Computer fundamentals

IB DP Computer Science A1 Computer fundamentals full package

What happens when you tap an app?

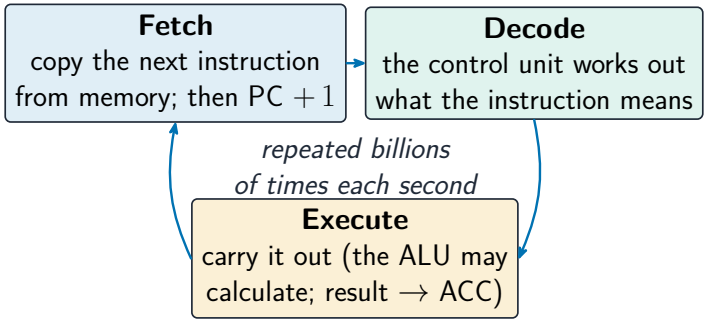
> One tap uses every part of the computer.

- Theme A1: hardware, data representation, logic and operating systems.
- Prerequisites: powers of 2, such as 2 to the 10th = 1024.

Inside the CPU

Part	Job
control unit (CU)	fetches and decodes; sends control signals
ALU	arithmetic and logic
registers	tiny, very fast stores inside the CPU

The fetch-decode-execute cycle



Fetch, decode, execute - billions of times a second.

CPU and GPU

CPU

A few powerful cores.

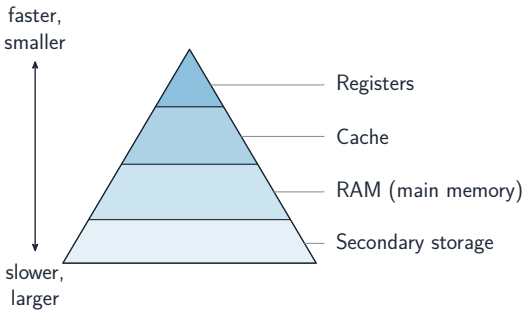
Varied tasks with many decisions: the OS and apps.

GPU

Thousands of simple cores.

The same calculation on lots of data: graphics, video, ML training.

The memory hierarchy



Faster and smaller at the top, slower and larger below.

Secondary storage

Type	Stores bits as	For / against
HDD	magnetised spots	cheap; moving parts, slower
SSD	charge in flash memory	fast, robust; dearer
optical	pits read by a laser	cheap to share; small, slow

Lossless or lossy?

Lossless

The original can be rebuilt exactly.

Text and program files; run-length encoding.

Lossy

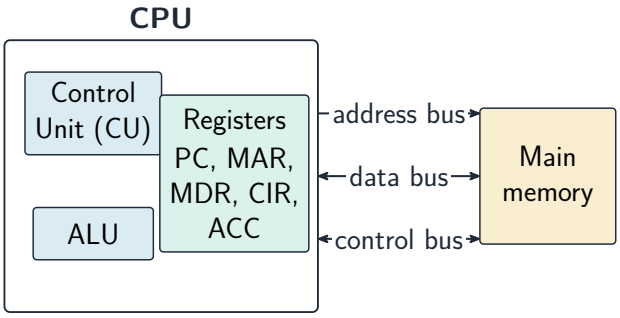
Removes detail people will not notice.

Photos (JPEG), music (MP3), video.

SaaS, PaaS and IaaS

Service	Provider supplies	Example
SaaS	a finished application	web e-mail
PaaS	a platform for your own apps	hosted database
IaaS	virtual machines and storage	servers by the hour

The von Neumann architecture



Program and data share one memory

182 in binary and hex

$$128 + 32 + 16 + 4 + 2 = 182$$

- n bits give 2 to the n values: 8 bits store 0 to 255.
- One hex digit is exactly 4 bits: 1011 0110 is B6 $= 11 \times 16 + 6 = 182$.

Two's complement: -42

1. **42** 00101010
2. **Flip every bit** 11010101
3. **Add 1** 11010110

Decision

Check: -128 + 64 + 16 + 4 + 2 = -42.

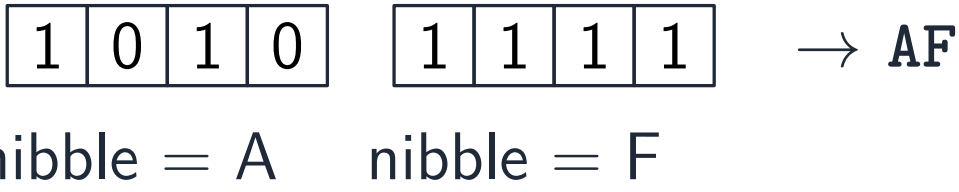
How big is the file?

1. **Image** 1920 x 1080 x 24 = 49,766,400 bits
2. **In bytes** / 8 = 6,220,800 bytes, about 6.2 MB
3. **Sound** 44,100 x 16 x 2 x 60 = 84,672,000 bits, about 10.6 MB

Decision

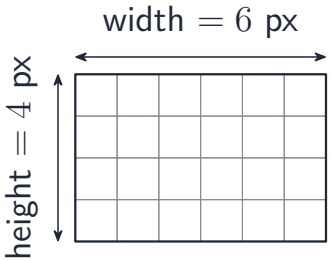
Multiply the parts, then divide by 8 for bytes.

Hexadecimal



Two hex digits per **byte**

The size of a bitmap



pixels $= 6 \times 4 = 24$

size $= 24 \times \mathbf{8} = \mathbf{192}$ bits

($\mathbf{8}$ = colour depth, bits per pixel)

> Size = width × height × **colour depth**

Logic gates

A B	AND	OR	XOR
0 0	0	0	0
0 1	0	1	1
1 0	0	1	1
1 1	1	1	0

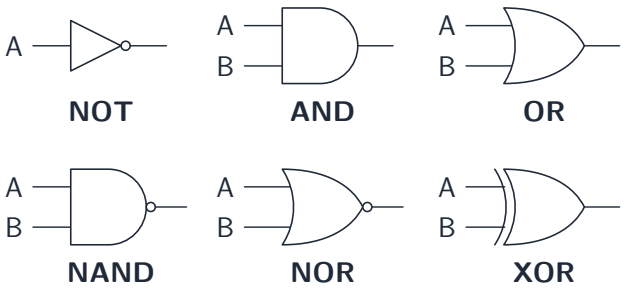
Tracing a circuit

1. **P** P = A AND B: 1 only in rows 110 and 111
2. **Q** Q = NOT C: 1 when C = 0
3. **X** X = P OR Q: 0 only when C = 1 and P = 0

Decision

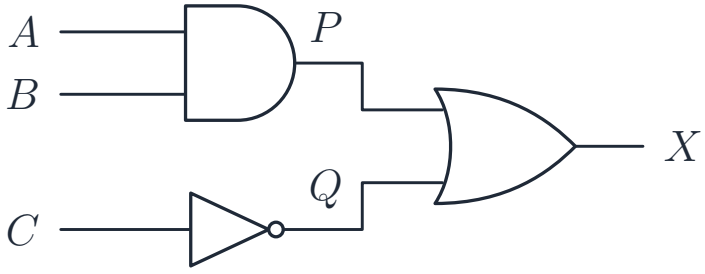
X = (A AND B) OR (NOT C) is 1 in five of the eight rows.

Logic gate symbols



Learn the symbols and their truth tables

Reading a logic circuit



$X = (A \wedge B) \vee \neg C$

What an operating system does

> It checks who you are before what you may do.

- Process, memory, file and device management (through drivers).
- A user interface, graphical or command line.

Round robin

P3 ends at 5 ms, P2 at 8 ms, P1 at 9 ms.

- First come, first served: a long job makes short ones wait.
- Priority: low-priority processes may never run.

Polling or interrupts?

Polling

The CPU checks each device again and again.

Simple, but wastes CPU time when nothing happens.

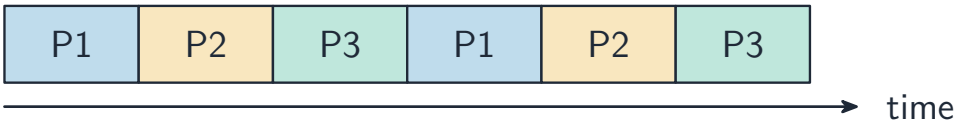
Interrupt

The device signals; the CPU saves its state and runs a handler.

Efficient for rare or unpredictable events.

Round-robin scheduling

each process gets a fixed time slice, then the next one runs



> Fair sharing, but switching has a cost

Practice

Your turn

Try these:

A. Describe the fetch stage, naming the registers.
B. Write -19 in 8-bit two's complement.
C. Find the size in bytes of an 800 × 600, 16-bit image.
D. P1 4 ms, P2 3 ms, 2 ms slice: when does each finish?

Check yourself

Fetch: PC → MAR → memory → MDR → CIR, and the PC increments

↓

Show place values for every conversion

↓

Formula, numbers, unit - then divide by 8 for bytes

↓

Truth tables: 2 to the n rows in **binary** order, a column per gate