Fast inverse square root

7 components of a computer (And more)

- Case
- Power supply
- Mother Board
- CPU
- RAM
- (Disk drive) Hard drive
- Graphics processing unit (GPU) Graphics card

Case, Power supply, Mother Board } Simple

CPU, RAM, Hard drive, Graphics card } Complex

Case - Hold everything

Power supply - plugs into Outlet, regulates power

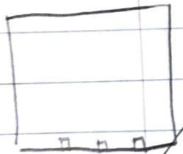
Mother Board - Circuit board, communicates between components
-connects everything

CPU - Central Processing unit, or Processor
- Doesn't store data - everything goes through it
- Reads & arranges calculations
- Programs run from it.

when new window opens, stored in RAM. when closed, freed

Run more/fancier programs

RAM - Random access memory, or memory
- Data/files stored ← currently using, get in and out quickly
- small storage space
- Fast access → long thin sticks



Hard drive - Data/files stored
- Large storage space ← Own but Not currently using
- Slow access → small wire

Have more installed

* CPU takes info out of Hard drive, then stores them in RAM for quick access

Graphics/video card - GPU
- Standalone computer
- what does it look like, or supposed to look like
- own dedicated RAM

Difference between a hard drive and a solid state drive?

Alternative to a graphics/video card is integrated graphics, where it borrows from RAM and is attached to the CPU

-Huge process power
- AI

Dumbdumb tips

PC - personal computer
Desktop - Keyboard & mouse separate
Laptop - all together

Voltage - Force of electrons
Amp - Amount of electrons

Optical drive - reads CDs & DVDs (outdated)

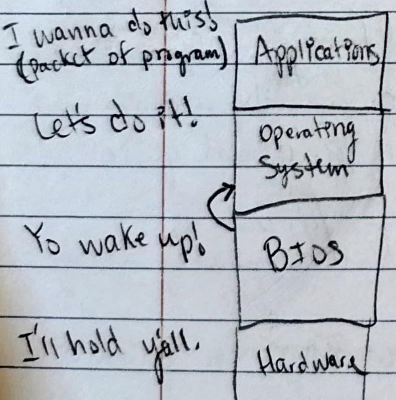
Input/Output devices - sends info in & out
- Mice, keyboards, camera
- Monitors, printer, speaker

Heat sinks - increases airflow away from hot device
Expansion slot - insert more

Networking card - Connects computer to other machines (Printer, Ethernet, Wi-Fi)

Sound card - converts digital to analog (inconsistent waves → sound)
- Sometimes in the motherboard

External connectors ("Ports") - Connects to the outside world (Peripherals)



Input, memory/storage, processing, output

Programs → software
physical parts → hardware

WHAT is BIOS!?

BIOS is a program that tells your operating system what to do. First thing to turn on when press power. (Firmware)

All components plug into - motherboard

CPU - central processing unit

RAM - random access memory (main memory) (Dynamic RAM DRAM) 4 GB

byte = 1/2 GPU - graphics processing unit

$K = 1,000 = 10^3$

$M = \text{mega} = 10^6$ CPU — one peice of info at a time

$G = \text{giga} = 10^9$ GPU — many pixels in parallel

$T = \text{tera} = 10^{12}$ Hard drive - large storage, slow speed 4TB Secondary Mass drive

$\text{Peta} = 10^{15}$ Brain = CPU

$\text{Exa} = 10^{18}$?? SSD = solid state drive → electronic transistors - no moving ??

$m = 10^{-3}$ not I/O - power supply

$\mu = 10^{-6}$ GPU Specialized for pixel display - true

A common cause of slow comp - too little memory

Disk? (Task manager)

Mega Hz? Hertz = measure of frequency, 1 cycle per second

Voltage

Static electricity = unbalaced of electrons

United for electric charge = Coulomb (c) = 6.242×10^{18} electrons

Ampere = 1C/Sec

electron = -1.602×10^{-19} C

proton = $+1.602 \times 10^{-19}$ C

Energy : J → What makes change happen

$1J = 1N \times 1m$ N, measure of force = $1\,kg \cdot m/s^2$

Voltage = measure of electric Potential between 2 points

The amount of work J a C can do?

$1 = V = \frac{1J}{1C}$ (energy/per charge

(holding, wants to go back)

$10V = \frac{10J}{1C}$

$9V = \frac{9J}{1C}$

(-1c) → (+1c)

10 J of electric potential energy

(-1c) → (+1c) (+1c) (+1c)

30 J

$\frac{30J}{3C} = 10V$

Voltage → pressure of electrons

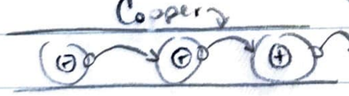
Even if there are no positive particles, the potential between the two points is still 10V.

If a battery 1.5V, then every C of charge does 1.5 J of work.

Electrostatic discharge (ESD) – Sudden flow of electricity neutralizing two staticly charged objects

Current electricity

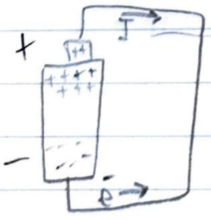
Copper



electric Current

Closed circuit

open → not closed



Coventional current = I
behaves as if there is positive
Charge carriers causing current flow

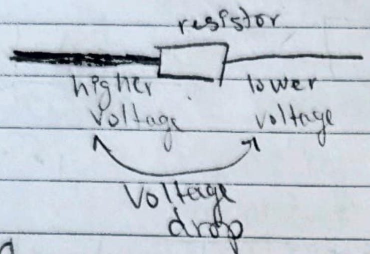
Question: If $1V = \frac{1\,kg \cdot m/s^2 \cdot m}{1C}$, does that mean that 1V = 1C of charge can accelerate 1kg at a rate of $1\,m/s^2$ for 1m?

8 = I · 2

I = 4

Ohm's law

resistor — restricts the flow of electrons



Resistor → electrical energy converted to heat
- toasters
- stoves
- heaters

$5 = I \ 500$

$\frac{5}{500}$ $\frac{1}{100}$

George Ohm

Voltage and current are proportional

$9 = .9 \cdot R$

$R = 9/.9$

V=IR

10 = I 200

5 = 0.5 · 10

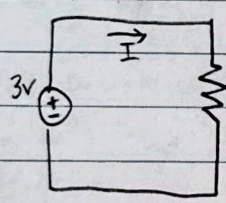
10 = .6

The voltage between 2 points = current flowing × resistance

$V = IR$

Resistance, Ω (ohm, omega)

10 = 10



Red = positive
Black = negative

If 3.03 volts, I = 0.03A
then R = 101 Ω

There is also resistance in the wire and the power supply itself. As resistance increases, current decreases.

Electric power → rate which energy is used

$1W = J/s$

$1V = 1 J/C$

$1A = 1 C/s$

$\frac{900J}{3s} = 300W$

$\frac{900J}{1s} = 900W$

$1V \times 1A = J/C \times C/S = J/S = 1W$

$\boxed{P(\text{power}) = V \times I}$

10 · 10

Ion - charged atom or particle

Grounding — allows safe route of electricity

Binary number system

Bit - Binary Digit
8 bits - byte → represent 256 values, from 0 to 255

8s	4s	2s	1s
2^3	2^2	2^1	2^0

2^4

no current = 0 current = 1

V=IR
=5·2

Hexadecimal

16^3	16^2	16^1	16^0

Represented by 16 characters
- easier to read
- Separate binary into bits of 4 then translate

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

bit [0/1]
nibble 4 bits
byte 8 bits

Binary to hexadecimal
- split into groups of 4
- find matching hex. sets (use table)

Decimal to hexadecimal
- divide by 16
- remainder in hex. form
- divide by 16 until result is 0

ex; 1128_{10} to X_{16}

1. take 1128 / greatest power of 16 possible
$1128/16^2 = 4.40625$ 3 place in hex
2. take remainder, convert back to dec., then do again in the next smallest power
$.40625 \cdot 16^2 = 104$ $104/16 = 6.5$
3. Do until can't. $.5 \cdot 16 = 8$ → 468_{16}

16^1 16^0
E7
160 ones

E sixteens + 7 ones
$\underbrace{14 \cdot 16}_{224} + \underbrace{7 \cdot 1}_{7} = 231$

10101 111
A F

7 C 3 F 9
16^4 16^3 16^2 16 0
$7 \cdot 16^4 + 12 \cdot 16^3 + 3 \cdot 16^2 + 16$
508688

1 0 0 0 1 0 1 1
128 64 32 16 8 4 2 1
32 + 8 + 2 + 1
43

0001,1100,1110
1 C E

3 C 1

64 + 4 + 2 + 1
71

0011, 1100, 0001

A.S.C.I.I.

"ask ee"

American Standard Code for Information Interchange

numbers = text

Letters, numbers, symbols — characters

A ⇄ 065   ! ⇄ 033

ASCII is a subset of unicode

Micro transistors → on or off (binary)

Single transistor = bit

transistor on = 1

off = 0

Spelling in binary → ASCII

How hard drives work

how - magnents

ferromagnetic material (can be magnetically charged)

- full of tiny magnetic region

↳ positive charge → 1

negative → 0

- arm sends tiny charges to the disk and reads the ones already there (reading & writing info)

How does it know where to find it? (cache + file Allocation Table)

- file allocation table

regions

↳ tracks

↳ Sectors

↳ clusters (multiple sectors)

Actuary arm → Checks file allocation table, then goes there

track 13,
Cluster 52

Cache - ~~more~~ way precise map
- faster loading time
- Hard drive, RAM, or CPU
- info very quickly & very often

Why store data?
what lack in speed, make up in space

(Solid State - made up of transistors) → Programs
↳ 500 GB ↳ operating system
Disk drive → Data
↳ 3000 GB

How does Blu-ray work
— to read → lasers
pits & flat areas
- pits → 0
- flats → 1 (a pit no)
when laser hits flat area, bounces off & hits a photoelectric cell which detects the light
- uses blue lasers → Blu-ray!
↳ smaller wavelegth → more can be read

Questions for the T.A.

What is a Volt?!

- "water" pressure — electric potential
- J/c
- = IR

• What is it used for? If it's not a measurement of power, why do we care?

• What does it mean to measure volts across a resistor? Not at a point? → water pressure?

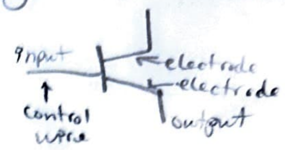
• What is a volt vs. a Watt?

Can you calculate resistance + amperes across a whole circuit or only at a point? (Demo)

If $V = \frac{1J}{1C}$,

then $V = \frac{N \cdot 1m}{1C} = \frac{1kg \cdot m/s^2 \cdot 1m}{1C}$?

Regular transistor:



Boolean logic & logic gates

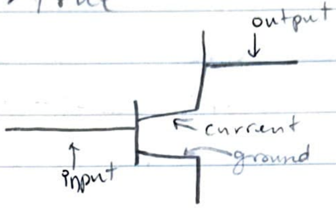
Boolean algebra → values of variables are true & false.

NOT, AND, OR

Not : True → false
False → true

A = A' = "Not A"

Not gate:



if input is on, output doesn't recieve current

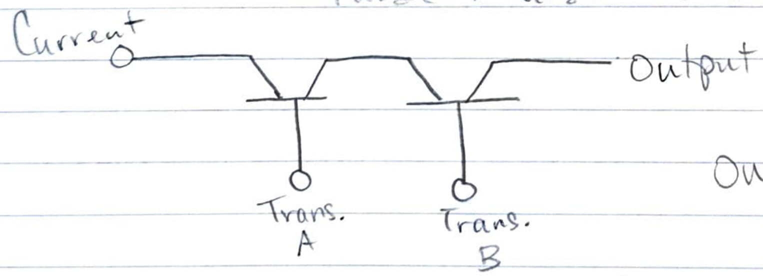
if input is off, output on

x = AB → "x = A and B"

multiply

X *

AND : True + true = true
True + false = false
~~F~~alse + true = false
False + False = false

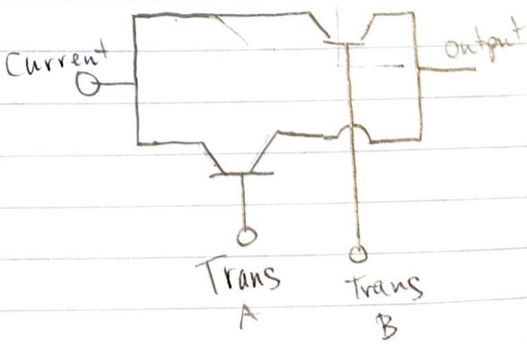


Output only if both are on

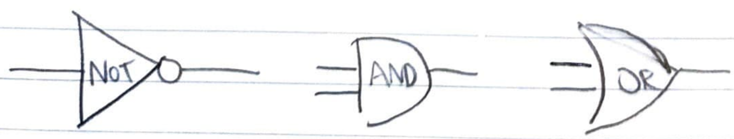
+ add

X = A+B+C → "x = A or B or C"

OR : T + T = T
T + F = T
T + F = T
F + T = T
F + F = F



If one or the other is on, the output is on.

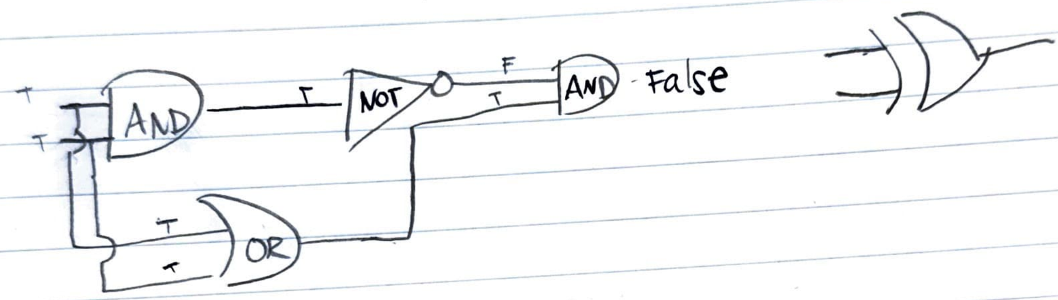


(exclusive or) $X = A \oplus B$ = "A exclusive or B"

XOR

Only true if one is true + one is false

T + T = F
T + F = T
F + T = T
F + F = F



1 → true
0 → false

AND

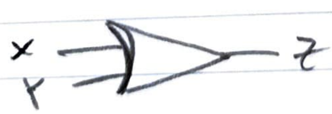
X Y	$Z = X \cdot Y$
0 0	0
0 1	0
1 0	0
1 1	1

OR

X Y	$Z = X + Y$
0 0	0
0 1	1
1 0	1
1 1	1

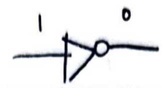
NOT

X	$Z = \bar{X}$
0	1
1	0

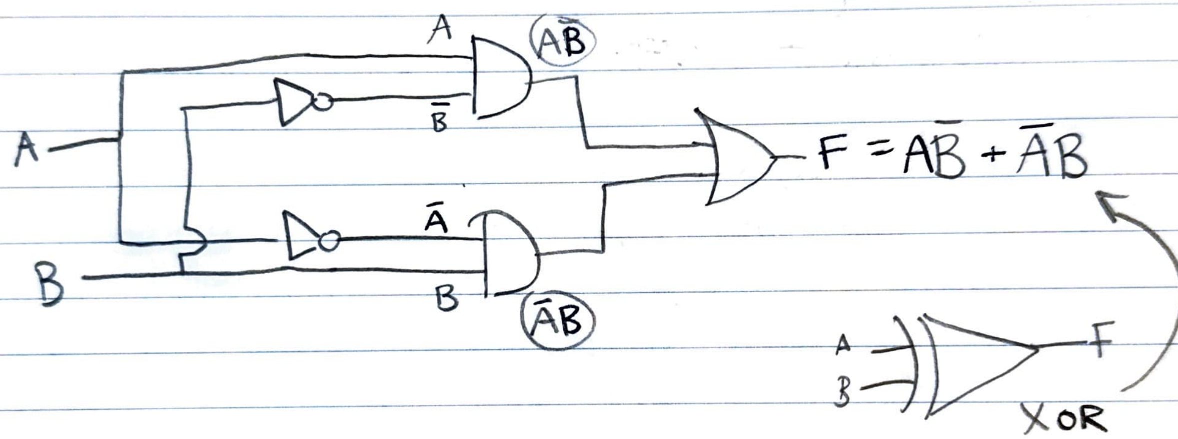


"A not"

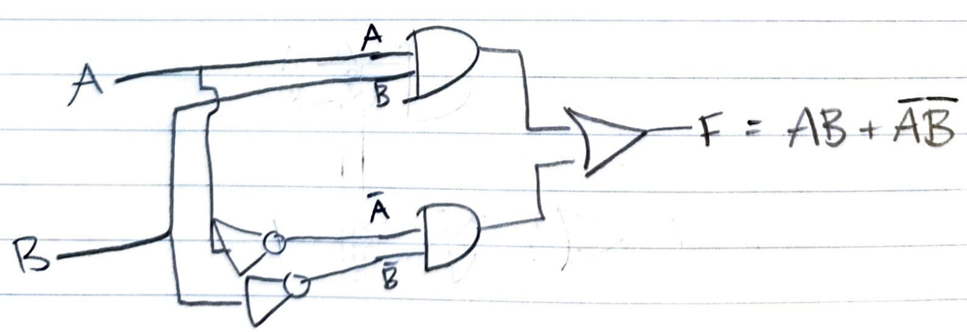
A A' $\bar{A}$



	And	OR	Not	
000	0	0	0	1
001	0	1	1	0
010	0	1		
011	0	1		
100	0	1		
101	0	1		
110	0	1		
111	1	1		

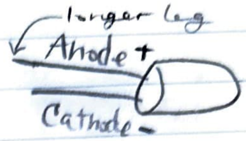


$F = AB + \bar{A}\bar{B}$



Resistors → check table for Ω value + accuracy

LED → Light-Emitting Diode
→ NEED a resistor!

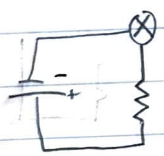


How much resistance?

- Component LED
- Voltage 9v
- Forward voltage → manufacturer ex; 3.2 v
  - how many volts the LED will use up
- Amps → manufacturer ex; 0.024A

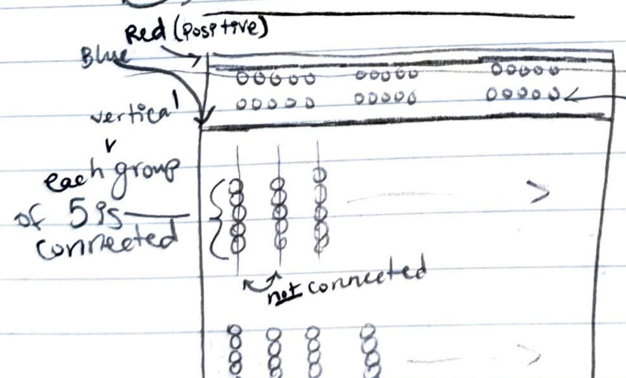
R = (Power source − fwd voltage) / Amps

ex; R = (9 − 3.2)/.024 = 240 Ω

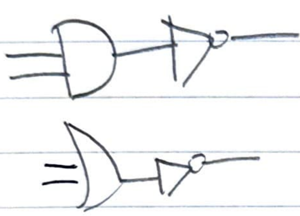


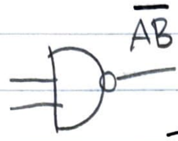
Question: does it matter which side the resistor is on?

Breadboards!



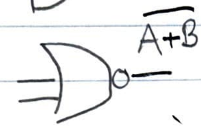
Multimeter - measures volts, resistance, and amps.
- auto-ranging - don't have to put in a range
OL → open leads (not connected)





"Nand gate" (opposite and)

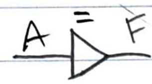
11	0
10	1
01	1
00	1



"Nor"

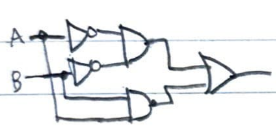
11	0
10	0
01	0
00	1

(opposite or)



Buffer →

0	0
1	1

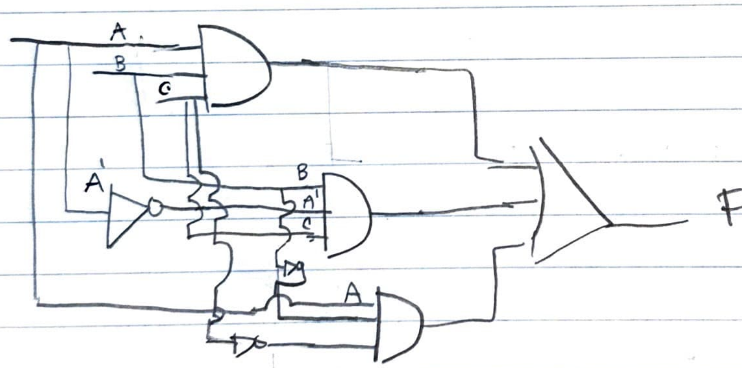


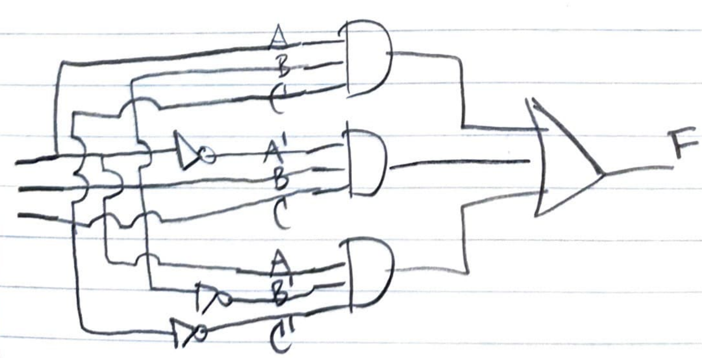
$F = \overline{A \oplus B}$ → Xnor

11	1
10	0
01	0
00	1

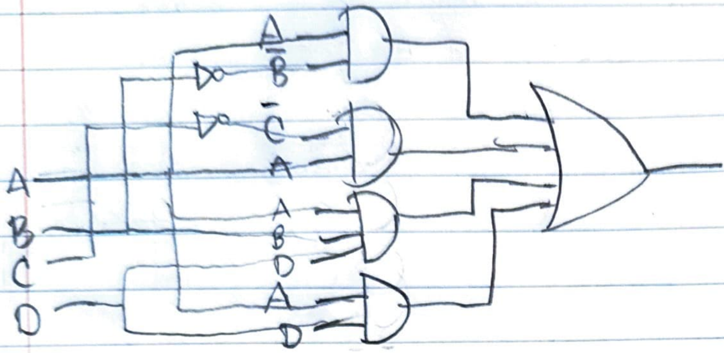
True only if both inputs are the same

$F = ABC + A'BC + AB'C'$





$F = A\bar{B} + \bar{C}A + ABD + A\bar{D}$



Truth table to logic equation

ABC	F
000	1
001	0
010	0
011	1
100	0
101	1
110	0
111	0

$F = \bar{A}\bar{B}\bar{C} + \bar{A}BC + A\bar{B}C$

000 + 011 + 101

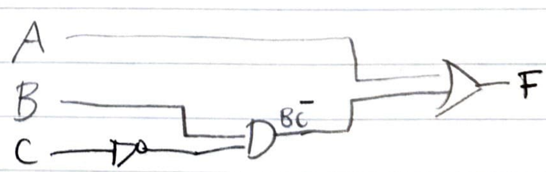
Logic equation to truth table

$F = \bar{A}\bar{B}C + \bar{A}BC + AB\bar{C} + ABC$

ABC	F
000	0
001	1
010	0
011	1
100	0
101	0
110	1
111	1

$F = A + B\bar{C}$

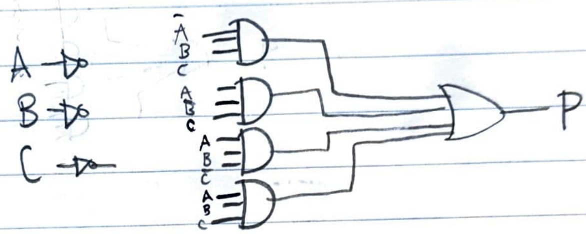
	ABC	F
	000	0
	001	0
	010	1
	011	0
A is 1	100	1
	101	1
	110	1
	111	1



Story Problems: Three board members vote on corporate policies. Their names are Adams (A) Baker (B) and Cheney (C). If two or more of them vote yes, the policy (P) is implemented. Create the Boolean equation.

A	B	C	P
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$P = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$



Fluid level, Temperature, pressure. Alarm sounds if T and P are both too high, or L is wrong.

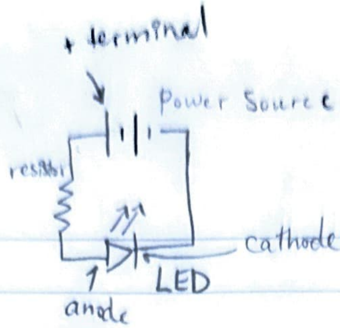
L	T	P	A
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

$A = \bar{L}TP + L\bar{T}\bar{P} + L\bar{T}P + LT\bar{P} + LTP$

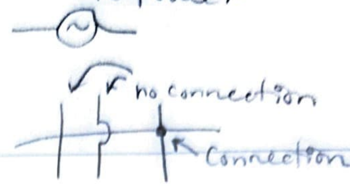
simplify $A = TP + L$

What is open source?

Schematics



AC power

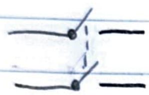


Switches

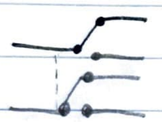
toggle switches, slide switch, dipswitches

SPST Switch (single pole, single throw)

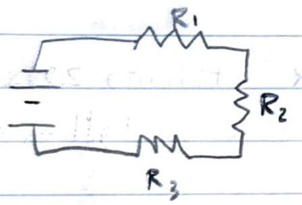
SPDT (single pole, double throw)

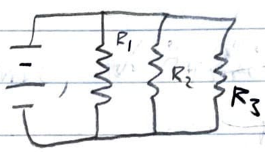


DPST (Double pole, single throw)



DPDT (Double pole, double throw)





$V = IR$

Series	Parallel
I same	I differ
V differ	V same
$R_T = R_1 + R_2 + R_3 \ldots$	$\frac{1}{R_T} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} \ldots$
add R, $R_T \uparrow$ $I \downarrow$	add R, $R_T \downarrow$ $I \uparrow$

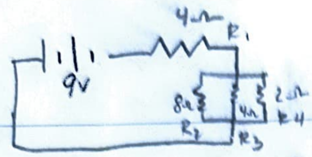
$$\frac{1}{R_T} = \frac{1}{8} + \frac{1}{4} + \frac{1}{2} = \frac{7}{8} \quad R_T = \frac{8}{7}$$

$$I = \frac{V}{8/7} = 9 \cdot \frac{7}{8} = 7.875\ A$$

$I_1 = V/R = 9/8\ A$

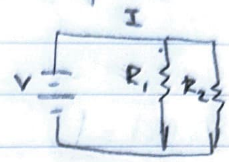
$I_2 = 9/4\ A$

$I_3 = 9/2\ A$



$R_T = \frac{8}{7}\Omega + 4\Omega = 4\frac{8}{7}\Omega$

proof:



$I_S = V/R = \frac{9}{4\frac{8}{7}\Omega} = 1.75\,A$

$V_{R_1} = (1.75)(4) = 7v$

So only 2v going through parallel part

$V = IR$
$I = V/R$

$I_2 = \frac{2}{8} = \frac{1}{4}\,A \qquad I_P = \frac{2}{\frac{8}{7}} = \frac{14}{8} = 1.75A$

$I_3 = \frac{2}{4} = \frac{1}{2}\,A$

$I_4 = \frac{2}{2} = 1A$

$R_1 = \frac{V}{I_1} \quad R_2 = \frac{V}{R_2}$

$I = I_1 + I_2$

$= \frac{V}{R_1} + \frac{V}{R_2}$

$I = V\left(\frac{1}{R_1} + \frac{1}{R_2}\right)$

$\frac{I}{V} = \frac{1}{R_1} + \frac{1}{R_2}$

$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2}$

$V = IR$

$V = 5 \quad R = 660$

Analog measures voltage from Ground to analog pin.

$5 = I \cdot (330 + R)$

$3.03 = I \cdot 330$

$V = IR$

$5 = I\,220$

Adding Base-2

Magic table
0+0=0
0+1=1
1+0=1
1+1=10

1+1+1=11

```
   10101
+  11101
  110010
```

```
  110
+ 101
 1011
```

```
 10101
 11101
     0
```

```
101101
101100
```

Subtract Binary

Magic table

1-1=0
0-0=0
1-0=1
0-1=1 * Carry required 10-1=1

```
  10111
- 10101
  00010
```

```
  10011
- 01101
  00110
```

```
  110
- 101
  001
```

```
  110010
-  10101
  100101
```

"Two's complement" → represent negative numbers

```
  10011
- 01101
```

→ Invert bottom → add 1 →

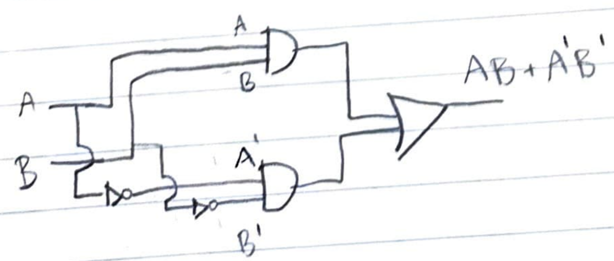
```
- 10011
  10011
```

→ add →

```
   10011
 + 10011
   00110
```

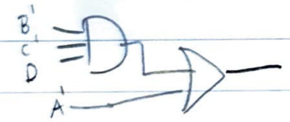
throw remainder away

↑ if 0, positive
if 1, negative



$F = A' + B'C'D$

ABCD	F	G
0000	1	0
0001	1	0
0010	1	0
0011	1	0
0100	1	0
0101	1	0
0110	1	0
0111	1	0
1000	0	1
1001	1	1
1010	0	1
1011	0	1
1100	0	1
1101	0	1
1110	0	1
1111	0	0



1011 1101 1110

$AB' + AC' + AD' = G$

A → pink
B → paper
C → jammed

Alarm = 2 or more are true

$AB + AC + BC = F$

ABC	F
000	0
001	0
010	0
011	1
100	0
101	1
110	1
111	1

Half-adder and full adder?

ALU - Arithmetic logic unit

0 + 0 + 0 = 1
0 + 0 + 1 = 1
0 + 1 + 0 = 1
0 + 1 + 1 = 0 carry 1
1 + 0 + 0 = 1
1 + 0 + 1 = 0 carry 1
1 + 1 + 0 = 0 carry 1
1 + 1 + 1 = 1 carry 1

What it started with before adding

110 ← carry in (C_{in})

  01
  11

 100

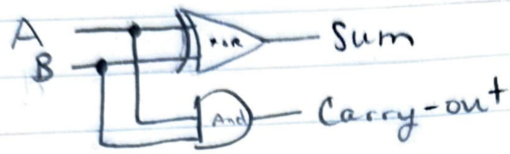
011 ← carry out (C_{out})

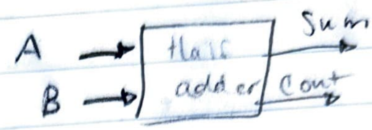
What was left after adding

Ones complement

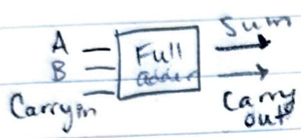
1001 → 0110

Half-adder - no carry-in number



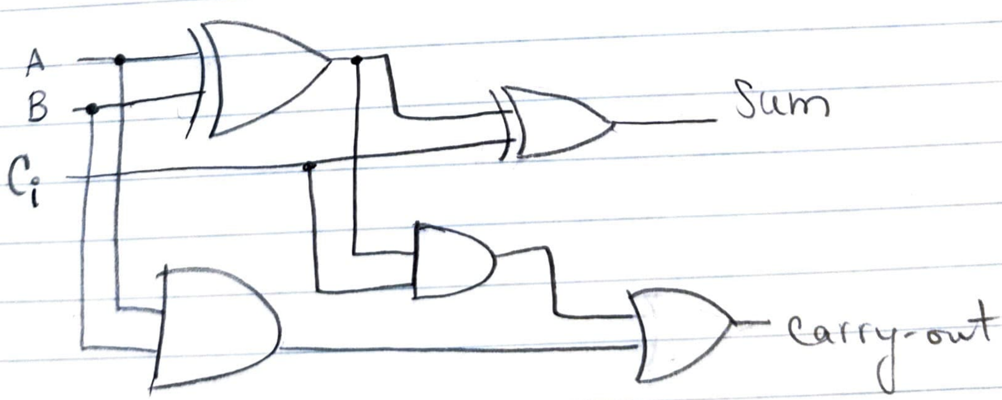


Full-adder - includes carry-in number



A	B	C_i	C_o	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

$$S = A'B'C_i + A'BC_i' + AB'C_i' + ABC$$
$$C_o = A'BC_i + AB'C_i + ABC_i' + ABC_i$$



Better explanation: Representing negative numbers

One's complement : 0000

has problem of 0 being + and −

0 → positive
1 → negative

Sign indicator flip; flip all bits for storing value

2 = 17
1 = 16
0 = 15
−1 = 14
−2 = 13
−3 = 12
−4 = 11
−5 = 10

$$\begin{array}{r} 3 \\ -3 \\ \hline 0 \end{array} \qquad \begin{array}{r} 3 \\ +12 \\ \hline 0 = 15 \end{array}$$

$$\begin{array}{r} 3 \\ -3 \\ \hline 0 \end{array} \qquad \begin{array}{r} 0011 \\ +1100 \\ \hline 1111 \end{array} \leftarrow -0?$$

Two's complement : 0000

→ discard (minus 16 again)

Flip sign indicator; flip and add 1

7 = 23
6 = 22
5 = 21
4 = 20
3 = 19
2 = 18
1 = 17
0 = 16
−1 = 15
−2 = 14
−3 = 13
−4 = 12
−5 = 11
−6 = 10
−7 = 9
−8 = 8

$$\begin{array}{r} 3 \\ -3 \\ \hline 0 \end{array} \qquad \begin{array}{r} 3 \\ +13 \\ \hline 16 \end{array}$$

$$\begin{array}{r} 3 \\ -1 \\ \hline \end{array} \qquad \begin{array}{r} 3 \\ +15 \\ \hline 18 \end{array}$$

$$\begin{array}{r} 3 \\ -2 \\ \hline 1 \end{array} \qquad \begin{array}{r} 3 \\ +14 \\ \hline 17 \end{array}$$

$$\begin{array}{r} 3 \\ -3 \\ \hline 0 \end{array} \qquad \begin{array}{r} 0011 \\ +1101 \\ \hline (1)\,0000 \end{array}$$
↳ discard

$$\begin{array}{r} 0100 \\ -0011 \end{array} \qquad \begin{array}{r} 4 \\ -3 \\ \hline 1 \end{array} \qquad \begin{array}{r} 0100 \\ +1101 \\ \hline (1)0001 \end{array}$$
positive

Take the number, add 16 (flip the bits and add 1) add them together, minus 16 (by discarding the remainder) (or by flipping +1 again) → only if answer is −
4th bit in original answer → sign indicator

42 ↑ Trevor's answer

$$0010 \quad \begin{array}{r} 5 \\ -2 \\ \hline \end{array} \qquad \begin{array}{r} 0011 \\ +1110 \\ \hline 0001 \end{array}$$

$$0101 \quad \begin{array}{r} 2 \\ -5 \\ \hline -3 \end{array} \qquad \begin{array}{r} 0010 \\ +1011 \\ \hline 1101 \end{array}$$
0011 → −3

$$\begin{array}{r} 10.1 \\ 1001\overline{)1010} \\ 100 \\ \hline 100 \end{array}$$

$$\begin{array}{r} 10 \\ -11 \\ \hline \end{array} \qquad \begin{array}{r} 10 \\ 01 \\ \hline 11 \end{array}$$

11

−00

0 4
−1 3
−2 2
−3 1

Binary Multiplication
(Same as base-ten)

$$\begin{array}{r} 10 \\ \times\ 2 \\ \hline 20 \end{array} \qquad \begin{array}{r} 1010 \\ \times\ 10 \\ \hline 0000 \\ 10100 \\ \hline 10100 \end{array} \qquad \begin{array}{r} 9 \\ \times 5 \\ \hline \end{array} \qquad \begin{array}{r} 1001 \\ \times\ 101 \\ \hline 1001 \\ 00000 \\ 100100 \\ \hline 101101 \end{array}$$

Binary division (You can start crying now)

$$\frac{10}{2} \qquad \begin{array}{r} 5 \\ 2\overline{)10} \\ -10 \\ \hline 0 \end{array} \qquad \begin{array}{r} 101 \\ 10\overline{)1010} \\ 10 \\ \hline 01 \\ 0 \\ \hline 10 \end{array} \qquad \frac{45}{5} \quad \begin{array}{r} 9 \\ 5\overline{)45} \end{array} \qquad \begin{array}{r} 1001 \\ 101\overline{)101101} \\ 101 \\ \hline 0101 \end{array}$$

"A hard one"

$$\frac{7}{4} = 1.75 \qquad \begin{array}{r} 1.75 \\ 4\overline{)7.00} \\ 4 \\ \hline 3.0 \\ 28 \\ \hline 20 \end{array} \qquad \begin{array}{r} 1.11 \\ 100\overline{)111.000} \\ 100 \\ \hline 110 \\ 100 \\ \hline 100 \\ 100 \end{array}$$

$$\ldots\ 8\quad 4\quad 2\quad 1.\quad \tfrac{1}{2}\quad \tfrac{1}{4}\quad \tfrac{1}{8}$$

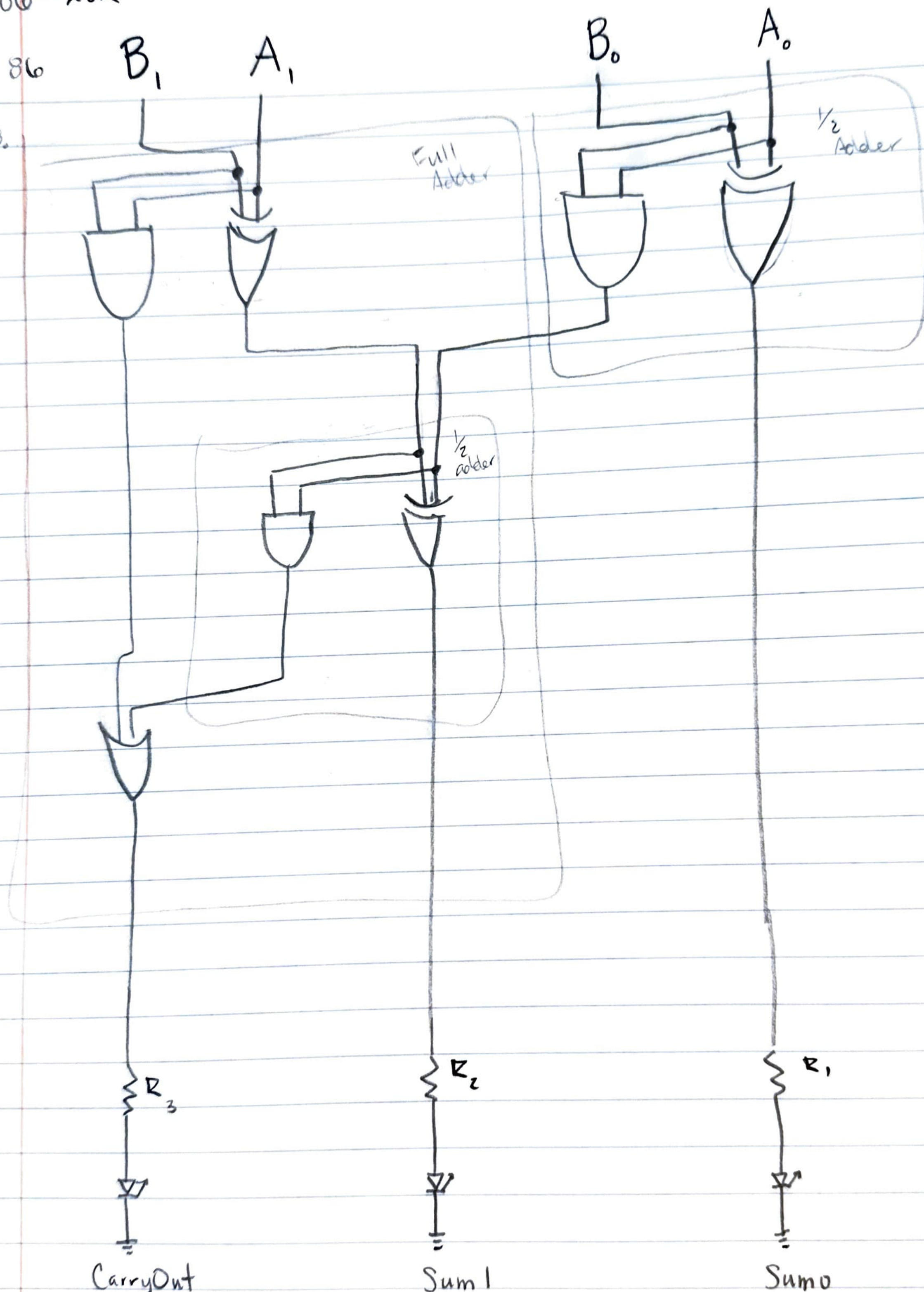
$$1.11 \longrightarrow 1 + \tfrac{1}{2} + \tfrac{1}{4} = 1\tfrac{3}{4} = 1.75$$

7408 - AND
7432 - OR
7486 - XOR

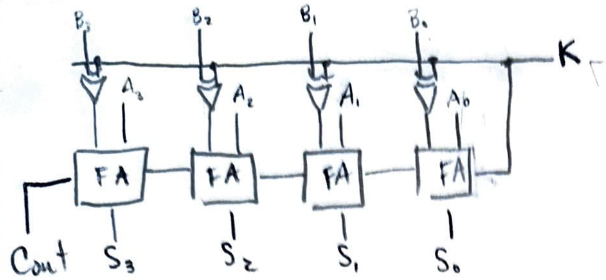
FULL ADDER

08 32 86

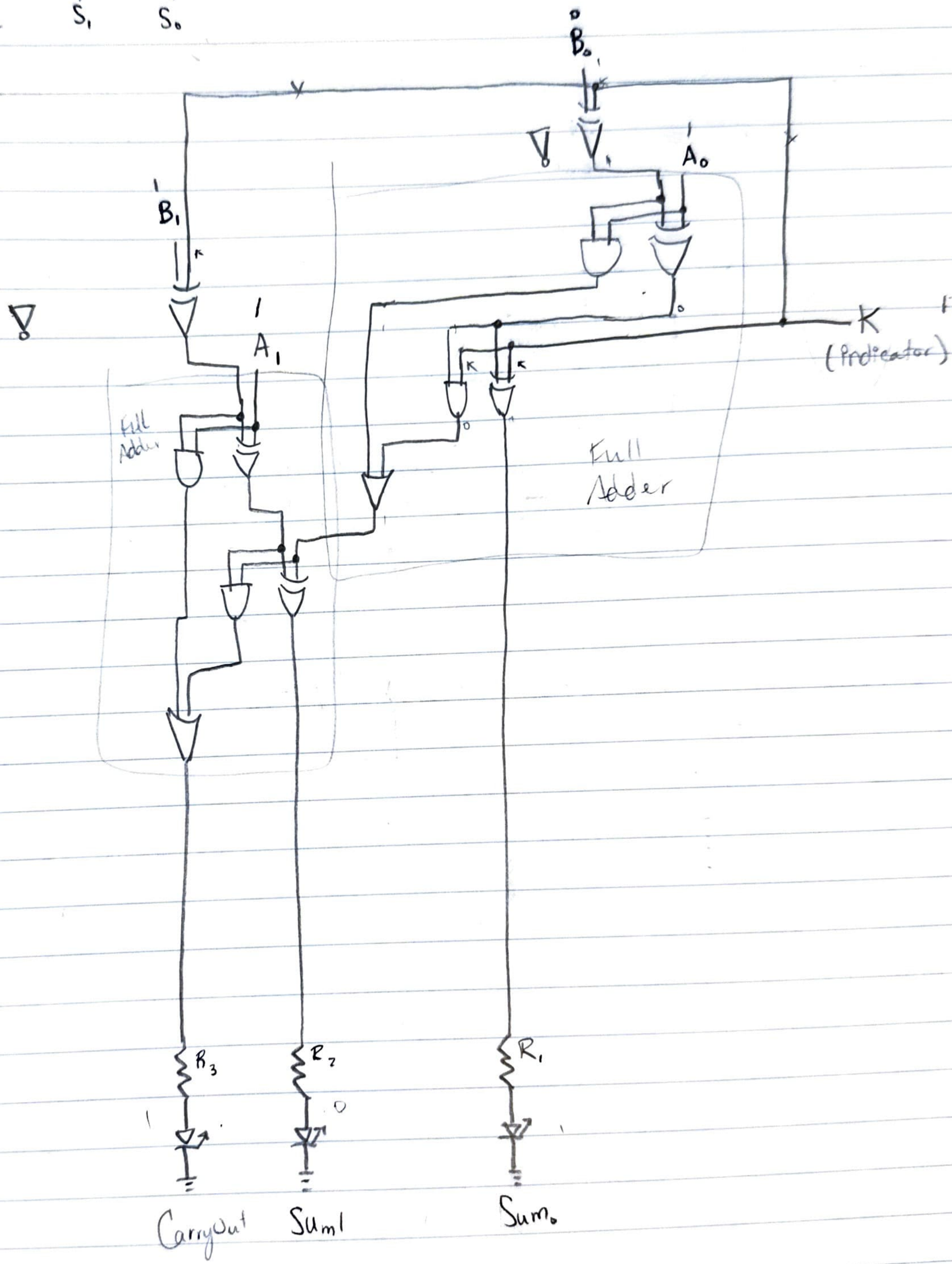
$A_1A_0 + B_1B_0$



$A_1A_0 + B_1B_0 = \text{CarryOut}\,\text{Sum}_1\,\text{Sum}_0$



FULL ADDER / SUBTRACTOR

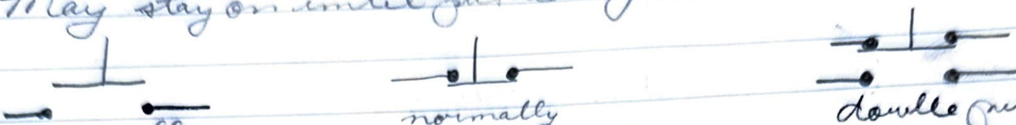


$A_1A_0 - B_1B_0$ = Carryout Sum1 Sum_0

(For only subtractor, change initial XOR gates to inverters and only 1 input K into second XOR in first full-adder)

Buttons

Tactile buttons
- May have more than one lead
- May stay on until pushed again

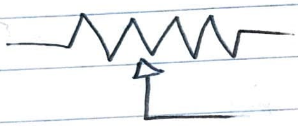


normally open push button — normally closed push button — double pull button

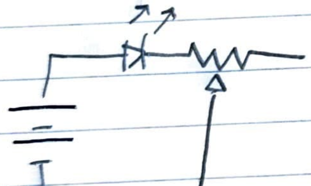
"wiper"

Potentiometers

- three metal leads + pole comes out to turn
- variable resistor 200—1000 Ω (resistive material)
  - wiper moves along track as turns pole
  - can hook up to one end and middle to control resistance



schematic symbol



Capacitors

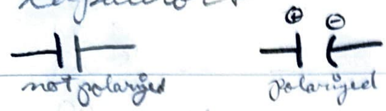
- stores electric energy
- not the same as a battery

Electrolytic capacitor
- longer lead → anode (like LED)

Ceramic Disc capacitor
- not polarized
- smaller values than

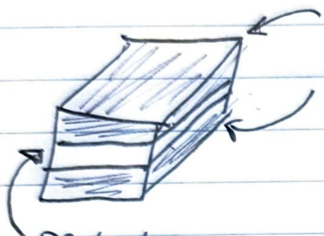
Poly film capacitors
- not polarized

capacitors



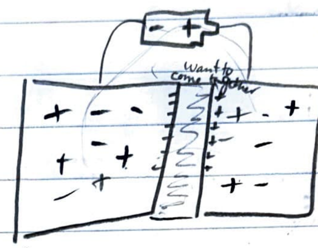
Farads (F)

$\mu F \rightarrow$ micro farads 0.000001 F



Conductive material

Dielectric Material (Mainly an insulator)



negative flows out, and leads this + charged

becomes more negative

When connected to something, it wants to equalize and acts like battery

No current flows "through" the capacitor

τ = Resistance (Ω) $\cdot$ Capacitance (F)

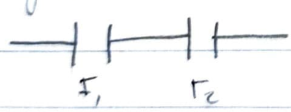
(s)

τ (towe) time to charge 63%

98% after 4τ

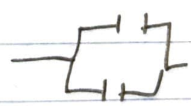
ex; $\tau = 10s$, $\tau \cdot 4 = 40s$ to 100% charge

Capacitors in series



$$\frac{1}{T_F} = \frac{1}{F_1} + \frac{1}{F_2}$$

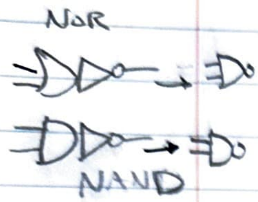
Parallel



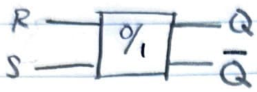
$T_F = F_1 + F_2 \ldots$

Latches

set, reset

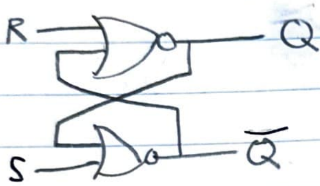


SR Latch — a one bit memory



Reset w/ high logic

latch w/ NOR gates: controlled by pulses

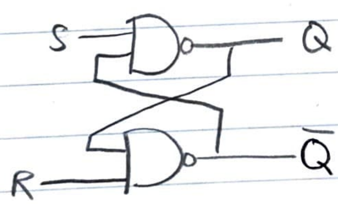


To reset, R put to 1 and $\bar{Q}$ is 1
To Set, S = 1 and Q is 1

S	R	Q	$\bar{Q}$	
0	0	1 / 0	0 / 1	
0	1	0	1	
1	0	1	0	
1	1	0	0	← INVALID. leads to "race" problem

Reset w/ low logic

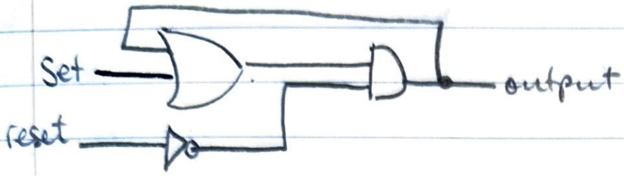
latches w/ NAND gates: "active-low SR latch"



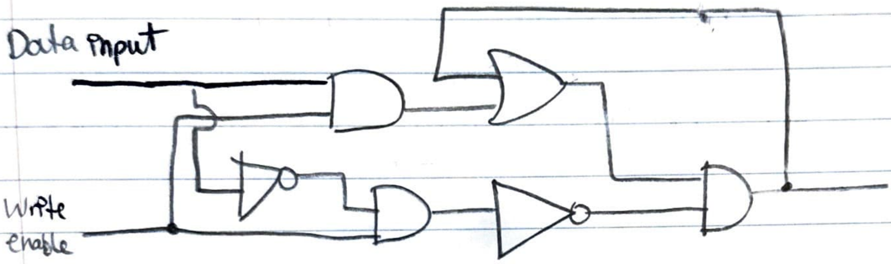
S	R	Q	$\bar{Q}$	
0	0	1	1	← INVALID
0	1	1	0	
1	0	0	1	
1	1	0 / 1	1 / 0	

Debouncing effect??

And-OR Latch



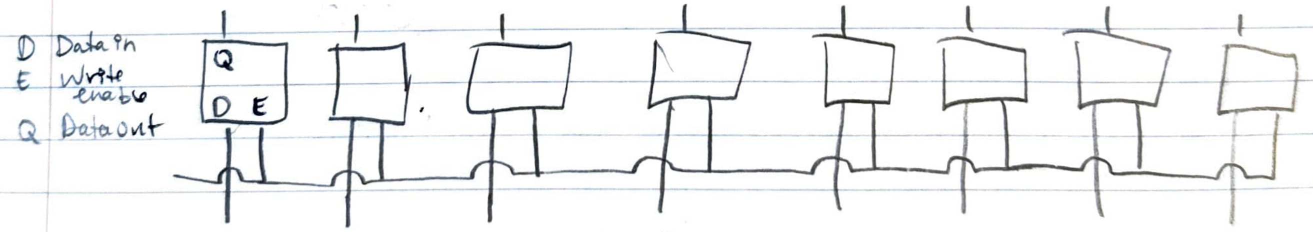
Gated latch



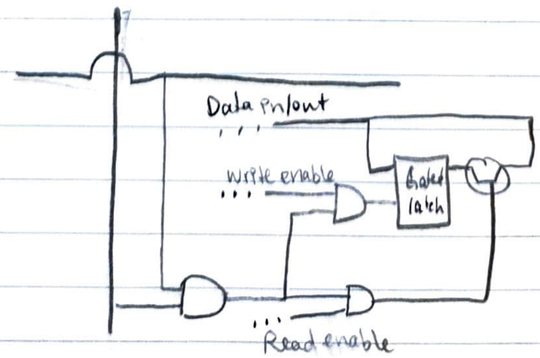
In order to change memory, the write enable needs to be on

Group of latches → register
the # of latches → width

8-BIT REGISTER

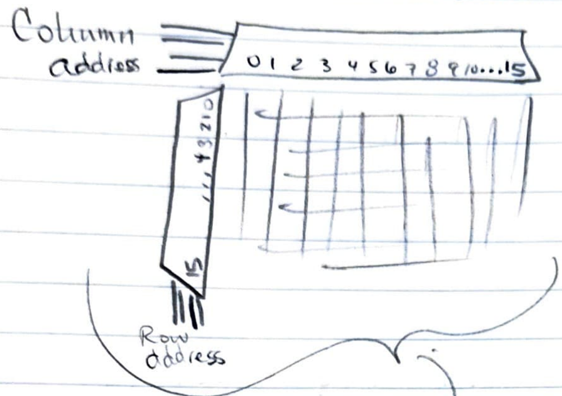


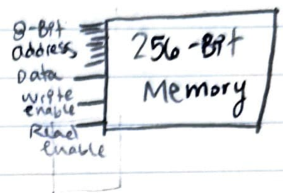
Lots of wires? solution → matrix
to turn on latch, turn on rows and column wires
AND write enable

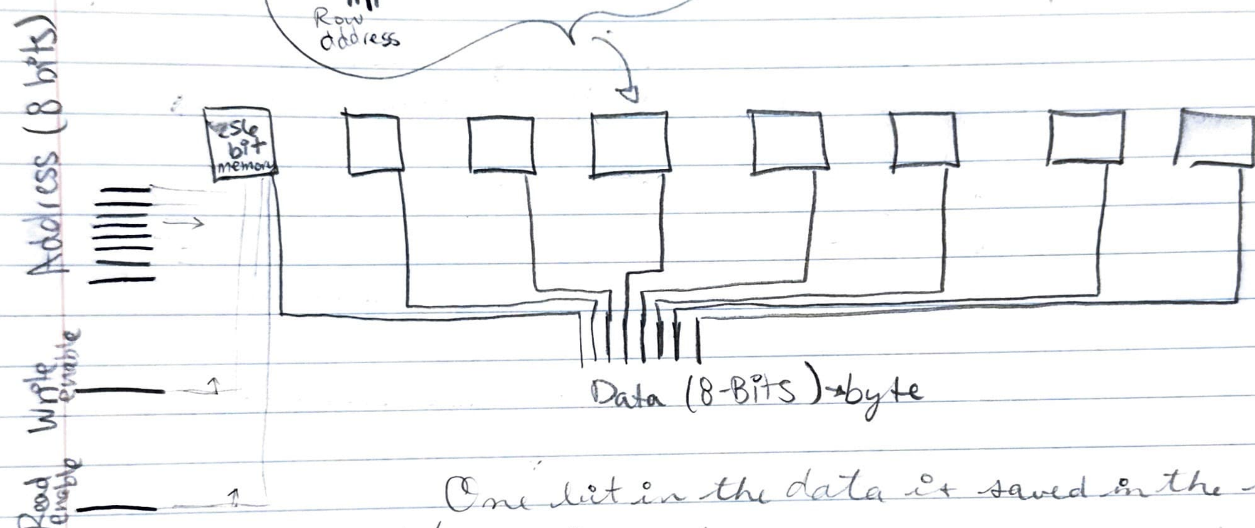


Memory address → binary
12,8
11001000

to choose row and column → multiplexer



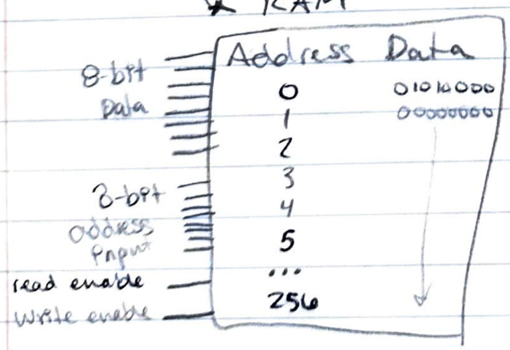




One bit in the data is saved in the same adress in each memory

Think of all of this as just one bank (SRAM)

RAM → Random Access Memory
(Can access it at any random time)
Short term memory

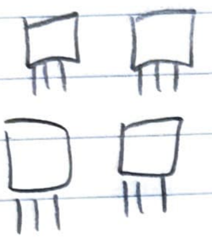


cache

Main memory

Static RAM
- Stores less
+ Faster access
+ Use less power
(Only transistors)

6
istors



= 1 Bit

Less storage because the transistors take up space

• Uses boolean logic
- constant state
- faster access

- Low Capacity
- Hard change
- Fast access
☆ Small but important info ☆

Dynamic RAM
+ Stores more
+ Faster rewrite
+ cheaper

Capacitor
Transistor

} 1 BIT

Capacitor loses charge super fast, but that's okay because RAM changes so fast anyway

- High Capacity
- Easy Change
- Slow access
☆ Everything else ☆

• SRAM is to RAM as RAM is to computer
• DRAM is to RAM as HardDrive is to computer

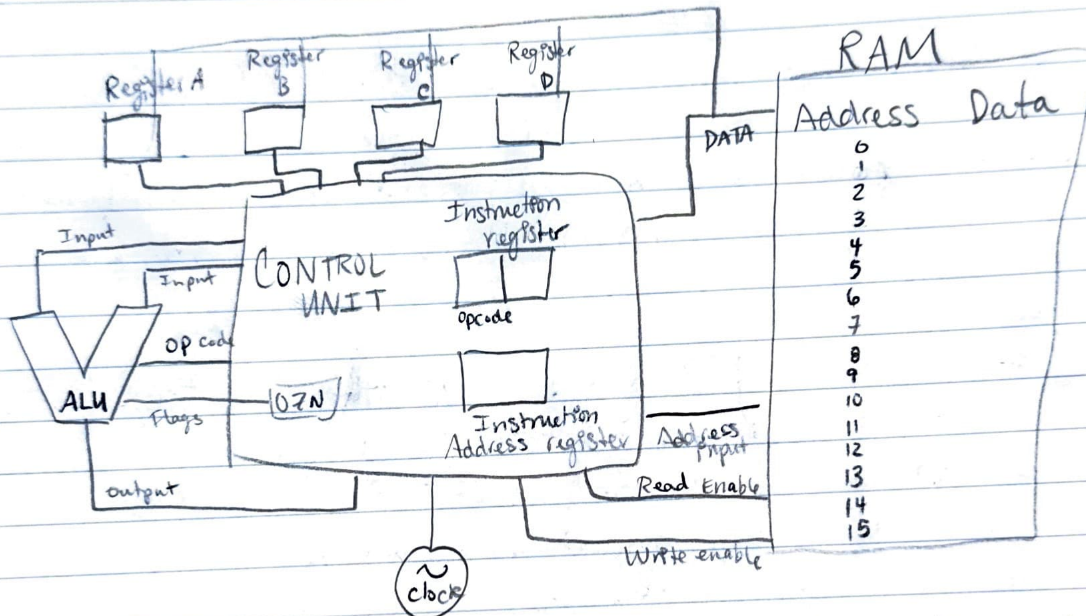
Flip-Flop

'edge-triggered' → has a clock that that changes periodically

CPU (Central Processing Unit)

- executes programs

Micro-architecture



O - overflow
Z - zero
N - negative

Fetch phase: retrieve instructions, Instructions address register connects to RAM, and its copied into instruction register

Decode Phase: First 4 bits → "opcode", which corresponds with instructions in instruction table
RAM address → last 4 bits
- Put into control unit

Execute phase: Performs instructions, and adds 1 to the instructions address register

Instruction table

		4-bit opcode	Address or registr
LOAD_A	Read RAM location into rA	0010	4-bit RAM address
LOAD_B	Read " " " rB	0001	"
STORE_A	Write from rA into RAM location	0100	"
ADD	Add two registers, store result into second register	1000	2-bit register ID, 2-bit register ID
...	...	...	...

Speed which a CPU can carry out – Clock Speed
– Measured in Hertz = 1 cycle/second

Variable length instructions? – instructions can be any length??

Pseudo-code → informal code instructions

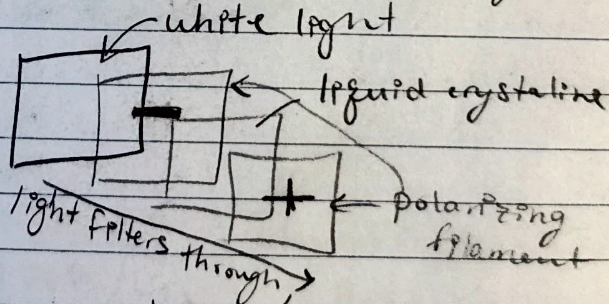
Von Neumann → instructions and data put in memory

Decode – type of code to be executed is determined
directly to machine language

Optical Mice – takes 2,000 pictures/second, and compares it to the previous one to compare where it has gone, 1600 pixels/in^2
– images not stored
– sends info to CPU, occurs in the "Kernel" operating system

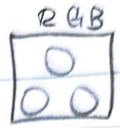
Computer monitors
RGB color-scheme → can make any color

LCD monitors –



and each pixel on the filter screen is charged by an electrical signal, allowing it to change the color

LED monitor, has a layer of LED pixels



RGB each has 255 diffrent shades it can be

- Computers uses hexadecimal

3F 1A B3

R G B

Touchscreens

Resistive - like a keyboard but it's pushing down a conducting polyester onto a conducting glass

Capacitive - multiple layers of conductive glass that change when a conductive finger touches it

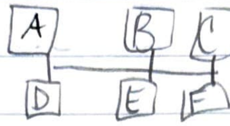
Infared - interrupts beams of light

Surface Acoustic Wave - uses Ultrasonic sound waves

Near field imaging - electric field changes and registers touch

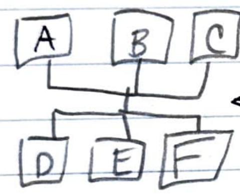
Computer Network

Ethernet



Mac address → to which computer

Bandwidth - rate a carrier can transmit data
Exponential backoff - if the line is in use, computer waits 1 second,... 2... 4... 8...



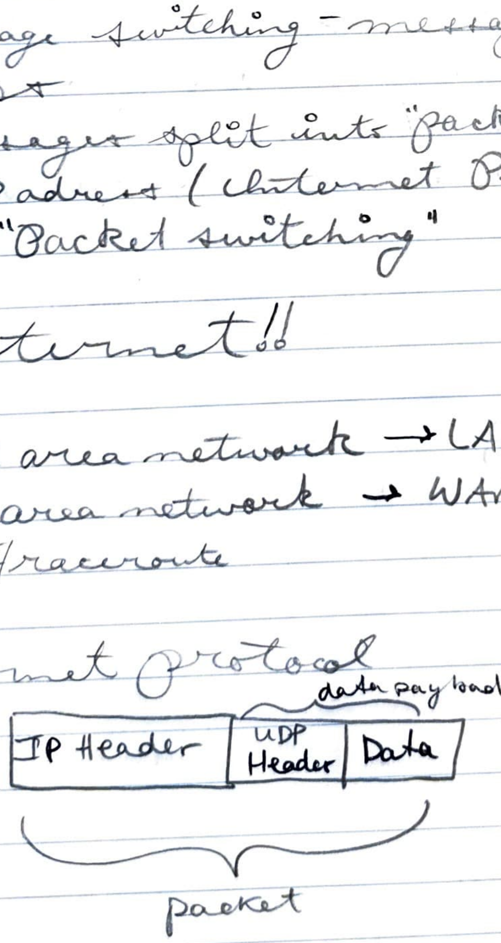
← 2 collision domain

Message switching - messages passed through several stops
Messages split into "packets", each has an address, or IP address (Internet Protocol)
- "Packet switching"

Internet!!

Local area network → LAN
Wide area network → WAN
Traceroute

Internet protocol



User datagram Protocol -
= Port checksum...
Port # → which application
Checksum → adds all the data, and checks to see if it was the same b4 & after. If not, something went wrong

Transmission Control Protocol — TCP/IP
- adress
- checksum
- given sequential numbers (can be un-scrambled)
- sent a an "ACK" (acknowledgment) that it was sent alright

Domain Name System → "phonebook" google.com → 172.217.7.238:80

Domain Structure

Top-level domains	.Org	.com
Second-level "	google.com	dftba.com
Sub-Domain of parent	drive.google.com	store.dftba.com

Layers of OSI Model
(Open system interconnection)

Application
Presentation
Session layer — open & close connection
Transport layer — UDP & TCP
Network layer — routing
Data link layer — mediate access to ↓
Physical layer — wires & air

server - computer that holds the data for a website

Coaxial - connects to great network

Modem + Router = Gateway

- Provides ethernet & wifi

Mac address → device

IP address → gateway

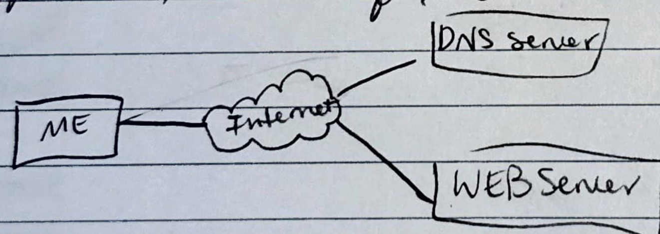
The cloud → data saved in a server backed up

The World Wide Web ≠ internet

↳ connected by hyperlinks (hypertext)

↳ specified by URL

Hypertext transfer Protocol → HTTP



Hypertext Markup Language - HTML

Search Engine → searches websites for words

Net neutrality - debate of whether all packets should be treated equally