  - produces custom IC one at a time

Artificial Intelligence and Machine Learning

- Artificial Intelligence (AI) and Machine Learning
  - *Symbolic AI:* programming *expert systems* (top-down)
    - Peaked in the 1980's with failed promises of translating Russian into English, etc.
    - Based on understanding and simulating complex reasoning processes.
  - *Connectionist AI* or *machine learning* (bottom-up)
    - Has proven more powerful than Symbolic AI.
    - Largely, neural nets.
    - Machine learns on its own.
    - Now what is meant by "*AI.*"
    - Particularly good for pattern recognition, computer vision.
    - Can function in ways not understood by humans.

Machine Learning

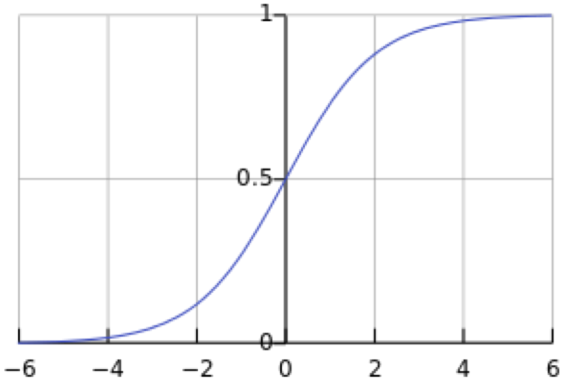


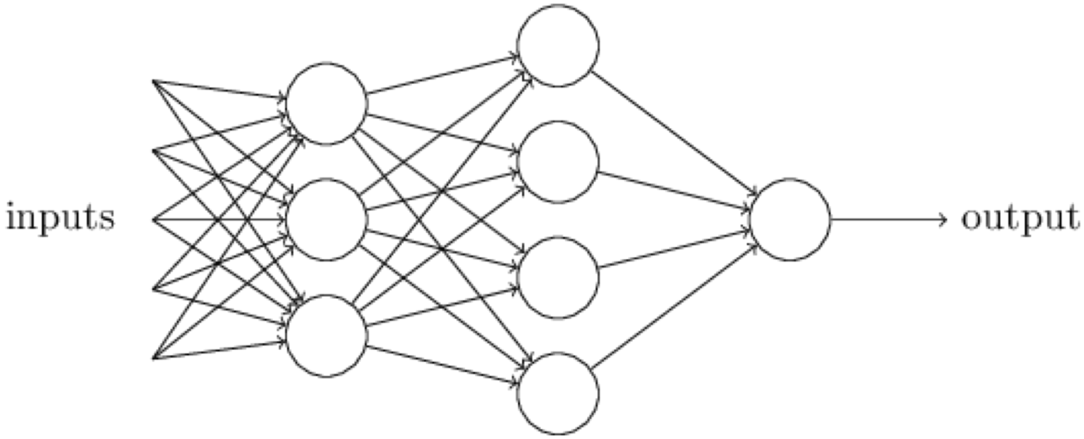
Computer learns to recognize patterns without being given explicit methods.

lecture by Eric Grimson https://www.youtube.com/watch?v=h0e2HAPTGF4

Neural Networks

- Originally called “perceptrons” (analog circuits)
- Non-linear (sigmoid) combinations of inputs.

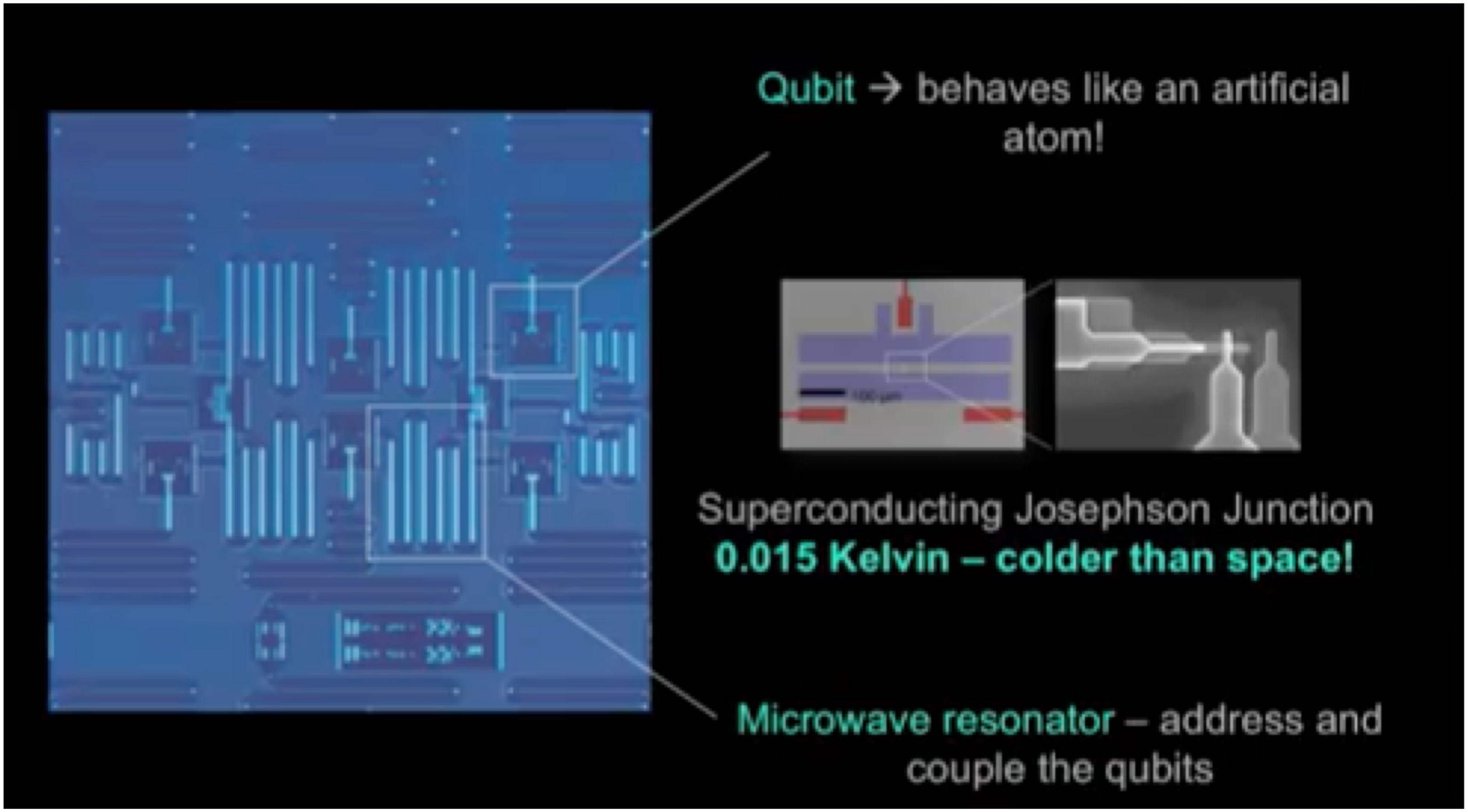




- Trained (optimized) by back-propagation (like Newton’s method).
- NN’s are trained on data sets that are labeled (*supervised learning*) or unlabeled (*unsupervised learning*).
- Convolutional NN’s are used in computer vision, and basically evolve their own kernels.
- NN’s can have many layers, in the case of *Deep Learning*.

Quantum Computing

- N “Qbits” coupled by quantum entanglement can test 2^N possible states simultaneously.



Qubit → behaves like an artificial atom!

Superconducting Josephson Junction

0.015 Kelvin – colder than space!

Microwave resonator – address and couple the qubits

Einstein called quantum entanglement, “Spooky action at a distance.”

https://www.youtube.com/watch?v=S52rxZG-zi0